FAST IMPLEMENTATION OF DOUBLE-COUPLED NONNEGATIVE CANONICAL POLYADIC DECOMPOSITION

Xiulin Wang[1,2] *Tapani Ristaniemi*[2] *Fengyu Cong*[1,2,*]

[1] School of Biomedical Engineering, Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian, China
[2] Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland

ABSTRACT

Real-world data exhibiting high order/dimensionality and various couplings are linked to each other since they share some common characteristics. Coupled tensor decomposition has become a popular technique for group analysis in recent years, especially for simultaneous analysis of multi-block tensor data with common information. To address the multi-block tensor data, we propose a fast double-coupled nonnegative Canonical Polyadic Decomposition (FDC-NCPD) algorithm in this study, based on the linked CP tensor decomposition (LCPTD) model and fast Hierarchical Alternating Least Squares (Fast-HALS) algorithm. The proposed FDC-NCPD algorithm enables simultaneous extraction of common components, individual components and core tensors from tensor blocks. Moreover, time consumption is greatly reduced without compromising the decomposition quality when handling large-scale tensor blocks. Simulation experiments of synthetic and real-world data are conducted to demonstrate the superior performance of the proposed algorithm.

Index Terms— Tensor decomposition, coupled tensor decomposition, Hierarchical Alternating Least Squares (HALS), linked CP tensor decomposition (LCPTD)

1. INTRODUCTION

Tensor decomposition has been successfully applied to an ensemble of disciplines including blind source separation, signal processing, classification, data mining and neuroscience [1, 2, 3, 4, 5]. For instance, in EEG data analysis, spatial, temporal and spectral information can be simultaneously considered via tensor decomposition, which in turn provides solutions with convincing physiological or pathological interpretations [5]. However, when it comes to joint analysis of multi-block tensor data, such as multiset or multimodal neurophysiological data fusion [6], conventional methods meet challenges in maintaining feature/component comparability and utilizing coupled information across tensors. Joint analysis of data from different samples can potentially reveal underlying structures and inner-relationships among data [7, 8]. Furthermore, joint analysis can take full advantage of prior information to improve the accuracy and stability of solutions [9]. Therefore, increasing recognition of joint analysis makes coupled tensor decomposition more extensively utilized. Coupled tensor decomposition can jointly analyze the multi-block tensors represented by various samples, meanwhile provide a simultaneous extraction of common components, individual components and core tensors [10]. To date, coupled tensor decomposition has been applied in the fields of neuroscience, multi-dimensional harmonic retrieval, array signal processing and metabolic physiology [8, 11, 12, 13].

Compared with tensor decomposition originally designed for single tensors [1], coupled tensor decomposition can utilize shared information among tensors to improve decomposition identifiability with keeping feature/component comparability. In addition, coupled tensor decomposition has the advantage of imposing constraints on particular modes or components compared to its matrix counterparts [14]. Any combination of constraints, including independence, sparsity, orthogonality and non-negativity, can be added more easily and flexibly [14]. Moreover, it is more reliable to consider the high-order feature of tensors in data analysis [5].

In the analysis of EEG data collected from multiple subjects under the same stimulus, the time consumed by the depletion process in the coupled algorithms (assuming that shared information exists in both modes of space and frequency) would go extremely heavy, due to the high-order, high-dimensional and nonnegative nature of EEG data. To address this problem, we propose a fast double-coupled nonnegative Canonical Polyadic Decomposition (FDC-NCPD) algorithm. This algorithm is based on linked CP tensor decomposition (LCPTD) model [10] and fast Hierarchical

This work was supported by the National Natural Science Foundation of China (Grant No. 81471742), the Fundamental Research Funds for the Central Universities [DUT16JJ(G)03] in Dalian University of Technology in China, and the scholarships from China scholarship Council (No. 201706060262). The authors would like to thank Dr. P. Toiviainen and Dr. M. Huotilainen for the use of EEG data in Experiment 4.2.

* Corresponding author: cong@dlut.edu.cn

Alternating Least Squares (Fast-HALS) algorithm [15]. With the proposed algorithm, time consumption is greatly reduced without losing decomposition quality in the analysis of large-scale tensor blocks.

The rest of this paper is organized as follows. Section 2 introduces the LCPTD model. In section 3, a fast double-coupled implementation of LCPTD model is proposed. In section 4, simulation experiments are conducted to verify the performance of the proposed algorithm. The last section concludes this paper.

2. REVIEW OF LCPTD MODEL

To deal with multi-block tensors with coupling information, researchers in [10] proposed a generalized model of simultaneous decomposition, namely LCPTD model, which is defined as follows:

$$\begin{aligned}\underline{\boldsymbol{X}}^{(s)} \approx \hat{\underline{\boldsymbol{X}}}^{(s)} &= \sum_{r=1}^{R} \lambda_r^{(s)} \boldsymbol{u}_r^{(1,s)} \circ \boldsymbol{u}_r^{(2,s)} \circ \cdots \circ \boldsymbol{u}_r^{(N,s)} \\ &= [\![\underline{\boldsymbol{G}}^{(s)}; \boldsymbol{U}^{(1,s)}, \boldsymbol{U}^{(2,s)}, \cdots, \boldsymbol{U}^{(N,s)}]\!] \\ \text{s.t.} \left\| \boldsymbol{u}_r^{(n,s)} \right\| &= 1, n = 1 \cdots N, r = 1 \cdots R, s = 1 \cdots S,\end{aligned} \tag{1}$$

where $\underline{\boldsymbol{X}}^{(s)} \in \Re^{I_1 \times I_2 \times \cdots I_N}$ and $\hat{\underline{\boldsymbol{X}}}^{(s)} \in \Re^{I_1 \times I_2 \times \cdots I_N}$ denote the original and estimated tensors, respectively. $\boldsymbol{U}^{(n,s)} = \left[\boldsymbol{u}_1^{(n,s)}, \cdots, \boldsymbol{u}_R^{(n,s)}\right] \in \Re^{I_n \times R}$ denotes the n-mode factor matrix of sth tensor. S, R, N are denoted as the number, rank and order of tensors, respectively. $\underline{\boldsymbol{G}}^{(s)} \in \Re^{R \times R \times \cdots R}$ denotes the sth core tensor with non-zero entries only on the super-diagonal. $\lambda_r^{(s)}$ is the (r,r,r)th element of $\underline{\boldsymbol{G}}^{(s)}$.

The LCPTD model assumes that each factor matrix $\boldsymbol{U}^{(n,s)} = [\boldsymbol{U}_C^{(n)} \ \boldsymbol{U}_I^{(n,s)}] \in \Re^{I_n \times R}$ consists of two parts: $\boldsymbol{U}_C^{(n)} \in \Re^{I_n \times L_n}$, $0 \le L_n \le R$ and $\boldsymbol{U}_I^{(n,s)} \in \Re^{I_n \times (R - L_n)}$. The former shared by all tensor blocks represents the coupling (same or highly correlated) information, whereas the latter corresponds to the individual characteristics of each tensor block.

3. ALGORITHM IMPLEMENTATION

In this section, an optimization criterion of squared Euclidean divergence minimization is used to evaluate the error between the original and estimated tensors. For simplicity, we assume that the element $\lambda_r^{(s)}$ in core tensors can be absorbed into the non-normalized component $\boldsymbol{u}_r^{(N,s)}$. Therefore, the cost function can be expressed in a simplified form as:

$$\begin{aligned}\min \sum_{s=1}^{S} \left\| \underline{\boldsymbol{X}}^{(s)} - \sum_{r=1}^{R} \boldsymbol{u}_r^{(1,s)} \circ \boldsymbol{u}_r^{(2,s)} \circ \cdots \circ \boldsymbol{u}_r^{(N,s)} \right\|_F^2 \\ \text{s.t. } \boldsymbol{u}_r^{(n,1)} = \cdots = \boldsymbol{u}_r^{(n,S)} \text{ for } r \le L_n,\end{aligned} \tag{2}$$

$$\left\| \boldsymbol{u}_r^{(n,s)} \right\| = 1, n = 1 \cdots N-1, r = 1 \cdots R, s = 1 \cdots S.$$

The above minimization problem can be transformed into R sub-problems via the HALS algorithm [16], which can be optimized sequentially and iteratively. More exactly, the learning rule of $\boldsymbol{u}_r^{(n,s)}$ can be formulated as follows:

$$\boldsymbol{u}_r^{(n,s)} = \begin{cases} \left[\sum_s \zeta_r^{(n,s)}\right] / \sum_s \gamma_r^{(n,s)}, & r \le L_n, \\ \zeta_r^{(n,s)} / \gamma_r^{(n,s)}, & r > L_n, \end{cases} \tag{3}$$

where $\zeta_r^{(n,s)} = \underline{\boldsymbol{Y}}_{r,(n)}^{(s)} \{\boldsymbol{u}_r^{(s)}\}^{\odot - n}$. $\underline{\boldsymbol{Y}}_{r,(n)}^{(s)}$ is the mode-n matricization of $\underline{\boldsymbol{Y}}_r^{(s)} \doteq \underline{\boldsymbol{X}}^{(s)} - \sum_{p \ne r}^{R} \boldsymbol{u}_p^{(1,s)} \circ \boldsymbol{u}_p^{(2,s)} \circ \cdots \circ \boldsymbol{u}_p^{(N,s)}$. '$\odot$' denotes the Khatri-Rao product. The scaling coefficients $\gamma_r^{(n,s)}$ can be calculated as:

$$\gamma_r^{(n,s)} = \begin{cases} \boldsymbol{u}_r^{(N,s)T} \boldsymbol{u}_r^{(N,s)}, & n \ne N. \\ 1, & n = N. \end{cases} \tag{4}$$

If $r \le L_n$, $\boldsymbol{u}_r^{(n,s)}$ will be calculated by combining all tensor information and assigned to each s. Otherwise, it needs to be calculated separately. The calculation of $\zeta_r^{(n,s)}$ in equation (3) which seems relatively simple may result in rather high computational cost, especially for large-scale problems [15]. To address the above issue, we further introduce the Fast-HALS algorithm [15] to LCPTD model instead of HALS algorithm (LCPTD-HALS algorithm only considered the introduction of HALS algorithm in the LCPTD model [10]). A detailed analysis of HALS and Fast-HALS algorithms can be found in [15, 16]. In the proposed algorithm, $\zeta_r^{(n,s)}$ in equation (3) can be represented as:

$$\zeta_r^{(n,s)} = \left[\underline{\boldsymbol{X}}_{(n)}^{(s)} \{\boldsymbol{U}^{(s)}\}^{\odot - n}\right]_r - \boldsymbol{U}^{(n,s)} \left[\xi_{(n)}^{(s)}\right]_r + \gamma_r^{(n,s)} \boldsymbol{u}_r^{(n,s)}, \tag{5}$$

where $\xi_{(n)}^{(s)} = (\boldsymbol{U}^{(s)T} \boldsymbol{U}^{(s)}) ^{\circledast} \oslash (\boldsymbol{U}^{(n,s)T} \boldsymbol{U}^{(n,s)})$. '$\circledast$' and '$\oslash$' are denoted as Hadamard (element-wise) product and element-wise division, respectively. $\boldsymbol{u}_r^{(n,s)}$, $n \ne N$ needs to be normalized to unit variance by $\boldsymbol{u}_r^{(n,s)} \leftarrow \boldsymbol{u}_r^{(n,s)} / \|\boldsymbol{u}_r^{(n,s)}\|_2$ in each iteration. Meanwhile, the denominators of equation (3) can be omitted due to the normalization of $\boldsymbol{u}_r^{(n,s)}$. In order to obtain the nonnegative components, a simple "half-rectifying" nonlinear projection is applied as $\boldsymbol{u}_r^{(n,s)} \leftarrow \|\boldsymbol{u}_r^{(n,s)}\|_+$ after (3). These R stages are updated alternatively one after another until convergence.

In the end, the (r,r,r)th element $\lambda_r^{(s)}$ of core tensors which has been absorbed into $\boldsymbol{u}_r^{(N,s)}$ can be obtained as $\lambda_r^{(s)} = \|\boldsymbol{u}_r^{(N,s)}\|_2$, and $\boldsymbol{u}_r^{(N,s)}$ needs to be normalized as $\boldsymbol{u}_r^{(N,s)} \leftarrow \boldsymbol{u}_r^{(N,s)} / \|\boldsymbol{u}_r^{(N,s)}\|_2$. Compared with the mode-n matricization $\underline{\boldsymbol{Y}}_{r,(n)}^{(s)}$ of $\underline{\boldsymbol{Y}}_r^{(s)}$ in (3), which is performed repeatedly in each iteration, mode-n matricization $\underline{\boldsymbol{X}}_{(n)}^{(s)}$ in (5) only needs to be executed once in the initialization before the iteration. This greatly improves the computational efficiency of the proposed algorithm. With the consideration

that tensors are only assumed to be coupled in two modes (i.d., $L_n = 0$, $n > 2$), this extended fast-HALS-based algorithm with nonnegative and CP constraints is termed as the fast double-coupled nonnegative CP Decomposition (FDC-NCPD). Please refer to [17] for the detailed description of standard notations and basic tensor operations due to the limited length of paper.

4. SIMULATION RESULTS

In this section, simulation experiments of synthetic and real-world ongoing EEG data are provided to illustrate and compare the performance of NTF-HALS [16], NTF-FastHALS [15], LCPTD-HALS [10] and FDC-NCPD algorithms.

The following experiments are done with the following computer configurations; CPU: Intel Core i5-7500 @ 3.40Hz 3.41Hz; Memory: 8.00 Gb; System: 64-bit Windows 10; Matlab R2016b.

4.1. Synthetic data

Data generation. The double-coupled nonnegative tensors with noisy disturbance are generated as follows:

$$\underline{\boldsymbol{X}}^{(s)'} = \sigma_s \frac{\underline{\boldsymbol{X}}^{(s)}}{\|\underline{\boldsymbol{X}}^{(s)}\|_F} + \sigma_n \frac{\underline{\boldsymbol{N}}^{(s)}}{\|\underline{\boldsymbol{N}}^{(s)}\|_F}, \ s = 1 \cdots S, \quad (6)$$

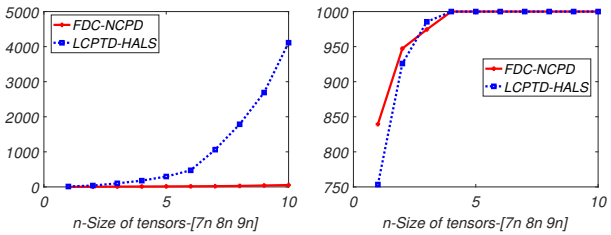
where $\underline{\boldsymbol{X}}^{(s)}$ is constructed as in equation (1). $\underline{\boldsymbol{N}}^{(s)}$ denotes noise term drawn from the standard uniform distribution on the open interval $(0, 1)$. Moreover, σ_s and σ_n represent the levels of signal and noise, respectively. Signal-to-noise ratio (SNR) is defined as $\mathrm{SNR} = 10\log_{10}(\sigma_s/\sigma_n)$.

Evaluation index. The performance index (PI) and Fit are used to evaluate the decomposition quality in the following experiments. More exactly, we use Fit value to measure the tensor reconstruction capability of algorithms, which is defined as: $Fit = \|\underline{\boldsymbol{X}}^{(s)} - \underline{\hat{\boldsymbol{X}}}^{(s)}\|_F / \|\underline{\boldsymbol{X}}^{(s)}\|_F$, where $\underline{\hat{\boldsymbol{X}}}^{(s)}$ is the reconstructed version of $\underline{\boldsymbol{X}}^{(s)}$. We use PI value to evaluate the accuracy of the estimated factor matrices [18].

Termination criteria. The iteration termination criteria for all algorithms are set as $|\mathrm{Fit}_{new} - \mathrm{Fit}_{old}| < \varepsilon, \varepsilon = 1e-6$ but no more than 1000 iterations .

4.1.1. *Convergence speed*

In this experiment, execution time and iteration number of LCPTD-HALS and FDC-NCPD algorithms are compared against the dimensionality of tensors $I_1 = 7n, I_2 = 8n, I_3 = 9n$ with n varying from 1 to 10. The target tensors are constructed as equation (6) under the noise scenario $\mathrm{SNR} = 20$ dB. The number of components, coupled components and tensors are fixed to $R = 4n$, $L_1 = L_2 = 2n$, $S = 10$, respectively. The performance curves averaged from 30 Monte Carlo runs are illustrated in Fig. 1.



(a) Execution time (b) Number of iterations

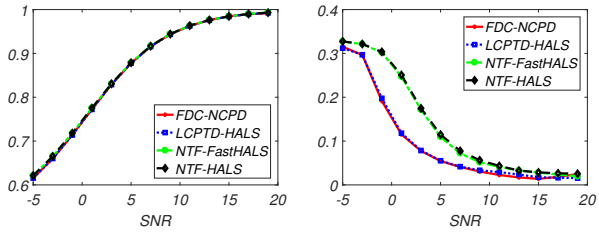
Fig. 1. Averaged execution time and iteration number versus dimensionality of tensors

From Fig.1 (a), we can see that the FDC-NCPD algorithm could greatly reduce the execution time compared with LCPTD-HALS algorithm. This advantage becomes more significant as the dimensionality of tensors increases. Furthermore, in Fig.2 (b), the difference in the number of iterations between two algorithms seems relatively small, especially when $n \geq 4$, which indicates that the FDC-NCPD algorithm outperforms significantly in terms of convergence speed in each iteration. The increase of convergence speed is in parallel with our analysis in section 3, replacing $\underline{\boldsymbol{Y}}^{(s)}_{r,(n)}$ in equation (3) by $\underline{\boldsymbol{X}}^{(s)}_{(n)}$ in equation (5) greatly reduces the calculation time.

4.1.2. *Decomposition quality*

In this experiment, we compare the Fit and PI performance of FDC-NCPD with LCPTD-HALS and two conventional tensor decomposition algorithms including NTF-HALS and NTF-FastHALS. The noisy double-coupled non-negative tensors are generated as equation (6) under different SNRs from -5 dB to 20 dB with a step size of 2 dB. For the dimensionality of tensors, we set $I_1 = 40, I_2 = 50, I_3 = 60$. The number of components, coupled components and tensors are fixed to $R = 30$, $L_1 = L_2 = 20$ and $S = 10$, respectively. The averaged performance curves obtained from 20 Monte Carlo runs are ploted as in Fig. 2.

As indicated in Fig.2 (a), the four algorithms provide nearly the same Fit performance under all SNRs. In Fig.2 (b), we can see that FDC-NCPD and LCPTD-HALS algorithms show better PI performance than the two conventional algorithms. Moreover, the coupled algorithms obtain equal performance in low SNRs (-5∼7 dB). When SNR is distributed in 7∼16 dB, FDC-NCPD algorithm yields slightly better performance than LCPTD-HALS algorithm. However, when SNR exceeds 17 dB, the proposed algorithm slightly underperforms its competitors. This experiment also verifies that joint/coupled analysis can effectively utilize prior information to improve the decomposition accuracy.



(a) Fit of tensors (b) PI of factor matrices

Fig. 2. Average tensor Fit and PI versus SNR

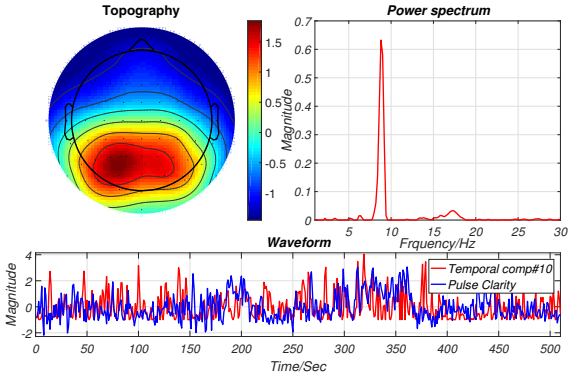


Fig. 3. The 10th temporal component and its corresponding spatial and spectral components from subject #1

4.2. Real-world ongoing EEG data

Data description. In this experiment, we apply the FDC-NCPD and LCPTD-HALS algorithms to the ongoing EEG data, collected from 14 subjects while listening to an 8.5-minute long tango music. We compare the performance of two algorithms in terms of execution time, data Fit and the number of components matched with musical features. The details of data collection, data preprocessing and music feature extraction can be found in [19]. Through short-time Fourier transform (STFT), 14 third-order tensors are formulated with size of $64 \times 146 \times 510$ (64 spatial channels, 146 frequency bins (1$\sim$30Hz) and 510 temporal samples from EEG data of each subject).

Parameter selection. For the selection of the number of components, smoothed DIFFIT [20] is used in this study with suggested $R = 36$. We assume that the coupling information exists both in brain activation areas and frequency oscillations among participants. Thus, the number of coupled components is set as $L_1 = L_2 = 20, L_3 = 0$ (after analyzing the components obtained by conventional method, we found that there were nearly 20 spatial or frequency components among participants with correlations of more than 0.96). Random initializations are used for both factor matrices and core tensors. Termination criteria of algorithms are identical with Experiment 4.1.

Correlation analysis. The temporal, spectral and spatial components can be extracted simultaneously via the FDC-NCPD and LCPTD-HALS algorithms. Correlation analyses are conducted between the temporal courses from EEG data and the temporal courses of musical features, aiming to find the brain activities corresponding to musical stimuli. In addition, we are interested in finding brain components with the significant correlation coefficients(at level $p < 0.05$). The method for determining significant correlation thresholds can be found in [21]. Fig. 3 shows an example of 10th EEG components (topography, spectrum and waveform) extracted from subject #1, in which the temporal course of temporal component is significantly correlated with the temporal course of musical feature termed as 'Pulse Clarity' ($0.1462 > 0.1167$). In addition, the corresponding spatial and spectral components indicate that the posterior area of subject #1 is activated with an alpha oscillation (8$\sim$13Hz) when listening to the tango music.

Table 1. Performance of two algorithms in ongoing EEG data analysis. I-total number of components matched with musical feature, II-execution time, III-data Fit.

	I	II	III
LCPTD-HALS	59.3	76442.65	0.7360
FDC-NCPD	65.6	350.97	0.7353

Results analysis. By 10 times of algorithm executions and correlation analyses, the averaged results of execution time, data Fit and the number of components matched with musical features are illustrated in Table 1. It can be noted that the FDC-NCPD algorithm extracts 6.3 interested components more than the latter on the total number of components. More importantly, the FDC-NCPD algorithm greatly reduces the execution time by nearly 200 times, while yielding equal performance on data Fit (the gap of 0.0007 can be negligible).

5. CONCLUSION

In this study, we introduced the Fast-HALS algorithm to LCPTD model and proposed the FDC-NCPD algorithm, in which the common components, individual components and core tensors can be extracted simultaneously. Simulation experiments of synthetic and real-world data were conducted, showing that the proposed algorithm can significantly reduce time consumption while retaining the decomposition quality. Besides, it can extract a larger number of interested components in the EEG data analysis. In the future studies, we can further analyze brain activation regions and frequency oscillations corresponding to the significantly correlated temporal components.

6. REFERENCES

[1] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*, John Wiley & Sons, 2009.

[2] N.D. Sidiropoulos, L. De Lathauwer, X. Fu, et al., "Tensor decomposition for signal processing and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, 2017.

[3] A.H. Phan and A. Cichocki, "Tensor decompositions for feature extraction and classification of high dimensional datasets," *Nonlinear theory and its applications, IEICE*, vol. 1, no. 1, pp. 37–68, 2010.

[4] M. Mørup, "Applications of tensor (multiway array) factorizations and decompositions in data mining," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 24–40, 2011.

[5] F.Y. Cong, Q.H. Lin, L.D. Kuang, et al., "Tensor decomposition of eeg signals: a brief review," *Journal of neuroscience methods*, vol. 248, pp. 59–69, 2015.

[6] N.M. Correa, T. Adalı, Y.O. Li, and V.D. Calhoun, "Canonical correlation analysis for data fusion and group inferences," *IEEE signal processing magazine*, vol. 27, no. 4, pp. 39–50, 2010.

[7] G.X. Zhou, A. Cichocki, Y. Zhang, and D.P. Mandic, "Group component analysis for multiblock data: Common and individual feature extraction," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 11, pp. 2426–2439, 2016.

[8] G.X. Zhou, Q.B. Zhao, Y. Zhang, et al., "Linked component analysis from matrices to high-order tensors: Applications to biomedical data," *Proceedings of the IEEE*, vol. 104, no. 2, pp. 310–331, 2016.

[9] X.F. Gong, X.L. Wang, and Q.H. Lin, "Generalized non-orthogonal joint diagonalization with lu decomposition and successive rotations.," *IEEE Trans. Signal Processing*, vol. 63, no. 5, pp. 1322–1334, 2015.

[10] T. Yokota, A. Cichocki, and Y. Yamashita, "Linked parafac/cp tensor decomposition and its fast implementation for multi-block tensor analysis," in *International Conference on Neural Information Processing*. Springer, 2012, pp. 84–91.

[11] M. Sørensen and L. De Lathauwer, "Multidimensional harmonic retrieval via coupled canonical polyadic decompositionpart ii: Algorithm and multirate sampling," *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 528–539, 2017.

[12] X.F. Gong, Q.H. Lin, F.Y. Cong, and L. De Lathauwer, "Double coupled canonical polyadic decomposition for joint blind source separation," *IEEE Transactions on Signal Processing*, vol. 66, no. 13, pp. 3475–3490, 2016.

[13] E. Acar, R. Bro, and A.K. Smilde, "Data fusion in metabolomics using coupled matrix and tensor factorizations," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1602–1620, 2015.

[14] A. Cichocki, "Tensor decompositions: a new concept in brain data analysis?," *arXiv preprint arXiv:1305.0395*, 2013.

[15] A. Cichocki and A.H. Phan, "Fast local algorithms for large scale nonnegative matrix and tensor factorizations," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 92, no. 3, pp. 708–721, 2009.

[16] A. Cichocki, R. Zdunek, and S. Amari, "Hierarchical als algorithms for nonnegative matrix and 3d tensor factorization," in *International Conference on Independent Component Analysis and Signal Separation*. Springer, 2007, pp. 169–176.

[17] T.G. Kolda and B.W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[18] A. Souloumiac, "Joint diagonalization: Is non-orthogonal always preferable to orthogonal?," in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2009 3rd IEEE International Workshop on*. IEEE, 2009, pp. 305–308.

[19] F.Y. Cong, V. Alluri, A.K. Nandi, et al., "Linking brain responses to naturalistic music through analysis of ongoing eeg and stimulus features," *IEEE Transactions on Multimedia*, vol. 15, no. 5, pp. 1060–1069, 2013.

[20] D.Q. Wang, Y.J. Zhu, T. Ristaniemi, and F.Y. Cong, "Extracting multi-mode erp features using fifth-order nonnegative tensor decomposition," *Journal of neuroscience methods*, vol. 308, pp. 240–247, 2018.

[21] V. Alluri, P. Toiviainen, I.P. Jääskeläinen, et al., "Large-scale brain networks emerge from dynamic processing of musical timbre, key and rhythm," *Neuroimage*, vol. 59, no. 4, pp. 3677–3689, 2012.