Fully-Connected Tensor Network Decomposition and Its Application to Higher-Order Tensor Completion

Yu-Bang Zheng, Ting-Zhu Huang, Xi-Le Zhao, Qibin Zhao, Tai-Xiang Jiang

1 School of Mathematical Sciences, University of Electronic Science and Technology of China, China
2 Tensor Learning Team, RIKEN Center for Advanced Intelligence Project (AIP), Japan
3 School of Automation, Guangdong University of Technology, China
4 School of Economic Information Engineering, Southwestern University of Finance and Economics, China

{zhengyubang, xzhao122003}@163.com, tingzhuhuang@126.com, qibin.zhao@riken.jp, taixiangjiang@gmail.com

Abstract

The popular tensor train (TT) and tensor ring (TR) decompositions have achieved promising results in science and engineering. However, TT and TR decompositions only establish an operation between adjacent two factors and are highly sensitive to the permutation of tensor modes, leading to an inadequate and inflexible representation. In this paper, we propose a generalized tensor decomposition, which decomposes an $N$th-order tensor into a set of $N$th-order factors and establishes an operation between any two factors. Since it can be graphically interpreted as a fully-connected network, we named it fully-connected tensor network (FCTN) decomposition. The superiorities of the FCTN decomposition lie in the outstanding capability for characterizing adequately the intrinsic correlations between any two modes of tensors and the essential invariance for transposition. Furthermore, we employ the FCTN decomposition to one representative task, i.e., tensor completion, and develop an efficient solving algorithm based on proximal alternating minimization. Theoretically, we prove the convergence of the developed algorithm, i.e., the sequence obtained by it globally converges to a critical point. Experimental results substantiate that the proposed method compares favorably to the state-of-the-art methods based on other tensor decompositions.

Introduction

The rapid advance in science and technology has given rise to the wide presence of higher-order data, e.g., multi-temporal, multi-spectral, and multi-scale data, which are usually expressed by higher-order tensors. Tensor decompositions focus on decomposing a higher-order tensor to a set of low-dimensional factors used to represent its latent features, which have powerful capability to capture the global correlations of tensors and have been widely applied in a variety of fields, such as signal processing, computer vision, and medical imaging (Kolda and Bader 2009; Mu et al. 2014; Anandkumar et al. 2014; Cong et al. 2015; Zhao, Zhang, and Cichocki 2015; Lu et al. 2016; Yokota and Hontani 2017; Yokota et al. 2019; Zheng et al. 2020). By designing different structures of latent factors and different multi-linear operations among them, various tensor decompositions have been proposed and attracted considerable attention. Among them, the Tucker decomposition and the CANDECOMP/PARAFAC (CP) decomposition as two most classical decompositions have achieved great success in the past decade (Kolda and Bader 2009; Gandy, Recht, and Yamada 2011; Liu et al. 2013, 2014; Zhao et al. 2016; Yokota, Zhao, and Cichocki 2016; Li, Ye, and Xu 2017; Xie et al. 2018; Yao et al. 2019; He et al. 2019; Phan et al. 2020).

More recently, an increasing number of tensor network-based tensor decompositions have emerged and shown great ability to deal with higher-order, especially beyond third-order, tensors. One of the most representative among them is the tensor train (TT) decomposition (Oseledets 2011), which decomposes an $N$th-order tensor into $N$-2 third-order tensors located at intermediate and two matrices located at both sides (see Figure 1(a)). Besides, from the first TT factor (matrix), each factor needs to conduct a multi-linear operation with its next factor, until the last one (matrix). Subsequently, as an extension of the TT decomposition, tensor ring (TR) decomposition (Zhao et al. 2016) replaced two matrices in TT factors by third-order tensors and established an additional multi-linear operation between them (see Figure 1(b)). Since TT and TR decompositions have the outstanding capability in super-compression and computational practicability, they have been employed in many applications, such as signal restoration, compression and reconstruction, and image/video recovery (Bengua et al. 2017; Imaizumi, Maehara, and Hayashi 2017; Ding et al. 2019; Zhao et al. 2019; Yuan et al. 2018; Yuan et al. 2019; Chen et al. 2020). Many of them can be regarded as a tensor completion (TC) problem, which aims to complete a tensor from its partial observation.

However, there are two limitations to TT and TR decompositions. First, these two decompositions only establish an operation/connection between adjacent two factors, rather than any two factors, which leads to a limited characterization for correlations of tensors. Second, TT decomposition keeps the invariance only when the modes of the target tensor make a reverse permuting, while TR decomposition keeps the invariance only when the modes of the target tensor make a circular shifting or a reverse permuting. These imply that these two decompositions are highly sensitive to the permutation of tensor modes, leading to the inflexibility of decompositions and applications.
To tackle the above two limitations, we propose a fully-connected tensor network (FCTN) decomposition, which decomposes an \( N \)-th-order tensor into a set of \( N \)-th-order factors and establishes a multi-linear operation/connection between any two factors (see Figure 1(c)). The proposed FCTN decomposition has the superior capability to characterize directly the intrinsic correlations between any two modes of tensors and is proved to be essentially invariant for any permutations of tensor modes. The main contributions of this paper are summarized as three-folds:

1) We propose an FCTN decomposition, which breaks through the limitations of TT and TR decompositions in terms of correlation characterization and transpositional invariance.

2) We employ the FCTN decomposition to the TC problem and develop an efficient proximal alternating minimization (PAM)-based algorithm to solve it.

3) We theoretically demonstrate the convergence of the developed algorithm by proving the sequence obtained by it globally converges to a critical point (local minima).

\section*{Notations and Preliminaries}

In this paper, we denote scalars, vectors, matrices, and tensors by \( x, \) \( x, \) \( X, \) and \( \mathcal{X}, \) respectively. For an \( N \)-th-order tensor \( \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}, \) we employ \( \mathcal{X}(i_1, i_2, \cdots, i_N) \) to denote its \((i_1, i_2, \cdots, i_N)\)th element. The Frobenius norm of \( \mathcal{X} \) is defined as \( \|\mathcal{X}\|_F = \sqrt{\sum_{i_1, i_2, \ldots, i_N} |\mathcal{X}(i_1, i_2, \cdots, i_N)|^2}. \) For the sake of clarity, we use \( \mathcal{X}_{i,d} \) to denote \((\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_d)\).

\section*{FCTN Decomposition}

\subsection*{Basic Theory}

Before proposing the FCTN decomposition, we first develop several basic definitions and theorems.

\begin{definition}[Generalized Tensor Transposition] \label{def:generalized_tensor_transposition}
Supposing that \( \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \) is an \( N \)-th-order tensor and the vector \( n \) is a reordering of the vector \((1, 2, \cdots, N)\). The vector \( n \)-based generalized tensor transposition of \( \mathcal{X} \) is defined as a tensor \( \overline{\mathcal{X}}^n \in \mathbb{R}^{I_{n_1} \times I_{n_2} \times \cdots \times I_{n_N}}, \) which is generated by rearranging the modes of \( \mathcal{X} \) in the order specified by the vector \( n \). We denote the corresponding operation and its inverse operation by \( \overline{\mathcal{X}}^n = \text{permute}(\mathcal{X}, n) \) and \( \mathcal{X} = \text{ipermute}(\overline{\mathcal{X}}^n, n), \) respectively.
\end{definition}

\begin{definition}[Generalized Tensor Unfolding] \label{def:generalized_tensor_unfolding}
Supposing that \( \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \) is an \( N \)-th-order tensor and the vector \( n \) is a reordering of the vector \((1, 2, \cdots, N)\). The generalized tensor unfolding of \( \mathcal{X} \) is defined as a matrix
\[
\mathcal{X} = \text{reshape}^l(I_{i_1}, \prod_{i=d+1}^{N} I_{i_1}, \prod_{i=1}^{d} I_{i_1}^d).
\]
We denote the corresponding operation and its inverse operation by \( \mathcal{X} = \text{GenUnfold}(\mathcal{X}, n_{d+1:N}) \) and \( \mathcal{X} = \text{GenFold}(\mathcal{X}, n_{d+1:N}), \) respectively.
\end{definition}

For example, the traditional mode-\( k \) unfolding (Kolda and Bader 2009) of \( \mathcal{X} \) is \( \mathcal{X}_{[k;1,\cdots,k-1,k+1,\cdots,N]} \), which is also simply denoted by \( \mathcal{X}(k) \).

\begin{definition}[Tensor Contraction] \label{def:tensor_contraction}
Supposing that vectors \( n \) and \( m \) are the reordering of vectors \((1, 2, \cdots, N)\) and \((1, 2, \cdots, M)\), respectively; \( \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \) and \( \mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M} \) are two tensors satisfied \( I_{n_i} = J_{m_i} \) with \( i = 1, 2, \cdots, d \). The tensor contraction along the \( n_{1:d} \)-th-modes of \( \mathcal{X} \) and the \( m_{1:d} \)-th-modes of \( \mathcal{Y} \) yields an \((N+M-2d)\)-th-order tensor
\[
\mathcal{Z} = \mathcal{X} \times^{m_{1:d}} \mathcal{Y} \in \mathbb{R}^{I_{n_{d+1}} \times \cdots \times I_{n_N} \times J_{m_{d+1}} \times \cdots \times J_{m_M}},
\]
whose elements
\[
\mathcal{Z} = \{ I_{n_{d+1}}, \cdots, I_{n_N}, I_{m_{d+1}}, \cdots, J_{m_M} \} = \sum_{i_{n_1}=1}^{I_{n_1}} \sum_{i_{n_2}=1}^{I_{n_2}} \cdots \sum_{i_{n_d}=1}^{I_{n_d}} \{ \overline{\mathcal{X}}^n(i_{n_1}, \cdots, i_{n_d}, i_{m_{d+1}}, \cdots, i_{m_N}) \} = \sum_{i_{n_1}=1}^{I_{n_1}} \sum_{i_{n_2}=1}^{I_{n_2}} \cdots \sum_{i_{n_d}=1}^{I_{n_d}} \{ \overline{\mathcal{X}}^n(i_{n_1}, \cdots, i_{n_d}, i_{m_{d+1}}, \cdots, i_{m_N}) \}.
\]
\end{definition}

\footnote{Matlab commands.}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{A graphical representation of TT, TR, and the proposed FCTN decompositions.}
\end{figure}
Furthermore, Theorem 1 delivers the relationship of the tensor contraction and the matrix multiplication.

**Theorem 1** Supposing that $X \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and $Y \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_M}$ are two tensors, we have

1) $X_{[n_1:d, n_{d+1:N}]^T} = X_{[n_{d+1:N}; n_{1:d}]}$;
2) $Z \leftrightarrow X^T \times_{n_{d+1:M}} Y$

Where vectors $n$ and $m$ have the same setting as in Definition 3.

**FCTN Decomposition**

**Definition 4 (FCTN Decomposition)** The FCTN decomposition aims to decompose an $N$th-order tensor $X \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ into a set of $N$-th order factor tensors $G_k \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, $k = 1, 2, \ldots, N$. More specifically, the element-wise form of the FCTN decomposition can be expressed as

$$X_{(i_1, i_2, \ldots, i_N)} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \cdots \sum_{r_{N-1}=1}^{R_{N-1}} \sum_{r_{N} = 1}^{R_N} \{G_1(i_1, r_2, \ldots, r_N), G_2(r_1, r_2, \ldots, r_N), \ldots, G_N(r_1, r_2, \ldots, r_{N-1}, i_N)\}$$

Moreover, denote the FCTN decomposition by $X = FCTN(G_1, G_2, \ldots, G_N)$ and call the vector collected by $R_{k,1} = 1 \leq k_1 < k_2 \leq N$ and $k_1, k_2 \in \mathbb{N}^+$ as the FCTN-ranks.

To illustrate the FCTN decomposition vividly, Figure 1(c) gives a graphical representation of it. It is not hard to see that for second-order tensors, the FCTN decomposition is actually the matrix factorization and the FCTN-rank is actually the matrix rank. Furthermore, for higher-order tensors, any two FCTN factors $G_k$ and $G_k$ have an equal-sized mode $R_{k,1}, k \geq 1$ used to conduct the tensor contraction operation, which enables the FCTN decomposition to characterize adequately the intrinsic correlations between any two modes of the target tensor. This indicates an essential advantage of the FCTN decomposition over the TT and TR decompositions, which establish only the connection between adjacent two factors, leading to a limited characterization for correlations of tensors. Besides, the FCTN decomposition can degenerate to the TT and TR decompositions by simply setting the corresponding modes of factors to 1.

In second-order case, it is well known that the matrix factorization is essentially invariable under the transpositional condition, i.e., $X = G_1G_2 \Leftrightarrow X^T = G_2^T G_1^T$. Naturally, it is expected to extend this property to higher-order tensors.

**Theorem 2 (Transpositional Invariance)** Supposing that an $N$th-order tensor $X$ has the following FCTN decomposition: $X = FCTN(G_1, G_2, \ldots, G_N)$. Then, its vector $n$-based generalized tensor transposition $X^n$ can be expressed as

$X^n = FCTN(G_{n_1}^n, G_{n_2}^n, \ldots, G_{n_N}^n)$, where $n = (n_1, n_2, \ldots, n_N)$ is a reordering of the vector $(1, 2, \ldots, N)$.

Theorem 2 illustrates another essential advantage of the FCTN decomposition as compared with the TT and TR decompositions. More specifically, the FCTN decomposition is essentially invariable, no matter how to permute the modes of the target tensor. But TR decomposition keeps the invariance only when the modes of the target tensor make a circular shifting or a reverse permuting. And TT decomposition keeps the invariance only when the modes of the target tensor make a reverse permuting.

The following theorem presents that the FCTN-ranks can bound the rank of all generalized tensor unfolding.

**Theorem 3** Supposing that an $N$th-order tensor $X$ can be represented by Equation (1), the following inequality holds:

$$\text{Rank}(X_{[n_1:d, n_{d+1:N}]}^n) \leq \prod_{i=1}^{d} \prod_{j=N}^{d+1} R_{n_i, n_j},$$

where $R_{n_i, n_j} = R_{n_j, n_i}$ if $n_i > n_j$ and $(n_1, n_2, \ldots, n_N)$ is a reordering of the vector $(1, 2, \ldots, N)$.

Since the FCTN decomposition aims to characterize the intrinsic correlations between any two modes by establishing a connection between any two factors, the factors have to be designed as $N$th-order tensors, which inevitably leads to the increment of the storage cost as compared to TT and TR decompositions. For an $N$th-order $X \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ whose FCTN-ranks are the same value $R_{k,1}$, the FCTN decomposition requires $O(NIR_{k,1}^{N-1})$ parameters to express it. It seems to stay on the same order of magnitude with that of the Tucker decomposition ($O(NIR_2 + R_N^N)$ parameters). But when we express real-world data, the required FCTN-rank $R_{k,1}$ is usually far less than Tucker-rank $R_2$, because the FCTN decomposition uses $R_{k,1}^{N-1}$ to bound Tucker-rank $R_2$ (as shown in Theorem 3). This indicates that the FCTN decomposition is superior to the Tucker decomposition regarding the storage cost.

**Definition 5 (FCTN Composition)** We call the process of generating $X$ by its FCTN factors $G_k$ ($k = 1, 2, \ldots, N$) as the FCTN composition, which is also denoted by $FCTN(G_k)_{k=1}^N$. Furthermore, if one of the factors $G_i$ ($i \in \{1, 2, \ldots, N\}$) does not participate in the composition, we denote it by $FCTN(G_k)_{k=1}^N / G_i$.

**Theorem 4** Supposing that $X = FCTN(G_k)_{k=1}^N$ and $M_i = FCTN(G_k)_{k=1}^N / G_i$, we obtain that

$$X_{(i)} = (G_i)^{(t)}(M_i)_{[m_1:N-1, m_{N-1}]}$$

where $m_i = \begin{cases} 2i, & \text{if } i < t, \\ 2i - 1, & \text{if } i \geq t, \end{cases}$ and $n_i = \begin{cases} 2i - 1, & \text{if } i < t, \\ 2i, & \text{if } i \geq t. \end{cases}$

Theorem 4 reveals the relationship between one FCTN factor and the composition of the other factors. It is of great importance to the computation of the FCTN decomposition since computing one factor usually needs to fix the others.

---

*More experimental evidence is provided in the supplementary material, which is available at https://yubangzheng.github.io.*
FCTN Decomposition-Based TC Method

Model and Solving Algorithm

Due to space limitations, we only apply the FCTN decomposition to one representative task, i.e., TC, which aims to recover missing elements of a higher-order tensor from its incomplete observation. Given an incomplete observation \( \mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \) of the target tensor \( \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \), the proposed FCTN decomposition-based TC (FCTN-TC) model can be formulated as

\[
\min_{\mathcal{X}, \mathcal{G}} \frac{1}{2} \| \mathcal{X} - \text{FCTN} (\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_N) \|^2_F + \iota_\Omega (\mathcal{X}), \tag{2}
\]

where \( \mathcal{G} = (\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_N) \) and

\[\iota_\Omega (\mathcal{X}) := \begin{cases} 0, & \text{if } \mathcal{X} \in \mathcal{S} \setminus \Omega, \\ \infty, & \text{otherwise}, \end{cases}\]

with \( \mathcal{S} := \{ \mathcal{X} : \mathcal{P}_\Omega (\mathcal{X} - \mathcal{F}) = 0 \} \).

Here \( \Omega \) denotes the index of the known elements and \( \mathcal{P}_\Omega (\mathcal{X}) \) is a projection operator which projects the elements in \( \Omega \) to themselves and all others to zeros.

Since all optimization variables are coupled with each other, we employ the framework of PAM (Attouch, Bolte, and Svaiter 2013) to solve (2), whose solution can be obtained by alternately updating

\[
\begin{align*}
G_k^{(s+1)} &= \arg \min_{G_k} f(G_{1:k-1}^{(s)}, G_k, G_{k+1:N}^{(s)}, X^{(s)}) \\
&+ \frac{\rho}{2} \| G_k - G_k^{(s)} \|^2_F, \quad k = 1, 2, \cdots, N, \tag{3}
\end{align*}
\]

where \( f(\mathcal{G}, \mathcal{X}) \) is the objective function of (2) and \( \rho > 0 \) is a proximal parameter.

1) Update \( \mathcal{G}_k \): According to Theorem 4, the \( \mathcal{G}_k \) \((k = 1, 2, \cdots, N)\)-subproblems can be rewritten as

\[
G_k^{(s+1)} = \arg \min_{G_k} \frac{\rho}{2} \| G_k - G_k^{(s)} \|^2_F + \frac{1}{2} \| X_k^{(s)} - (M_k^{(s)})^{(m_1:N-1;m_1:N-1)} \|^2_F \tag{4}
\]

where \( M_k^{(s)} = \text{FCTN}(G_{1:k-1}^{(s)}, G_k, G_{k+1:N}^{(s)}, \mathcal{G}_k) \) and vectors \( m \) and \( n \) have the same setting as in Theorem 4. The problem (4) can be directly solved as

\[
G_k^{(s+1)} = (M_k^{(s)})^{(m_1:N-1;m_1:N-1)} + \rho \mathcal{G}_k^{(s)}
\]

and \( \mathcal{G}_k^{(s+1)} = \text{GenFold}(G_k^{(s+1)}), k = 1, \cdots, k - 1, k + 1, \cdots, N \).

2) Update \( \mathcal{X} \): The \( \mathcal{X} \)-subproblem has the following closed-form solution since it is a least square problem:

\[
\mathcal{X}^{(s+1)} = \mathcal{P}_\Omega \left( \frac{\text{FCTN}(G_k^{(s+1)}, X_k^{(s)})}{1 + \rho} + \mathcal{P}_\Omega (\mathcal{F}) \right) \tag{6}
\]

Algorithm 1 PAM-Based Solver for the FCTN-TC Model.

1: Input: The incomplete tensor \( \mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N} \), the index \( \Omega \), the maximal FCTN-rank \( R_{\text{max}} \), and \( \rho = 0.1 \).

2: Initialization: \( s = 0 \), \( s_{\text{max}} = 1000 \), \( \mathcal{X}^{(0)} = \mathcal{F} \), the initial FCTN-rank \( R = \max\{\text{ones}(N(N-1)/2, 1), R_{\text{max}} - 5\} \), and \( G_k^{(0)} = \text{rand}(1, k, R_{\text{max}} - 5) \), where \( k = 1, 2, \cdots, N \).

3: while not converged and \( s < s_{\text{max}} \) do

4: Update \( G_k^{(s+1)} \) via (5).

5: Update \( \mathcal{X}^{(s+1)} \) via (6).

6: Let \( R = \min\{R + 1, R_{\text{max}}\} \) and expand \( G_k^{(s+1)} \) if \( \| \mathcal{X}^{(s+1)} - \mathcal{X}^{(s)} \|_F / \| \mathcal{X}^{(s)} \|_F < 10^{-2} \).

7: Check the convergence condition:

\[\| \mathcal{X}^{(s+1)} - \mathcal{X}^{(s)} \|_F / \| \mathcal{X}^{(s)} \|_F < 10^{-5} \]

8: Let \( s = s + 1 \).

9: end while

10: Output: The reconstructed tensor \( \mathcal{X} \).

The whole process of the PAM-based solver for the FCTN-TC model is summarized in Algorithm 1. Especially, if the observed tensor \( \mathcal{F} \) is complete (no missing elements), the way for iteratively solving the factors \( G_k \) in the proposed FCTN-TC method can be regarded as a strategy for obtaining its one FCTN decomposition.

Computational Complexity Analysis

For an \( N \)-th order incomplete tensor \( \mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_3} \), we analyze the computational complexity of the proposed FCTN-TC method by simply setting the FCTN-ranks \( R_{k_1, k_2} \) \((1 \leq k_1 < k_2 \leq N) \) as the same value \( R \). The computational cost lies on two part: 1) updating \( G_k \) \((k = 1, 2, \cdots, N) \) and 2) updating \( \mathcal{X} \). In (5), updating \( G_k \) involves the FCTN composition, the matrix multiplication, and the matrix inversion, which costs \( O(N \sum_{k=2}^{N} I^k R^{(N-k)+k-1} + N R^2(N-1) + N R^3(N-1)) \). In (6), updating \( \mathcal{X} \) requires the FCTN composition costing \( O(\sum_{k=2}^{N} I^k R^2(N-k)+k-1) \). Therefore, the whole computational complexity at each iteration in the Algorithm 1 is \( O(N \sum_{k=2}^{N} I^k R^{(N-k)+k-1} + N R^2(N-1) + N R^3(N-1)) \).

Convergence Analysis

In this section, we provide a theoretical guarantee for the convergence of the developed PAM-based algorithm.

Theorem 5 The sequence \( \{G(s), X(s)\}_{s \in \mathbb{N}} \) obtained by the Algorithm 1 globally converges to a critical point of (2).

To prove the Theorem 5, we only need to justify that the following four conditions hold (Attouch, Bolte, and Svaiter 2013):

1) \( G(s) \) and \( X(s) (s \in \mathbb{N}) \) are bounded;
2) \( f(G, X) \) is a proper lower semi-continuous function;
3) \( f(G, X) \) satisfies the K-L property at \( \{G^{(s)}, X^{(s)}\}_{s \in N} \);
4) \( \{G^{(s)}, X^{(s)}\}_{s \in N} \) satisfies Lemmas 1 and 2.

**Lemma 1 (Sufficient Decrease)** Let \( \{G^{(s)}, X^{(s)}\}_{s \in N} \) be the sequence obtained by the Algorithm 1, then it satisfies
\[
\frac{f(G^{(s+1)}\mid_{k}, G^{(s+1)}\mid_{k+1:N}, X^{(s)})}{2} + \frac{\rho}{2} \|G^{(s+1)}\mid_{k} - G^{(s)}\mid_{k}\|^2_F \\
\leq \frac{f(G\mid_{k}, X^{(s)})}{2} + \|X^{(s)} - X^{(s)}\|^2_F.
\]

**Lemma 2 (Relative Error)** Letting \( \{G^{(s)}, X^{(s)}\}_{s \in N} \) be the sequence obtained by the Algorithm 1, then there exists \( A^{(s+1)} \) such that \( A^{(s+1)} + \nabla h(G^{(s)}\mid_{k}, X^{(s)}) \) satisfies
\[
\|A^{(s+1)} + \nabla h(G^{(s)}\mid_{k}, X^{(s)})\|^2_F \\
\leq \rho \|G^{(s+1)}\mid_{N} - G^{(s)}\mid_{N}\|^2_F, \quad k = 1, 2, \ldots, N;
\]
\[
\|A^{(s+1)} + \nabla h(G^{(s)}\mid_{k}, X^{(s)})\|^2_F \\
\leq \rho \|X^{(s)} - X^{(s)}\|^2_F,
\]
where \( h(G, X) = \frac{1}{2} \|X - FCTN(G^{(s)}\mid_{k=1:N})\|^2_F \).

The detailed proof for the above four conditions is presented in the supplementary materials.

**Numerical Experiments**

We test the performance of the proposed FCTN-TC method \(^3\) by conducting synthetic data and real data experiments. The missing ratio (MR) is defined as the ratio of the number of missing elements and the total elements.

**Synthetic Data Experiments**

This section mainly aims to verify the superiorities of the proposed FCTN decomposition over the TT and TR decompositions by contrasting the performance of their corresponding TC methods, i.e., FCTN-TC, TT-TC, and TR-TC. All methods are solved by PAM to get rid of the influence of the algorithm. Since TT decomposition is a special case of TR and FCTN decompositions, we generate

---

\(^3\)The code is available at https://yubangzheng.github.io.

---

<table>
<thead>
<tr>
<th>Dataset</th>
<th>MR</th>
<th>95%</th>
<th>90%</th>
<th>80%</th>
<th>Mean time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observed</td>
<td>8.7149</td>
<td>8.9503</td>
<td>9.4607</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>14.490</td>
<td>18.507</td>
<td>22.460</td>
<td>36.738</td>
<td></td>
</tr>
<tr>
<td>TMac</td>
<td>25.092</td>
<td>27.035</td>
<td>29.778</td>
<td>91.114</td>
<td></td>
</tr>
<tr>
<td>t-SVD</td>
<td>25.070</td>
<td>28.130</td>
<td>31.402</td>
<td>74.807</td>
<td></td>
</tr>
<tr>
<td>TMacTT</td>
<td>24.699</td>
<td>27.492</td>
<td>31.546</td>
<td>465.75</td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>22.558</td>
<td>27.823</td>
<td>31.447</td>
<td>891.96</td>
<td></td>
</tr>
<tr>
<td>FCTN-TC</td>
<td>26.392</td>
<td>29.523</td>
<td>33.048</td>
<td>473.50</td>
<td></td>
</tr>
<tr>
<td>Observed</td>
<td>4.5969</td>
<td>4.8315</td>
<td>5.3421</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>18.617</td>
<td>21.556</td>
<td>25.191</td>
<td>34.528</td>
<td></td>
</tr>
<tr>
<td>TMac</td>
<td>26.941</td>
<td>26.142</td>
<td>32.533</td>
<td>1224.4</td>
<td></td>
</tr>
<tr>
<td>t-SVD</td>
<td>28.814</td>
<td>34.912</td>
<td>39.722</td>
<td>71.510</td>
<td></td>
</tr>
<tr>
<td>TMacTT</td>
<td>28.131</td>
<td>31.282</td>
<td>37.088</td>
<td>450.70</td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>30.631</td>
<td>32.512</td>
<td>38.324</td>
<td>640.41</td>
<td></td>
</tr>
<tr>
<td>FCTN-TC</td>
<td>30.806</td>
<td>37.326</td>
<td>42.974</td>
<td>412.72</td>
<td></td>
</tr>
<tr>
<td>Observed</td>
<td>3.8499</td>
<td>4.0847</td>
<td>4.5946</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>16.651</td>
<td>20.334</td>
<td>24.813</td>
<td>38.541</td>
<td></td>
</tr>
<tr>
<td>TMac</td>
<td>26.753</td>
<td>28.648</td>
<td>31.010</td>
<td>500.70</td>
<td></td>
</tr>
<tr>
<td>t-SVD</td>
<td>21.810</td>
<td>27.252</td>
<td>30.975</td>
<td>63.994</td>
<td></td>
</tr>
<tr>
<td>TMacTT</td>
<td>25.918</td>
<td>28.880</td>
<td>32.232</td>
<td>204.64</td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>27.120</td>
<td>28.361</td>
<td>32.133</td>
<td>592.13</td>
<td></td>
</tr>
<tr>
<td>FCTN-TC</td>
<td>27.780</td>
<td>30.835</td>
<td>34.391</td>
<td>455.71</td>
<td></td>
</tr>
<tr>
<td>Observed</td>
<td>6.4291</td>
<td>6.6638</td>
<td>7.1736</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>14.561</td>
<td>19.128</td>
<td>23.396</td>
<td>32.882</td>
<td></td>
</tr>
<tr>
<td>TMac</td>
<td>25.464</td>
<td>28.169</td>
<td>30.525</td>
<td>779.78</td>
<td></td>
</tr>
<tr>
<td>t-SVD</td>
<td>21.552</td>
<td>26.094</td>
<td>30.344</td>
<td>66.294</td>
<td></td>
</tr>
<tr>
<td>TMacTT</td>
<td>26.252</td>
<td>29.512</td>
<td>33.096</td>
<td>264.15</td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>27.749</td>
<td>29.034</td>
<td>33.224</td>
<td>652.03</td>
<td></td>
</tr>
<tr>
<td>FCTN-TC</td>
<td>28.337</td>
<td>32.230</td>
<td>36.135</td>
<td>468.25</td>
<td></td>
</tr>
<tr>
<td>Observed</td>
<td>8.4157</td>
<td>8.6506</td>
<td>9.1594</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>12.008</td>
<td>22.140</td>
<td>29.000</td>
<td>15.456</td>
<td></td>
</tr>
<tr>
<td>TMac</td>
<td>31.151</td>
<td>36.749</td>
<td>41.663</td>
<td>93.065</td>
<td></td>
</tr>
<tr>
<td>t-SVD</td>
<td>35.328</td>
<td>40.379</td>
<td>45.801</td>
<td>30.533</td>
<td></td>
</tr>
<tr>
<td>TMacTT</td>
<td>36.478</td>
<td>41.561</td>
<td>46.581</td>
<td>131.31</td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>37.352</td>
<td>41.794</td>
<td>46.211</td>
<td>4776.5</td>
<td></td>
</tr>
<tr>
<td>FCTN-TC</td>
<td>43.256</td>
<td>48.175</td>
<td>52.701</td>
<td>501.03</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: The PSNR values and the running times of all utilized methods on the CV and HSV datasets.
Real Data Experiments

This section mainly aims to test the performance of the proposed FCTN-TC method on two types of real data by contrasting it with the state-of-the-art methods based on different tensor decompositions, including HaLRTC (Liu et al. 2013), TMac (Xu et al. 2015), t-SVD (Zhang and Aeron 2017), TMacTT (Bengua et al. 2017), and TRLRF (Yuan et al. 2019). All hyper-parameters involved in all compared methods are manually adjusted to achieve optimal performance following the authors’ recommendations and codes.

Figure 2 shows the reconstructed results of different methods on the synthetic data under different MRs, where the RSE value under each case is averaged over the values obtained by 50 independent tests. We observe that 1) the performance of the proposed FCTN-TC method is pronouncedly robust to different permutations, while that of TT-TC and TR-TC are sensitive; and 2) the proposed FCTN-TC method always achieves the lowest RSE values among three methods under different datasets, MRs, and permutations. This is because the rearranging of tensor modes shifts the correlation among them (e.g., the correlation between the first and second modes shifts to the first and third modes), leading to the change in the performance of TT and TR decomposition-based methods. But owing that the proposed FCTN decomposition can characterize the correlations between any two modes and is essentially invariant for tensor transposition, the FCTN-TC method obtains the best and robust results. These testing results provide empirical evidence for the fore theoretical analysis regarding the superiorities of the FCTN decomposition.

For instance, in TMac, TMacTT, and TRLRF, the hyper-parameters are mainly Tucker-rank, TT-rank, and TR-rank. We adjust them in a certain range. In HaLRTC and t-SVD, the hyper-parameters are mainly the threshold for the singular value thresholding operation, which is selected from the candidate set \( \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}\).

Video Data Completion. The testing video dataset includes four color videos\(^4\) (CVs) of size 144 x 176 x 3 x 50 (spatial height x spatial width x color channel x frame) and a hyperspectral video\(^5\) (HSV) of size 60 x 60 x 2 x 20 x 20 (spatial height x spatial width x band x frame) (Mian and Hartley 2012). For each data, we test three MRs: 80%, 90%, and 95%, and employ the peak signal-to-noise ratio (PSNR) as the quantitative metric. The methodology for selecting the missing elements is purely random sampling and the pixel values of the testing data are normalized into \([0, 1]\).

For the proposed FCTN-TC method, on CV dataset, we set \(R_{1,4}^{\text{max}}\) and \(R_{2,4}^{\text{max}}\) as the same value since they all directly characterize the correlation between the spatial modes (height and width, respectively) and the temporal mode. And we set \(R_{1,3}^{\text{max}}, R_{2,3}^{\text{max}}, \) and \(R_{3,4}^{\text{max}}\) as the same value, since the third mode represents the color channel. Therefore, only \(R_{1,2}^{\text{max}}, R_{1,3}^{\text{max}}, \) and \(R_{1,4}^{\text{max}}\) need to adjust, which are recommended to select from the candidate sets \( \{10, 15, 20, 25, 30, 35, 40, 45, 50\}\), \( \{2, 3\}\), and \( \{4, 5, 6, 7, 8\}\), respectively. On the HSV dataset, we simply set \(R_{k_1,k_2}^{\text{max}}\) (1 \( \leq k_1 < k_2 \leq 4\) and \( k_1, k_2 \in \mathbb{N}^+\)) as the same value recommended to select from the candidate set \( \{4, 5, 6, 7, 8\}\).

\(^4\)The data is available at http://trace.eas.asu.edu/yuv/.
\(^5\)The data is available at http://openremotesensing.net/kb/data/.

Figure 3: Reconstructed results on two testing CVs with MR=90%. From top to bottom: the odd-numbered rows are the visual results at the 1st frame of the CV containe and the 35th frame of the CV bunny, respectively; the even-numbered rows are the corresponding residual images average over three color channels.
Figure 4: Reconstructed results on the traffic flow dataset with MR=40%. The first and the second rows are the results on the 2nd day and the RSE value of the whole dataset. As observed, the proposed FCTN-TC method achieves the best approximation to the ground truth among different methods. This illustrates the superior of the proposed FCTN decomposition on the slice missing problem.

In summary, the above experimental results imply the superior capability of the proposed FCTN decomposition for capturing the intrinsic information of higher-order tensors, as compared to other tensor decompositions.

Table 1 reports the PSNR values and the running times of all utilized methods under different MRs on the testing CVs and HSV. In terms of effect (PSNR), we see that the proposed FCTN-TC method compares favorably to the other methods on both the testing CVs and HSV. Especially on the HSV, the FCTN-TC method is ahead by about 5dB of PSNR compared to the second-best method. In terms of efficiency (running time), we observe that on the CV dataset, the mean of the running time of the FCTN-TC method stays roughly on the same order of magnitude with that of the compared decomposition-based methods, i.e., TMac, TMacTT, and TRLRF. The main reason is that although the computational complexity of the FCTN-TC method is theoretically higher, the setting rank of it for obtaining the optimal result is usually lower than that of the compared ones.

Furthermore, Figure 3 shows the reconstructed spatial images and their corresponding residual images (the absolute difference between the reconstructed image and the ground truth) of two CVs. From Figure 3, we observe that the results obtained by the proposed FCTN-TC method are markedly superior to those by the compared ones, especially for the recovery of local details, such as ripples in the CV containe and grasses in the CV bunny.

Traffic Data Completion. This experiment was realized with traffic data provided by the NeCS team from the Grenoble Traffic Lab (GTL). The testing traffic flow dataset is collected from 19 road segments within 31 days from January 1, 2019, to January 31, 2019, and the time interval is 2 minutes. The size of it is $30 \times 24 \times 31 \times 19$ (minute\times\text{hour}\times\text{day}\times\text{segment}). In the traffic dataset, the damage to detectors usually leads to the data missing over some time. Therefore, we consider the slice (made up of the first and second modes) missing problem. The MR is set to be 40% and the RSE is employed as the quantitative metric. We simply set $F_{k_1, k_2}^\text{max}$ ($1 \leq k_1 < k_2 \leq 4$ and $k_1, k_2 \in \mathbb{N}^+$) as the same value, i.e., 5.

In Figure 4, we present the reconstructed results on the 2nd day and the corresponding residual results, respectively.

---

6Homepage: http://gtl.inrialpes.fr/.

Acknowledgments

This work was supported by NSFC (Nos. 61772003, 61876203, 62071132, 12001446), Key Project of Applied Basic Research in Sichuan Province (No. 2020J0216), Project of Applied Basic Research in Sichuan Province (No. 21YYJC3042), National Key Research and Development Program of China (No. 2020YFA0714001), and JSPS KAKENHI (No. 20H04249). The authors thank the CNRS/INRIA NeCS team for providing traffic data from the GTL.