



**HAL**  
open science

# Online Canonical Polyadic Decomposition: Application of Fluorescence Tensors with Nonnegative Orthogonality and Sparse Constraint

Isaac Wilfried Sanou, Xavier Luciani, Roland Redon, Stéphane Mounier

► **To cite this version:**

Isaac Wilfried Sanou, Xavier Luciani, Roland Redon, Stéphane Mounier. Online Canonical Polyadic Decomposition: Application of Fluorescence Tensors with Nonnegative Orthogonality and Sparse Constraint. Optimization Algorithms - Classics and Recent Advances, IntechOpen, 2024, 10.5772/intechopen.110123 . hal-04647468

**HAL Id: hal-04647468**

**<https://hal.science/hal-04647468>**

Submitted on 14 Jul 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

7,100

Open access books available

188,000

International authors and editors

205M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



Chapter

# Online Canonical Polyadic Decomposition: Application of Fluorescence Tensors with Nonnegative Orthogonality and Sparse Constraint

*Isaac Wilfried Sanou, Xavier Luciani, Roland Redon and Stéphane Mounier*

## Abstract

The canonical polyadic decomposition (CPD) is now widely used in signal processing to decompose multi-way arrays. In many applications, it is important to add constraints to quickly converge on an optimal solution. In contrast to classical CPD, we then focus on online CPD. In this context, the number of relevant factors is usually unknown and can vary with time. We propose two algorithms to compute the online CPD based on sparse dictionary learning. We also introduce an application example in environmental sciences and evaluate the performances of the proposed approaches in this context on real data.

**Keywords:** third order tensor decomposition, online tensor decomposition, PARAFAC, rank variation, fluorescence spectroscopy

## 1. Introduction

In many fields such as psychometric [1], data mining [2], neuroscience [3], chemometric [4], telecommunications [5], computer vision [6], and biomedical image processing [7], use canonical polyadic decomposition (CPD) also known as PARAllel Factor analysis (PARAFAC).

The data from these fields can be put in a multidimensional data array (or tensor). Then, the CPD permits to decompose this array into factors (which is composed of the number of components or rank depending to the application) that can be interpreted by the user.

CPD algorithms can be summarized in three mains methods:

- *Direct methods.* These methods are based on algebraic computation such as Direct TriLinear Decomposition (DTLD) [8], SEMi-algebraic framework for approximate

CP decompositions via Simultaneous matrix diagonalization (SECSI) [9], and Direct ALgorithm for canonical polyadic decomposition (DIAG) [10].

- *Alternating methods.* These methods are based on least square methods, and we can cite alternating least square (ALS) [1], hierarchical alternating least square (HALS) [11–13], and alternating direction method of multipliers (ADMM) [14].
- *Descent methods.* These methods are based on traditional gradient descent. We can cite some algorithms using stochastic gradient descent [15–17] or Nadam optimizer [18].

By considering these algorithms, we can say that the rank of the decomposition, namely the relevant number of factors, is known, but we find in practice that this is not the case. Indeed, this value is unknown so we make an estimate on the order of magnitude while a bad choice can have an impact on the factorial estimates [19]. Traditionally, two opposing approaches are used to address this situation.

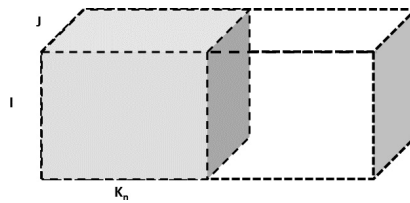
- *Rank estimation.* Rank estimation consists in estimating the appropriate CPD rank before the decomposition. Among these methods, we can cite the CORE CONSistency DIAGnostic (CORCONDIA) [20], split half validation [21], or AutoTen [22]. Though these approaches do not always allow to clearly decide between several possible rank values.
- *Overfactoring.* In this method, an overestimated value is chosen, i.e., one higher than the current rank and the CPD algorithms are designed to produce additional factors with zero contribution [19]. Thus, in contrast, overfactoring is a posterior rank estimation method because the appropriate rank value is inferred from the CPD.

At this stage, we have talked about offline CPD. However, in this chapter, we are interested in online CPD. In online CPD, the data increases with the time, and the decomposition must follow this augmentation [23].

In the context of fluorescence data and environmental sciences, the online CPD can be summarized as follow:

- 3-way array is used. This tensor is built by concatenation of a new matrix on the last mode. At each time we got a tensor call sub-tensor as shown in **Figure 1**.
- In practice, some components or the rank of the decomposition can change at interval time from one sub-tensor to another. This phenomenon can be seen as appear and/or disappear of component. We talk about the rank variation but these variations are unknown.
- The goal of online CPD in this context is to update at each new time interval the CPD factors using the factors estimated previously without performing the CPD of the whole.

In the literature, some online CPD algorithms exist. For example, in [23], the authors introduced two adaptive algorithms: simultaneous diagonalization tracking (SDT) which tracks the SVD of the unfolded tensor and recursive least squares tracking (RLST). In [24], grid-based tensor factorization algorithm (GridTF) is introduced for the large tensor. Also, in [25] authors proposed OnlineCP based on ALS



**Figure 1.**  
*Example of a sub-tensor.*

algorithms. In [26] authors introduced an online Tucker decomposition with a fixed rank. This list is not exhaustive, and more online algorithms can be found in [27].

For the rest of the chapter, we propose an approach based on sparse dictionary learning to compute the CPD with sparsity, nonnegativity, and orthogonality constraint. These constraints permit to handling unknown rank variation over time and the better convergence. Considering the most general case, we make no assumptions about these variations. The goal of the proposed approach is to combine dictionary learning with LASSO [28] in a simple but appropriate way for nonnegative online CPD. We have selected the factors of the two fixed modes from two dictionaries which are learned and updated throughout the online process. The appropriate number of vectors is selected from a sparsity constraint on the two atoms matrices (i.e., the CPD rank). We derived from this solution three different algorithms (one for offline CPD and the rest for online CPD). They differ in the way that the dictionaries are updated from one sub-tensor to another. Next, we propose an experiment semi-controlled to obtain real fluorescence data online with rank variations. We provide out an evaluation of our approaches with the state of the art.

Notations are introduced while recalling some basic definitions and the problem of online CPD in the particular context of fluorescence spectroscopy and environmental sciences in Section 2. In Section 3, the new proposed approach is described in the form of three algorithms for computing the nonnegative CPD of a third-order tensor. Two of these algorithms were already presented in a previous conference paper [18] and this journal paper [27]. In Section 4, an experimental design is proposed to evaluate the approaches. Results and comparisons with reference approach are provided and discussed. Section 5 concludes.

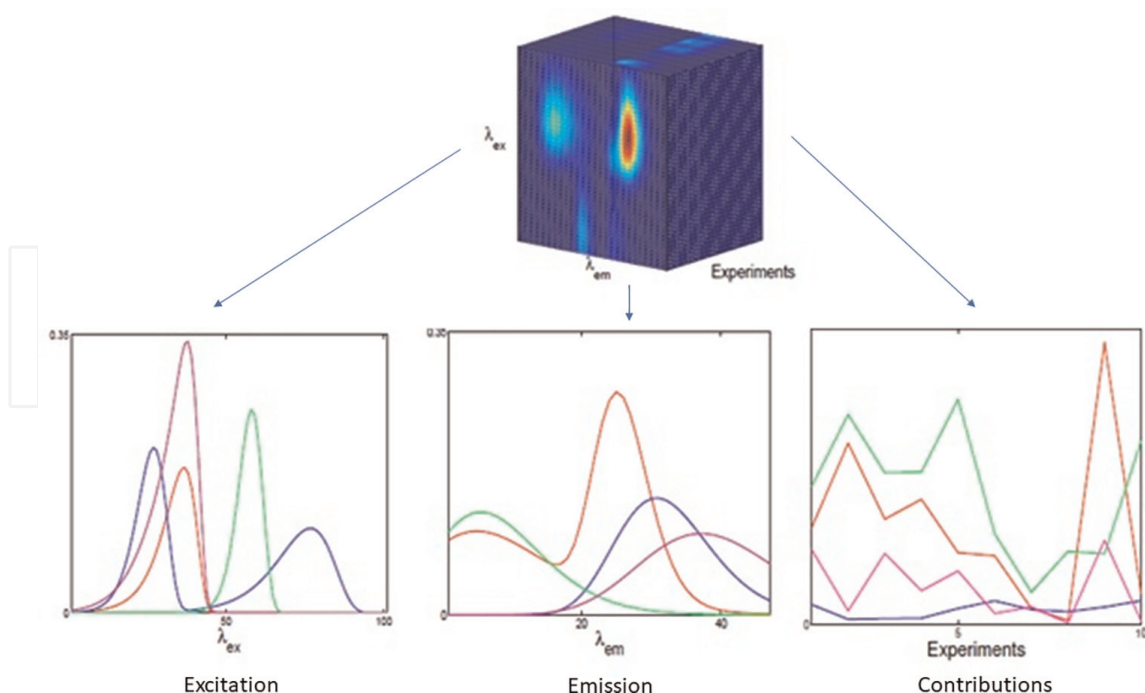
## 2. Problem formulation

We denote tensors and sub-tensors with letter  $\mathcal{T}$  and are of size  $(I \times J \times K)$  meaning that  $\mathcal{T}$  gather  $K$  matrix of size  $(I \times J)$  on its last mode.

### 2.1 Offline CPD: Example of a fluorescence tensor

In environmental sciences, especially for fluorescence data tensors the third-order tensor is most used. The CPD of this tensor can be used in order to characterize fluorescent dissolved organic matters (FDOM) in natural water samples [21, 27, 29–31] (see **Figure 2**).

Fluorescence data tensors are a set of 2D signals called emission and excitation matrices (EEMs) of fluorescence measured from a set of samples. Each sample is then a mixture of an unknown number of fluorescent components (fluorophores), and each part of an EEM corresponds to the fluorescence intensity of one sample at a



**Figure 2.**  
 Example of the rank 4 CPD of the fluorescence tensor [18, 27].

given couple of excitation and emission wavelengths. At low concentrations, the nonlinear model based on the Beer–Lambert law can be linearized, and it then follows the CPD model [32]. To recover each individual emission and excitation spectra of the fluorophores present in the different samples along with their respective contributions can be found using the CPD. In some applications like in our case, the decomposition factors have physical meaning and are known to be nonnegative. This constraint allows better convergence especially when the columns of the factors are collinear. Nonnegativity constraint can be imposed on the factors during optimization by projecting the values of each factor matrix on  $\mathbb{R}^+$  [33]. This method of applying nonnegativity constraint sometimes shows its limits in certain cases due to the change of scale. In [34], the authors propose to solve this problem by involving two factors,  $\epsilon$  and  $\theta \in \{-1; 1\}$ , which multiply each factors. Nonnegativity can also be imposed by choosing an optimization method [14, 17, 35] or even using an exponential variable change [31]. In the literature, the projection method is mostly used such as in approach [14, 17, 18, 33, 36, 37]. In signal or data processing, the nonnegative CPD is usually used as a mathematical model to fit a data tensor  $\mathcal{T}$  of size  $(I, J, K)$ :

$$\forall i, j, k, \quad \mathcal{T}_{i,j,k} \simeq \widehat{\mathcal{T}}_{i,j,k}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \sum_{r=1}^R \mathbf{A}_{ir} \mathbf{B}_{jr} \mathbf{C}_{kr} \quad (1)$$

The matrices  $\mathbf{A} \in \mathbb{R}_+^{I \times R}$ ,  $\mathbf{B} \in \mathbb{R}_+^{J \times R}$  and  $\mathbf{C} \in \mathbb{R}_+^{K \times R}$  are named the factor matrices.  $R$  is the CPD rank. Each column of the factor matrices  $\mathbf{A}$ ,  $\mathbf{B}$ , and  $\mathbf{C}$ , defines the CPD factors. We can distinguish two values of  $R$  if the CPD factors have a physical meaning:

- The define value of the tensor  $\mathcal{T}$ .
- The maximal value of  $R$  for which all the factors have a physical meaning. We name this value the physical rank of  $\mathcal{T}$ . It is usually much smaller than the tensor rank.

In the considered application, these matrices are usually full column rank so that the Kruskal condition [38] is fulfilled and guarantees that the decomposition is unique up to trivial scaling and permutation indeterminacy.

The *euclidia* distance  $F$  is the most used to the reconstruction error. With nonnegative constraint, the offline CPD problem is thus given by

$$\min \left\{ F(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \|\mathcal{T} - \widehat{\mathcal{T}}(\mathbf{A}, \mathbf{B}, \mathbf{C})\|_F^2 \right\} \quad (2)$$

$$s.t. \mathbf{A} \geq 0, \mathbf{B} \geq 0, \mathbf{C} \geq 0$$

## 2.2 Fluorescence tensor in online CPD

In the online case, tensor  $\mathcal{T}$  can be seen as

- A composition of **sub-tensors**  $\mathcal{T}_n$  of size  $(I, J, K_n)$  acquired at various time intervals.
- Each sub-tensor can have zero matrix in common between last and the present sub-tensor. We call this case partition with *nooverlapping*.
- In opposite case, we talk about overlapping case. As explained on **Figure 3**, two consecutive sub-tensors share some matrices, *i.e.*, in this case,  $\mathcal{T}_n$  also contains the last  $\eta$  EEMs of  $\mathcal{T}_{n-1}$ . Sub-tensor overlapping should help the factor estimations but then the online decomposition becomes slower because we find ourselves with a larger number of sub-tensors.
- Rank variation or fluorophore appearance and disappearance can be explained by sea currents or pollution events, natural degradation ... , in a natural marine environment.

The physical rank of sub-tensor  $\mathcal{T}_n$  will be denoted  $\tilde{R}_n$  in this chapter, and we recall that it is unknown. The online CPD problem with rank variation has been clearly described in [18, 27, 39].

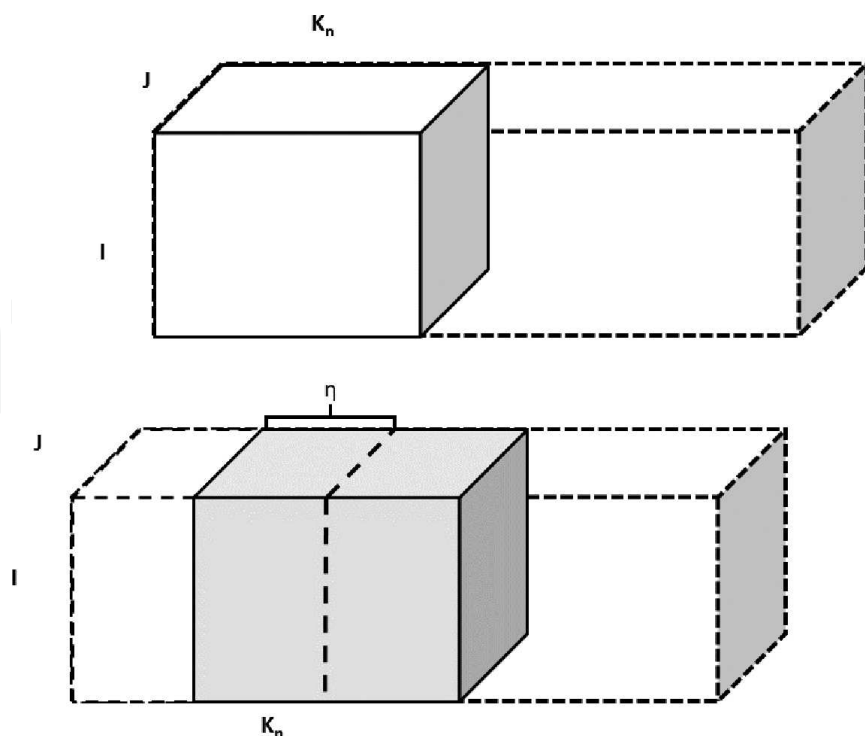
In short, the online CPD consists by solving Eq. (2) for each sub-tensor  $\mathcal{T}_n$  with the nonnegative constraint. We speak of: NonNegative Online Canonical Polyadic Decomposition (NNOnline CPD). The factor matrices estimated at time  $t_{n-1}$  with the sub-tensor  $\mathcal{T}_{n-1}$  are used to update the sub-tensor  $\mathcal{T}_n$  in order to reduce the computational cost.

## 3. Algorithms for computing the nonnegative online canonical polyadic decomposition with sparsity and orthogonality constraint

### 3.1 NNCPD for the first tensor acquired at time $t_0$

The first sub-tensor acquired at time  $t_0$  is denoted  $\mathcal{T}_0$ . At this time, we do not know the value of  $\tilde{R}_0$  but we suppose that,  $\forall n, \tilde{R}_n \ll R$ . We aim to rewrite the matrices  $\mathbf{A}$  and  $\mathbf{B}$  as the product of sparse dictionaries  $\mathbf{D}^A$  and  $\mathbf{D}^B$  by atoms  $\mathbf{V}^A$  and  $\mathbf{V}^B$  :

$$\mathbf{A} = \mathbf{D}^A \mathbf{V}^A \text{ and } \mathbf{B} = \mathbf{D}^B \mathbf{V}^B \quad (3)$$



**Figure 3.**  
Example of a sub-tensor.

with  $\mathbf{D}^A(I, R)$  and  $\mathbf{D}^B(J, R)$ , respectively, and  $\mathbf{V}^A(R, R)$  and  $\mathbf{V}^B(R, R)$ . We expect that  $\mathbf{D}^A$  and  $\mathbf{D}^B$  contain  $\tilde{R}$  true components on their columns. The factors without physical meaning are  $R - \tilde{R}$ . While  $\mathbf{V}^A$  and  $\mathbf{V}^B$  have  $R - \tilde{R}$  null columns and that the other columns form generalized permutation matrix. For instance, for  $R = 3$  and  $\tilde{R}_0 = 2$ , we could have ideally

$$\mathbf{A} = \overbrace{\begin{pmatrix} 9 & 7 & 2 \\ 1 & 4 & 2 \\ 2 & 1 & 2 \end{pmatrix}}^{\mathbf{D}^A} \cdot \overbrace{\begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0.9 & 0 \end{pmatrix}}^{\mathbf{V}^A} = \begin{pmatrix} 7.2 & 1.8 & 0 \\ 0.8 & 1.8 & 0 \\ 1.6 & 1.8 & 0 \end{pmatrix}$$

We can easily estimate  $\tilde{R}_0$  by counting the number of non-null columns of  $\mathbf{A}$  or  $\mathbf{B}$ . A similar problem has been studied in depth by Cohen and Gillis in [40] but for the case in which the dictionary is known and the atom is a selection matrix. To have zeros columns on the atoms, we thus promote the sparsity constraints on the matrices  $\mathbf{V}^A$  and  $\mathbf{V}^B$ . In this purpose, we will aim at minimizing their  $L_{1,1}$  norm. Regarding the sparsity constraint, it is generally used for sparse tensors or to ensure the overestimation of the rank, as mentioned above. This constraint is imposed according to the goal we are looking for, and in our case, it is the overestimation of the rank. The constraint can be applied to the matrices factors in the form of a regularization term:

$$\min_x \underbrace{F(x)}_{\text{fonction}} + \underbrace{R(x)}_{\text{regularization}} \quad (4)$$



$F(x)$  corresponds here to Eq. (2) and is the attachment part.  $R(x)$  is the regularization part on the matrix  $\mathbf{a}$ ,  $\mathbf{b}$ , and  $\mathbf{c}$ .

$L_{1,1}$  norm is a classically used to promote the sparsity [41]. Indeed, it is more tractable than the  $L_{0,0}$  norm. It has been used in similar contexts in [17, 35, 36, 40]. We can thus rewrite our problem as a LASSO regression problem by adding two penalty terms to the cost function (see Appendix A).

For  $\mathcal{T} = \mathcal{T}_0$ , we solve

$$\begin{aligned} & \min \{F_1(\mathbf{D}^A, \mathbf{V}^A, \mathbf{D}^B, \mathbf{V}^B, \mathbf{C})\} \text{ s.t. } \mathbf{D}^A, \mathbf{V}^A, \mathbf{D}^B, \mathbf{V}^B, \mathbf{C} \geq 0 \\ & \text{where } F_1 = \frac{1}{2} \|\mathcal{T} - \widehat{\mathcal{T}}(\mathbf{D}^A \mathbf{V}^A, \mathbf{D}^B \mathbf{V}^B, \mathbf{C})\|_F^2 + \alpha \|\mathbf{V}^A\|_{1,1} + \alpha \|\mathbf{V}^B\|_{1,1} \end{aligned} \quad (5)$$

$\alpha > 0$  is a penalty term.

This approach is named sparse nonnegative CPD (SNNCPD). We resort to a stochastic gradient descent (SGD) algorithm called Nadam [42, 43] in order to solve Eq. (5). Details of the Nadam algorithm are given in Appendix B. In order to ensure the nonnegativity of matrix entries, all the element of matrices  $\mathbf{D}^A$ ,  $\mathbf{D}^B$ ,  $\mathbf{C}$ ,  $\mathbf{V}^A$ , and  $\mathbf{V}^B$  are projected on  $\mathbb{R}_+$  at each iteration [44]. We can thus differentiate the  $L_{1,1}$  norm, and the gradients with respect to the different variables are given by

$$\begin{aligned} \frac{\partial F_1}{\partial \mathbf{V}^A} &= -(\mathbf{D}^A)^\top (\mathbf{T}_1 - \mathbf{D}^A \mathbf{V}^A (\mathbf{L}_1)^\top) \mathbf{L}_1 + \alpha \mathbf{1}_{R,R} \\ \frac{\partial F_1}{\partial \mathbf{V}^B} &= -(\mathbf{D}^B)^\top (\mathbf{T}_2 - \mathbf{D}^B \mathbf{V}^B (\mathbf{L}_2)^\top) (\mathbf{L}_2) + \alpha \mathbf{1}_{R,R} \\ \frac{\partial F_1}{\partial \mathbf{D}^A} &= -(\mathbf{T}_1 - \mathbf{D}^A \mathbf{V}^A (\mathbf{L}_1)^\top) \mathbf{L}_1 (\mathbf{V}^A)^\top \\ \frac{\partial F_1}{\partial \mathbf{D}^B} &= -(\mathbf{T}_2 - \mathbf{D}^B \mathbf{V}^B (\mathbf{L}_2)^\top) (\mathbf{L}_2) (\mathbf{V}^B)^\top \\ \frac{\partial F_1}{\partial \mathbf{C}} &= -(\mathbf{T}_3 - \mathbf{C} (\mathbf{L}_3)^\top) (\mathbf{L}_3) \end{aligned} \quad (6)$$

Matrices  $\mathbf{T}_1$ ,  $\mathbf{T}_2$ ,  $\mathbf{T}_3$  are obtained by unfolding the tensor  $\mathcal{T}$  with respect to the first, second, and third modes, respectively.  $\mathbf{L}_1 = \mathbf{C} \odot (\mathbf{D}^B \mathbf{V}^B)$ ,  $\mathbf{L}_2 = \mathbf{C} \odot (\mathbf{D}^A \mathbf{V}^A)$  and  $\mathbf{L}_3 = (\mathbf{D}^B \mathbf{V}^B) \odot (\mathbf{D}^A \mathbf{V}^A)$ . The matrices  $\mathbf{V}^A$  and  $\mathbf{V}^B$  are initialized as the identity matrix. The matrices  $\mathbf{D}^A$  and  $\mathbf{D}^B$  and  $\mathbf{C}$  are initialized with nonnegative random values. The different steps of SNNCPD are summarized in Algorithm 1.

---

**Algorithm 1.** Sparse NonNegative Canonical Polyadic Decomposition (SNNCPD).  
Initialization step

---

• **Input:**  $\mathcal{T}$ ,  $R$  overestimated

**while** a convergence criterion is not reached **do**.

    Update the dictionaries  $\mathbf{D}^A$ ,  $\mathbf{D}^B$ , and  $\mathbf{C}$  using Nadam optimizer.

    Update the atoms  $\mathbf{V}^A$  and  $\mathbf{V}^B$  using Nadam optimizer.

**end while**

• **Output:**  $(\mathbf{D}_a, \mathbf{D}_b, \mathbf{C}, \mathbf{V}_a, \mathbf{V}_b)$

---

### 3.2 Online step: NN-CPD for the first tensor acquired at time $t_n$ with $n > 0$

In this step, we assume that the factors matrices  $\mathbf{A}_n$  and  $\mathbf{B}_n$  are well estimated.

#### 3.2.1 Algorithm 1

In this approach, we allow some modification of the factors estimated at time  $t_{n-1}$  thanks to linear combinations as explained below. We first replace all the null columns of  $\mathbf{A}_{n-1}$  and  $\mathbf{B}_{n-1}$  by columns of random numbers drawn from the standard normal distribution. We then look for  $\mathbf{A}_n$  and  $\mathbf{B}_n$  as

$$\mathbf{A}_n = \mathbf{U}^A \mathbf{A}_{n-1} \mathbf{V}^A \text{ and } \mathbf{B}_n = \mathbf{U}^B \mathbf{B}_{n-1} \mathbf{V}^B \quad (7)$$

The matrices  $\mathbf{U}^A$  and  $\mathbf{U}^B$  are square with sizes  $I$  and  $J$ , respectively. Here,  $\mathbf{U}^A \mathbf{A}_{n-1}$  and  $\mathbf{U}^B \mathbf{B}_{n-1}$  are our dictionaries, while  $\mathbf{V}^A$  and  $\mathbf{V}^B$  matrices are still sparse atoms. Therefore, the optimization problem becomes

$$\begin{aligned} & \min \{ F_2(\mathbf{U}^A, \mathbf{V}^A, \mathbf{U}^B, \mathbf{V}^B, \mathbf{C}) \} \text{ s.t. } \mathbf{U}^A, \mathbf{U}^B, \mathbf{C}, \mathbf{V}^B, \mathbf{V}^B \geq 0 \\ & \text{with } F_2 = \frac{1}{2} \|\mathcal{T} - \widehat{\mathcal{T}}(\mathbf{U}^A \mathbf{A}_{n-1} \mathbf{V}^B, \mathbf{U}^B \mathbf{B}_{n-1} \mathbf{V}^B, \mathbf{C})\|_F^2 \\ & + \alpha \|\mathbf{V}^A\|_1 + \alpha \|\mathbf{V}^B\|_1 \end{aligned} \quad (8)$$

We call this algorithm online sparse and nonnegative CPD (OSNCPD). The different steps are.

summarized in Algorithm 2. Most details can be found in [18, 27].

---

#### Algorithm 2. Online Sparse and NonNegative CPD

---

- **STEP 1: Initialization phase**

*Input:*  $\mathcal{T}_0$ ,  $R$  overestimated.  
*Solve Eq. (5) with  $\mathcal{T} = \mathcal{T}_0$ .*  
*Compute matrices  $\mathbf{A}_0$  and  $\mathbf{B}_0$  using Eq. (3).*  
*Output:*  $\mathbf{A}_0, \mathbf{B}_0$  and  $\mathbf{C}_0$

- **STEP 2: Online phase at time  $t_n$  with  $n > 0$**

*Input:*  $\mathcal{T}_n$ , and  $\mathbf{A}_{n-1}, \mathbf{B}_{n-1}$ .  
*Fill null columns of  $\mathbf{A}_{n-1}, \mathbf{B}_{n-1}$  by random numbers and Solve Eq. (8).*  
*Update matrices  $\mathbf{A}_n$  and  $\mathbf{B}_n$  using Eq. (7).*  
*Output:*  $(\mathbf{A}_n, \mathbf{B}_n, \mathbf{C}_n$  and  $\tilde{R}_n)$

- **STEP 3: Return to Step 2 with  $n = n + 1$**

---

The advantage of this approach is therefore its flexibility. The inconvenient is that the matrices to optimize are larger which make the algorithm more expensive in time.

We call this algorithm online sparse and nonnegative CPD (OSNCPD). The different stages are summarized in the Algorithm 2.

### 3.2.2 Algorithm 2

By taking the Eq. (7) as well as the cost function 8, the optimization of the algorithm becomes costly because of the dictionaries  $\mathbf{U}^A$  and  $\mathbf{U}^B$ . This is caused by the fact that no constraint is applied to these matrices to restrict the research space. In this approach, we will apply an orthogonality constraint on these dictionaries. The orthogonality constraint on the factors is also useful in certain applications [26, 45]. It is possible to apply the orthogonality constraint to the factors by using the Eq. 4. If, for example,  $\mathbf{A}$  must be orthogonal then:  $\mathbf{I}_A = \mathbf{A}^t \mathbf{A}$ . We therefore seek to minimize

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \left\| \mathcal{T} - \widehat{\mathcal{T}}(\mathbf{A}, \mathbf{B}, \mathbf{C}) \right\|_F^2 + \left\| \mathbf{A}^T \mathbf{A} - \mathbf{I}_A \right\|_F + \left\| \mathbf{B}^T \mathbf{B} - \mathbf{I}_B \right\|_F + \left\| \mathbf{C}^T \mathbf{C} - \mathbf{I}_C \right\|_F \quad (9)$$

To impose orthogonality constraint on the factors matrices for  $\mathcal{T} = \mathcal{T}_N$ , the cost function becomes

$$\begin{aligned} & \min \{ E_2(\mathbf{U}_n^A, \mathbf{V}_n^A, \mathbf{U}_n^B, \mathbf{V}_n^B, \mathbf{C}) \} \\ \text{où } E_2 &= \frac{1}{2} \left\| \mathcal{T} - \widehat{\mathcal{T}}(\mathbf{U}_n^A \mathbf{A}_{n-1} \mathbf{V}_n^B, \mathbf{U}_n^B \mathbf{B}_{n-1} \mathbf{V}_n^A, \mathbf{C}) \right\|_F^2 + \alpha \left\| \mathbf{V}_n^A \right\|_{1,1} + \alpha \left\| \mathbf{V}_n^B \right\|_{1,1} + \\ & \left\| \mathbf{I}_A - (\mathbf{U}_n^A)^T \mathbf{U}_n^A \right\|_F^2 + \left\| \mathbf{I}_B - (\mathbf{U}_n^B)^T \mathbf{U}_n^B \right\|_F^2 \end{aligned} \quad (10)$$

The gradient of the cost Eq. (10) is given by

$$\begin{aligned} \frac{\partial E_2}{\partial \mathbf{V}_n^A} &= -(\mathbf{U}_n^A \mathbf{A}_{n-1})^T \left( \mathbf{T}^A - \mathbf{U}_n^A \mathbf{A}_{n-1} \mathbf{V}_n^A (\mathbf{Z}^A)^T \right) \mathbf{Z}^A + \alpha \mathbf{1}_R \\ \frac{\partial E_2}{\partial \mathbf{V}_n^B} &= -(\mathbf{U}_n^B \mathbf{B}_{n-1})^T \left( \mathbf{T}^B - \mathbf{U}_n^B \mathbf{B}_{n-1} \mathbf{V}_n^B (\mathbf{Z}^B)^T \right) \mathbf{Z}^B + \alpha \mathbf{1}_R \\ \frac{\partial E_2}{\partial \mathbf{U}_n^A} &= - \left( \mathbf{T}^A - \mathbf{U}_n^A \mathbf{A}_{n-1} \mathbf{V}_n^B (\mathbf{Z}^A)^T \right) \mathbf{Z}^A (\mathbf{V}_n^A)^T (\mathbf{A}_{n-1})^T - \left( 2\mathbf{U}_n^A \left( \mathbf{I}_A^T - (\mathbf{U}_n^A)^T \mathbf{U}_n^A \right) \right) + \\ & \left( 2\mathbf{U}_n^A \left( \mathbf{I}_A \mathbf{U}_n^A - (\mathbf{U}_n^A)^T \mathbf{U}_n^A \right) \right) \\ \frac{\partial E_2}{\partial \mathbf{U}_n^B} &= - \left( \mathbf{T}^B - \mathbf{U}_n^B \mathbf{B}_{n-1} \mathbf{V}_n^A (\mathbf{Z}^B)^T \right) \mathbf{Z}^B (\mathbf{V}_n^B)^T (\mathbf{B}_{n-1})^T - \left( 2\mathbf{U}_n^B \left( \mathbf{I}_B^T - (\mathbf{U}_n^B)^T \mathbf{U}_n^B \right) \right) + \\ & \left( 2\mathbf{U}_n^B \left( \mathbf{I}_B \mathbf{U}_n^B - (\mathbf{U}_n^B)^T \mathbf{U}_n^B \right) \right) \\ \frac{\partial E_2}{\partial \mathbf{C}} &= - \left( \mathbf{T}^C - \mathbf{C} (\mathbf{Z}^C)^T \right) \mathbf{Z}^C \end{aligned} \quad (11)$$

The advantage of this approach is its speed of convergence because we reduce the research space for our dictionaries. For example, if  $\mathbf{A}_n$  is similar to  $\mathbf{A}_{n-1}$ , the matrix  $\mathbf{U}_n^A$  is a diagonal matrix. We call this algorithm orthogonality online sparse CPD (OOSCPD). The different steps are similar with the Algorithm 2.

## 4. Experiment for fluorescence real data acquisition

We plan an experiment to obtain real data acquisition. Four well-known fluorophores are injected on a reservoir at different time intervals under quasi-real conditions. At all, we get 50 matrices or EEMs of size  $[21, 36]$  online with a spectrofluorimeter *Hitachi F7000*. The corresponding fluorescence tensor was partitioned into four successive sub-tensors ( $\mathcal{T}_0 \dots \mathcal{T}_3$ ). Two kinds of partitions are considered. In the first partition (50% overlapping), each sub-tensor contains 20 EEMs, and two consecutive sub-tensors have ten EEMs in common. In the second partition (no overlapping, see **Figure 3**), sub-tensors have no EEMs in common. The initialization tensor  $\mathcal{T}_0$  contains 20 EEMs. The other sub-tensors contain ten EEMs. During the acquisition, an extra fluorophore appears in all the EEMs due to the experimental device. A preliminary study showed that this extra fluorophore can be represented by a single additional factor in the CPD model. Thus, the rank of the sub-tensors can be 3, 4, or 5. We recall that SNNCPD is used for sub-tensors  $\mathcal{T}_0$  in initialization phase.

### 4.1 Results and discussions

OSNCPD has already been compared with others online algorithms in [18, 27]. We compare the values of the physical ranks estimated by the two online approaches (OSNCPD, OOSCPD) with NN-CPD algorithm proposed in [31] and the actual values. Results are reported in **Tables 1** and **2** for the 50% overlapping case and no overlapping case, respectively. The penalty coefficient term ( $\alpha$ ) used in our two algorithms and in NN-CPD is also given. We can see that NN-CPD overestimated the rank in the last sub-tensors in both cases and this means that NN-CPD creates

Symbols	Definition
$\mathcal{T}, \mathcal{T}_n$	Tensor, sub-tensor at time $t_n$
$A, A^T, a, a$	Matrix, transposed matrix, column vector, scalar
$\mathbf{1}_R$	A matrix ( $R, R$ ) of 1
$I_A$	Identity matrix with size of matrix A
$\mathbb{R}$	Set of real numbers
$\ \cdot\ _F, \ \cdot\ _{1,1}$	Frobenius norm, $L_{1,1}$ norm
$A \geq 0$	Means that all the elements of matrix A are nonnegative
$\tilde{R}_n, R_n$	Physical rank of the sub-tensor $\mathcal{T}_n$ , CPD rank of the sub-tensor
$\odot$	Khatri-Rao product

**Table 1.**  
Main notations used in the paper.

Sub-tensor	0	1	2	3	$\alpha$
True rank	3	4	5	5	—
NN-CPD	3	4	7	10	0.5
OSNCPD & OOSCPD	3	4	5	5	1

**Table 2.**  
Rank estimation in the overlapping case.

duplicate factors. In contrast, our approaches (OSNCPD and OOSCPD) find the correct rank.

When the null columns of  $\hat{\mathbf{A}}$  and  $\hat{\mathbf{B}}$  and the corresponding columns of  $\hat{\mathbf{C}}$  are removed, we can compare the factor matrices estimated by our algorithms with the true factors by means of normalized root mean squared errors:

$$E_A = \frac{\|\mathbf{A} - \hat{\mathbf{A}}\|_F}{\|\mathbf{A}\|_F}, E_B = \frac{\|\mathbf{B} - \hat{\mathbf{B}}\|_F}{\|\mathbf{B}\|_F} \text{ and } E_C = \frac{\|\mathbf{C} - \hat{\mathbf{C}}\|_F}{\|\mathbf{C}\|_F} \quad (12)$$

Results are reported in **Tables 3** and **4** for the 50% overlapping case and no overlapping case, respectively **Table 5**.

In addition, to give some physical meaning to these error terms that may seem high, we have plotted in **Figure 4** the factors obtained from our approach in the sub-tensor  $\mathcal{T}_4$  along with the true factors. We observe a good agreement between the true and estimated factors.

In the overlapping case, both algorithms give similar results for sub-tensors, and this can be explained that the algorithms use past information. Also, these are close to those obtained from the initialization sub-tensor  $\mathcal{T}_0$  meaning that the online phase has been performed correctly.

Sub-tensor	0	1	2	3	$\alpha$
True rank	3	4	5	5	—
NN-CPD	3	4	5	10	0.1
OSNCPD & OOSCPD	3	4	5	5	0.5

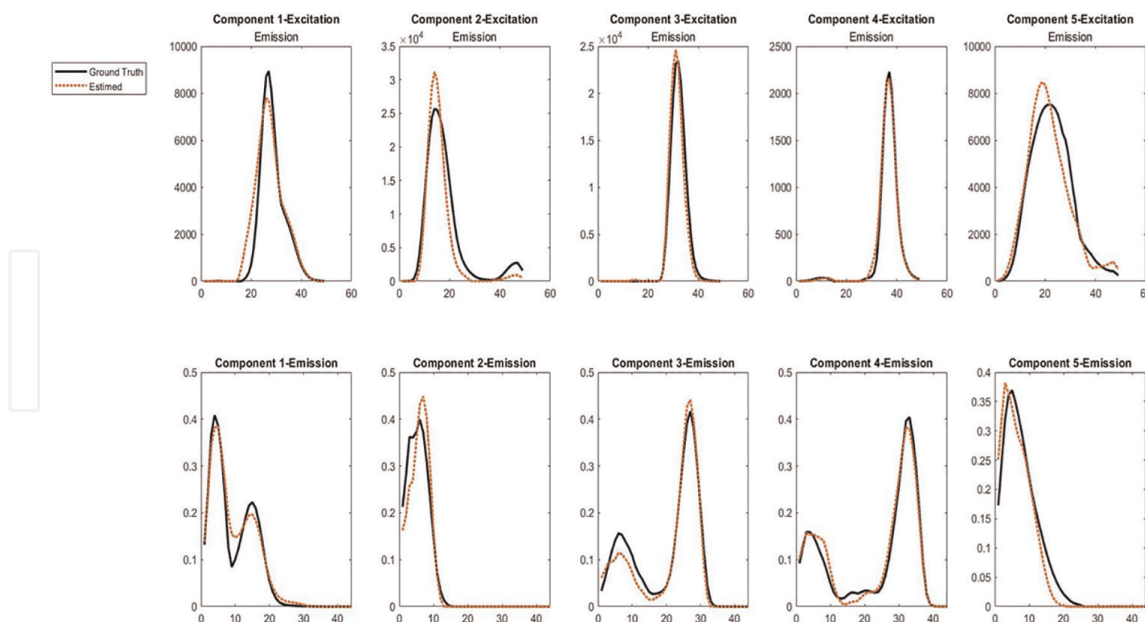
**Table 3.**  
Rank estimation in the no overlapping case.

Sub-tensor	Algorithm 1 (OSNCPD)				Algorithm 2 (OOSCPD)			
	0	1	2	3	0	1	2	3
Mean $E_A$	0.17	0.19	0.19	0.2	0.17	0.18	0.2	0.16
Mean $E_B$	0.14	0.15	0.15	0.16	0.14	0.16	0.14	0.14
Mean $E_C$	0.25	0.25	0.16	0.20	0.25	0.21	0.16	0.18

**Table 4.**  
Mean estimations errors in the overlapping case.

Sub-tensor	Algorithm 1 (OSNCPD)				Algorithm 2 (OOSCPD)			
	0	1	2	3	0	1	2	3
Mean $E_A$	0.17	0.19	0.20	0.20	0.17	0.18	0.21	0.21
Mean $E_B$	0.14	0.15	0.16	0.19	0.14	0.16	0.19	0.23
Mean $E_C$	0.25	0.32	0.31	0.32	0.25	0.31	0.32	0.25

**Table 5.**  
Mean estimations errors in the no overlapping case.



**Figure 4.** Emission and excitation spectra of the fluorophores for  $T_4$ . Top: Emission spectra, bottom: Excitation spectra. Red dots: estimated spectra, black lines: true spectra.

In the no overlapping case OSNCPD and OOSCPD give good results although no sub-tensors are share any information. The error in matrix C seems to be high but in the no overlapping case, because the algorithms do not have the a priori information.

## 5. Conclusions

We have introduced two algorithms (OSNCPD and OOSCPD) for the online nonnegative canonical polyadic decomposition of sub-tensors. We have also introduced an offline nonnegative algorithm with sparsity constraint. Our algorithms are based on dictionary learning and can deal with unknown rank variations. OOSCPD incorporates orthogonality constraint and can be seen as an extension of OSNCPD which incorporates nonnegativity and sparsity constraint for a better convergence.

These algorithms are presented in the particular case of the nonnegative online CPD of third-order fluorescence tensors, but they are not limited to this application field, and they can be easily extended to higher-order tensors. A real online fluorescence spectroscopy experiment was conducted in laboratory to validate our approach and compare with state-of-the-art approaches. The proposed algorithms allow to correctly follow the rank variations in most of the considered situations contrary to reference approaches. Eventually, these encouraging results allow to plan, for example, a monitoring in the natural environment in future work.

## Conflict of interest

“The authors declare no conflict of interest.”

## Abbreviations

ALS	alternating least square
ADMM	alternating direction method of multipliers
CPD	canonical polyadic decomposition
NNCPD	nonnegative canonical polyadic decomposition
ELS	enhanced line search
PARAFAC	parallel factor analysis
DOM	dissolve organisme matter
EEM	emission and excitation matrices
pH	potential Hydrogen
NMF	nonnegative matrix factorization
SGD	stochastique gradient descent
Nadam	nesterov accelerated adaptive moment estimation
F	fluorescence
TRP	tryptophane
5S8HQ	5-sulfate-8-hydroxyquinine
RHO	RHOdamine
SVD	singular value decomposition

## Appendix A

It is useful to mention that the CPD can incorporate different constraints depending on the field from which the data comes from. For example, it can be constraints of nonnegativity, orthogonality, or sparsity to name only the most commonly used.

*Nonnegativity constraint* Nonnegativity constraint is imposed in the CPD when the data to be processed is linked to physical quantities. In this sense, we can cite the domain of imagery [46] where nonnegativity is present because we know that optical signals are positive or zero. This is also the case of fluorescence spectroscopy where the data from the sensor is nonnegative [4, 30, 31, 47]. Nonnegativity constraint can be imposed on the factors during optimization by projecting the values of each factor matrix on  $\mathbb{R}^+$  [33]. This method of imposing nonnegativity sometimes shows its limits in certain cases because of the change of scale. In [34], the authors propose to remedy this problem by involving two factors,  $\epsilon$  and  $\theta \in \{-1; 1\}$  which multiply every factor matrix. Nonnegativity can also be imposed by choosing an optimization method [14, 17, 35] or even using an exponential variable change [48] or square [31].

*Sparsity constraint* Regarding the constraint of parsimony, it is generally used for parsimonious tensors or to ensure the overestimation of the rank, as mentioned above. This constraint is imposed according to the goal we are looking for, and in our case, it is the overestimation of the rank. The constraint can be applied to the matrices factors in the form of a regularization term:

$$\min_x \underbrace{f(x)}_{\text{function}} + \underbrace{r(x)}_{\text{regularization}} \quad (13)$$

$F(x)$  corresponds here to  $\left\| \mathcal{T} - \widehat{\mathcal{T}}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \right\|_F^2$  and is the attachment part.  $R(x)$  is the regularization part on the matrix  $\mathbf{a}$ ,  $\mathbf{b}$ , and  $\mathbf{c}$ .

In the literature, the standard  $l_{1,1}$  is the most used as in [17, 35, 41, 46, 49], and it can be defined according to the Eq. (A1) as following:

$$\min_{\mathbf{a}, \mathbf{b}, \mathbf{c}} \left\| \mathcal{T} - \widehat{\mathcal{T}}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \right\|_F^2 + \|\mathbf{a}\|_{1,1} + \|\mathbf{B}\|_{1,1} + \|\mathbf{C}\|_{1,1} \quad (14)$$

Standard  $l_{1,1}$  is conventionally used compared to standard  $l_{0,0}$  to promote parsimony because it is easier to handle than standard  $l_{0,0}$ . In order to minimize the rank of a matrix, we can speak of the nuclear standard which makes it possible to impose a small rank [50–52] and can be defined as a function of the Eq. (A1) as follows:

$$\min_{\mathbf{a}, \mathbf{b}, \mathbf{c}} \left\| \mathcal{T} - \widehat{\mathcal{T}}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \right\|_F^2 + \|\mathbf{a}\|_* + \|\mathbf{B}\|_* + \|\mathbf{C}\|_* \quad (15)$$

In [53], the authors show that the mixed standard provides a better convex envelope of the rank function than the nuclear standard. It therefore makes it possible to minimize the rank of a matrix. In the literature, the mixed standard is increasingly used as in [54–56]. It can be defined as a function of the Eq. (A1) as follows:

$$\min_{\mathbf{a}, \mathbf{b}, \mathbf{c}} \left\| \mathcal{T} - \widehat{\mathcal{T}}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \right\|_F^2 + \|\mathbf{a}\|_{2,1} + \|\mathbf{B}\|_{2,1} + \|\mathbf{C}\|_{2,1} + \|\mathbf{a}\|_{1,2} + \|\mathbf{b}\|_{1,2} + \|\mathbf{c}\|_{1,2} \quad (16)$$

## Appendix B: Stochastique gradient descent (SGD)

Take a differentiable function  $f(\mathbf{x})$ . For each iteration  $k$ , the update of  $\mathbf{x}$  in gradient descent is given by

$$\mathbf{x}(k+1) = \mathbf{x}(k) - \gamma \mathbf{f}'(\mathbf{x}(k)) \quad (17)$$

with  $\gamma > 0$ , the step-size or the learning rate.

### B.1 Adaptive moment estimation (ADAM)

Adaptive Moment estimation (ADAM) [43]. Adam optimizer is a variant of SGD. It acts on the gradient component using the exponential mobile average of the gradients and the step-size component by dividing the step-size by the square root of  $\mathbf{v}$ , the square of exponential average gradients.

Let take a function  $f(\mathbf{x})$ . For each iteration  $k$ , the update of  $\mathbf{x}$  in gradient descent is given by

$$\begin{aligned} \mathbf{g}_k &= \mathbf{f}'(\mathbf{x}(k)) \\ \mathbf{m}_k &= \beta_1 \mathbf{m}_{k-1} + (1 - \beta_1) \mathbf{g}_k \\ \mathbf{v}_k &= \beta_2 \mathbf{v}_{k-1} + (1 - \beta_2) \mathbf{g}_k^2 \\ \widehat{\mathbf{m}}_k &= \frac{\mathbf{m}_k}{1 - (\beta_1)^k} \\ \widehat{\mathbf{v}}_k &= \frac{\mathbf{v}_k}{1 - (\beta_2)^k} \\ \mathbf{x}(k+1) &= \mathbf{x}(k) - \frac{\gamma}{\sqrt{\widehat{\mathbf{v}}_k} + \epsilon} \widehat{\mathbf{m}}_k \end{aligned} \quad (18)$$

with  $\gamma > 0$  the step-size,  $\beta_1 \in [0, 1]$ ,  $\beta_2 \in [0, 1]$  and  $\mathbf{m}$ ,  $\mathbf{v}$  is initialized to 0.



## B.2 Nesterov accelerated adaptive moment estimation (Nadam)

Nadam optimizer is an extension of the Adam optimization algorithm. The algorithm was described in the 2016 article by Timothy Dozat [42]. For each iteration  $k$  of a function  $f(\mathbf{x})$ , the update of  $\mathbf{x}$  in gradient descent is given by:

$$\begin{aligned}\mathbf{g}_k &= \mathbf{f}'(\mathbf{x}(k-1)) \\ \mathbf{m}_k &= \beta_1 \mathbf{m}_{k-1} + (1 - \beta_1) \mathbf{g}_k \\ \mathbf{n}_k &= \beta_2 \mathbf{n}_{k-1} + (1 - \beta_2) \mathbf{g}_k^2 \\ \hat{\mathbf{m}}_k &= \frac{\beta_1 \mathbf{m}_k}{1 - (\beta_1)^k} + \frac{(1 - \beta_1) \mathbf{g}_k}{1 - (\beta_1)^k} \\ \hat{\mathbf{n}}_k &= \frac{\beta_2 \mathbf{n}_k}{1 - \beta_2} \\ \mathbf{x}(k+1) &= \mathbf{x}(k) - \frac{\gamma \hat{\mathbf{m}}_k}{\sqrt{\hat{\mathbf{n}}_k + \epsilon}}\end{aligned}\tag{19}$$

### Author details

Isaac Wilfried Sanou<sup>1\*†</sup>, Xavier Luciani<sup>2†</sup>, Roland Redon<sup>3†</sup> and Stéphane Mounier<sup>3†</sup>

1 Université de Bourgogne France-comté, Bourgogne, France


2 CNRS, LIS, Université de Toulon, Aix-Marseille Université, Toulon, France

3 CNRS/INSU, Université de Toulon, Aix-Marseille Université, Toulon, France

\*Address all correspondence to: [isaac-wilfried.sanou@u-bourgogne.fr](mailto:isaac-wilfried.sanou@u-bourgogne.fr)

† These authors contributed equally.

### IntechOpen

© 2023 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

## References

- [1] Harshman RA. Foundation of PARAFAC procedure: Models and conditions for an ‘explanatory’ multi-mode factor analysis. In: *UCLA Working Papers in Phonetics*. 1970. pp. 1-84
- [2] Wang H, Ahuja N. Compact representation of multidimensional data using tensor rank-one decomposition. *Vectors*. 2004;**1**:5
- [3] Field AS, Graupe D. Topographic component (parallel factor) analysis of multichannel evoked potentials: Practical issues in trilinear spatiotemporal decomposition. *Brain Topography*. 1991;**3**:407-423
- [4] Bro R. PARAFAC. Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems*. 1997;**38**: 149-171
- [5] Nion D, De Lathauwer L. An enhanced line search scheme for complex-valued tensor decompositions. application in DS-CDMA. *Signal Processing*. 2008;**88**:749-755
- [6] Vasilescu MAO, Terzopoulos D. Multilinear analysis of image ensembles: Tensorfaces. In: *European Conference on Computer Vision*. Springer; 2002. pp. 447-460
- [7] Kopriva I, Cichocki A. Blind multispectral image decomposition by 3d nonnegative tensor factorization. *Optics Letters*. 2009;**34**:2210-2212
- [8] Sanchez E, Kowalski BR. Tensorial resolution: A direct trilinear decomposition. *Journal of Chemometrics*. 1990;**4**:29-45
- [9] Roemer F, Haardt M. A semi-algebraic framework for approximate CP decompositions via simultaneous matrix diagonalizations (SECSI). *Signal Processing*. 2013;**93**:2722-2738
- [10] Luciani X, Albera L. Canonical polyadic decomposition based on joint eigenvalue decomposition. *Chemometrics and Intelligent Laboratory Systems*. 2014;**132**:152-167
- [11] Cichocki A, Zdunek R, Phan AH, Amari SI. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. John Wiley & Sons; 2009
- [12] Fu X, Huang K, Sidiropoulos ND, Ma WK. Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications. *IEEE Signal Processing Magazine*. 2019;**36**:59-80
- [13] Phan AH, Cichocki A. Multi-way nonnegative tensor factorization using fast hierarchical alternating least squares algorithm (HALS). In: *Proc. of The 2008 International Symposium on Nonlinear Theory and its Applications*. 2008
- [14] Liavas AP, Sidiropoulos ND. Parallel algorithms for constrained tensor factorization via alternating direction method of multipliers. *IEEE Transactions on Signal Processing*. 2015;**63**:5450-5463
- [15] Kolda TG, Bader BW. Tensor decompositions and applications. *SIAM Review*. 2009;**51**:455-500
- [16] Tomasi G, Bro R. A comparison of algorithms for fitting the PARAFAC model. *Computational Statistics & Data Analysis*. 2006;**50**:1700-1734
- [17] Xu Y, Yin W. A block coordinate descent method for regularized

- multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences*. 2013;**6**: 1758-1789
- [18] Sanou IW, Redon R, Luciani X, Mounier S. Online nonnegative canonical polyadic decomposition: Algorithms and application. In: 2021 29th European Signal Processing Conference (EUSIPCO). Vol. 2021. IEEE. pp. 1805-1809
- [19] André R, Luciani X, Albera L, Moreau E. A two-step algorithm for joint eigenvalue decomposition-application to canonical polyadic decomposition of fluorescence spectra. *Chemometrics and Intelligent Laboratory Systems*. 2020; **206**:104065
- [20] Bro R, Kiers HA. A new efficient method for determining the number of components in PARAFAC models. *Journal of Chemometrics*. 2003;**17**: 274-286
- [21] Stedmon CA, Bro R. Characterizing dissolved organic matter fluorescence with parallel factor analysis: A tutorial. *Limnology and Oceanography: Methods*. 2008;**6**:572-579
- [22] Papalexakis EE. Automatic unsupervised tensor mining with quality assessment. In: Proceedings of the 2016 SIAM International Conference on Data Mining, SIAM. 2016. pp. 711-719
- [23] Nion D, Sidiropoulos ND. Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor. *IEEE Transactions on Signal Processing*. 2009;**57**:2299-2310
- [24] Phan AH, Cichocki A. PARAFAC algorithms for large-scale problems. *Neurocomputing*. 2011;**74**: 1970-1984
- [25] Zhou S, Vinh NX, Bailey J, Jia Y, Davidson I. Accelerating online cp decompositions for higher order tensors. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM. 2016. pp. 1375-1384
- [26] Traoré A, Berar M, Rakotomamonjy A. Online multimodal dictionary learning. *Neurocomputing*. 2019;**368**: 163-179
- [27] Sanou IW, Redon R, Luciani X, Mounier S. Online nonnegative and sparse canonical polyadic decomposition of fluorescence tensors. *Chemometrics and Intelligent Laboratory Systems*. 2022;**225**:104550
- [28] Tibshirani R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*. 1996;**58**:267-288
- [29] Mounier S, Redon R. The use of 3-d fluorescence and its decomposition in environmental organic matter studies. In: *Encyclopedia of Analytical Chemistry: Applications, Theory and Instrumentation*. 2006. pp. 1-16
- [30] Murphy KR, Stedmon CA, Graeber D, Bro R. Fluorescence spectroscopy and multi-way techniques. *PARAFAC Analytical Methods*. 2013;**5**:6557-6566
- [31] Royer JP, Thirion-Moreau N, Comon P, Redon R, Mounier S. A regularized nonnegative canonical polyadic decomposition algorithm with preprocessing for 3D fluorescence spectroscopy. *Journal of Chemometrics*. 2015;**29**:253-265
- [32] Lakowicz J.. Principles of Fluorescence Spectroscopy. 2006. Vol. 1. ISBN: 978-0-387-31278-1. doi:10.1007/978-0-387-46312-4

- [33] Huang K, Sidiropoulos ND, Liavas AP. A flexible and efficient algorithmic framework for constrained matrix and tensor factorization. *IEEE Transactions on Signal Processing*. 2016;**64**:5052-5065
- [34] Jouni M, Mura MD, Comon P. Some issues in computing the cp decomposition of nonnegative tensors. In: *International Conference on Latent Variable Analysis and Signal Separation*. Springer; 2018. pp. 57-66
- [35] Vu X, Chaux C, Thirion-Moreau N, Maire S, Carstea EM. A new penalized nonnegative third-order tensor decomposition using a block coordinate proximal gradient approach: Application to 3d fluorescence spectroscopy. *Journal of Chemometrics*. 2017;**31**:2859
- [36] Royer JP, Thirion-Moreau N, Comon P. Computing the polyadic decomposition of nonnegative third order tensors. *Signal Processing*. 2011;**91**: 2159-2171
- [37] Vu, X.T., Chaux, C., Maire, S., Thirion-Moreau, N., 2014. 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: *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, IEEE. pp. 1–6.
- [38] Kruskal JB. Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*. 1977;**18**:95-138
- [39] Pasricha R, Gujral E, Papalexakis EE. Identifying and alleviating concept drift in streaming tensor decomposition. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer; 2018. pp. 327-343
- [40] Cohen JE, Gillis N. Dictionary-based tensor canonical polyadic decomposition. *IEEE Transactions on Signal Processing*. 2017;**66**:1876-1889
- [41] Lee H, Battle A, Raina R, Ng AY. Efficient sparse coding algorithms. In: *Advances in Neural Information Processing Systems*. Citeseer; 2007. pp. 801-808
- [42] Dozat T. Incorporating Nesterov momentum into Adam. In: *Proceedings of 4th International Conference on Learning Representations, Workshop, Track*. 2016
- [43] Kingma DP, Ba J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. 2014
- [44] Zdunek R, Cichocki A. Fast nonnegative matrix factorization algorithms using projected gradient approaches for large-scale problems. *Computational Intelligence and Neuroscience*. 2008, 2008
- [45] Olliver G, Absil PA, De Lathauwer L. Variable projection applied to block term decomposition of higher-order tensors. In: *International Conference on Latent Variable Analysis and Signal Separation*. Springer; 2018. pp. 139-148
- [46] Zhang Q, Wang H, Plemmons RJ, Pauca VP. Tensor methods for hyperspectral data analysis: A space object material identification study. *Journal of the Optical Society of America A*. 2008;**25**:3001-3012
- [47] Coble PG, Lead J, Baker A, Reynolds DM, Spencer RG. *Aquatic Organic Matter Fluorescence*. Cambridge University Press; 2014

- [48] Royer JP. Identification aveugle de mélanges et décomposition canonique de tenseurs: Application à l'analyse de l'eau. [theses]. Université Nice Sophia Antipolis. 2013
- [49] Cohen JE, Gillis N. Dictionary-based tensor canonical polyadic decomposition. *IEEE Transactions on Signal Processing*. 2018;**66**:1876-1889. DOI: 10.1109/tsp.2017.2777393
- [50] Gu S, Zhang L, Zuo W, Feng X. Weighted nuclear norm minimization with application to image denoising. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014. pp. 2862-2869
- [51] Recht B, Fazel M, Parrilo PA. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*. 2010;**52**: 471-501
- [52] Yuan M, Zhang CH. On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics*. 2016;**16**: 1031-1068
- [53] Han X, Albera L, Kachenoura A, Senhadji L, Shu H. Low rank canonical polyadic decomposition of tensors based on group sparsity. In: *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE; 2017. pp. 668-672
- [54] Chen Y, He W, Yokoya N, Huang TZ. Hyperspectral image restoration using weighted group sparsity-regularized low-rank tensor decomposition. *IEEE Transactions on Cybernetics*. 2019;**50**:3556-3570
- [55] Gramfort A, Kowalski M, Hämmäläinen M. Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods. *Physics in Medicine & Biology*. 2012;**57**:1937
- [56] de Morais Goulart JH, de Oliveira PMR, Farias RC, Zarzoso V, Comon P. Alternating group lasso for block-term tensor decomposition and application to ECG source separation. *IEEE Transactions on Signal Processing*. 2020; **68**:2682-2696