Coupled Nonnegative CANDECOMP/PARAFAC Decomposition for Multi-Block Tensor Analysis

Xiulin Wang[1,2,3,4], Jing Liu[1,2,✉], and Fengyu Cong[3,4,5]

[1] The First Affiliated Hospital of Dalian Medical University, Dalian 116011, China
[2] Dalian Innovation Institute of Stem Cell and Precision Medicine, Dalian, China
xiulin.wang@foxmail.com;liujing@dmu.edu.cn
[3] School of Biomedical Engineering, Faculty of Medicine, Dalian University of Technology, Dalian 116024, China
[4] Faculty of Information Technology, University of Jyväskylä, Jyväskylaä, Finland
[5] Key Laboratory of Social Computing and Cognitive Intelligence (Dalian University of Technology), Ministry of Education, China
cong@dlut.edu.cn

Abstract. Nonnegative tensor decomposition imposes nonnegative constraints on its latent factors, providing a part-based tensor representation that can extract meaningful and convincing information. This approach has been used widely across applications like signal processing, neuroscience, and other areas. For multi-block tensor group analysis, including multiple-subject or multiple-modal medical data, traditional single tensor decomposition fails to maintain feature comparability or explore the coupled information across tensors. This study introduces a novel coupled CANDECOMP/PARAFAC tensor decomposition method using the non-negativity constraints and the alternating proximal gradient strategy, termed CoNCPD-APG. The proposed algorithm enables the group analysis of two or more tensors that are fully- or partially-coupled, allowing for the simultaneous acquisition of shared, individual information, and core tensors. Experiment results of synthetic and real event-related potential data confirm the effectiveness of the proposed coupled tensor decomposition algorithm in discovering meaningful latent patterns and relationships from/among complex multi-block tensors.

Keywords: Alternating proximal gradient · Coupled · Nonnegative CANDECOMP/PARAFAC decomposition · Simultaneous decomposition · Tensor decomposition.

1 Introduction

Tensor decomposition, such as CANDECOMP/PARAFAC or Canonical Polyadic (CP, [1, 2]) model, can decompose the high-order tensor into the product of a core tensor along with small-size factor matrices, thus have found successful applications across multiple fields [3–5]. Nonnegative CP decomposition (NCPD) imposes nonnegative constraints on the latent components, providing a part-based representation of the tensor [6]. For example, the spatial, temporal, and

subject features from the multi-channel EEG data can be simultaneously acquired through CP decomposition [7]. When time-frequency analysis is included, non-negativity naturally becomes a feature of the EEG data, and it needs to be addressed by the NCPD [7]. Even though traditional tensor decomposition has achieved good performance, it faces challenges in the group analysis of multi-block tensors. Specifically, maintaining feature comparability between tensors and exploiting coupling information between tensors have proven to be difficult for traditional methods [8, 9].

Tensor decomposition can be extended to joint decomposition across multi-block tensors, i.e., coupled tensor decomposition, which provides a solution to analyze heterogeneous tensors that share common information. [8–10]. This method can jointly analyze tensors derived from different scenarios, possibly uncovering underlying data structures and interrelationships [9, 11], while making full use of prior knowledge to enhance both the accuracy and robustness of the solution [10]. Coupled tensor decomposition can achieve more unique solutions than its matrix-based counterparts [12, 13], and has its advantages in imposing constraints on hidden components [14]. Moreover, for the tensor data, performing the analysis via two-way matrix approaches inevitably leads to the loss of potential interactions embedded within the multi-dimensional structure of tensors [7]. When datasets are gathered under identical scenarios, it is reasonable to assume shared or high-correlated parts between them, which is a crucial condition for using coupled tensor decomposition.

For example, algebraic (double) coupled CP decomposition (C-CPD/DC-CPD, [10, 15, 16]) algorithms are coupled on one or two of the modes and have been successfully applied to applications such as multidimensional harmonic retrieval and joint blind source separation. However, algebraic algorithms require that some of the factor matrices must be full rank, and are more suitable for noise-free or high signal-to-noise ratio environments. The stability of coupled tensor decomposition depends largely on the choice of the number of components. The robust coupled tensor factorization (RCTF, [17]) algorithm can dynamically adjust the number, but it is only suitable for the group analysis of two datasets. In certain cases, methods like coupled matrix and tensor factorizations (CMTF, [18, 19]) and their derivatives [20, 21] have demonstrated superiority when comparing with ICA-based matrix methods in fusing multi-model datasets, assuming that subject dimensions between the data are coupled. The linked tensor decomposition, which is based on CP and Tucker models, has also performed well in fields such as image and signal processing [8, 22–24]. However, the higher the noise, the larger the number of coupled components, so how to choose the number is also a difficult problem. The common and individual feature extraction (CIFE, [11]) framework used the common orthogonal basis extraction method to identify shared features across all datasets, even when the number of components shared by tensors is unknown in advance. In addition, coupled tensor decomposition also shows greater advantages than tensor decomposition in practical applications such as hyperspectral and multispectral image fusion [25, 26].

In total, existing coupled tensor decomposition methods face the challenges of slow convergence speed and low decomposition accuracy. To address these problems, we propose a novel CP-based coupled tensor decomposition algorithm, incorporating nonnegative constraints and utilizing the alternating proximal gradient (APG, [27, 28]) method, aiming to achieve an efficient joint analysis of various tensors. This algorithm allows for the simultaneous decomposition of core tensors, the shared and unique information, ensuring high accuracy and efficiency throughout the decomposition.

The rest of this manuscript is listed below. Fundamental notations and tensor operations are presented in the Section 2, along with the coupled NCPD model and its solution. In Section 3, we demonstrate the algorithm's performance by conducting experiments on both computer-designed and real datasets. The last part summarizes this study.

2 Coupled Nonnegative CANDECOMP/PARAFAC Decomposition

2.1 Notation definition and tensor operations

In this study, we use lowercase to represent scalars, boldface lowercase to represent vectors, boldface uppercase to represent matrices, calligraphic boldface uppercase letters to represent tensors, respectively, e.g. y, $\boldsymbol{y}$, $\boldsymbol{Y}$ and $\boldsymbol{\mathcal{Y}}$. The operations $\boldsymbol{Y}^T$ and $\|\boldsymbol{Y}\|_F$ mean the transpose and Frobenius norm of $\boldsymbol{Y}$.'$\circ$', '$\circledast$', and '$\odot$' denote the products of outer, Hadamard, and Khatri-Rao, while the symbol '$\oslash$' denotes the element-wise division. The expressions $\{\boldsymbol{F}\}^{\odot}$ and $\{\boldsymbol{F}\}^{\odot_{-n}}$ are the abbreviated forms of $\boldsymbol{F}^{(N)} \odot \boldsymbol{F}^{(N-1)} \odot \cdots \boldsymbol{F}^{(2)} \odot \boldsymbol{F}^{(1)}$ and $\boldsymbol{F}^{(N)} \odot \boldsymbol{F}^{(N-1)} \cdots \odot \boldsymbol{F}^{(n+1)} \odot \boldsymbol{F}^{(n-1)} \cdots \boldsymbol{F}^{(2)} \odot \boldsymbol{F}^{(1)}$, and the format is also applies to $\boldsymbol{F}^{\circledast}$ and $\boldsymbol{F}^{\circledast_{-n}}$. The matrix $\boldsymbol{\mathcal{Y}}_{(n)}$ with the two-way size of $I_n \times (I_1 I_2 \cdots I_{n-1} I_{n+1} \cdots I_{N-1} I_N)$ denotes the nth-mode unfolding of $\boldsymbol{\mathcal{Y}} \in \Re_+^{I_1 \times I_2 \times \cdots \times I_N}$. More details about standard notations and tensor mathematical operations can be found in [29].

2.2 Model

For a collection of Nth-order non-negative tensors $\boldsymbol{\mathcal{Y}}^{(z)} \in \mathbb{R}_+^{I_1 \times I_2 \times \cdots \times I_N}$, $z = 1, 2, \cdots, Z$, the expression for the coupled NCPD model is defined as follows,

$$\begin{aligned} \boldsymbol{\mathcal{Y}}^{(z)} \approx \hat{\boldsymbol{\mathcal{Y}}}^{(z)} &= \sum_{r=1}^{R} d_r^{(z)} \boldsymbol{f}_r^{(1,z)} \circ \boldsymbol{f}_r^{(2,z)} \circ \cdots \circ \boldsymbol{f}_r^{(N,z)} \\ &= \left[\!\!\left[\boldsymbol{\mathcal{D}}^{(z)}; \boldsymbol{F}^{(1,z)}, \boldsymbol{F}^{(2,z)}, \cdots, \boldsymbol{F}^{(N,z)} \right]\!\!\right] \\ &= \boldsymbol{\mathcal{D}}^{(z)} \times_1 \boldsymbol{F}^{(1,z)} \times_2 \boldsymbol{F}^{(2,z)} \times \cdots \times_N \boldsymbol{F}^{(N,z)}, \end{aligned} \tag{1}$$

where $\hat{\boldsymbol{\mathcal{Y}}}^{(z)} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$ represents the estimated item for $\boldsymbol{\mathcal{Y}}^{(z)}$. $\boldsymbol{f}_r^{(n,z)}$ represents rth-column of nth-factor matrix $\boldsymbol{F}^{(n,z)} = \left[\boldsymbol{f}_1^{(n,z)}, \boldsymbol{f}_2^{(n,z)}, \cdots, \boldsymbol{f}_R^{(n,z)}\right] \in$

$\mathbb{R}_{+}^{I_n \times R}$ of zth-tensor. $\boldsymbol{\mathcal{D}}^{(z)}$ denotes the zth core tensor with the size of $R \times R \cdots \times R$, where only the super-diagonal locations have the non-zero values as $d_r^{(z)}$. In this model, it is assumed that every factor matrix $\boldsymbol{F}^{(n,z)}$ takes the form $\boldsymbol{F}^{(n,z)} = \left[\boldsymbol{F}_C^{(n,z)}\ \boldsymbol{F}_I^{(n,z)}\right]$. Specifically, $\boldsymbol{F}_C^{(n,z)} \in \mathbb{R}_{+,0 \le L_n \le R}^{I_n \times L_n}$ denotes the shared information across all tensor blocks, thus $\boldsymbol{F}_C^{(n)} = \boldsymbol{F}_C^{(n,1)} = \cdots = \boldsymbol{F}_C^{(n,Z)}$, and $\boldsymbol{F}_I^{(n,z)} \in \mathbb{R}^{I_n \times (R - L_n)}$ represents the unique information corresponding to specific tensor.

2.3 Algorithm

Based on the block coordinate descend (BCD) framework [27, 28, 30], the APG method was proposed to solve the nonnegative matrix factorization (NMF) and NCPD problems, and the performance of this method has been proven to be superior to its many competitors [28]. Therefore, it is very promising to extend the APG from NCPD to coupled NCPD problem.

In this part, a solution based on APG updates is provided for coupled tensor decomposition. To measure the gap between the original and recovered tensors, Euclidean divergence minimization is applied as the optimization criterion. Thus, the cost function for the model is represented as,

$$\min_{\boldsymbol{\mathcal{D}}^{(z)}, \boldsymbol{F}^{(n,z)}} \xi \frac{1}{2} \sum_{z=1}^{Z} \left\| \boldsymbol{\mathcal{Y}}^{(z)} - \left[\!\left[\boldsymbol{\mathcal{D}}^{(z)}; \boldsymbol{F}^{(1,z)}, \cdots, \boldsymbol{F}^{(N,z)} \right]\!\right] \right\|_F^2 . \tag{2}$$

$$\text{s.t. } \boldsymbol{\mathcal{D}}^{(z)} \in \mathbb{R}_{+}^{I_1 \times \cdots \times I_N},\ \boldsymbol{U}^{(n,z)} \in \mathbb{R}_{+}^{I_n \times R}$$

$$\left\| \boldsymbol{f}_r^{(n,z)} \right\| = 1, n = 1 \cdots N, r = 1 \cdots R, z = 1 \cdots Z.$$

First, for the core tensor $\boldsymbol{\mathcal{D}}^{(z)} \in \mathbb{R}_{+}^{I_1 \times \cdots \times I_N}$, inspired by the APG algorithm, we can calculate it according to the following minimization problem,

$$\boldsymbol{\mathcal{D}}^{(z)} = \underset{\boldsymbol{\mathcal{D}}^{(z)} \ge 0}{\operatorname{argmin}} \left[\left\langle \hat{\boldsymbol{\mathcal{G}}}^{(z)}, \boldsymbol{\mathcal{D}}^{(z)} - \hat{\boldsymbol{\mathcal{D}}}^{(z)} \right\rangle + \frac{L_d^{(z)}}{2} \left\| \boldsymbol{\mathcal{D}}^{(z)} - \hat{\boldsymbol{\mathcal{D}}}^{(z)} \right\|_F^2 \right], \tag{3}$$

whose closed form can be simply rewritten as,

$$\boldsymbol{\mathcal{D}}^{(z)} = \max\left(0,\ \hat{\boldsymbol{\mathcal{D}}}^{(z)} - \hat{\boldsymbol{\mathcal{G}}}^{(z)} / L_d^{(z)}\right). \tag{4}$$

Here, $\hat{\boldsymbol{\mathcal{D}}}^{(z)}$ represents an extrapolated point, while $L_d^{(z)}$ is a Lipschitz constant for the block-partial gradient at $\boldsymbol{\mathcal{D}}^{(z)}$ of equation (2). $\hat{\boldsymbol{\mathcal{G}}}^{(z)}$ denotes the block-partial gradient of equation (2) at the point $\hat{\boldsymbol{\mathcal{D}}}^{(z)}$.

Next, we derive the updatings for solving the factor matrix $\boldsymbol{F}^{(n,z)} \in \mathbb{R}_{+}^{I_n \times R}$ (excluding coupled information) as,

$$\boldsymbol{F}^{(n,z)} = \underset{\boldsymbol{F}^{(n,z)} \ge 0}{\operatorname{argmin}} \left[\left\langle \hat{\boldsymbol{G}}^{(n,z)}, \boldsymbol{F}^{(n,z)} - \hat{\boldsymbol{F}}^{(n,z)} \right\rangle + \frac{L_f^{(n,z)}}{2} \left\| \boldsymbol{F}^{(n,z)} - \hat{\boldsymbol{F}}^{(n,z)} \right\|_F^2 \right], \tag{5}$$

and the closed-form expression is represented as,

$$\boldsymbol{F}^{(n,z)} = \max\left(0,\ \hat{\boldsymbol{F}}^{(n,z)} - \hat{\boldsymbol{G}}^{(n,z)}/L_f^{(n,z)}\right), \tag{6}$$

where $\hat{\boldsymbol{F}}^{(n,z)}$ means the extrapolated point corresponding to $\boldsymbol{F}^{(n,z)}$. $L_f^{(n,z)}$ denotes a Lipschitz constant at $\boldsymbol{F}^{(n,z)}$ of the block-partial gradient of equation (2). $\hat{\boldsymbol{G}}^{(n,z)}$ means the block-partial gradient at the matrix point $\hat{\boldsymbol{F}}^{(n,z)}$ for equation (2) .

Third, for the partially-coupled factor matrix $\boldsymbol{F}^{(n,z)} \in \mathbb{R}_+^{I_n \times R}$, due to its two parts including $\boldsymbol{F}_C^{(n,z)}$ and $\boldsymbol{F}_I^{(n,z)}$, and we should calculate the solutions of $\boldsymbol{U}_C^{(n)}$ and $\boldsymbol{U}_I^{(n,z)}$ separately. It should be noted that calculating the solution of $\boldsymbol{F}_C^{(n)}$ requires merging all tensors together, whereas calculating the solution for $\boldsymbol{F}_I^{(n,z)}$ only considers the corresponding tensor for the mth set, and they can be calculated a follows:

$$\boldsymbol{F}_C^{(n)} = \underset{\boldsymbol{F}_C^{(n,z)} \geq 0}{\operatorname{argmin}} \sum_{z=1}^{Z} \left[\left\langle \hat{\boldsymbol{G}}_C^{(n,z)}, \boldsymbol{F}_C^{(n,z)} - \hat{\boldsymbol{F}}_C^{(n,z)} \right\rangle + \frac{L_f^{(n,z)}}{2} \left\| \boldsymbol{F}_C^{(n,z)} - \hat{\boldsymbol{F}}_C^{(n,z)} \right\|_F^2 \right] \tag{7}$$

and

$$\boldsymbol{F}_I^{(n,z)} = \underset{\boldsymbol{F}_I^{(n,z)} \geq 0}{\operatorname{argmin}} \left[\left\langle \hat{\boldsymbol{G}}_I^{(n,z)}, \boldsymbol{F}_I^{(n,z)} - \hat{\boldsymbol{F}}_I^{(n,z)} \right\rangle + \frac{L_f^{(n,z)}}{2} \left\| \boldsymbol{F}_I^{(n,z)} - \hat{\boldsymbol{F}}_I^{(n,z)} \right\|_F^2 \right], \tag{8}$$

whose closed forms can be represented as,

$$\boldsymbol{F}_C^{(n)} = \max\left(0,\ \hat{\boldsymbol{F}}_C^{(n)} - \frac{\sum_{z=1}^{Z} \hat{\boldsymbol{G}}_C^{(n,z)}}{\sum_{z=1}^{Z} L_f^{(n,z)}}\right) \tag{9}$$

and

$$\boldsymbol{F}_I^{(n,z)} = \max\left(0,\ \hat{\boldsymbol{F}}_I^{(n,z)} - \frac{\hat{\boldsymbol{G}}_I^{(n,z)}}{L_f^{(n,z)}}\right), \tag{10}$$

where $\hat{\boldsymbol{F}}_C^{(n,z)}$ and $\hat{\boldsymbol{F}}_I^{(n,z)}$ represent extrapolated points with respect to factor matrices $\boldsymbol{F}_C^{(n,z)}$ and $\boldsymbol{F}_I^{(n,z)}$. Similarly, $\hat{\boldsymbol{G}}_C^{(n,z)}$ and $\hat{\boldsymbol{G}}_I^{(n,z)}$ represent block-partial gradients of equation (2) at $\hat{\boldsymbol{F}}_C^{(n,z)}$ and $\hat{\boldsymbol{F}}_I^{(n,z)}$. Most importantly, the calculation of $\boldsymbol{F}_C^{(n)}$ depends on all tensors and is then assigned to each tensor.

During each iteration, the updates are performed sequentially in the following order $\boldsymbol{F}^{(1,1)}, \boldsymbol{F}^{(1,2)} \dots \boldsymbol{F}^{(1,Z)} \dots \boldsymbol{F}^{(N,1)}, \boldsymbol{F}^{(N,2)} \dots \boldsymbol{U}^{(N,Z)}$ and $\boldsymbol{\mathcal{D}}^{(1)}, \boldsymbol{\mathcal{D}}^{(2)} \dots \boldsymbol{\mathcal{D}}^{(Z)}$. Therefore, $\boldsymbol{\mathcal{D}}^{(z)}$, and $\boldsymbol{F}^{(n,z)}$ are alternately updated based on (4), (6), (9), and (10) until the algorithm converges. Moreover, the algorithm proposed here, which combines coupled NCPD model and APG update, is referred to as the CoNCPD-APG algorithm, and its detailed parameter settings are given below.

At the kth iteration, for the updates of $\boldsymbol{F}^{(n,z)}$ and $\boldsymbol{\mathcal{D}}^{(z)}$, we set Lipschitz constants $L_{f,k-1}^{(n,z)}$ and $L_{d,k-1}^{(z)}$ as outlined in [28] as follows,

$$L_{f,k-1}^{(n,z)} = \max\left(1, \left\| \boldsymbol{D}_{k-1}^{(z)} (\boldsymbol{F}_{k-1}^{(z)\odot_{-n}})^T \boldsymbol{F}_{k-1}^{(z)\odot_{-n}} (\boldsymbol{D}_{k-1}^{(z)})^T \right\|\right) \tag{11}$$

and

$$L_{d,k-1}^{(z)} = \max\left(1, \left\| (\boldsymbol{F}_{k-1}^{(z)\odot})^T \boldsymbol{F}_{k-1}^{(z)\odot} \right\|\right). \tag{12}$$

The extrapolation weights are taken as

$$\omega_{d,k-1}^{(z)} = \min\left(\hat{\omega}_{k-1},\ \delta_\omega \sqrt{\frac{L_{d,k-2}^{(z)}}{L_{d,k-1}^{(z)}}}\right) \tag{13}$$

and

$$\omega_{f,k-1}^{(n,z)} = \min\left(\hat{\omega}_{k-1},\ \delta_\omega \sqrt{\frac{L_{f,k-2}^{(n,z)}}{L_{f,k-1}^{(n,z)}}}\right), \tag{14}$$

where $\delta_\omega < 1$ is pre-selected (e.g., [28] set it to 0.9999), and $\hat{\omega}_{k-1} = \frac{\tau_{k-1}-1}{\tau_k}$ with $\tau_0 = 1$ and $\tau_k = \left(1 + \sqrt{1 + 4\tau_{k-1}^2}\right)/2$. Furthermore, the extrapolations at $\boldsymbol{F}_{k-1}^{(n,z)}$ and $\boldsymbol{\mathcal{D}}_{k-1}^{(z)}$ are defined as,

$$\hat{\boldsymbol{F}}_{k-1}^{(n,z)} = \boldsymbol{F}_{k-1}^{(n,z)} + \omega_{f,k-1}^{(n,z)} \left(\boldsymbol{F}_{k-1}^{(n,z)} - \boldsymbol{F}_{k-2}^{(n,z)}\right) \tag{15}$$

and

$$\hat{\boldsymbol{\mathcal{D}}}_{k-1}^{(z)} = \boldsymbol{\mathcal{D}}_{k-1}^{(z)} + \omega_{d,k-1}^{(z)} \left(\boldsymbol{\mathcal{D}}_{k-1}^{(z)} - \boldsymbol{\mathcal{D}}_{k-2}^{(z)}\right). \tag{16}$$

For equation (2), the block-partial gradient $\hat{\boldsymbol{\mathcal{G}}}_{k-1}^{(z)}$ at $\hat{\boldsymbol{\mathcal{D}}}_{k-1}^{(z)}$ is expressed as follows,

$$\begin{aligned} \hat{\boldsymbol{\mathcal{G}}}_{k-1}^{(z)} = {} & \text{dddiag}\left[\left(\boldsymbol{F}^{(z)^T}\boldsymbol{F}^{(z)}\right)^{\circledast} \text{ddiag}\left(\hat{\boldsymbol{\mathcal{D}}}_{k-1}^{(z)}\right)\right] \\ & - \text{dddiag}\left[\left(\boldsymbol{F}^{(z)\odot}\right)^T \text{vect}\left(\boldsymbol{\mathcal{Y}}^{(z)}\right)\right], \end{aligned} \tag{17}$$

where $\text{vect}(\boldsymbol{\mathcal{Y}}^{(z)})$ indicates vectorizing the tensor $\boldsymbol{\mathcal{Y}}^{(z)}$ into a vector. The operation $\text{ddiag}(\hat{\boldsymbol{\mathcal{D}}}_{k-1}^{(z)})$ represents vectorizing the super-diagonal non-zero entries of the tensor $\hat{\boldsymbol{\mathcal{D}}}^{(z)}$ to a vector, while the notation 'dddiag' is the reverse process of 'ddiag'.

Similarly, at $\hat{\boldsymbol{F}}_{k-1}^{(n,z)}$, the block-partial gradient of equation (2) is defined as,

$$\hat{\boldsymbol{G}}_{k-1}^{(n,z)} = \hat{\boldsymbol{F}}_{k-1}^{(n,z)} \boldsymbol{D}^{(z)} \left(\boldsymbol{F}^{(z)^T}\boldsymbol{F}^{(z)}\right)^{\circledast_{-n}} \boldsymbol{D}^{(z)^T} - \boldsymbol{\mathcal{Y}}_{(n)}^{(z)} \boldsymbol{F}^{(z)\odot_{-n}} \boldsymbol{D}^{(z)^T}, \tag{18}$$

where $\boldsymbol{D}^{(z)}$ represents a diagonal matrix in which elements come directly from the super-diagonal of $\boldsymbol{\mathcal{D}}^{(z)}$. $\hat{\boldsymbol{G}}_{k-1}^{(n,z)} = \left[\hat{\boldsymbol{G}}_{C,k-1}^{(n,z)}\ \hat{\boldsymbol{G}}_{I,k-1}^{(n,z)}\right]$, which means $\hat{\boldsymbol{G}}_{C,k-1}^{(n,z)} = \left[\hat{\boldsymbol{G}}_{k-1}^{(n,z)}\right]_{:,1:L_n}$ and $\hat{\boldsymbol{G}}_{I,k-1}^{(n,z)} = \left[\hat{\boldsymbol{G}}_{k-1}^{(n,z)}\right]_{:,L_{n+1}:R}$.

After substituting (11), (12), (15), (16), (17) and (18) into (4), (6), (9) and (10) we can obtain the solutions of $\boldsymbol{\mathcal{D}}_k^{(z)}$, $\boldsymbol{F}_k^{(n,z)}$, $\boldsymbol{F}_{C,k}^{(n,z)}$ and $\boldsymbol{F}_{I,k}^{(n,z)}$.

3 Simulation and results

In this part, we first compared and demonstrated the performances of CoNCPD-APG algorithm by analyzing synthetic data. The strategies compared included multiplicative updating (MU, [6]) proposed by Lee et al., the alternating direction method of multipliers (ADMM, [31,32]), and the fast hierarchical alternating least squares (fHALS, [23,33]) proposed by Cichocki et al.. Then we applied the coNCPD-APG algorithm to the real event-related potential (ERP) data. For all algorithms, the iteration stopping criteria were kept consistent as: the maximum number of iterations cannot exceed 1000 as $k \leq 1000$ and the change of average relative error (aRelErr) satisfies $|\text{aRelErr}_k - \text{aRelErr}_{k-1}| < 1e-8$. The parameter aRelErr is defined as,

$$\text{aRelErr}_k \triangleq \frac{1}{Z}\sum_{z=1}^{Z}\frac{\left\|\boldsymbol{\mathcal{Y}}^{(z)} - \left[\!\left[\boldsymbol{\mathcal{D}}_k^{(z)}; \boldsymbol{F}_k^{(1,z)}, \boldsymbol{F}_k^{(2,z)} \cdots, \boldsymbol{F}_k^{(N,z)}\right]\!\right]\right\|_F}{\|\boldsymbol{\mathcal{Y}}^{(z)}\|_F}, \tag{19}$$

where $\left[\!\left[\boldsymbol{\mathcal{D}}_k^{(z)}; \boldsymbol{F}_k^{(1,z)}, \cdots, \boldsymbol{F}_k^{(N,z)}\right]\!\right]$ denotes the estimation item of the tensor $\boldsymbol{\mathcal{Y}}^{(z)}$. Tensor fitting, referred to as Fit, is defined as,

$$\text{Fit}_k \triangleq \frac{1}{Z}\sum_{z=1}^{Z}\left[1 - \frac{\left\|\boldsymbol{\mathcal{Y}}^{(z)} - \left[\!\left[\boldsymbol{\mathcal{D}}_k^{(z)}; \boldsymbol{F}_k^{(1,z)}, \boldsymbol{F}_k^{(2,z)} \cdots, \boldsymbol{F}_k^{(N,z)}\right]\!\right]\right\|_F}{\|\boldsymbol{\mathcal{Y}}^{(z)}\|_F}\right], \tag{20}$$

and the Fit is applied to measure the similarity between the original and reconstructed matrices/tensors. The Performance Index (PI), which evaluates the accuracy between original and recovered matrices, is given as,

$$\text{PI} \triangleq \frac{1}{2R(R-1)}\left[\sum_{i=1}^{R}\left(\sum_{j=1}^{R}\frac{|w_{ij}|}{\max_k|w_{ik}|}\right) + \sum_{i=1}^{R}\left(\sum_{j=1}^{R}\frac{|w_{ji}|}{\max_k|w_{ki}|}\right)\right]. \tag{21}$$

Here, w_{ij} represents the element at position (i,j) in matrix $\boldsymbol{W} = \tilde{\boldsymbol{F}}^{\dagger}\boldsymbol{F}$, where $\tilde{\boldsymbol{F}}$ denotes the recovery of the matrix $\boldsymbol{F}$, $\dagger$ means the pseudo-inverse. The closer PI values are to 0 and Fit values to 1, the more accurate the estimations.

For initializing core tensors and factor matrices, we used a MATLAB function `rand` that follows a uniform distribution. We imposed the noise item to the tensors using signal-to-noise ratio (SNR), calculated via $\text{SNR} \triangleq 10\log_{10}(\phi_s/\phi_n)$, with ϕ_s and ϕ_n representing the signal and noise levels.

The experimental simulations were conducted using the following system specifications: System-64bit-Windows11; CPU-Intel-i9-14900KF @3.20GHz; Memory 128-GB; Software-MATLAB-R2022b.

3.1 Synthetic data

In this part, we generated 20 third-order tensors that were partially/fully coupled in one, two, or three modes according to (1) with a fixed SNR=10. The sizes of tensors were set as $I_{1,2,3} = [100, 100, 100]$, and the rank of tensors was set to 10. To ensure the reliability of the algorithms, we performed 100 independent runs and reported the average results, see Table 1. The four algorithms perform almost the same on the Fit index for various coupled types. From PI values and running time, we can conclude that the proposed coupled algorithm is very competitive in decomposition speed and the accuracy of the recovered factor matrix.

Table 1. Performance comparison of coupled nonnegative CANDECOMP/PARAFAC decomposition models that are coupled in one, two, and three modes optimized using different algorithms

	type	coupled in three modes		coupled in two modes		coupled in one mode	
	$L_{1,2,3}$	[10,10,10]	[5,5,5]	[10,10,0]	[5,5,0]	[10,0,0]	[5,0,0]
Fit	ADMM	0.9534	0.9528	0.9536	0.9536	0.9535	0.9536
	fHALS	0.9534	0.9531	0.9536	0.9536	0.9533	0.9535
	MU	0.9534	0.9484	0.9526	0.9532	0.9525	0.9527
	APG	0.9534	0.9524	0.9536	0.9536	0.9531	0.9535
PI	ADMM	0.0040	0.0185	0.0039	0.0043	0.0041	0.0042
	fHALS	0.0039	0.0092	0.0045	0.0054	0.0055	0.0059
	MU	0.0040	0.0593	0.0183	0.0093	0.0188	0.0147
	APG	0.0039	0.0199	0.0040	0.0043	0.0055	0.0042
Time	ADMM	15.993	19.387	9.1603	9.3587	9.9832	9.2803
	fHALS	6.8604	12.200	7.7722	3.4584	8.5071	6.7957
	MU	51.409	38.017	19.8343	22.979	20.789	20.298
	APG	8.5072	18.440	5.2630	5.6902	5.6701	5.9041

3.2 Event-related potential data

In this part, we demonstrate how the CoNCPD-APG algorithm performs in jointly analyzing two groups of event-related potential (ERP) data. One group consists of 21 children that have reading disability (RD), while the other group includes 21 children of attention deficit (AD). The aim of this study is to explore

ERP multi-domain features that more effectively differentiate between the two groups. For detailed information about the data, please refer to [34]. We use complex Morlet wavelet transform to create 42 third-order tensors, each sized 9 (space) $\times$ 71 (frequency) $\times$ 60 (time), corresponding to 42 subjects (21 RD & 21 AD). Following [7], we set 36 components for every tensor. Given the characteristics of the data, these tensors can be seen as coupled across the modes of space, frequency, and time, so the number of components shared across the tensors is also fixed as 36.

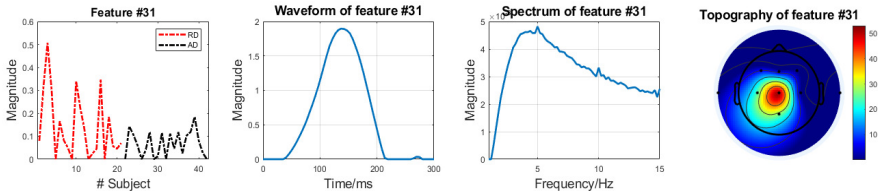

Fig. 1. Illustration of ERP multi-domain features representing subject, time, frequency, and space extracted by the CoNCPD-APG algorithm

Generally, the ERP data are obtained by repeatedly presenting stimuli, providing a rough understanding of their properties in domains of time, frequency, and space before finally extracting them. Therefore, based on the prior information from [7], the ERP multi-domain features and relevant components can be obtained, as shown in Fig. 1. The subject features underwent a t-test analysis, revealing the significant difference between the two groups of RD and AD ($t_{20} = 2.419$, $p = 0.025$.) We can notice that the latency and spectrum peaks are respectively around 150 ms and 5 Hz, which are both well consistent with the property of mismatch negativity component we desired. The topographical results suggest that differences between the two groups may be located in the central and left hemispheres. The mean tensor Fit and running Time of 100 Monte Carlo runs for CoNCPD-APG algorithm are 0.8491 and 56 seconds, respectively.

4 Conclusion

To sum up, this study proposed an efficient APG-based coupled NCPD algorithm, namely CoNCPD-APG algorithm, to solve the group decomposition problem of multiple tensor blocks. The algorithm can simultaneously extract shared information, individual information, and core tensors, where shared components represent coupling information among all tensors and the individual components represent the unique information of a specific tensor. Experiments conducted on synthetic and real-world ERP data certify the excellent performance of the CoNCPD-APG algorithm we proposed. Next, how to further accelerate the optimization convergence of the algorithm and fully prove the performance of the

algorithm will be the direction of our future work.

Acknowledgments. This study was supported by the Dalian Science and Technology Talent Innovation Support Project (2023RY034), the Dalian life and health guidance program project (2024033), and the Natural Science Foundation of Liaoning Province (2024-BS-181).
✉ Corresponding author: Jing Liu, liujing@dmu.edu.cn

References

1. Hitchcock, F.L.: The expression of a tensor or a polyadic as a sum of products. Journal of Mathematics and Physics **6**(1-4), 164–189 (1927)
2. Harshman, R.A., et al.: Foundations of the parafac procedure: Models and conditions for an "explanatory" multi-modal factor analysis. UCLA working papers in phonetics **16**(1), 84 (1970)
3. Cichocki, A., Zdunek, R., Phan, A.H., Amari, S.i.: Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation. John Wiley & Sons (2009)
4. Sidiropoulos, N.D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E.E., Faloutsos, C.: Tensor decomposition for signal processing and machine learning. IEEE Transactions on Signal Processing **65**(13), 3551–3582 (2017)
5. Liu, W., Wang, X., Ristaniemi, T., Cong, F.: Identifying task-based dynamic functional connectivity using tensor decomposition. In: International Conference on Neural Information Processing. pp. 361–369. Springer (2020)
6. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. nature **401**(6755), 788–791 (1999)
7. Cong, F., Lin, Q.H., Kuang, L.D., Gong, X.F., Astikainen, P., Ristaniemi, T.: Tensor decomposition of eeg signals: a brief review. Journal of neuroscience methods **248**, 59–69 (2015)
8. Yokota, T., Cichocki, A., Yamashita, Y.: Linked parafac/cp tensor decomposition and its fast implementation for multi-block tensor analysis. In: International Conference on Neural Information Processing. pp. 84–91. Springer (2012)
9. Zhou, G., Zhao, Q., Zhang, Y., Adalı, T., Xie, S., Cichocki, A.: Linked component analysis from matrices to high-order tensors: Applications to biomedical data. Proceedings of the IEEE **104**(2), 310–331 (2016)
10. Gong, X., Lin, Q., Cong, F., De Lathauwer, L.: Double coupled canonical polyadic decomposition for joint blind source separation. IEEE Transactions on Signal Processing **66**(13), 3475–3490 (2018)
11. Zhou, G., Cichocki, A., Zhang, Y., Mandic, D.P.: Group component analysis for multiblock data: Common and individual feature extraction. IEEE transactions on neural networks and learning systems **27**(11), 2426–2439 (2015)
12. Gong, X.F., Wang, X.L., Lin, Q.H.: Generalized non-orthogonal joint diagonalization with lu decomposition and successive rotations. IEEE Transactions on Signal Processing **63**(5), 1322–1334 (2015)
13. Chen, X., Wang, Z.J., McKeown, M.: Joint blind source separation for neurophysiological data analysis: Multiset and multimodal methods. IEEE Signal Processing Magazine **33**(3), 86–107 (2016)
14. Cichocki, A.: Tensor decompositions: a new concept in brain data analysis? arXiv preprint arXiv:1305.0395 (2013)

15. Sørensen, M., De Lathauwer, L.: Multidimensional harmonic retrieval via coupled canonical polyadic decomposition—part i: Model and identifiability. IEEE Transactions on Signal Processing **65**(2), 517–527 (2016)
16. Sørensen, M., De Lathauwer, L.: Multidimensional harmonic retrieval via coupled canonical polyadic decomposition—part ii: Algorithm and multirate sampling. IEEE Transactions on Signal Processing **65**(2), 528–539 (2016)
17. Genicot, M., Absil, P.A., Lambiotte, R., Sami, S.: Coupled tensor decomposition: a step towards robust components. In: 2016 24th European Signal Processing Conference (EUSIPCO). pp. 1308–1312. IEEE (2016)
18. Acar, E., Kolda, T.G., Dunlavy, D.M.: All-at-once optimization for coupled matrix and tensor factorizations. arXiv preprint arXiv:1105.3422 (2011)
19. Acar, E., Levin-Schwartz, Y., Calhoun, V.D., Adali, T.: Acmtf for fusion of multimodal neuroimaging data and identification of biomarkers. In: 2017 25th European Signal Processing Conference (EUSIPCO). pp. 643–647. IEEE (2017)
20. Jonmohamadi, Y., Muthukumaraswamy, S., Chen, J., Roberts, J., Crawford, R., Pandey, A.: Extraction of common task features in eeg-fmri data using coupled tensor-tensor decomposition. Brain Topography **33**, 636–650 (2020)
21. Schenker, C., Cohen, J.E., Acar, E.: A flexible optimization framework for regularized matrix-tensor factorizations with linear couplings. IEEE Journal of Selected Topics in Signal Processing **15**(3), 506–521 (2020)
22. Yokota, T., Cichocki, A.: Linked tucker2 decomposition for flexible multi-block data analysis. In: Neural Information Processing: 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part III 21. pp. 111–118. Springer (2014)
23. Zdunek, R., Fonał, K., Wołczowski, A.: Linked cp tensor decomposition algorithms for shared and individual feature extraction. Signal Processing: Image Communication **73**, 37–52 (2019)
24. Wang, X., Liu, W., Toiviainen, P., Ristaniemi, T., Cong, F.: Group analysis of ongoing eeg data based on fast double-coupled nonnegative tensor decomposition. Journal of neuroscience methods **330**, 108502 (2020)
25. Kanatsoulis, C.I., Fu, X., Sidiropoulos, N.D., Ma, W.K.: Hyperspectral superresolution: A coupled tensor factorization approach. IEEE Transactions on Signal Processing **66**(24), 6503–6517 (2018)
26. Xu, T., Huang, T.Z., Deng, L.J., Xiao, J.L., Broni-Bediako, C., Xia, J., Yokoya, N.: A coupled tensor double-factor method for hyperspectral and multispectral image fusion. IEEE Transactions on Geoscience and Remote Sensing (2024)
27. 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 **6**(3), 1758–1789 (2013)
28. Xu, Y.: Alternating proximal gradient method for sparse nonnegative tucker decomposition. Mathematical Programming Computation **7**(1), 39–70 (2015)
29. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM review **51**(3), 455–500 (2009)
30. Kim, J., He, Y., Park, H.: Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. Journal of Global Optimization **58**, 285–319 (2014)
31. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine learning **3**(1), 1–122 (2011)

32. Wang, X., Liu, W., Wang, X., Mu, Z., Xu, J., Chang, Y., Zhang, Q., Wu, J., Cong, F.: Shared and unshared feature extraction in major depression during music listening using constrained tensor factorization. Frontiers in Human Neuroscience **15**, 799288 (2021)
33. Wang, X., Ristaniemi, T., Cong, F.: Fast implementation of double-coupled non-negative canonical polyadic decomposition. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 8588–8592. IEEE (2019)
34. Cong, F., Phan, A.H., Zhao, Q., Huttunen-Scott, T., Kaartinen, J., Ristaniemi, T., Lyytinen, H., Cichocki, A.: Benefits of multi-domain feature of mismatch negativity extracted by non-negative tensor factorization from eeg collected by low-density array. International journal of neural systems **22**(06), 1250025 (2012)