A Blind Block Term Decomposition of High Order Tensors

Yunfeng Cai, Ping Li
Cognitive Computing Lab
Baidu Research
No. 10 Xibeiwang East Road, Beijing 100193, China
10900 NE 8th St. Bellevue, Washington 98004, USA
{caiyunfeng, liping11}@baidu.com

Abstract

Tensor decompositions have found many applications in signal processing, data mining, machine learning, etc. In particular, the block term decomposition (BTD), which is a generalization of CP decomposition and Tucker decomposition/HOSVD, has been successfully used for the compression and acceleration of neural networks. However, computing BTD is NP-hard, and optimization based methods usually suffer from slow convergence or even fail to converge, which limits the applications of BTD. This paper considers a "blind" block term decomposition (BBTD) of high order tensors, in which the block diagonal structure of the core tensor is unknown. Our contributions include: 1) We establish the necessary and sufficient conditions for the existence of BTD, characterize the condition when a BTD solves the BBTD problem, and show that the BBTD is unique under a "low rank" assumption. 2) We propose an algebraic method to compute the BBTD. This method transforms the problem of determining the block diagonal structure of the core tensor into a clustering problem of complex numbers, in polynomial time. And once the clustering problem is solved, the BBTD can be obtained via computing several matrix decompositions. Numerical results show that our method is able to compute the BBTD, even in the presence of noise to some extent, whereas optimization based methods (e.g., MINF and NLS in TENSORLAB) may fail to converge.

1 Introduction

Tensors are multi-dimensional arrays, and they are powerful for the representation of high dimensional data. Just like matrix decompositions play important roles in matrix computations (linear system solving, least-square problem, eigenvalue problem, etc.), tensor decompositions nowadays become fundamental tools in multi-dimensional data analysis. Though tensors and matrices have many similarities (since tensors are natural generalizations of matrices), they also have significant differences, e.g., a real-valued tensor may have different tensor ranks over $\mathbb{R}$ and $\mathbb{C}$, in fact, the problem of the determination of the tensor rank is NP-hard (Håstad 1990); the rank decomposition of high order tensors are unique under mild conditions (Kruskal 1977, 1989), whereas matrix decompositions are not; the best low rank approximation of a high order tensor may not exist. We also refer readers to Kolda and Bader (2009); Sidiropoulos et al. (2017).

Though tensors are still not well understood as matrices, tensors and tensor decompositions have already found many applications in signal processing, data mining, machine learning, chemometrics, psychometrics, etc. For example, classification (Rendle 2010), recommender system (Acar, Kolda, and Dunlavy 2011; Karatzoglou et al. 2010), learning latent variables (Anandkumar et al. 2014), knowledge graph embedding (Balazevic, Allen, and Hospedales 2019), tensor completion (Cai and Li 2020b). We refer the readers to Acar and Yener (2008); Kolda and Bader (2009); Sidiropoulos et al. (2017); Papalexakis, Faloutsos, and Sidiropoulos (2017) for more detailed applications. Tensor decompositions are also found useful in speeding-up, compressing and understanding neural networks (Calvi et al. 2019; Cohen, Sharir, and Shashua 2016; Lebedev et al. 2015; Novikov et al. 2015; Stoudenmire and Schwab 2016; Tjandra, Sakti, and Nakamura 2017, 2018). Many numerical methods, mostly optimization based, are also developed to compute tensor decompositions, see Cichocki et al. (2015); Grasedyck, Kressner, and Tobler (2013); Lu, Plataniotis, and Venetsanopoulos (2011) and reference therein.

There are two fundamental tensor decompositions – CPD (Carroll and Chang 1970; Cattell 1952; Harshman 1970; Hitchcock 1928), and Tucker decomposition/HOSVD (Tucker 1966). BTD (De Lathauwer 2008a,b; De Lathauwer and Nion 2008), which is popularized by De Lathauwer and his collaborators, unifies CPD and HOSVD. To introduce the focus of this paper – the "blind" BTD (BBTD), we consider a model from multidimensional independent component analysis (MICA) (Cardoso 1998; Póczos and Lőrincz 2005; Theis 2006):

$$\mathbf{x} = \mathbf{M}\mathbf{s} + \mathbf{n},$$

where $\mathbf{x} = [x_1, \ldots, x_m]^T \in \mathbb{R}^m$ is the observed mixture, $\mathbf{M} \in \mathbb{R}^{m\times n}$ is a full column rank matrix, $\mathbf{s} \in \mathbb{R}^n$ is the source signal, and $\mathbf{n} \in \mathbb{R}^m$ is the noise vector. The task is to recover $\mathbf{s}$ from $\mathbf{x}$. Let $\mathbf{s} = \left[\mathbf{s}_1^T, \ldots, \mathbf{s}_R^T\right]^T$ with $\mathbf{s}_r \in \mathbb{R}^{n_r}$ and $\mathbf{n} = [\nu_1, \ldots, \nu_m]^T$. Assume that all $\mathbf{s}_r$'s are centered and independent of each other. Also the noises $\nu_1, \ldots, \nu_m$ are i.i.d. Gaussian and independent of the sources. Basically, if $\mathbf{M}$ is known, we can recover the sources. Now consider the fourth order cumulant $\boldsymbol{\mathcal{C}}_x$ of $\mathbf{x}$, which is a fourth order tensor with dimension m, and its entries can be given by

$$\boldsymbol{\mathcal{C}}_x(i,j,k,l) = \mathbb{E}(x_i x_j x_k x_l) - \mathbb{E}(x_i x_j)\mathbb{E}(x_k x_l) \\ - \mathbb{E}(x_i x_k)\mathbb{E}(x_j x_l) - \mathbb{E}(x_i x_l)\mathbb{E}(x_j x_k).$$

The following equality holds:

$$\boldsymbol{\mathcal{C}}_x = \boldsymbol{\mathcal{C}}_s \times_1 \mathbf{M} \times_2 \mathbf{M} \times_3 \mathbf{M} \times_4 \mathbf{M}, \qquad (1)$$

where $\boldsymbol{\mathcal{C}}_s$ is the fourth order cumulant of $\mathbf{s}$, $\times_n$ is the modal product, see e.g., Van Loan and Golub (2012, Chapter 12). Based on the assumptions on $\mathbf{s}$, we know that $\boldsymbol{\mathcal{C}}_s$ is a block diagonal tensor with R blocks, and the sizes of the blocks are $n_r \times n_r \times n_r \times n_r$ for $r = 1, \ldots, R$. When all $n_r \equiv 1$, then (1) is the CPD of $\boldsymbol{\mathcal{C}}_x$; When all n_r's are known, then (1) is a BTD of $\boldsymbol{\mathcal{C}}_x$. Consequently, by computing a tensor decomposition, we obtain an estimation for $\mathbf{M}$. However, when R and n_r's are unknown, the problem becomes even more challenging than computing BTD. Hereafter, we call BTD with unknown block structure as BBTD.

In the past two decades, BTD has found many applications in MICA (Cardoso 1998; De Lathauwer, De Moor, and Vandewalle 2000; Theis 2005, 2006) and semidefinite programming (Bai et al. 2009; De Klerk, Pasechnik, and Schrijver 2007; De Klerk and Sotirov 2010; Gatermann and Parrilo 2004). And recently, BTD is also successfully used to clustering (Wang and Zeng 2019) and the compression and acceleration of neural networks (Li et al. 2017; Ye et al. 2018; Ben-younes et al. 2019). However, in the training of neural networks, the block diagonal structure of the core tensor is unknown. And people treat the block diagonal structure as hyperparameters, which need to be carefully tuned. Therefore, once BBTD is solved, the solution can be beneficial in tuning the hyperparameters.

Related Work. BTD and its variants, especially the so-called joint block diagonalization (JBD) problem, have been studied for decades, both theoretically and algorithmically. Sufficient conditions for the uniqueness have been proposed, e.g., De Lathauwer (2008b, 2011); Yang (2014); Sørensen and De Lathauwer (2015). Various optimization based methods have been developed, e.g., Sorber, Van Barel, and De Lathauwer (2015); Sorber, Barel, and Lathauwer (2012). However, there are far less studies for the case when the block diagonal structure is unknown, i.e., the BBTD problem. For a special BBTD – the blind JBD problem, was studied from algebraic point of views (Maehara and Murota 2011; De Klerk, Dobre, and Ṗasechnik 2011; Cai and Liu 2017; Cai and Li 2020a); in addition, Nion (2011) proposed to determine the block diagonal structure via a generalized eigenproblem; Cai, Cheng, and Shi (2019) proposed to solve blind JBD problem via a matrix polynomial.

Our Contribution. In this paper, we mathematically formulate the BBTD problem. We establish a necessary and sufficient condition for the existence of the BTD, and then we characterize the condition under which a BTD is a solution to the BBTD problem. We show that the BBTD is unique under proper assumptions. Finally, we propose a numerical method to compute the BBTD. Numerical simulations show that our method outperforms state-of-the-arts optimization based methods.

Organization. Section 2 gives preliminary definitions and propositions and raises the BBTD problem. Section 3 establishes the necessary and sufficient conditions for the existence and uniqueness of the decomposition and proposes an algebraic method to compute the decomposition. The approximate BBTD and some variants of BBTD are discussed in Sections 4 and 5, respectively. Numerical simulations are presented in Section 6. Section 7 concludes the paper.

Notation. Lowercase letters denote scalars (e.g., a, b), boldface lowercase letters denote column vectors (e.g., $\mathbf{a}, \mathbf{b}$), boldface uppercase letters denote matrices (e.g., $\mathbf{A}, \mathbf{B}$), and boldface calligraphic letters denote tensors (e.g., $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}$). If a boldface letter is used to denote a vector/matrix/tensor, then the corresponding lowercase letter with a subscript refers to its entry, e.g., a_i, a_{ij} and a_{ijk} stand for the ith entry of a vector $\mathbf{a}$, the (i,j) entry of a matrix $\mathbf{A}$ and the (i,j,k) entry of a third order tensor $\boldsymbol{\mathcal{A}}$, respectively. Indices typically range from 1 to their italic uppercase version, e.g., $i = 1, 2, \ldots, I$. The MATLAB colon notation is used to indicate submatrices of a given matrix or subtensors of a given tensor, e.g., $\mathbf{A}_{(i_1:i_2, j_1:j_2)}$ denotes a submatrix of $\mathbf{A}$, which consists of row i_1 to row i_2 and column j_1 to column j_2. The symbol $\otimes$ denotes the Kronecker product. The identity matrix of order N is denoted by $\mathbf{I}_N$. For a real square matrix $\mathbf{A}$, $\lambda(\mathbf{A})$ denotes the eigenvalue set of $\mathbf{A}$, and $\lambda_d(\mathbf{A})$ denote the distinct eigenvalues of $\mathbf{A}$ within the closed upper complex plane, e.g., $\mathbf{A} = \operatorname{diag}(1+\imath, 1-\imath, 2, 2, 3)$, $\lambda(\mathbf{A}) = \{1 \pm \imath, 2, 2, 3\}$, $\lambda_d(\mathbf{A}) = \{1+\imath, 2, 3\}$. For a (rectangular) matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, its singular values are denoted by $\sigma_1(\mathbf{A}) \geq \cdots \geq \sigma_{\min\{m,n\}}(\mathbf{A}) \geq 0$, and $\sigma_{\min\{m,n\}}(\mathbf{A})$ is usually denoted by $\sigma_{\min}(\mathbf{A})$. The 2-norm and Frobenius norm are denoted by $\|\cdot\|_2$ and $\|\cdot\|_F$, respectively.

2 Preliminary

In this section, we first briefly review CPD, HOSVD, and BTD, together with some definitions, followed by the mathematical formulation of the BBTD problem.

CPD. The CANDECOMP/PARAFAC decomposition (CPD) of a tensor $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is a decomposition of $\boldsymbol{\mathcal{T}}$ as a linear combination of rank-1 terms:

$$\boldsymbol{\mathcal{T}} = [\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}]\!] = \sum_{r=1}^{R} \lambda_r \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \cdots \circ \mathbf{a}_r^{(N)},$$

where $\boldsymbol{\lambda} = \operatorname{diag}(\lambda_1, \ldots, \lambda_R)$ is a diagonal tensor, and $\mathbf{A}^{(n)} = [\mathbf{a}_1^{(n)}, \ldots, \mathbf{a}_R^{(n)}]$ with $\|\mathbf{a}_1^{(n)}\|_2 = \cdots = \|\mathbf{a}_R^{(n)}\|_2 = 1$, $n = 1, 2, \ldots, N$. The smallest integer R such that the equality holds is the tensor rank, denoted by $\operatorname{rank}(\boldsymbol{\mathcal{T}})$. See Figure 1 for an illustration.

The mode-n unfolding of a tensor $\boldsymbol{\mathcal{T}}$ is denoted by $\mathbf{T}_{(n)}$, the mode-n product between a tensor $\boldsymbol{\mathcal{T}}$ and a matrix $\mathbf{A}$ is denoted by $\boldsymbol{\mathcal{T}} \times_n \mathbf{A}$; see Van Loan and Golub (2012, Chapter 12) for the definitions of modal unfolding, modal product and also their properties.

HOSVD. Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ and the economic singular value decompositions of the modal-n unfoldings of $\boldsymbol{\mathcal{T}}$ be $\boldsymbol{\mathcal{T}}_{(n)} = \mathbf{U}^{(n)} \boldsymbol{\Sigma}^{(n)} (\mathbf{V}^{(n)})^T$ for $n = 1, \ldots, N$, where $\mathbf{U}^{(n)}$

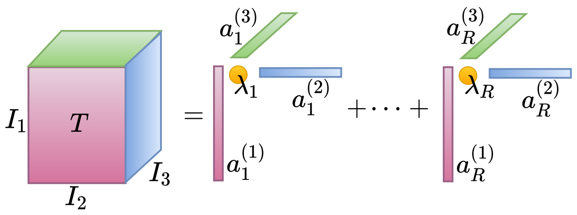

Figure 1: CPD of a third order tensor

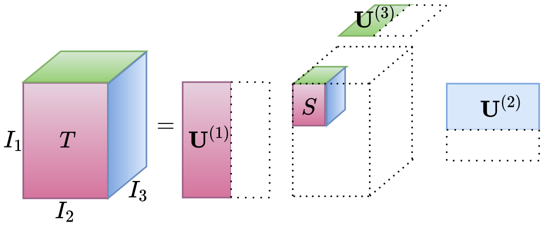

Figure 2: HOSVD of a third order tensor

and $\mathbf{V}^{(n)}$ have orthonormal columns, $\mathbf{\Sigma}^{(n)}$ has the nonzero singular values of $\mathcal{T}_{(n)}$ on its main diagonal. The high order singular value decomposition (HOSVD) of $\mathcal{T}$ is given by

$$\mathcal{T} = [\![\mathcal{S}; \mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(N)}]\!] = \mathcal{S} \times_1 \mathbf{U}^{(1)} \cdots \times_N \mathbf{U}^{(N)}, \quad (2)$$

where $\mathcal{S} = \mathcal{T} \times_1 (\mathbf{U}^{(1)})^T \times_2 (\mathbf{U}^{(2)})^T \cdots \times_N (\mathbf{U}^{(N)})^T$ is the core tensor. See Figure 2 for an illustration.

Definition 1. *The multilinear rank of a tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ is a vector of modal unfolding ranks:*

$$\text{rank}_*(\mathcal{T}) = [\text{rank}(\mathcal{T}_{(1)}), \text{rank}(\mathcal{T}_{(2)}), \ldots, \text{rank}(\mathcal{T}_{(N)})].$$

$\mathcal{T}$ is of **full multilinear rank** *if* $\text{rank}_*(\mathcal{T}) = [I_1, \ldots, I_N]$.

Note here that $\text{rank}(\mathcal{T}_{(n)})$ is the number of columns of $\mathbf{U}^{(n)}$. Thus, the core tensor $\mathcal{S}$ is in fact dimension $\text{rank}(\mathcal{T}_{(1)})$-by-$\text{rank}(\mathcal{T}_{(2)})$-$\cdots$-$\text{rank}(\mathcal{T}_{(N)})$.

BTD. A block term decomposition (BTD) of a tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ in rank-$[M_r^{(n)}]$ terms is a decomposition of $\mathcal{T}$ of the form:

$$\begin{aligned} \mathcal{T} &= [\![\mathcal{S}; \mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}]\!] = \sum_{r=1}^{R} [\![\mathcal{S}_r; \mathbf{U}_r^{(1)}, \ldots, \mathbf{U}_r^{(N)}]\!] \\ &= \sum_{r=1}^{R} \mathcal{S}_r \times_1 \mathbf{A}_r^{(1)} \times_2 \mathbf{A}_r^{(2)} \cdots \times_N \mathbf{A}_r^{(N)}, \end{aligned} \quad (3)$$

where $\mathcal{S} = \text{diag}(\mathcal{S}_1, \ldots, \mathcal{S}_R)$ with $\mathcal{S}_r \in \mathbb{R}^{M_r^{(1)} \times \cdots \times M_r^{(N)}}$, and $\mathbf{A}^{(n)} = [\mathbf{A}_1^{(n)}, \ldots, \mathbf{A}_R^{(n)}]$ with $\mathbf{A}_r^{(n)} \in \mathbb{R}^{I_n \times M_r^{(n)}}$. See Figure 3 for an illustration.

We can see that BTD unifies the CPD and HOSVD, and it also provides a unifying view on tensor rank (De Lathauwer 2008b), since not strictly speaking, when each term of BTD is rank one, BTD becomes CPD; when BTD has only one term, BTD becomes HOSVD.

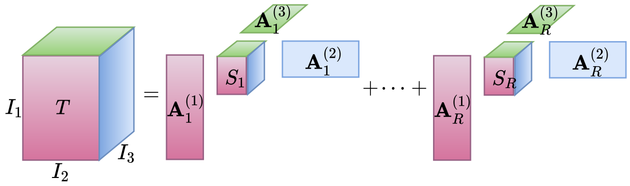

Figure 3: BTD of a third order tensor

Next, we give several definitions to make our further discussions more precise.

Definition 2. *We call* $\tau = (M_r^{(n)}) = \begin{pmatrix} M_1^{(1)} & \ldots & M_R^{(1)} \\ \vdots & \ddots & \vdots \\ M_1^{(N)} & \ldots & M_R^{(N)} \end{pmatrix}$ *a partition of positive integer-valued vector $(I_1, \ldots, I_N)$ if $M_r^{(n)}$ for $1 \le r \le R$, $1 \le n \le N$ are all positive integers and $\sum_{r=1}^{R} M_r^{(n)} = I_n$ for $1 \le n \le N$. The integer R is called the cardinality of τ, denoted by* $\text{card}(\tau)$. *The set of all partitions of $(I_1, \ldots, I_N)$ is denoted by $\mathbb{T}(I_1, \ldots, I_N)$. In particular, if $N = 1$, we call $\tau = (M_1, \ldots, M_R)$ a partition of I if $\sum_{r=1}^{R} M_r = I$, and the set of all partitions of I is denoted by $\mathbb{T}(I)$.*

The partition $\tau = (M_r^{(n)}) \in \mathbb{T}(I_1, \ldots, I_N)$ essentially determines the BTD in (3): the number of rows of τ is the order of the tensor, the number of columns of τ is the number of terms in BTD, and each row vector of τ corresponds to the dimension of a diagonal block of the core tensor. *Hereafter we say that $\mathcal{T}$ has a τ-BTD with the meaning that* (3) *holds,* which is more informative.

Definition 3. *Given a partition $\tau = (M_r^{(n)}) \in \mathbb{T}(I_1, \ldots, I_N)$ with* $\text{card}(\tau) = R$ *and an Nth order tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$. Partition $\mathcal{T}$ as $\mathcal{T} = [\mathcal{T}_{r_1 r_2 \ldots r_N}]$ with $\mathcal{T}_{r_1 r_2 \ldots r_N} \in \mathbb{R}^{M_{r_1}^{(1)} \times \cdots \times M_{r_N}^{(N)}}$ for $1 \le r_1, r_2, \ldots, r_N \le R$. We call $\mathcal{T}$ a τ-block diagonal tensor if $\mathcal{T}_{r_1 r_2 \ldots r_N} \neq 0$ is possible only if $r_1 = r_2 = \cdots = r_N$. Such a τ-block diagonal tensor is denoted by* $\text{diag}_\tau(\mathcal{T}_{1\ldots 1}, \ldots, \mathcal{T}_{R\ldots R})$. *In particular, if $M_r^{(n)} = 1$ for all r and n, we call $\mathcal{T}$ a diagonal tensor.*

Definition 4. *Given $\tau = (M_r^{(n)}) \in \mathbb{T}(I_1, \ldots, I_N)$ with* $\text{card}(\tau) = R$ *and an Nth order tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$. The τ-off-block diagonal part of $\mathcal{T}$ is defined as* $\text{offblkdiag}_\tau(\mathcal{T}) = \mathcal{T} - \text{diag}_\tau(\mathcal{T})$.

In Section 3, we will see that a tensor $\mathcal{T}$ has a BTD under very restrictive conditions. Therefore, in Section 4, we consider the BTD in approximate sense, with the meaning that the core tensor $\mathcal{S}$ in (3) is approximately τ-block diagonal, i.e., $\|\text{offblkdiag}_\tau(\mathcal{S})\|_F$ is small.

The uniqueness of tensor decomposition is of great importance. To discuss the uniqueness of BTD, we need the following definitions.

Definition 5. *Given a partition $\tau = (M_1, \ldots, M_R) \in \mathbb{T}(I)$, we call a matrix $\mathbf{A} \in \mathbb{R}^{I \times I}$ τ-block diagonal if $\mathbf{A} =$* $\text{diag}(\mathbf{A}_1, \ldots, \mathbf{A}_R)$ *with $\mathbf{A}_r \in \mathbb{R}^{M_r \times M_r}$ for $r = 1, \ldots, R$.*

Definition 6. *Given a partition $\tau = (M_1,\dots,M_R) \in \mathbb{T}(I)$ and a permutation π: $\{1,2,\dots,R\} \to \{\pi(1),\pi(2),\dots,\pi(R)\}$. We call $\mathbf{\Pi} \in \mathbb{R}^{I\times I}$ a τ-block permutation matrix corresponding with π if for any $\mathbf{A} = [\mathbf{A}_1,\mathbf{A}_2,\dots,\mathbf{A}_R]$ with $\mathbf{A}_r$ having M_r columns, $\mathbf{A\Pi} = [\mathbf{A}_{\pi(1)},\mathbf{A}_{\pi(2)},\dots,\mathbf{A}_{\pi(R)}]$.*

Now we may state the uniqueness for BTD.

Uniqueness. Let $\mathcal{T} = [\![\mathcal{S};\mathbf{A}^{(1)},\dots,\mathbf{A}^{(N)}]\!]$ have a τ-BTD of $\mathcal{T}$ as in (3). For $1 \le n \le N$, let $\tau_n = (M_1^{(n)},\dots,M_R^{(n)}) \in \mathbb{T}(I_n)$, for any permutation π of $\{1,2,\dots,R\}$, let $\mathbf{D}^{(n)}$, $\mathbf{\Pi}^{(n)}$ be τ_n-block diagonal matrices and τ_n-block permutation matrices corresponding with π, respectively. Then $\mathcal{T} = [\![\widehat{\mathcal{S}};\widehat{\mathbf{A}}^{(1)},\dots,\widehat{\mathbf{A}}^{(N)}]\!]$ is a $\hat{\tau}$-BTD of $\mathcal{T}$, where $\widehat{\mathcal{S}} = \mathcal{S}\times_1(\mathbf{D}^{(1)}\mathbf{\Pi}^{(1)})^{-1}\cdots\times_N(\mathbf{D}^{(N)}\mathbf{\Pi}^{(N)})^{-1}$, $\widehat{\mathbf{A}}^{(n)} = \mathbf{A}^{(n)}\mathbf{D}^{(n)}\mathbf{\Pi}^{(n)}$, and $\hat{\tau} = (M^{(n)}_{\pi(r)})$. We say the τ-BTD is *unique* if the τ-BTD is unique up to the above trivial/natural indeterminacy.

When $\tau \in \mathbb{T}(I_1,\dots,I_N)$ is prescribed, computing a τ-BTD of $\mathcal{T}$ is naturally an optimization problem, e.g.,

$$\min_{\mathcal{S}_r,\mathbf{A}_r^{(n)}} \frac{1}{2}\|\mathcal{T} - \sum_{r=1}^{R}\mathcal{S}_r \times_1 \mathbf{A}_r^{(1)}\cdots\times_N \mathbf{A}_r^{(N)}\|_F^2.$$

However, for BBTD, τ is unknown, solving BBTD without exhaustive search for τ is difficult. We propose to consider the following BBTD problem:

BBTD. Given an Nth order tensor $\mathcal{T} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ and a positive integer-valued vector $[J_1,J_2,\dots,J_N]$, find a τ-BTD of $\mathcal{T}$ with as many terms as possible, and satisfying

$$\sum_{r=1}^{R}[M_r^{(1)},M_r^{(2)},\dots,M_r^{(N)}] = [J_1,J_2,\dots,J_N].$$

With particular choices of $[J_1,J_2,\dots,J_N]$, the BBTD may become CP and HOSVD. Specifically, if $J_1 = J_2 = \cdots = J_N = \operatorname{rank}(\mathcal{T})$, the BBTD is the CPD; if $[J_1,J_2,\dots,J_N] = \operatorname{rank}_*(\mathcal{T})$ and $R = 1$, the BBTD becomes the HOSVD.

In this paper, we try to answer the following questions:

Q1 Under what conditions does a tensor $\mathcal{T}$ has a τ-BTD? And under what conditions the τ-BTD is a solution to the BBTD problem?

Q2 Is the solution to the BBTD problem unique? Under what conditions?

Q3 How to compute a solution?

Q4 How about the approximate BBTD problem?

While full answers to the above questions are beyond our knowledge, in this paper we make the following assumptions to simplify the problem:

A $[J_1,J_2,\dots,J_N] \overset{entrywise}{\le} \operatorname{rank}_*(\mathcal{T})$.

Assumption **A** is used to simulate the role of "low rank" in matrix decomposition. Under Assumption **A**, the BBTD can be simplified, based on the following proposition:

Proposition 1. *Let $\mathcal{T} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ have a τ-BTD of the form* (3) *and also a HOSVD of the form* (2)*. If the assumption* $\mathbf{A}$ *holds, then $[J_1,J_2,\dots,J_N] = \operatorname{rank}_*(\mathcal{T})$ and $(\mathbf{U}^{(n)})^T\mathbf{A}^{(n)}$ for $n = 1,\dots,N$ are all nonsingular.*

Denote the core tensor of HOSVD as $\mathcal{S}_{\text{hosvd}}$, $\mathbf{B}^{(n)} = (\mathbf{U}^{(n)})^T\mathbf{A}^{(n)}$. Using (2) and (3), we have

$$\mathcal{S}_{\text{hosvd}} = \operatorname{diag}_\tau(\mathcal{S}_1,\dots,\mathcal{S}_R)\times_1\mathbf{B}^{(1)}\cdots\times_N\mathbf{B}^{(N)}. \quad (4)$$

In other words, under assumption **A**, that $\mathcal{T}$ has a τ-BTD is equivalent to that the core tensor $\mathcal{S}_{\text{hosvd}}$ has a τ-BTD with nonsingular transformations along all modes. Once we find a τ-BTD for the core tensor $\mathcal{S}_{\text{hosvd}}$ as in (4), we know that

$$\mathcal{T} = \operatorname{diag}_\tau(\mathcal{S}_1,\dots,\mathcal{S}_R)\times_1(\mathbf{U}^{(1)}\mathbf{B}^{(1)})\cdots\times_N(\mathbf{U}^{(N)}\mathbf{B}^{(N)}),$$

i.e., $\mathcal{T}$ has a τ-BTD. Therefore, hereafter, we only consider the BBTD problem for a tensor with full multilinear rank.

3 Exact BBTD

In this section, we will give answers to the questions **Q1-Q3**.

3.1 Existence and Uniqueness

We first give the condition for the existence of BTD, then the condition for the uniqueness of BBTD.

The following linear subspace plays the central role in our analysis:

$$\mathscr{N}(\mathcal{T}) \triangleq \{(\mathbf{Z}^{(1)},\dots,\mathbf{Z}^{(N)}) \mid \mathcal{T}\times_1\mathbf{Z}^{(1)} = \mathcal{T}\times_2(\mathbf{Z}^{(2)})^T = \cdots = \mathcal{T}\times_N(\mathbf{Z}^{(N)})^T\}.$$

Now we can give the condition for the existence of BTD.

Theorem 1. *Let $\mathcal{T} \in \mathbb{R}^{I_1\times\cdots\times I_N}$ have full multilinear rank. Then $\mathcal{T}$ has a τ-BTD of the form* (3) *if and only if there exists a matrix-valued vector $(\mathbf{Z}^{(1)},\dots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\mathcal{T})$, where $\mathbf{Z}^{(n)}$ can be factorized as*

$$\begin{aligned}\mathbf{Z}^{(1)} &= \mathbf{A}^{(1)}\mathbf{\Gamma}^{(1)}(\mathbf{A}^{(1)})^{-1},\\ (\mathbf{Z}^{(n)})^T &= \mathbf{A}^{(n)}\mathbf{\Gamma}^{(n)}(\mathbf{A}^{(n)})^{-1}, \quad n > 1,\end{aligned} \quad (5)$$

where $\mathbf{\Gamma}^{(n)} = \operatorname{diag}(\mathbf{\Gamma}_1^{(n)},\dots,\mathbf{\Gamma}_R^{(n)})$ with $\lambda(\mathbf{\Gamma}_r^{(n)}) \cap \lambda(\mathbf{\Gamma}_{r'}^{(n)}) = \emptyset$ for $r \ne r'$, $\lambda_d(\mathbf{\Gamma}_r^{(n)}) = \lambda_d(\mathbf{\Gamma}_r^{(n')})$ for $n \ne n'$.

Theorem 1 gives a necessary and sufficient condition for a tensor $\mathcal{T}$ having a BTD of form (3). We can see that the size of each diagonal block of the core tensor $\mathcal{S}$ is determined by the eigenvalues of $\mathbf{Z}^{(n)}$. One might wonder if there is any eigenvalue $\lambda \in \lambda(\mathbf{Z}^{(n)})$ and $\lambda \notin \lambda(\mathbf{Z}^{(n')})$ for certain $n' \ne n$. Interestingly, the answer is negative as shown below.

Proposition 2. *Let $\mathcal{T} \in \mathbb{R}^{I_1\times\cdots\times I_N}$ have full multilinear rank. For any $(\mathbf{Z}^{(1)},\dots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\mathcal{T})$, it holds*

$$\lambda_d(\mathbf{Z}^{(1)}) = \cdots = \lambda_d(\mathbf{Z}^{(N)}).$$

Remark 1. *Theorem 1 together with Proposition 2 tells that $\cup_{n=1}^N\lambda_d(\mathbf{Z}^{(n)})$ has R distinct numbers, which is the number of diagonal blocks in the core tensor; Denote the R distinct numbers by $c_1,\dots,c_R$, and let the number of eigenvalues of $\mathbf{Z}^{(n)}$ correspond with c_r be $M_r^{(n)}$, then the sizes of the diagonal tensors will be $M_1^{(1)}$-by-$\cdots$-by-$M_1^{(N)}$, $\dots$, $M_R^{(1)}$-by-$\cdots$-by-$M_R^{(N)}$.*

Next, we characterize when a BTD of form (3) is a solution to the BBTD problem.

Theorem 2. *Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ have full multilinear rank. A τ-BTD of $\boldsymbol{\mathcal{T}}$ of the form* (3) *is a solution to the BBTD problem if and only if*

$$R = \max_{(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})} \operatorname{card} \lambda_d(\mathbf{Z}^{(n)}).$$

And $\operatorname{card} \lambda_d(\mathbf{Z}^{(n)})$ is maximized for almost all $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})$.

When a tensor $\boldsymbol{\mathcal{T}}$ has a BTD of form (3), one may ask if the BTD is a solution to the BBTD problem. The following proposition provides a necessary condition.

Proposition 3. *Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ have full multilinear rank. If a τ-BTD of $\boldsymbol{\mathcal{T}}$ of the form* (3) *is a solution to BBTD, then for any $(\mathbf{Z}_r^{(1)},\ldots,\mathbf{Z}_r^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{S}}_r)$, $|\lambda_d(\mathbf{Z}_r^{(n)})| = 1$, i.e., the eigenvalues of $\mathbf{Z}_r^{(n)}$ are the same real number or a complex conjugate pair, for all $1 \le n \le N$, $1 \le r \le R$.*

The following Theorem tells that the solution to the BBTD problem is unique.

Theorem 3. *Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ have full multilinear rank. Then the solution to the BBTD problem is unique.*

Remark 2. *The uniqueness condition here holds for general high order tensors, whereas the uniqueness conditions established in De Lathauwer (2008b); Yang (2014) are for third order tensors.*

3.2 Computing a Solution

Since the solution is unique, we only need to find one solution. To do so, we need to compute $\mathscr{N}(\boldsymbol{\mathcal{T}})$. Noticing that $\mathscr{N}(\boldsymbol{\mathcal{T}})$ is just a linear system of equations, which has $\sum_{n=1}^{N} I_n^2$ unknowns and $(N-1)I_1 I_2 \cdots I_N$ equations. In general, the linear system has only the trivial solution zero. This tells us that a tensor $\boldsymbol{\mathcal{T}}$, in general, does not have a BTD with more than one diagonal block. But it doesn't mean that BTD is useless. Because (i) the tensor $\boldsymbol{\mathcal{T}}$ is usually structured (e.g., $\boldsymbol{\mathcal{C}}_x$ in (1)), as a result, the coefficient matrix for the linear system is rank deficient; (ii) the linear system can be solved in the least square sense, then $\boldsymbol{\mathcal{T}}$ will have an approximate BTD, see in Section 4.

By Theorem 2, we only need to find a "generic" $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})$ rather than find a basis for $\mathscr{N}(\boldsymbol{\mathcal{T}})$. By "generic", we mean that $\operatorname{card} \lambda_d(\mathbf{Z}^{(n)})$ is maximized. Now our computational task becomes:

(S1) Find a "generic" $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})$;

(S2) Compute the eigenvalue decomposition (5) of $\mathbf{Z}^{(n)}$.

(S2) is simple – first, compute the Schur decomposition of $\mathbf{Z}^{(n)} = \mathbf{Q}^{(n)}\mathbf{T}^{(n)}(\mathbf{Q}^{(n)})^T$; second, determine $M_r^{(n)}$ via the algebraic multiplicities of the distinct eigenvalues of $\mathbf{T}^{(n)}$; third, compute the decomposition of $\mathbf{Z}^{(n)}$ as in (5) via Van Loan and Golub (2012, Alg. 7.6.3). Our central computational task is **(S1)** – find a "generic" $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})$. For any $1 \le n < n' \le N$, denote

$$\mathbf{i}_{nn'} \triangleq \{(i_1,\ldots,i_{n-1},:,i_{n+1},\ldots,i_{n'-1},:,i_{n'+1},\ldots,i_N) \mid 1 \le i_{n''} \le I_{n''}, n'' \neq n, n'\}.$$

Then for any $i_{1n} \in \mathbf{i}_{1n}$, $\boldsymbol{\mathcal{T}}_{i_{1n}}$ is an I_1-by-I_n matrix, and $\boldsymbol{\mathcal{T}} \times_1 \mathbf{Z}^{(1)} = \boldsymbol{\mathcal{T}} \times_n (\mathbf{Z}^{(n)})^T$ is equivalent to $\mathscr{T}_{i_{1n}}\mathbf{z} = 0$, for all $i_{1n} \in \mathbf{i}_{1n}$, where

$$\mathscr{T}_{i_{1n}} = [\boldsymbol{\mathcal{T}}_{i_{1n}}^T \otimes \mathbf{I}_{I_1}, \underbrace{0,\ldots,0}_{n-2}, -\mathbf{I}_{I_n} \otimes \boldsymbol{\mathcal{T}}_{i_{1n}}, \underbrace{0,\ldots,0}_{N-n}],$$

$$\mathbf{z} = [\operatorname{vec}(\mathbf{Z}^{(1)})^T, \operatorname{vec}(\mathbf{Z}^{(2)})^T, \ldots, \operatorname{vec}(\mathbf{Z}^{(N)})^T]^T. \quad (6)$$

Stack all $\mathscr{T}_{i_{1n}}$ for all $i_{1n} \in \mathbf{i}_{1n}$, and denote it by $\mathscr{T}_{\mathbf{i}_{1n}}$. We know that $\boldsymbol{\mathcal{T}} \times_1 \mathbf{Z}^{(1)} = \boldsymbol{\mathcal{T}} \times_n (\mathbf{Z}^{(n)})^T$ is equivalent to $\mathscr{T}_{\mathbf{i}_{1n}}\mathbf{z} = 0$. Stack all $\mathscr{T}_{\mathbf{i}_{1n}}$ for all $2 \le n \le N$, and denote it by $\mathscr{T}$. Then that $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)})$ belongs to $\mathscr{N}(\boldsymbol{\mathcal{T}})$ is equivalent to

$$\mathscr{T}\mathbf{z} = 0. \quad (7)$$

Therefore, finding a generic $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)})$ in $\mathscr{N}(\boldsymbol{\mathcal{T}})$ is essentially finding a generic $\mathbf{z}$ in the null space of $\mathscr{T}$. One can, of course, accomplish the task by computing the SVD of $\mathscr{T}$. Note that $\mathscr{T}$ is an $(N-1)I_1 I_2 \cdots I_N$-by-$\sum_{n=1}^{N} I_n^2$ matrix, which is huge when I_n's and N are large. Therefore, computing the full SVD is computationally overwhelming. Since only the null space of $\mathscr{T}$ is needed, and $\mathscr{T}$ is extremely sparse and structured, we can use the restarted Lanczos bidiagonalization method (Baglama and Reichel 2005), which is available in MATLAB with command `svds`, to compute a few smallest singular values and the corresponding right singular vectors. Then a random linear combination of the right singular vectors is "generic", from which we can construct a generic $(\mathbf{Z}^{(1)},\ldots,\mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})$. The computational cost of the method is dominated by the matrix-vector multiplications $\mathscr{T}v$ and $\mathscr{T}^T u$. Let $I_n = d$ for all n, the computational complexity for $\mathscr{T}v$ and $\mathscr{T}^T u$ is $\mathcal{O}(N \mathtt{nnz}(\boldsymbol{\mathcal{T}})d)$. Assuming that `svds` converges in $\mathcal{O}(1)$ steps, the overall computational complexity will be $\mathcal{O}(N \mathtt{nnz}(\boldsymbol{\mathcal{T}})d)$, which scales linearly with respect to the order of the tensor N, the number of nonzero entries of the tensor $\mathtt{nnz}(\boldsymbol{\mathcal{T}})$, and the dimension d. The method is thus suitable for large scale computation.

The overall algorithm is summarized in Algorithm 1. Later, we will see that the algorithm can also be used to solve the approximate BBTD problem.

Implementation Details. Line 3 can be skipped if $\boldsymbol{\mathcal{T}}$ has full multilinear rank. In Line 4, since the restarted Lanczos bidiagonalization method only requires two matrix-vector multiplications $\mathscr{T}v$ and $\mathscr{T}^T u$, it is also OK to construct two subroutines which return $\mathscr{T}v$ and $\mathscr{T}^T u$, instead of constructing $\mathscr{T}$ explicitly. In Line 5, K is a small integer, we recommend to set it as $\widehat{R}+1$, where $\widehat{R}$ is a guess for R. In Line 6, ζ is a parameter used to truncate $\hat{\sigma}_k$. Noticing that $\hat{\sigma}_1 \equiv 0$ since $(\operatorname{vec}(\mathbf{I}_{I_1}),\ldots,\operatorname{vec}(\mathbf{I}_{I_N})) \in \mathscr{N}(\mathscr{T})$, in our implementation, we set ζ as a multiple of $\hat{\sigma}_2$, e.g., $\zeta = 1.2 \times \hat{\sigma}_2$. Line 9, this step essentially determines the block diagonal structure. Any clustering algorithm can be used, e.g., k-means, DBSCAN, etc.

Algorithm 1 SVDS for BBTD (SVDS4BBTD for short)

```
Input: $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$.
Output: A solution to the BBTD problem of $\boldsymbol{\mathcal{T}}$.
Compute the HOSVD: $\boldsymbol{\mathcal{T}} = [\![\boldsymbol{\mathcal{S}}_0; \mathbf{A}_0^{(1)}, \dots, \mathbf{A}_0^{(N)}]\!]$, set $\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{S}}_0$;
Construct $\mathscr{T}$ as in (7);
Compute the $K$ smallest singular values $\hat{\sigma}_1, \dots, \hat{\sigma}_K$ (in a non-decreasing order) and the corresponding right singular vectors $\hat{\mathbf{v}}_1, \dots, \hat{\mathbf{v}}_K$ of $\mathscr{T}$;
Compute $\mathbf{z} = \sum_{k=1}^{K} \mathbb{I}_{\{\hat{\sigma}_k < \zeta\}} \alpha_k \mathbf{v}_k$, where $\zeta$ is parameter, $\alpha_k$'s are random real numbers;
Construct $\mathbf{Z}^{(n)}$ from $\mathbf{z}$ via (6);
For all $n$, compute the eigenvalues of $\mathbf{Z}^{(n)}$ via computing its Schur decomposition;
Cluster real($\lambda(\mathbf{Z}^{(1)}) \cup \cdots \cup \lambda(\mathbf{Z}^{(N)})$);
For all $n$, compute $\mathbf{Z}^{(n)} = \mathbf{A}^{(n)} \Gamma^{(n)} (\mathbf{A}^{(n)})^{-1}$, where $\Gamma^{(n)}$ satisfies the condition in Theorem 4;
Compute $\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{T}} \times_1 (\mathbf{A}^{(1)})^{-1} \cdots \times_N (\mathbf{A}^{(N)})^{-1}$;
For $1 \leq n \leq N$, compute $\mathbf{A}^{(n)} = \mathbf{A}_0^{(n)} \mathbf{A}^{(n)}$.
```

4 Approximate BBTD

In this section, we give the answer to **Q4**. The idea appears simple: solve a $(\mathbf{Z}^{(1)}, \dots, \mathbf{Z}^{(N)})$ that is approximately in $\mathscr{N}(\boldsymbol{\mathcal{T}})$, then follow the way for the exact BBTD problem.

Recall that for the exact BBTD problem, $\cup_{n=1}^{N} \lambda_d(\mathbf{Z}^{(n)})$ has R distinct numbers. For the approximate BBTD problem, we can show that $\cup_{n=1}^{N} \lambda_d(\mathbf{Z}^{(n)})$ can be divided into R clusters. Then we have the following theorem.

Theorem 4. *Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ have full multilinear rank. Suppose that*

$$\max_{\substack{\|\mathbf{v}^{(n'')}\|_2 = 1 \\ n'' \neq n, n'}} \|\boldsymbol{\mathcal{E}}^{(nn')} \times_{\substack{n''=1,\dots,N \\ n'' \neq n,n'}} (\mathbf{v}^{(n'')})^T)\|_2 \leq \delta, \qquad (8)$$

where $\boldsymbol{\mathcal{E}}^{(nn')} = \boldsymbol{\mathcal{T}} \times_n (\mathbf{Z}^{(n)})^T - \boldsymbol{\mathcal{T}} \times_1 (\mathbf{Z}^{(n')})^{\star}$, $\star = T$ for $n' = 1$, $\star$ is void otherwise, δ is a real constant, $\mathbf{Z}^{(n)}$ can be factorized as in (5)*, with $\mathbf{\Gamma}^{(n)}$ of the form in Theorem 1. Assume that the eigenvalues of all $\mathbf{Z}^{(n)}$ can be divided into R clusters, denoted by $\mathcal{C}_1, \dots, \mathcal{C}_R$. For $1 \leq n \leq N$ and $1 \leq r \leq R$, the eigenvalues of the rth diagonal block $\mathbf{\Gamma}_r^{(n)}$ of $\mathbf{\Gamma}^{(n)}$ belong to $\mathcal{C}_r$. For any $1 \leq n \neq n' \leq N$ and $1 \leq r \neq r' \leq R$, assume that there exists a positive constant g such that*

$$\min_{\mathbf{X}} \frac{\|\mathbf{\Gamma}_r^{(n)} \mathbf{X} - \mathbf{X} (\mathbf{\Gamma}_{r'}^{(n')})^T\|_F}{\|X\|_F} \geq g. \qquad (9)$$

Denote $C = \sqrt{\frac{R^{N-2}(R-1)}{R^{N-1}-1}} \frac{\|(\mathbf{A}^{(1)})^{-1}\|_2 \dots \|(\mathbf{A}^{(N)})^{-1}\|_2}{g}$. Then $\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{S}} \times_1 \mathbf{A}^{(1)} \cdots \times_N \mathbf{A}^{(N)}$ with

$$\|\operatorname{offblkdiag}_\tau(\boldsymbol{\mathcal{S}})\|_F \leq C\delta.$$

Remark 3. a) *The parameter δ in* (8) *is a measurement for how well $(\mathbf{Z}^{(1)}, \dots, \mathbf{Z}^{(N)})$ approximately lies in $\mathscr{N}(\boldsymbol{\mathcal{T}})$. In particular, if $\delta = 0$, then $(\mathbf{Z}^{(1)}, \dots, \mathbf{Z}^{(N)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}})$.* b) *The parameter g in* (9) *is used to measure the gap between the eigenvalues of $\mathbf{\Gamma}_r^{(n)}$ and $\mathbf{\Gamma}_{r'}^{(n')}$. The larger g is, the better the eigenvalues separate.*

Remark 4. *Theorem 4 tells that the Frobenius norm of the τ-off-block diagonal part of $\boldsymbol{\mathcal{S}}$ is at the order of δ. If $\delta = 0$, $\boldsymbol{\mathcal{S}}$ will be exactly block diagonal. The constant C is determined by the number of diagonal blocks R, the condition number of the transformation matrix $\mathbf{A}^{(n)}$, and the separation of clusters g. This tells us that when cluster $\cup_{n=1}^{N} \lambda_d(\mathbf{Z}^{(n)})$, we should make sure that $\mathbf{A}^{(n)}$'s are good conditioned and the gap between different clusters should be large.*

5 Variants of BBTD

In this section, we briefly discuss two variants of BBTD, which can be solved similarly.

Partial Modes BBTD. The partial modes decomposition of a tensor $\boldsymbol{\mathcal{T}}$ has the following form:

$$\boldsymbol{\mathcal{T}} = \sum_{r=1}^{R} \boldsymbol{\mathcal{S}}_r \times_{n_1} \mathbf{A}_r^{(n_1)} \times_{n_2} \mathbf{A}_r^{(n_2)} \cdots \times_{n_\ell} \mathbf{A}_r^{(n_\ell)},$$

where $1 \leq n_1 < \cdots < n_\ell \leq N$. The problem can be solved by considering the following null space:

$$\mathscr{N}_p(\boldsymbol{\mathcal{T}}) \triangleq \{(\mathbf{Z}^{(n_1)}, \dots, \mathbf{Z}^{(n_\ell)}) \mid \boldsymbol{\mathcal{T}} \times_{n_1} (\mathbf{Z}^{(n_1)})^{\star} = \boldsymbol{\mathcal{T}} \times_{n_2} (\mathbf{Z}^{(n_2)})^T = \cdots = \boldsymbol{\mathcal{T}} \times_{n_\ell} (\mathbf{Z}^{(n_\ell)})^T\},$$

where $\star = T$ if $n_1 = 1$, is void otherwise.

Coupled Modes BBTD. Partition $\{1, \dots, N\}$ into L nonintersect subsets $\mathcal{N}_1, \dots, \mathcal{N}_L$. The coupled modes decomposition of a tensor $\boldsymbol{\mathcal{T}}$ has the form (3), where $\mathbf{A}^{(n_1)} = \mathbf{A}^{(n_2)}$ for any $n_1, n_2 \in \mathcal{N}_\ell$, $\forall 1 \leq \ell \leq L$. The problem can be solved by considering the subspace of $\mathscr{N}(\boldsymbol{\mathcal{T}})$:

$$\mathscr{N}_c(\boldsymbol{\mathcal{T}}) \triangleq \{(\mathbf{Z}^{(n_1)}, \dots, \mathbf{Z}^{(n_\ell)}) \in \mathscr{N}(\boldsymbol{\mathcal{T}}) \mid \mathbf{A}^{(n_1)} = \mathbf{A}^{(n_2)}, \forall n_1, n_2 \in \mathcal{N}_\ell, \forall 1 \leq \ell \leq L\}.$$

Not surprisingly, a partial mode BBTD can also be coupled, which can be treated similarly. For example, in Cai and Liu (2017); Cai and Li (2020a), a coupled modes 1 and 2 BBTD of third order tensor, which has many application in independent subspace analysis and semidefinite programming, is discussed with a different name, called general/blind matrix joint block diagonalization.

6 Numerical Experiment

We present several numerical examples to illustrate the performance of our method. All the numerical tests were carried out using MATLAB R2018a, with machine $\epsilon = 2.2 \times 10^{-16}$. We compare the performance of our algorithm with the nonlinear unconstrained optimization (MINF) and the nonlinear least squares algorithm (NLS) (Sorber, Barel, and Lathauwer 2012; Sorber, Van Barel, and De Lathauwer 2013, 2015), which are available in TENSORLAB 3.0 (available online at https://www.tensorlab.net).

We emphasize here that in order to perform optimization based methods, the block diagonal structure of the core tensor needs to be given, meanwhile, our method does not need such structure information.

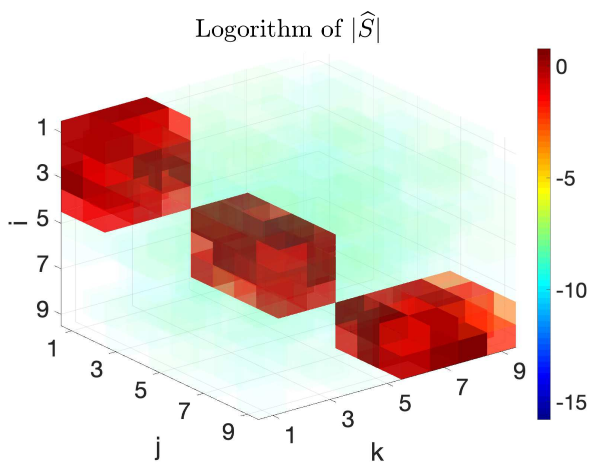

Figure 4: The revealed block diagonal structure

We generate our data tensor as

$$\mathcal{T} = \mathcal{T}_0 + \sigma\mathcal{N} = [\![\mathcal{S}; \mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}]\!] + \sigma\mathcal{N},$$

where $\mathcal{T}$, $\mathcal{T}_0$, $\mathcal{N}$ are the noisy tensor, noiseless tensor and noise tensor, respectively, $\mathbf{A}^{(n)}$'s are random matrices, $\mathcal{S}$ is a random τ-block diagonal tensor with $\tau = (M_r^{(n)})$, $\mathcal{N}$ is a random tensor, $\sigma \geq 0$ is parameter to control noise level. Here all random variables are *i.i.d.* from the standard normal distribution. We consider 5 settings, using $N = 3$, $\sigma = \eta\|\mathcal{T}_0\|_F/\|\mathcal{N}\|_F$, where $\eta = 10^{-3}$ for settings 1 and 2, $\eta = [10^{-6}, 10^{-3}]$ for the settings 3-5. The rest of the parameters are specified below.

1. $I = 20$, $\tau = \left(\begin{smallmatrix} 2 & 3 & 4 \\ 4 & 2 & 3 \\ 3 & 4 & 2 \end{smallmatrix}\right)$;
2. $I = 30$, $M_r^{(n)} \equiv 3$, $R = [2, 4, 6, 8, 10]$;
3. $I = 20$, $M_r^{(n)} \equiv 1, R = 6$;
4. $I = 20$, $\tau = \left(\begin{smallmatrix} 3 & 3 & 1 \\ 4 & 4 & 1 \\ 5 & 5 & 1 \end{smallmatrix}\right)$, $R = 3$;
5. $I = 20$, $\tau = \left(\begin{smallmatrix} 3 & 3 & 3 \\ 4 & 4 & 4 \\ 5 & 5 & 5 \end{smallmatrix}\right)$, $R = 3$.

We use setting 1 to illustrate our algorithm – SVDS4BBTD. We generate the data 10 times, the block diagonal structure, as shown in Figure 4, is successfully revealed 10 times, the average relative residual `err` $= \frac{\|\mathcal{T} - [\![\mathrm{diag}_\tau(\widehat{\mathcal{S}}); \widehat{\mathbf{A}}^{(1)}, \ldots, \widehat{\mathbf{A}}^{(N)}]\!]\|_F}{\|\mathcal{T}\|_F}$ is 9.85×10^{-4}, and the CPU time is 0.12 second.

Using setting 2, for each R, we perform our algorithm 20 times, the average CPU time and the rate of successful recoveries of the block diagonal structure are reported in Table 1. We can see that as R increases, the CPU time increases, and for $R = 2, 4, 6, 8$, the rate of successful recovery is 100%, for $R = 10$, the rate decreases to 40%.

In setting 3, $\mathcal{T}$ has a CPD with rank 6. The results are reported in Table 2. For different noise levels, our method is able to recover the block diagonal structure successfully, with residuals and CPU times comparable to MINF and NLS.

R	2	4	6	8	10
time (s)	0.08	0.32	1.44	6.47	24.8
rate (%)	100	100	100	100	40

Table 1: Results for setting 2

η	1e-6		1e-3	
	`err`	`time(s)`	`err`	`time(s)`
MINF	9.8e-6	0.14	9.8e-3	0.14
NLS	9.8e-6	0.14	9.8e-3	0.12
BBTD	9.9e-6	**0.08**	9.9e-3	**0.07**

Table 2: Results for setting 3 (10 times average)

η	1e-6		1e-3	
	`err`	`time(s)`	`err`	`time(s)`
MINF	6.4e-2	0.81	1.0e-1	0.78
NLS	8.8e-2	0.99	1.1e-1	0.89
BBTD	**1.7e-5**	**0.15**	**9.9e-4**	**0.14**

Table 3: Results for setting 4 (10 times average)

η	1e-6		1e-3	
	`err`	`time(s)`	`err`	`time(s)`
MINF	2.9e-1	2.40	3.2e-1	2.39
NLS	2.9e-1	26.4	2.6e-1	26.2
BBTD	**1.1e-6**	**0.34**	**1.1e-3**	**0.27**

Table 4: Results for setting 5 (10 times average)

In setting 4, $\mathcal{T}$ has a decomposition in rank $(L_r, L_r, 1)$ terms. The results are reported in Table 3. For different noise levels, our method recovers the block diagonal structure successfully, with much smaller residuals and less CPU times.

In setting 5, $\mathcal{T}$ has a τ-BTD. The results are reported in Table 4. For different noise levels, MINF and NLS fail to converge to the desired tolerance in 500 and 200 iterations, respectively. Our method recovers the block diagonal structure successfully, with significantly smaller residuals and times.

7 Concluding Remarks

In this paper, we propose a blind BTD for high order tensors. We characterize the necessary and sufficient conditions for the existence of BTD via $\mathscr{N}(\mathcal{T})$. Under a "low rank" assumption, we show that BBTD is unique. And it is shown that a solution to BBTD can be obtained from an arbitrary $(\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(N)}) \in \mathscr{N}(\mathcal{T})$ almost surely. An algebraic method is proposed to compute the decomposition, in which the problem of determination of the block diagonal structure of the core tensor is transformed into a problem of solving a clustering problem for the eigenvalues of $\mathbf{Z}^{(n)}$, in polynomial time. Numerical simulations show that our method not only is able to reveal the block diagonal structure, but also outperforms state-of-the-art optimization-based methods, in terms of the accuracy and also the efficiency.

Acknowledgments

The authors sincerely thank the anonymous reviewers. The first author was supported in part by NSFC-11671023.

References

Acar, E.; Kolda, T. G.; and Dunlavy, D. M. 2011. All-at-once optimization for coupled matrix and tensor factorizations. *arXiv:1105.3422 & KDD Workshop Min. Learn. Graphs* .

Acar, E.; and Yener, B. 2008. Unsupervised multiway data analysis: A literature survey. *IEEE Trans. Knowl. Data Eng.* 21(1): 6–20.

Anandkumar, A.; Ge, R.; Hsu, D.; Kakade, S. M.; and Telgarsky, M. 2014. Tensor decompositions for learning latent variable models. *J. Machine Learning Res.* 15(1): 2773–2832.

Baglama, J.; and Reichel, L. 2005. Augmented implicitly restarted Lanczos bidiagonalization methods. *SIAM J. Sci. Comput.* 27(1): 19–42.

Bai, Y.; de Klerk, E.; Pasechnik, D.; and Sotirov, R. 2009. Exploiting group symmetry in truss topology optimization. *Optim. Engrg.* 10(3): 331–349.

Balazevic, I.; Allen, C.; and Hospedales, T. M. 2019. TuckER: Tensor Factorization for Knowledge Graph Completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5184–5193. Hong Kong, China.

Ben-younes, H.; Cadène, R.; Thome, N.; and Cord, M. 2019. BLOCK: Bilinear Superdiagonal Fusion for Visual Question Answering and Visual Relationship Detection. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, 8102–8109. Honolulu, HI.

Cai, Y.; Cheng, G.; and Shi, D. 2019. Solving the general joint block diagonalization problem via linearly independent eigenvectors of a matrix polynomial. *Numer. Linear Algebra Appl.* 26(4): e2238.

Cai, Y.; and Li, P. 2020a. Identification of Matrix Joint Block Diagonalization. *arXiv:2011.01111* .

Cai, Y.; and Li, P. 2020b. Tensor Completion via Tensor Networks with a Tucker Wrapper. *arXiv:2010.15819* .

Cai, Y.; and Liu, C. 2017. An Algebraic Approach to Nonorthogonal General Joint Block Diagonalization. *SIAM J. Matrix Anal. Appl.* 38(1): 50–71.

Calvi, G. G.; Moniri, A.; Mahfouz, M.; Yu, Z.; Zhao, Q.; and Mandic, D. P. 2019. Tucker tensor layer in fully connected neural networks. *arXiv:1903.06133* .

Cardoso, J. 1998. Multidimensional independent component analysis. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1941–1944. Seattle, WA.

Carroll, J. D.; and Chang, J.-J. 1970. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika* 35(3): 283–319.

Cattell, R. B. 1952. The three basic factor-analytic research designs—their interrelations and derivatives. *Psych. Bull.* 49(5): 499.

Cichocki, A.; Mandic, D.; De Lathauwer, L.; Zhou, G.; Zhao, Q.; Caiafa, C.; and Phan, H. A. 2015. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *IEEE Signal Process. Mag.* 32(2): 145–163.

Cohen, N.; Sharir, O.; and Shashua, A. 2016. On the Expressive Power of Deep Learning: A Tensor Analysis. In *Proceedings of the 29th Conference on Learning Theory (COLT)*, 698–728. New York, NY.

De Klerk, E.; Dobre, C.; and Ṗasechnik, D. V. 2011. Numerical block diagonalization of matrix $*$-algebras with application to semidefinite programming. *Math. Program.* 129(1): 91–111.

De Klerk, E.; Pasechnik, D. V.; and Schrijver, A. 2007. Reduction of symmetric semidefinite programs using the regular $*$-representation. *Math. Program.* 109(2-3): 613–624.

De Klerk, E.; and Sotirov, R. 2010. Exploiting group symmetry in semidefinite programming relaxations of the quadratic assignment problem. *Math. Program.* 122(2): 225–246.

De Lathauwer, L. 2008a. Decompositions of a higher-order tensor in block terms-Part I: Lemmas for partitioned matrices. *SIAM J. Matrix Anal. Appl.* 30(3): 1022–1032.

De Lathauwer, L. 2008b. Decompositions of a higher-order tensor in block terms-Part II: Definitions and uniqueness. *SIAM J. Matrix Anal. Appl.* 30(3): 1033–1066.

De Lathauwer, L. 2011. Blind separation of exponential polynomials and the decomposition of a tensor in rank-(L_r,L_r,1) terms. *SIAM J. Matrix Anal. Appl.* 32(4): 1451–1474.

De Lathauwer, L.; De Moor, B.; and Vandewalle, J. 2000. Fetal electrocardiogram extraction by blind source subspace separation. *IEEE Trans. Biomedical Engrg.* 47(5): 567–572.

De Lathauwer, L.; and Nion, D. 2008. Decompositions of a higher-order tensor in block terms-part III: Alternating least squares algorithms. *SIAM J. Matrix Anal. Appl.* 30(3): 1067–1083.

Gatermann, K.; and Parrilo, P. A. 2004. Symmetry groups, semidefinite programs, and sums of squares. *J. Pure Appl. Algebra* 192(1): 95–128.

Grasedyck, L.; Kressner, D.; and Tobler, C. 2013. A literature survey of low-rank tensor approximation techniques. *GAMM-Mitteilungen* 36(1): 53–78.

Harshman, R. A. 1970. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics* 16: 1–84.

Håstad, J. 1990. Tensor rank is NP-complete. *Journal of Algorithms* 11(4): 644–654.

Hitchcock, F. L. 1928. Multiple invariants and generalized rank of a p-way matrix or tensor. *J. Math. Phys.* 7(1-4): 39–79.

Karatzoglou, A.; Amatriain, X.; Baltrunas, L.; and Oliver, N. 2010. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys)*, 79–86. Barcelona, Spain.

Kolda, T. G.; and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM Rev.* 51(3): 455–500.

Kruskal, J. B. 1977. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra Appl.* 18(2): 95–138.

Kruskal, J. B. 1989. Rank, decomposition, and uniqueness for 3-way and N-way arrays. *Multiway data analysis* 7–18.

Lebedev, V.; Ganin, Y.; Rakhuba, M.; Oseledets, I. V.; and Lempitsky, V. S. 2015. Speeding-up Convolutional Neural Networks Using Fine-tuned CP-Decomposition. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. San Diego, CA.

Li, G.; Ye, J.; Yang, H.; Chen, D.; Yan, S.; and Xu, Z. 2017. Bt-nets: Simplifying deep neural networks via block term decomposition. *arXiv:1712.05689* .

Lu, H.; Plataniotis, K. N.; and Venetsanopoulos, A. N. 2011. A survey of multilinear subspace learning for tensor data. *Pattern Recognition* 44(7): 1540–1551.

Maehara, T.; and Murota, K. 2011. Algorithm for error-controlled simultaneous block-diagonalization of matrices. *SIAM J. Matrix Anal. Appl.* 32(2): 605–620.

Nion, D. 2011. A tensor framework for nonunitary joint block diagonalization. *IEEE Trans. Signal Process.* 59(10): 4585–4594.

Novikov, A.; Podoprikhin, D.; Osokin, A.; and Vetrov, D. P. 2015. Tensorizing neural networks. In *Advances in neural information processing systems*, 442–450.

Papalexakis, E. E.; Faloutsos, C.; and Sidiropoulos, N. D. 2017. Tensors for data mining and data fusion: Models, applications, and scalable algorithms. *ACM Transactions on Intelligent Systems and Technology (TIST)* 8(2): 16.

Póczos, B.; and Lőrincz, A. 2005. Independent subspace analysis using k-nearest neighborhood distances. In *Artificial Neural Networks: Formal Models and Their Applications-ICANN 2005*, 163–168. Springer.

Rendle, S. 2010. Factorization Machines. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM)*, 995–1000. Sydney, Australia.

Sidiropoulos, N. D.; De Lathauwer, L.; Fu, X.; Huang, K.; Papalexakis, E. E.; and Faloutsos, C. 2017. Tensor decomposition for signal processing and machine learning. *IEEE Trans. Signal Process.* 65(13): 3551–3582.

Sorber, L.; Barel, M. V.; and Lathauwer, L. D. 2012. Unconstrained optimization of real functions in complex variables. *SIAM J. Optim.* 22(3): 879–898.

Sorber, L.; Van Barel, M.; and De Lathauwer, L. 2013. Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-(L_r,L_r,1) terms, and a new generalization. *SIAM J. Optim.* 23(2): 695–720.

Sorber, L.; Van Barel, M.; and De Lathauwer, L. 2015. Structured data fusion. *IEEE Journal of Selected Topics in Signal Processing* 9(4): 586–600.

Sørensen, M.; and De Lathauwer, L. 2015. Coupled Canonical Polyadic Decompositions and (Coupled) Decompositions in Multilinear Rank-(Lr,n,Lr,n,1) Terms–Part I: Uniqueness. *SIAM J. Matrix Anal. Appl.* 36(2): 496–522.

Stoudenmire, E. M.; and Schwab, D. J. 2016. Supervised Learning with Tensor Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 4799–4807. Barcelona, Spain.

Theis, F. J. 2005. Blind signal separation into groups of dependent signals using joint block diagonalization. In *International Symposium on Circuits and Systems (ISCAS)*, 5878–5881. Kobe, Japan.

Theis, F. J. 2006. Towards a general independent subspace analysis. In *Advances in Neural Information Processing Systems*, 1361–1368. MIT Press, Cambridge, MA.

Tjandra, A.; Sakti, S.; and Nakamura, S. 2017. Compressing recurrent neural network with tensor train. In *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN)*, 4451–4458. Anchorage, AK.

Tjandra, A.; Sakti, S.; and Nakamura, S. 2018. Tensor Decomposition for Compressing Recurrent Neural Network. In *Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN)*, 1–8. Rio de Janeiro, Brazil.

Tucker, L. R. 1966. Some mathematical notes on three-mode factor analysis. *Psychometrika* 31(3): 279–311.

Van Loan, C. F.; and Golub, G. H. 2012. *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD, 4th edition.

Wang, M.; and Zeng, Y. 2019. Multiway clustering via tensor block models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 713–723. Vancouver, Canada.

Yang, M. 2014. On partial and generic uniqueness of block term tensor decompositions. *Annali Dell'Universita'Di Ferrara* 60(2): 465–493.

Ye, J.; Wang, L.; Li, G.; Chen, D.; Zhe, S.; Chu, X.; and Xu, Z. 2018. Learning Compact Recurrent Neural Networks With Block-Term Tensor Decomposition. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 9378–9387. Salt Lake City, UT.