SIAM J. MATRIX ANAL. APPL.
Vol. 42, No. 1, pp. 299–329

TRIPLE DECOMPOSITION AND TENSOR RECOVERY OF THIRD ORDER TENSORS*

LIQUN QI†, YANNAN CHEN‡, MAYANK BAKSHI§, AND XINZHEN ZHANG¶

Abstract. Motivated by the Tucker decomposition, in this paper we introduce a new tensor decomposition for third order tensors, which decomposes a third order tensor to three third order factor tensors. Each factor tensor has two low dimensions. We call such a decomposition the triple decomposition, and the corresponding rank the triple rank. The triple rank of a third order tensor is not greater than the middle value of the Tucker rank. The number of parameters in the bilevel form of standard triple decomposition is less than the number of parameters of Tucker decomposition in substantial cases. The theoretical discovery is confirmed numerically. Numerical tests show that third order tensor data from practical applications such as internet traffic and image are of low triple ranks. A tensor recovery method based on low rank triple decomposition is proposed. Its convergence and convergence rate are established. Numerical experiments confirm the efficiency of this method.

Key words. Tucker decomposition, triple decomposition, tensor recovery, Tucker rank, triple rank

AMS subject classifications. 15A69, 15A83

DOI. 10.1137/20M1323266

1. Introduction.

Higher order tensors have found many applications in recent years. Third order tensors are very common and useful higher order tensors in applications [1, 11, 16, 17, 21, 22]. Tensor decomposition has emerged as a valuable tool for analyzing and computing such tensors [12]. For example, a key idea behind tensor recovery algorithms is that many practical datasets are highly structured in the sense that the corresponding tensors can be approximately represented through a low rank decomposition.

The two most well-known tensor decompositions are the Tucker decomposition and the CANDECOMP/PARAFAC (CP) decomposition [12]. Their corresponding ranks are called CP rank and Tucker rank [10], respectively.

Suppose that we have a third order tensor $\mathcal{X} \in \Re^{n_1 \times n_2 \times n_3}$, where n_1, n_2, and n_3 are positive integers. The CP rank of $\mathcal{X}$ may be higher than $\max\{n_1, n_2, n_3\}$. For example, the CP rank of a $9 \times 9 \times 9$ tensor given by Kruskal is between 18 and 23. See [12]. It is known [12] that an upper bound of the CP rank is $\min\{n_1 n_2, n_1 n_3, n_2 n_3\}$.

The Tucker decomposition decomposes $\mathcal{X}$ into a core tensor $\mathcal{D} \in \Re^{r_1 \times r_2 \times r_3}$ multiplied by three factor matrices $U \in \Re^{n_1 \times r_1}$, $V \in \Re^{n_2 \times r_2}$, and $W \in \Re^{n_3 \times r_3}$ along three

*Received by the editors March 3, 2020; accepted for publication (in revised form) by E. Acar December 14, 2020; published electronically March 4, 2021.
https://doi.org/10.1137/20M1323266
Funding: The work of the second author was supported by the National Natural Science Foundation of China grant 11771405 and 12071159. The work of the fourth author was supported by the National Natural Science Foundation of China grant 11871369.

†Future Network Theory Lab, 2012 Labs Huawei Technologies Investment Co., Ltd, Shatin, New Territory, Hong Kong, China; Department of Mathematics, School of Science, Hangzhou Dianzi University, Hangzhou, 310018, China; and Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China (liqun.qi@polyu.edu.hk).
‡Corresponding author. School of Mathematical Sciences, South China Normal University, Guangzhou 510631, China (ynchen@scnu.edu.cn).
§Future Network Theory Lab, 2012 Labs Huawei Technologies Investment Co., Ltd, Shatin, New Territory, Hong Kong, China (mayank.bakshi@huawei.com).
¶School of Mathematics, Tianjin University, Tianjin, 300354, China (xzzhang@tju.edu.cn).

modes, i.e.,

$$\mathcal{X} = \mathcal{D} \times_1 U \times_2 V \times_3 W.$$

The minimum possible values of r_1, r_2, and r_3 are called the Tucker rank of $\mathcal{X}$ [12]. Then $r_i \leq n_i$ for $i = 1, 2, 3$. Thus, the Tucker rank is relatively smaller.

In tensor approximation or completion via Tucker decomposition, it is usually computationally expensive to update the Tucker core $\mathcal{D}$ [19, 18]. Furthermore, in some tensor recovery applications, such as transportation and internet data recovery, each mode of the third order tensor $\mathcal{X}$ in the problem has a different meaning [16, 17, 21]. Different features of the problems, such as temporal stability, spatial correlation, and traffic periodicity, may be reflected in the three modes [17]. The three factor matrices U, V, and W of the Tucker decomposition thus inherit such features and can be utilized in the tensor recovery process [17, 21]. But the Tucker core $\mathcal{D}$ is not decomposed. Can we have a decomposition which decomposes $\mathcal{X}$ to three pieces, and which also decomposes the Tucker core $\mathcal{D}$ to three pieces in the same time, to inherit such features of the three modes? In this paper, motivated by this, we introduce a new tensor decomposition for third order tensors, which decomposes a third order tensor to a product of three third order factor tensors. Each factor tensor has two low dimensions. The Tucker core is decomposed simultaneously with the original third order tensor. We call such a decomposition triple decomposition, and the corresponding rank the triple rank. The triple rank of a third order tensor is not greater than the middle value of the Tucker rank of that tensor. It is also not greater than the triple rank of the Tucker core of that tensor, and equality holds under mild conditions. We introduce the standard triple decomposition of a third order tensor. The number of parameters in the bilevel form of the standard triple decomposition is less than the number of parameters of the Tucker decomposition in substantial cases. Hence, triple decomposition does not cost more than Tucker decomposition, the three low rank tensors inherit the features of the three modes, and the Tucker core is also decomposed simultaneously.

The rest of this paper is distributed as follows. Preliminary knowledge on CP decomposition and Tucker decomposition is presented in the next section. In section 3, we introduce triple decomposition and triple rank and prove the above-mentioned key properties and some other theoretical properties. We present an algorithm in section 4 to check if a given third order tensor can be approximated by a third order tensor of low triple rank such that the relative error is reasonably small. In section 5, we show that practical data of third order tensors from internet traffic and image are of low triple ranks. A tensor recovery method is proposed in section 6, based on such low rank triple decomposition. Its convergence and convergence rate are also established in that section. Numerical tests of our method are presented in section 7. Some concluding remarks are made in section 8.

2. CP decomposition, Tucker decomposition, and related tensor ranks. We use small letters to denote scalars, small bold letters to denote vectors, capital letters to denote matrices, and calligraphic letters to denote tensors. In this paper, we only study third order tensors.

Perhaps the most well-known tensor decomposition is CP decomposition [12]. Its corresponding tensor rank is called the CP rank.

DEFINITION 2.1. *Suppose that* $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$. *Let* $A = (a_{ip}) \in \Re^{n_1 \times r}$,

$B = (b_{jp}) \in \Re^{n_2 \times r}$, *and* $C = (c_{tp}) \in \Re^{n_3 \times r}$. *Here* n_1, n_2, n_3, r *are positive integers. If*

$$x_{ijt} = \sum_{p=1}^{r} a_{ip} b_{jp} c_{tp} \tag{2.1}$$

for $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, *and* $t = 1, \ldots, n_3$, *then* $\mathcal{X}$ *has a CP decomposition* $\mathcal{X} = [[A, B, C]]$. *The smallest integer* r *such that* (2.1) *holds is called the CP rank of* $\mathcal{X}$, *and denoted as* $CPRank(\mathcal{X}) = r$.

A well-known tensor decomposition is Tucker decomposition [12]. Its corresponding tensor rank is called the Tucker rank. Higher order SVD (HOSVD) decomposition [9] can be regarded as a special variant of Tucker decomposition.

DEFINITION 2.2. *Suppose that* $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$, *where* n_1, n_2, *and* n_3 *are positive integers. We may unfold* $\mathcal{X}$ *to a matrix* $X_{(1)} = (x_{i,jt}) \in \Re^{n_1 \times n_2 n_3}$, *or a matrix* $X_{(2)} = (x_{j,it}) \in \Re^{n_2 \times n_1 n_3}$, *or a matrix* $X_{(3)} = (x_{t,ij}) \in \Re^{n_3 \times n_1 n_2}$. *Denote the matrix ranks of* $X_{(1)}, X_{(2)}$, *and* $X_{(3)}$ *as* r_1, r_2, *and* r_3, *respectively. Then the triplet* (r_1, r_2, r_3) *is called the Tucker rank of* $\mathcal{X}$, *and is denoted as* $TuckRank(\mathcal{X}) = (r_1, r_2, r_3)$ *with* $TuckRank(\mathcal{X})_i = r_i$ *for* $i = 1, 2, 3$.

The CP rank and the Tucker rank are called the rank and n-rank in some papers [12]. Here, we follow [10] to distinguish them from other tensor ranks.

DEFINITION 2.3. *Suppose that* $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$. *Let* $U = (u_{ip}) \in \Re^{n_1 \times r_1}$, $V = (v_{jq}) \in \Re^{n_2 \times r_2}$, $W = (w_{ts}) \in \Re^{n_3 \times r_3}$, *and* $\mathcal{D} = (d_{pqs}) \in \Re^{r_1 \times r_2 \times r_3}$. *Here* $n_1, n_2, n_3, r_1, r_2, r_3$ *are positive integers. If*

$$x_{ijt} = \sum_{p=1}^{r_1} \sum_{q=1}^{r_2} \sum_{s=1}^{r_3} u_{ip} v_{jq} w_{ts} d_{pqs} \tag{2.2}$$

for $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, *and* $t = 1, \ldots, n_3$, *then* $\mathcal{X}$ *has a Tucker decomposition* $\mathcal{X} = [[\mathcal{D}; U, V, W]]$. *The matrices* U, V, W *are called factor matrices of the Tucker decomposition, and the tensor* $\mathcal{D}$ *is called the Tucker core. We may also denote the Tucker decomposition as*

$$\mathcal{X} = \mathcal{D} \times_1 U \times_2 V \times_3 W. \tag{2.3}$$

The Tucker ranks r_1, r_2, r_3 of $\mathcal{X}$ are the smallest integers such that (2.2) holds [12]. Nonnegative tensor recovery methods via Tucker decomposition can be found in [19].

3. Triple decomposition, triple rank, and their properties. Let $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$. As in [12], we use $\mathcal{X}(i, :, :)$ to denote the ith horizontal slice, $\mathcal{X}(:, j, :)$ to denote the jth lateral slice, and $\mathcal{X}(:, :, t)$ to denote the tth frontal slice. We say that $\mathcal{X}$ is a third order horizontally square tensor if all of its horizontal slices are square, i.e., $n_2 = n_3$. Similarly, $\mathcal{X}$ is a third order laterally square tensor (resp., frontally square tensor) if all of its lateral slices (resp., frontal slices) are square, i.e., $n_1 = n_3$ (resp., $n_1 = n_2$).

For three real numbers α, β, and γ, let $\text{mid}\{\alpha, \beta, \gamma\}$ be the second largest value of these three numbers.

DEFINITION 3.1. *Let* $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$ *be a nonzero tensor. We say that* $\mathcal{X}$ *is the triple product of a third order horizontally square tensor* $\mathcal{A} = (a_{iqs}) \in \Re^{n_1 \times r \times r}$,

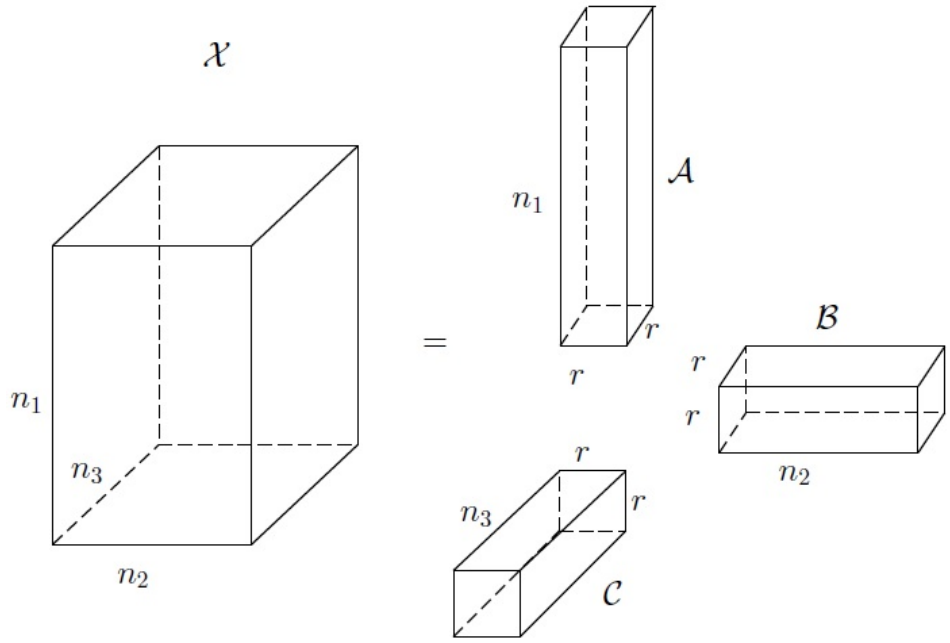

FIG. 1. *Low rank triple decomposition.*

a third order laterally square tensor $\mathcal{B} = (b_{pjs}) \in \Re^{r\times n_2\times r}$*, and a third order frontally square tensor* $\mathcal{C} = (c_{pqt}) \in \Re^{r\times r\times n_3}$*, and denote*

$$\mathcal{X} = [\![\mathcal{ABC}]\!] \tag{3.1}$$

if for $i = 1, \ldots, n_1$*,* $j = 1, \ldots, n_2$*, and* $t = 1, \ldots, n_3$*, we have*

$$x_{ijt} = \sum_{p=1}^{r}\sum_{q=1}^{r}\sum_{s=1}^{r} a_{iqs}b_{pjs}c_{pqt}. \tag{3.2}$$

If

$$r \le \text{mid}\{n_1, n_2, n_3\}, \tag{3.3}$$

then we call (3.1) *a low rank triple decomposition of* $\mathcal{X}$*, and call* $\mathcal{A}, \mathcal{B}$*, and* $\mathcal{C}$ *factor tensors of* $\mathcal{X}$*. See Figure* 1 *for a visualization.*

The smallest value of r *such that* (3.2) *holds is called the triple rank of* $\mathcal{X}$*, and is denoted as TriRank*$(\mathcal{X}) = r$*. For a zero tensor, we define its triple rank as zero.*

Note that TriRank($\mathcal{X}$) is zero if and only if it is a zero tensor. This is analogous to the matrix case.

In (3.2), a_{iqs} contributes its first index to x_{ijt}, b_{pjs} and c_{pqt} contribute their second and third indices to x_{ijt}, respectively, while p, q, and s are "link" indices. In practice, different modes may have different meanings such as time and space [16, 17, 21]. Thus, factor tensors $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ inherit information of modes 1, 2, and 3 from $\mathcal{X}$, respectively.

The triple decomposition can be regarded as a special case (with equal ranks for third order tensors) of the tensor ring decomposition [20], which, in turn, is closely

related to the tensor train decomposition. When the order is higher than three or when the ranks in the tensor ring decomposition are not equal, the tensor ring decomposition may not have good properties. For example, for a fourth order tensor, the tensor ring decomposition linked as modes 1-2-3-4-1 will be different from those linked as modes 1-3-2-4-1. There are even more combinations of the mode orders for higher order tensors. There are also some arguments on tensor ring decompositions [4]. We will not be involved with such arguments, and will concentrate on triple decomposition of third order tensors.

THEOREM 3.2. *Low rank triple decomposition and triple ranks are well defined. A third order nonzero tensor $\mathcal{X}$ always has a low rank triple decomposition* (3.1)*, satisfying* (3.3)*. Also, the number of parameters of triple decomposition can be restricted such that it is not greater than the number of parameters in the third order tensor.*

Proof. Without loss of generality, we may assume that we have a third order nonzero tensor $\mathcal{X} \in \Re^{n_1 \times n_2 \times n_3}$ and $n_1 \geq n_2 \geq n_3 \geq 1$. Thus, $\text{mid}\{n_1, n_2, n_3\} = n_2$. Let $r = n_2$. Let $\mathcal{A} \in \Re^{n_1 \times r \times r}$, $\mathcal{B} \in \Re^{r \times n_2 \times r}$, and $\mathcal{C} \in \Re^{r \times r \times n_3}$ be such that $a_{iqs} = x_{isq}$ if $q \leq n_3$, $a_{iqs} = 0$ if $q > n_3$, $b_{1js} = \delta_{js}$ and $b_{pjs} = 0$ for $p > 1$, $c_{1qt} = \delta_{qt}$ and $c_{pqt} = 0$ for $p > 1$, for $i = 1, \ldots, n_1$, $j, p, q, s = 1, \ldots, n_2$, and $t = 1, \ldots, n_3$, where δ_{js} and δ_{qt} are the Kronecker symbol such that $\delta_{jj} = 1$ and $\delta_{js} = 0$ if $j \neq s$. Then,

$$\sum_{p=1}^{r}\sum_{q=1}^{r}\sum_{s=1}^{r} a_{iqs}b_{pjs}c_{pqt} = \sum_{q=1}^{\min\{r,n_3\}}\sum_{s=1}^{r} x_{isq}\delta_{js}\delta_{qt} = x_{ijt}$$

for $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, and $t = 1, \ldots, n_3$, i.e., (3.2) holds for the above choices of $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$. Thus, the triple decomposition always exists with $r \leq n_2$.

In the above choices of $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$, excluding zero and one, the number of parameters in $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ is still $n_1 n_2 n_3$, the number of parameters in $\mathcal{X}$. Hence, triple decomposition does not increase the number of parameters. ∎

Note that one cannot change (3.3) to

$$r \leq \min\{n_1, n_2, n_3\}. \tag{3.4}$$

The above assertion can be seen through the following argument. Let $n_1 = n_2 = 3$ and $n_3 = 1$. Suppose that $\mathcal{X}$ is chosen to have nine independent entries. If (3.4) is required, then with $r = 1$, the decomposition consists of $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ that can have only a maximum of seven independent entries in total. Thus, we cannot find $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$, satisfying (3.1), (3.2), and (3.4).

For the relation between triple decomposition and CP decomposition, we have the following theorem.

THEOREM 3.3. *Suppose that $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$. Then we may regard its CP decomposition as a special case of its triple decomposition. In particular, we have*

$$\text{TriRank}(\mathcal{X}) \leq \text{CPRank}(\mathcal{X}) \leq (\text{TriRank}(\mathcal{X}))^3.$$

Proof. Suppose that $\mathcal{X} = [[A, B, C]]$ with $A = (a_{ip}) \in \Re^{n_1 \times r}$, $B = (b_{jp}) \in \Re^{n_2 \times r}$, and $C \in \Re^{n_3 \times r}$ is a CP decomposition. Denote $\mathcal{A} = (\bar{a}_{ipq}) \in \Re^{n_1 \times r \times r}$, $\mathcal{B} = (\bar{b}_{sjq}) \in \Re^{r \times n_2 \times r}$, and $\mathcal{C} = (c_{spt}) \in \Re^{r \times r \times n_3}$ with

$$\bar{a}_{ipq} = \begin{cases} a_{ip} & \text{if } p = q, \\ 0 & \text{otherwise,} \end{cases} \qquad \bar{b}_{sjq} = \begin{cases} b_{jq} & \text{if } s = q, \\ 0 & \text{otherwise,} \end{cases}$$

$$\bar{c}_{spt} = \begin{cases} c_{tp} & \text{if } s = p, \\ 0 & \text{otherwise.} \end{cases}$$

Then for all $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, and $t = 1, \ldots, n_3$, there holds

$$(\llbracket \mathcal{ABC} \rrbracket)_{ijt} = \sum_{p=1}^{r} \sum_{q=1}^{r} \sum_{s=1}^{r} \bar{a}_{ipq} \bar{b}_{sjq} \bar{c}_{spt} = \sum_{p=1}^{r} a_{ip} b_{jp} c_{tp} = x_{ijt}.$$

This means that $\mathcal{X} = \llbracket \mathcal{ABC} \rrbracket$, i.e., we may regard its CP decomposition as a special case of its triple decomposition. Furthermore, we have $\mathrm{TriRank}(\mathcal{X}) \leq \mathrm{CPRank}(\mathcal{X})$ from the definition of the triple rank.

On the other hand, suppose that $\mathcal{X}$ is of the form $x_{ijt} = \sum_{p=1}^{\bar{r}} \sum_{q=1}^{\bar{r}} \sum_{s=1}^{\bar{r}} a_{iqs} b_{pjs} c_{pqt}$. Then, $\mathcal{X}$ can be represented as a sum of $\bar{r}^3$ rank-one tensors. Hence, the last inequality in the theorem holds by setting $\bar{r} = TriRank(\mathcal{X})$. ∎

Equality may occur in the second inequality of the theorem, i.e., it may be possible that $\mathrm{CPRank}(\mathcal{X}) = (\mathrm{TriRank}(\mathcal{X}))^3$. For example, let $\mathcal{A} \in \Re^{4\times 2\times 2}$, $\mathcal{B} \in \Re^{2\times 4\times 2}$, $\mathcal{C} \in \Re^{2\times 2\times 4}$, and $\mathcal{X} = \llbracket \mathcal{ABC} \rrbracket$. Then $\mathrm{TriRank}(\mathcal{X}) \leq 2$, but $\mathrm{CPRank}(\mathcal{X})$ may be 8. See Example 3.5 below. It is a little hard to prove this conjecture as the task to calculate the CP rank is not easy. The number of parameters of triple decomposition is $(n_1 + n_2 + n_3)(\mathrm{TriRank}(\mathcal{X}))^2$. The number of parameters of CP decomposition is $(n_1 + n_2 + n_3)\mathrm{CPRank}(\mathcal{X})$. In the case that $\mathrm{CPRank}(\mathcal{X})$ is close to $(\mathrm{TriRank}(\mathcal{X}))^3$, triple decomposition is much better. In this case it happens that $\mathcal{X}$ is sparse, which appears more in practice.

We now study the relation between triple decomposition and Tucker decomposition. We have the following theorem.

THEOREM 3.4. *Suppose that* $\mathcal{X} = (x_{ijt}) \in \Re^{n_1\times n_2\times n_3}$ *and* $\mathcal{X} = \llbracket \mathcal{ABC} \rrbracket$ *with* $TriRank(\mathcal{X}) = R$, $\mathcal{A} \in \Re^{n_1\times R\times R}, \mathcal{B} \in \Re^{R\times n_2\times R}$, *and* $\mathcal{C} \in \Re^{R\times R\times n_3}$. *Furthermore,*

$$\mathcal{X} = \mathcal{D} \times_1 U \times_2 V \times_3 W$$

is a Tucker decomposition of $\mathcal{X}$ *with* $\mathcal{D} \in \Re^{r_1\times r_2\times r_3}$ *and factor matrices* $U \in \Re^{n_1\times r_1}$, $V \in \Re^{n_2\times r_2}, W \in \Re^{n_3\times r_3}$. *Then*

$$\mathrm{TriRank}(\mathcal{X}) \leq \mathrm{TriRank}(\mathcal{D}) \leq \mathrm{mid}\{r_1, r_2, r_3\}. \tag{3.5}$$

Furthermore, if U^TU, V^TV, *and* W^TW *are invertible, then we have*

$$\mathrm{TriRank}(\mathcal{X}) = \mathrm{TriRank}(\mathcal{D}). \tag{3.6}$$

Thus, we always have

$$\mathrm{TriRank}(\mathcal{X}) \leq \mathrm{mid}\{\mathrm{TuckRank}(\mathcal{X})_1, \mathrm{TuckRank}(\mathcal{X})_2, \mathrm{TuckRank}(\mathcal{X})_3\}. \tag{3.7}$$

Proof. For notational convenience, let $\mathrm{TriRank}(\mathcal{D}) = r$. By (3.3), we have the second inequality of (3.5).

We first show that $r \geq R$. Assume that $\mathcal{D} = \llbracket \bar{\mathcal{A}}\bar{\mathcal{B}}\bar{\mathcal{C}} \rrbracket$ with $\bar{\mathcal{A}} \in \Re^{r_1\times r\times r}$, $\bar{\mathcal{B}} \in \Re^{r\times r_2\times r}$, and $\bar{\mathcal{C}} \in \Re^{r\times r\times r_3}$. Then

$$\mathcal{X} = (\llbracket \bar{\mathcal{A}}\bar{\mathcal{B}}\bar{\mathcal{C}} \rrbracket) \times_1 U \times_2 V \times_3 W = \left\llbracket (\bar{\mathcal{A}} \times_1 U)(\bar{\mathcal{B}} \times_2 V)(\bar{\mathcal{C}} \times_3 W) \right\rrbracket.$$

Clearly, $\bar{\mathcal{A}} \times_1 U \in \Re^{n_1\times r\times r}$, $\bar{\mathcal{B}} \times_2 V \in \Re^{r\times n_2\times r}$, and $\bar{\mathcal{C}} \times_3 W \in \Re^{r\times r\times n_3}$. Hence, $r \geq R$ from the definition of TriRank. This proves the first inequality of (3.5).

Now we assume that U^TU, V^TV, and W^TW are invertible. From $\mathcal{X} = \mathcal{D} \times_1 U \times_2 V \times_3 W$, we have that

$$\begin{aligned} & \mathcal{X} \times_1 (U^TU)^{-1}U^T \times_2 (V^TV)^{-1}V^T \times_3 (W^TW)^{-1}W^T \\ = \; & (\mathcal{D} \times_1 U \times_2 V \times_3 W) \times_1 (U^TU)^{-1}U^T \times_2 (V^TV)^{-1}V^T \times_3 (W^TW)^{-1}W^T \\ = \; & \mathcal{D} \times_1 (U^TU)^{-1}(U^TU) \times_2 (V^TV)^{-1}(V^TV) \times_3 (W^TW)^{-1}(W^TW) \\ = \; & \mathcal{D} \times_1 I_{r_1} \times_2 I_{r_2} \times_3 I_{r_3} = \mathcal{D}. \end{aligned}$$

Hence, it holds that

$$\begin{aligned} \mathcal{D} = \; & (\llbracket \mathcal{A}, \mathcal{B}, \mathcal{C} \rrbracket) \times_1 (U^TU)^{-1}U^T \times_2 (V^TV)^{-1}V^T \times_3 (W^TW)^{-1}W^T \\ = \; & \left\llbracket (\mathcal{A} \times_1 (U^TU)^{-1}U^T)(\mathcal{B} \times_2 (V^TV)^{-1}V^T)(\mathcal{C} \times_3 (W^TW)^{-1}W^T) \right\rrbracket. \end{aligned}$$

It is easy to see that $\mathcal{A} \times_1 (U^TU)^{-1}U^T \in \Re^{r_1 \times R \times R}$, $\mathcal{B} \times_2 (V^TV)^{-1}V^T \in \Re^{R \times r_2 \times R}$, and $\mathcal{C} \times (W^TW)^{-1}W^T \in \Re^{R \times R \times r_3}$. From the definition of TriRank, we have $r \le R$. Therefore, $r = R$ and (3.6) holds.

Note that the condition that $\text{TuckRank}(\mathcal{X}) = (r_1, r_2, r_3)$ can be always realized. For example, in HOSVD [9], all factor matrices are orthogonal, and we always have $\text{TuckRank}(\mathcal{X}) = (r_1, r_2, r_3)$. This shows that (3.7) always holds. ∎

Suppose that we have a third order tensor $\mathcal{X} \in \Re^{n_1 \times n_2 \times n_3}$ with a triple decomposition $\mathcal{X} = \llbracket \mathcal{A}\mathcal{B}\mathcal{C} \rrbracket$, where $\text{TriRank}(\mathcal{X}) = r$, $\mathcal{A} \in \Re^{n_1 \times r \times r}$, $\mathcal{B} \in \Re^{r \times n_2 \times r}$, $\mathcal{C} \in \Re^{r \times r \times n_3}$. By the above theorem, we see that there always exist matrices $U \in \Re^{n_1 \times r_1}$, $V \in \Re^{n_2 \times r_2}$, and $W \in \Re^{n_3 \times r_3}$, and third order tensors $\hat{\mathcal{A}} \in \Re^{r_1 \times r \times r}$, $\hat{\mathcal{B}} \in \Re^{r \times r_2 \times r}$, $\hat{\mathcal{C}} \in \Re^{r \times r \times r_3}$, such that

$$\mathcal{A} = \hat{\mathcal{A}} \times_1 U, \quad \mathcal{B} = \hat{\mathcal{B}} \times_2 V, \quad \mathcal{C} = \hat{\mathcal{C}} \times_3 W, \tag{3.8}$$

and $r_i = \text{TuckRank}(\mathcal{X})_i$ for $i = 1, 2, 3$. We call $\mathcal{X} = \llbracket \mathcal{A}\mathcal{B}\mathcal{C} \rrbracket$ and (3.8) a bilevel form of triple decomposition of $\mathcal{X}$, and $\hat{\mathcal{A}}, \hat{\mathcal{B}}$, and $\hat{\mathcal{C}}$ the inner factor tensors. For any triple decomposition of a third order tensor, we may always write it in a bilevel form as we always can let U, V, and W be identity matrices and $\hat{\mathcal{A}} = \mathcal{A}$, $\hat{\mathcal{B}} = \mathcal{B}$, and $\hat{C} = \mathcal{C}$.

We now give an example that $\text{TriRank}(\mathcal{X}) < \min\{ \text{TuckRank}(\mathcal{X})_1, \text{TuckRank}(\mathcal{X})_2, \text{TuckRank}(\mathcal{X})_3 \}$.

Example 3.5. Let $n_1 = n_2 = n_3 = 4$ and $r = 2$. Consider $\mathcal{A} = (a_{iqs}) \in \Re^{4 \times 2 \times 2}$, $\mathcal{B} = (b_{pjs}) \in \Re^{2 \times 4 \times 2}$, and $\mathcal{C} = (c_{pqt}) \in \Re^{2 \times 2 \times 4}$ such that $a_{111} = a_{212} = a_{321} = a_{422} = 1$ and $a_{iqs} = 0$ otherwise, $b_{111} = b_{122} = b_{231} = b_{242} = 1$ and $b_{pjs} = 0$ otherwise, and $c_{111} = c_{122} = c_{213} = c_{224} = 1$ and $c_{pqt} = 0$ otherwise. Then $\text{TuckRank}(\mathcal{A})_1 = \text{TuckRank}(\mathcal{B})_2 = \text{TuckRank}(\mathcal{C})_3 = 4$. Let $\mathcal{X} = \llbracket \mathcal{A}\mathcal{B}\mathcal{C} \rrbracket$. Then $\text{TriRank}(\mathcal{X}) \le 2$ and $\mathcal{X} \in \Re^{4 \times 4 \times 4}$. We have $x_{111} = x_{133} = x_{221} = x_{243} = x_{312} = x_{334} = x_{422} = x_{444} = 1$ and $x_{ijt} = 0$ otherwise. We may easily check that $\text{TuckRank}(\mathcal{X})_1 = \text{TuckRank}(\mathcal{X})_2 = \text{TuckRank}(\mathcal{X})_3 = 4$. Thus, $\text{TriRank}(\mathcal{X}) \le 2 < \text{TuckRank}(\mathcal{X})_1 = \text{TuckRank}(\mathcal{X})_2 = \text{TuckRank}(\mathcal{X})_3 = 4$.

Suppose that $\mathcal{X} = \llbracket \mathcal{A}\mathcal{B}\mathcal{C} \rrbracket$, where $\mathcal{A} = \mathcal{F} \times_1 \widetilde{A}$, $\mathcal{F} \in \Re^{r_1 \times r \times r}$, $\widetilde{A} \in \Re^{n_1 \times r_1}$, $\mathcal{B} = \mathcal{G} \times_2 \widetilde{B}$, $\mathcal{G} \in \Re^{r \times r_2 \times r}$, $\widetilde{B} \in \Re^{n_2 \times r_2}$, $\mathcal{C} = \mathcal{H} \times_3 \widetilde{C}$, $\mathcal{H} \in \Re^{r \times r \times r_3}$, and $\widetilde{C} \in \Re^{n_3 \times r_3}$. Then, we have

$$x_{ijk} = \sum_{p=1}^{r}\sum_{q=1}^{r}\sum_{s=1}^{r} a_{iqs}b_{pjs}c_{pqk} = \sum_{u=1}^{r_1}\sum_{v=1}^{r_2}\sum_{w=1}^{r_3} \widetilde{A}_{iu}\widetilde{B}_{jv}\widetilde{C}_{kw} \underbrace{\sum_{p=1}^{r}\sum_{q=1}^{r}\sum_{s=1}^{r} F_{uqs}G_{pvs}H_{pqw}}_{\text{a core tensor } \mathcal{F}\mathcal{G}\mathcal{H}}. \tag{3.9}$$

Thus, this is a formulation of the Tucker decomposition. This will be useful in the proof of the following theorem relating the triple rank to the Tucker rank.

THEOREM 3.6. *Suppose that* $\mathcal{X} = (x_{ijk}) \in \Re^{n_1 \times n_2 \times n_3}$ *and* $\mathcal{X} = [\![\mathcal{ABC}]\!]$ *with* $\mathcal{A} \in \Re^{n_1 \times r \times r}, \mathcal{B} \in \Re^{r \times n_2 \times r}$, *and* $\mathcal{C} \in \Re^{r \times r \times n_3}$. *Then*

$$\text{TuckRank}(\mathcal{X})_1 \le \text{TuckRank}(\mathcal{A})_1 \le (\text{TriRank}(\mathcal{A}))^2 \le (\text{TriRank}(\mathcal{X}))^2, \tag{3.10}$$

$$\text{TuckRank}(\mathcal{X})_2 \le \text{TuckRank}(\mathcal{B})_2 \le (\text{TriRank}(\mathcal{B}))^2 \le (\text{TriRank}(\mathcal{X}))^2, \tag{3.11}$$

$$\text{TuckRank}(\mathcal{X})_3 \le \text{TuckRank}(\mathcal{C})_3 \le (\text{TriRank}(\mathcal{C}))^2 \le (\text{TriRank}(\mathcal{X}))^2. \tag{3.12}$$

Furthermore, equality may occur for all these inequalities.

Proof. Let $\text{TriRank}(\mathcal{X}) = r$, $\text{TuckRank}(\mathcal{A})_1 = r_1$, $\text{TuckRank}(\mathcal{B})_2 = r_2$, and $\text{TuckRank}(\mathcal{C})_3 = r_3$. Let $\mathcal{A} = \mathcal{F} \times_1 U \times_2 U_2 \times_3 U_3$ be a Tucker decomposition of $\mathcal{A}$ with core tensor $\mathcal{F} \in \Re^{r_1 \times s_2 \times s_3}$ and factor matrices $U \in \Re^{n_1 \times r_1}, U_2 \in \Re^{r \times s_2}, U_3 \in \Re^{r \times s_3}$. Denote $\bar{\mathcal{A}} = \mathcal{F} \times_2 U_2 \times_3 U_3 \in \Re^{r_1 \times r \times r}$. Then $\mathcal{A} = (\mathcal{F} \times_2 U_2 \times_3 U_3) \times_1 U = \bar{\mathcal{A}} \times_1 U$.

Similarly, there exist $\bar{\mathcal{B}} \in \Re^{r \times r_2 \times r}, \bar{\mathcal{C}} \in \Re^{r \times r \times r_3}$, $V \in \Re^{n_2 \times r_2}, W \in \Re^{n_3 \times r_3}$ such that

$$\mathcal{A} = \bar{\mathcal{A}} \times_1 U, \quad \mathcal{B} = \bar{\mathcal{B}} \times_2 V, \quad \mathcal{C} = \bar{\mathcal{C}} \times_3 W.$$

Hence, $\mathcal{X} = [\![\mathcal{ABC}]\!] = ([\![\bar{\mathcal{A}}\bar{\mathcal{B}}\bar{\mathcal{C}}]\!]) \times_1 U \times_2 V \times_3 W$ according to (3.9). From definition of the Tucker rank, we have the first inequalities of (3.10)–(3.12).

Assume that $\text{TriRank}(\mathcal{A}) = \bar{r}$. Then there are tensors $\hat{\mathcal{A}} \in \Re^{n_1 \times \bar{r} \times \bar{r}}$, $\hat{\mathcal{B}} \in \Re^{\bar{r} \times r \times \bar{r}}$, and $\hat{\mathcal{C}} \in \Re^{\bar{r} \times \bar{r} \times r}$ such that $\mathcal{A} = [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!]$. Replacing $\mathcal{X}$ and $\mathcal{A}$ in the first inequality of (3.10) by $\mathcal{A}$ and $\hat{A}$, we have $\text{TuckRank}(\mathcal{A})_1 \le \text{TuckRank}(\hat{\mathcal{A}})_1$. Note that $\hat{A} \in \Re^{n_1 \times \bar{r} \times \bar{r}}$. By the definition of the Tucker rank, $\text{TuckRank}(\hat{\mathcal{A}})_1$ is the matrix rank of an $n_1 \times \bar{r}^2$ matrix. Hence, $\text{TuckRank}(\hat{\mathcal{A}})_1 \le \bar{r}^2$. This proves the second inequality of (3.10).

Since $\mathcal{A} = [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!]$ and $\mathcal{A} \in \Re^{n_1 \times r \times r}$, by (3.3), $\text{TriRank}(\mathcal{A}) \le r = \text{TriRank}(\mathcal{X})$. Then the third inequality of (3.10) holds.

The second and third inequalities of (3.11) and (3.12) hold similarly.

Considering Example 3.5, we conclude that equality may occur for all these inequalities. ∎

This theorem shows that in the bilevel form (3.8), we may let $r = \text{TriRank}(\mathcal{X})$ and $r_i = \text{TuckRank}(\mathcal{X})_i$ for $i = 1, 2, 3$. We call such a triple decomposition a standard triple decomposition. Every third order tensor has a standard triple decomposition.

The minimum number of parameters of Tucker decomposition of a third order tensor $\mathcal{X} \in \Re^{n_1 \times n_2 \times n_3}$ is $n_1 r_1 + n_2 r_2 + n_3 r_3 + r_1 r_2 r_3$, where $r_i = \text{TuckRank}(\mathcal{X})_i$ for $i = 1, 2, 3$. On the other hand, the number of parameters of the bilevel form (3.8) of a standard triple decomposition of $\mathcal{X}$ is $n_1 r_1 + n_2 r_2 + n_3 r_3 + (r_1 + r_2 + r_3) r^2$, where $r = \text{TriRank}(\mathcal{X})$. In Example 3.5, $r_1 = r_2 = r_3 = 4$ and $r = 2$. Then $(r_1 + r_2 + r_3) r^2 = 48 < r_1 r_2 r_3 = 64$. Actually, we may regard the third order tensor $\mathcal{X}$ in Example 3.5 as the Tucker core of a larger tensor. Then there are substantial cases where the number of parameters of the bilevel form (3.8) of a standard triple decomposition is strictly less than the minimum number of parameters of Tucker decomposition.

As Theorem 3.2 shows, the number of parameters in triple decomposition can be restricted such that it is not greater than the number of parameters in the original third order tensor. The above discussion shows that there are substantial cases where

the number of parameters of the bilevel form (3.8) of a standard triple decomposition is strictly less than the minimum number of parameters of Tucker decomposition. In [18], sparse nonnegative Tucker decomposition was proposed. It is considerable to extend the approach in [18] to sparse triple decomposition in the bilevel form.

This also explains a part of our motivation to introduce triple decomposition. We are motivated by the Tucker decomposition. We aim to decompose the Tucker core further by triple decomposition.

In the literature, there are essential uniqueness results for CP decomposition [8]. These results were further extended to block term decomposition [8]. Suppose that a third order tensor $\mathcal{X} \in \Re^{n_1 \times n_2 \times n_3}$ has a triple decomposition $[\![\mathcal{ABC}]\!]$, where $\mathcal{A} \in \Re^{n_1 \times r \times r}$, $\mathcal{B} \in \Re^{r \times n_2 \times r}$, $\mathcal{C} \in \Re^{r \times r \times n_3}$, and r is the triple rank of $\mathcal{X}$. If we replace $\mathcal{A}$ and $\mathcal{B}$ by $\bar{A} = \mathcal{A} \times_3 E$ and $\bar{\mathcal{B}} = \mathcal{B} \times_3 (E^\top)^{-1}$, respectively, where E is an $r \times r$ nonsingular matrix, then we have $\mathcal{X} = [\![\bar{\mathcal{A}}\bar{\mathcal{B}}\mathcal{C}]\!]$. We may say that the triple decomposition of $\mathcal{X}$ is essentially unique if it is subject only to such indeterminacies. Recently, under some conditions, Chang and Chan [6] showed that if $\mathcal{C}$ is determined, then $\mathcal{A}$ and $\mathcal{B}$ are essentially unique.

4. Approximate a third order tensor by low rank triple decomposition. We now consider approximating a given third order tensor $\mathcal{X} \in \Re^{n_1 \times n_2 \times n_3}$ by $[\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W$, i.e., to consider the following minimization problem:

$$\min f(\hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{C}}, U, V, W) \equiv \left\| [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W - \mathcal{X} \right\|_F^2, \tag{4.1}$$

where $\hat{\mathcal{A}} \in \Re^{r_1 \times r \times r}$, $\hat{\mathcal{B}} \in \Re^{r \times r_2 \times r}$, $\hat{\mathcal{C}} \in \Re^{r \times r \times r_3}$, $U \in \Re^{n_1 \times r_1}$, $V \in \Re^{n_2 \times r_2}$, and $W \in \Re^{n_3 \times r_3}$. Here, the positive integers r_1, r_2, r_3, and r satisfy $r_1 \le n_1, r_2 \le n_2, r_3 \le n_3$, $\text{mid}\{r_1, r_2, r_3\} \ge r$, and

$$(r_1 + r_2 + r_3)r^2 < r_1 r_2 r_3. \tag{4.2}$$

Let $\mathcal{D} = [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!]$. Then we recognize that this is very close to approximating $\mathcal{X}$ by the Tucker decomposition $\mathcal{D} \times_1 U \times_2 V \times_3 W$. In [18], it was proposed to update the core tensor $\mathcal{D}$ and factor matrices alternatingly in the order of $\mathcal{D}, U, \mathcal{D}, V, \mathcal{D}, W$. It is argued there that since the core tensor interacts with all U, V, and W, updating it more frequently is expected to speed up the convergence of the algorithm. We call such a method TuckD. As the inner factor tensors $\hat{\mathcal{A}}, \hat{\mathcal{B}}$, and $\hat{\mathcal{C}}$ interact with U, V, and W, respectively, in the bilevel form of triple decomposition, we may modify this approach by using the updating order $\hat{\mathcal{A}}, U, \hat{\mathcal{B}}, V, \hat{\mathcal{C}}, W$. We call such a method TriD. Then, the complexity of TriD in each iteration is much lower than TuckD as $\hat{\mathcal{A}}, \hat{\mathcal{B}}$, and $\hat{\mathcal{C}}$ are much smaller than $\mathcal{D}$. TriD is still a variant of the block coordinate descent (BCD) method proposed in [19]. Hence, the convergence of TriD is also guaranteed by Theorems 2.8 and 2.9 of [19].

In [18], sparsity and nonnegativity constraints are considered. In TuckD, we omit these additional considerations. Our purpose is to compare TriD with TuckD.

There are two basic steps in TriD.

1. Update the inner factor tensors $\hat{\mathcal{A}}, \hat{\mathcal{B}}$, and $\hat{\mathcal{C}}$. Since they are similar, we describe the procedure for updating $\hat{\mathcal{A}}$ as an example. We have

$$\hat{\mathcal{A}}_{new} = \text{argmin} \left\| [\![\hat{\mathcal{A}}\hat{\mathcal{B}}_{old}\hat{\mathcal{C}}_{old}]\!] \times_1 U_{old} \times_2 V_{old} \times_3 W_{old} - \mathcal{X} \right\|_F^2 + \mu_A \left\| \hat{\mathcal{A}} - \hat{\mathcal{A}}_{old} \right\|_F^2, \tag{4.3}$$

where $\mu_A > 0$ is a regularization parameter. Let $\hat{\mathcal{A}}_{(1)}$ and $\mathcal{X}_{(1)}$ be matrices corresponding to the mode-1 unfolding of $\hat{\mathcal{A}}$ and $\mathcal{X}$, respectively. Then, the above optimization

problem could be rewritten as

$$[\hat{\mathcal{A}}_{new}]_{(1)} = \text{argmin} \left\| U_{old}\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)} \right\|_F^2 + \mu_A \left\| \hat{\mathcal{A}}_{(1)} - [\hat{\mathcal{A}}_{old}]_{(1)} \right\|_F^2,$$

where F is an $n_2n_3 \times r^2$ matrix. The optimal solution satisfies the following matrix equation:

$$U_{old}^T U_{old}[\hat{\mathcal{A}}_{new}]_{(1)}F^TF + \mu_A[\hat{\mathcal{A}}_{new}]_{(1)} = U_{old}^T\mathcal{X}_{(1)}F + \mu_A[\hat{\mathcal{A}}_{old}]_{(1)}.$$

Let $U_{old}^TU_{old} = Q_U\text{diag}(\mathbf{s}_U)Q_U^T$ and $F^TF = Q_F\text{diag}(\mathbf{s}_F)Q_F^T$ be eigendecompositions of $U_{old}^TU_{old}$ and F^TF, respectively. Define $\tilde{A} = Q_U^T[\hat{\mathcal{A}}_{new}]_{(1)}Q_F$. It holds that

$$\text{diag}(\mathbf{s}_U)\tilde{A}\text{diag}(\mathbf{s}_F) + \mu_A\tilde{A} = Q_U^T\left(U_{old}^T\mathcal{X}_{(1)}F + \mu_A[\hat{\mathcal{A}}_{old}]_{(1)}\right)Q_F.$$

We note that this matrix equation has a closed-form solution

$$[\tilde{A}]_{ij} = \frac{\left[Q_U^T\left(U_{old}^T\mathcal{X}_{(1)}F + \mu_A[\hat{\mathcal{A}}_{old}]_{(1)}\right)Q_F\right]_{ij}}{[\mathbf{s}_U]_i[\mathbf{s}_F]_j + \mu_A}. \tag{4.4}$$

The denominator is positive because $\mathbf{s}_U \geq 0$, $\mathbf{s}_F \geq 0$, and $\mu_A > 0$. Then, we immediately obtain $[\hat{\mathcal{A}}_{new}]_{(1)} = Q_U\tilde{A}Q_F^T$ and hence the tensor $\hat{\mathcal{A}}_{new}$.

Similarly, we may obtain $\hat{\mathcal{B}}_{new}$ and $\hat{\mathcal{C}}_{new}$.

2. Update the factor matrices U, V, and W. They are similar. We now illustrate the procedure for updating U. We have

$$U_{new} = \text{argmin} \left\| [\![\hat{\mathcal{A}}_{new}\hat{\mathcal{B}}_{old}\hat{\mathcal{C}}_{old}]\!] \times_1 U \times_2 V_{old} \times_3 W_{old} - \mathcal{X} \right\|_F^2 + \lambda_U \left\| U - U_{old} \right\|_F^2, \tag{4.5}$$

which could be rewritten in a matrix form

$$U_{new} = \text{argmin} \left\| U[\hat{\mathcal{A}}_{new}]_{(1)}F^T - \mathcal{X}_{(1)} \right\|_F^2 + \lambda_U \left\| U - U_{old} \right\|_F^2, \tag{4.6}$$

where $\lambda_U > 0$ is a regularization parameter to make the objective function of (4.6) uniformly convex. In (4.6), only matrix operations are involved. We have the explicit solution

$$U_{new} = \left(\mathcal{X}_{(1)}F[\hat{\mathcal{A}}_{new}]_{(1)}^T + \lambda_U U_{old}\right)\left([\hat{\mathcal{A}}_{new}]_{(1)}F^TF[\hat{\mathcal{A}}_{new}]_{(1)}^T + \lambda_U I_{r_1}\right)^{-1}. \tag{4.7}$$

Similarly, we have V_{new} and W_{new}.

From these, we may construct TriD. See Algorithm 1.

Here, we use the extrapolation technique with step size $\gamma \in [1, 2)$ to deal with the swamp effect [7]. TriD may terminate if the difference between two iterates is small enough or the iteration arrives a preset maximal iterative number.

Next, we analyze the computational cost for computing $\hat{\mathcal{A}}_{new}$. First, it requires $\mathcal{O}(n_1r_1^2 + n_2r_2^2 + n_3r_3^2)$ flops to calculate symmetric matrices $U_{old}^TU_{old}$, $V_{old}^TV_{old}$, and $W_{old}^TW_{old}$. Second, we define a third order tensor $\mathcal{F} \in \Re^{r^2\times r_2\times r_3}$ with entries $[\mathcal{F}]_{mjk} = \sum_{p=1}^r[\hat{\mathcal{B}}_{old}]_{pjs}[\hat{\mathcal{C}}_{old}]_{pqk}$, where $m = q + (s-1)r$. Let $\bar{F} = (\mathcal{F}_{(1)})^T \in \Re^{r_2r_3\times r^2}$. Then, it holds that $F = (W_{old} \otimes V_{old})\bar{F}$ and hence

$$F^TF = \bar{F}^T[(W_{old}^TW_{old}) \otimes (V_{old}^TV_{old})]\bar{F} = \left[\mathcal{F} \times_2 (V_{old}^TV_{old}) \times_3 (W_{old}^TW_{old})\right]_{(1)}\bar{F},$$

Algorithm 1 Triple Decomposition Algorithm (TriD).

1: Set $\gamma \in [1, 2)$. Choose positive integers r_1, r_2, r_3, and r as specified early and initial points $\hat{\mathcal{A}}^0 \in \Re^{r_1 \times r \times r}$, $\hat{\mathcal{B}}^0 \in \Re^{r \times r_2 \times r}$, $\hat{\mathcal{C}}^0 \in \Re^{r \times r \times r_3}$, $U^0 \in \Re^{n_1 \times r_1}$, $V^0 \in \Re^{n_2 \times r_2}$. Set $k \leftarrow 0$.
2: Compute $\widetilde{\mathcal{A}}^k$ and set $\hat{\mathcal{A}}^{k+1} = \gamma\widetilde{\mathcal{A}}^k + (1-\gamma)\hat{\mathcal{A}}^k$.
3: Compute $\widetilde{U}^k$ and set $U^{k+1} = \gamma\widetilde{U}^k + (1-\gamma)U^k$.
4: Compute $\widetilde{\mathcal{B}}^k$ and set $\hat{\mathcal{B}}^{k+1} = \gamma\widetilde{\mathcal{B}}^k + (1-\gamma)\hat{\mathcal{B}}^k$.
5: Compute $\widetilde{V}^k$ and set $V^{k+1} = \gamma\widetilde{V}^k + (1-\gamma)V^k$.
6: Compute $\widetilde{\mathcal{C}}^k$ and set $\hat{\mathcal{C}}^{k+1} = \gamma\widetilde{\mathcal{C}}^k + (1-\gamma)\hat{\mathcal{C}}^k$.
7: Compute $\widetilde{W}^k$ and set $W^{k+1} = \gamma\widetilde{W}^k + (1-\gamma)W^k$.
8: Set $k \leftarrow k+1$ and goto step 2.

where $\otimes$ stands for the Kronecker product. This step costs about $\mathcal{O}((r_2+r_3)r_2r_3r^2 + r_2r_3r^4)$ flops. Third, eigendecompositions of $U_{old}^T U_{old}$ and F^TF need $\mathcal{O}(r_1^3 + r^6)$ flops. Fourth, we calculate $\mathcal{X}_{(1)}F = [\mathcal{X} \times_2 V_{old} \times_3 W_{old}]_{(1)}\bar{F}$, which costs about $\mathcal{O}(n_1n_2n_3r_2 + n_1n_3r_2r_3 + n_1r_2r_3r^2)$ flops. Fifth, it costs about $\mathcal{O}(n_1r_1r^2 + r_1^2r^2 + r_1r^4)$ flops to compute $Q_U^T\left(U_{old}^T\mathcal{X}_{(1)}F + \mu_A[\hat{\mathcal{A}}_{old}]_{(1)}\right)Q_F$. Finally, it costs about $\mathcal{O}(r_1r^2)$ flops to obtain $\tilde{A}$ by (4.4) and hence $\hat{\mathcal{A}}_{new}$.

Since F^TF and $\mathcal{X}_{(1)}F$ are available, the computational costs for $\mathcal{X}_{(1)}F[\hat{\mathcal{A}}_{new}]_{(1)}^T$ and $[\hat{\mathcal{A}}_{new}]_{(1)}F^TF[\hat{\mathcal{A}}_{new}]_{(1)}^T$ are about $\mathcal{O}(n_1r_1r^2)$ and $\mathcal{O}(r_1r^4 + r_1^2r^2)$, respectively. Then, computing U_{new} by (4.7) is about $\mathcal{O}(r_1^3 + n_1r_1^2)$.

Suppose that integers r_1, r_2, r_3, and r are of the same order of magnitude and they are much smaller than n_1, n_2, and n_3. We claim that the total computational cost of the proposed TriD algorithm is about $\mathcal{O}(n_1n_2n_3r + (n_1+n_2+n_3)r^4 + r^6)$ flops in each iteration.

5. Practical data of third order tensors. In this section, we investigate practical data from applications and show that they can be approximated by triple decomposition of low triple ranks very well.

5.1. Abilene internet traffic data. The first application we consider is the internet traffic data. The data set is the Abilene data set[1] [17].

The Abilene data arises from the backbone network located in North America. There are 11 routers: Atlanta, GA; Chicago, IL; Denver, CO; Houston, TX; Indianapolis, IN; Kansas City, MO; Los Angeles, CA; New York, NY; Sunnyvale, CA; Seattle, WA; and Washington, DC. These routers send and receive data. Thus we get 121 original-destination (OD) pairs. For each OD pair, we record the internet traffic data of every five minutes in a week from Dec. 8, 2003 to Dec. 14, 2003. Hence, there are $7 \times 24 \times 60/5 = 2016$ numbers for each OD pair. In this way, we get a third order tensor $\mathcal{X}_{Abil}$ with size $11 \times 11 \times 2016$. This model was used in [1, 21] for internet traffic data recovery.

At the beginning, we should choose proper parameters $\lambda_U, \lambda_V, \lambda_W, \mu_A, \mu_B, \mu_C$, and γ. For simplicity, we set $\lambda_U = \lambda_V = \lambda_W = \mu_A = \mu_B = \mu_C$ and take their values from $1, 0.1, 0.01, 0.001$, and 0, where the value 0 corresponds to a basic algorithm. For the tensor $\mathcal{X}_{Abil}$ with $r_1 = r_2 = 10, r_3 = 100$, and $r = 9$, the performance of TriD

[1]The Abilene observatory data collections: http://abilene.internet2.edu/observatory/data-collections.html

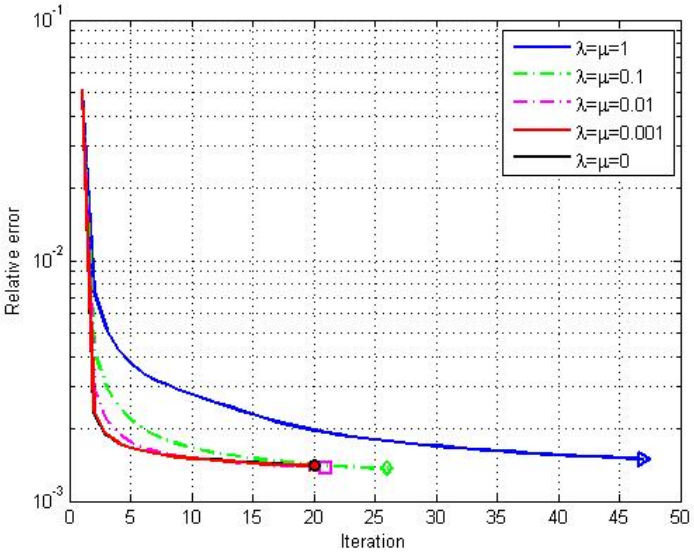

FIG. 2. *Performance of TriD with various* $\lambda_U, \lambda_V, \lambda_W, \mu_A, \mu_B$, *and* μ_C.

with $\gamma = 1$ and different values of $\lambda_U, \lambda_V, \lambda_W, \mu_A, \mu_B$, and μ_C is illustrated in Figure 2. Small values of $\lambda_U, \lambda_V, \lambda_W, \mu_A, \mu_B$, and μ_C work well. We set $\lambda_U = \lambda_V = \lambda_W = \mu_A = \mu_B = \mu_C = 0.001$ for the purpose of ensuring that subproblems on $\hat{\mathcal{A}}_{new}$, $\hat{\mathcal{B}}_{new}$, $\hat{\mathcal{C}}_{new}$, U_{new}, V_{new}, and W_{new} are uniformly convex and have unique solutions.

We also compare TriD with the parameter γ varying from 1 to 1.8, where $\gamma = 1$ corresponds to a basic algorithm. Using $\mathcal{X}_{Abil}$ with $r_1 = r_2 = 10$, $r_3 = 100$, and $r = 9$ again, the performance of TriD with various values of γ is reported in Figure 3. Obviously, TriD with $\gamma = 1.6$ uses 15 iterations which is less than 20 iterations required by a basic algorithm. Hence, the extrapolation parameter γ could accelerate the proposed algorithm. Whereafter, we fix these parameters in all experiments.

Now, we examine the triple decomposition approximation of the tensor $\mathcal{X}_{Abil} \in \Re^{11\times 11\times 2016}$ with different upper bounds of the triple rank r from 2 to 10. For each r, we set $r_1 = r_2 = r+1$ and $r_3 = 10(r+1)$, which satisfies $r_1 r_2 r_3 \geq (r_1 + r_2 + r_3)r^2$. Then, we compute the triple decomposition approximation $[\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W$ by Algorithm 1 and calculate the relative error of low triple rank approximation:

$$\text{RelativeError} = \frac{\|\mathcal{X}_{Abil} - [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W\|_F}{\|\mathcal{X}_{Abil}\|_F}.$$

Figure 4 illustrates the relative error of the low rank approximations via the number of used parameters. For comparison, the relative error of Tucker approximation is also reported in Figure 4. When we take $r = 8$, the relative errors of the triple approximation and Tucker approximation are about 5.32% and 5.13%, respectively. This shows that the Abilene data can be approximated by triple decomposition of low triple rank as well as the Tucker approximation. For the CPU time, TuckD and TriD cost totally about 50.66 seconds and 7.99 seconds, respectively. It is worth noting that TriD is faster than TuckD.

Next, we consider another Abilene traffic dataset $\tilde{\mathcal{X}}_{Abil} \in \Re^{121\times 96\times 21}$, which is

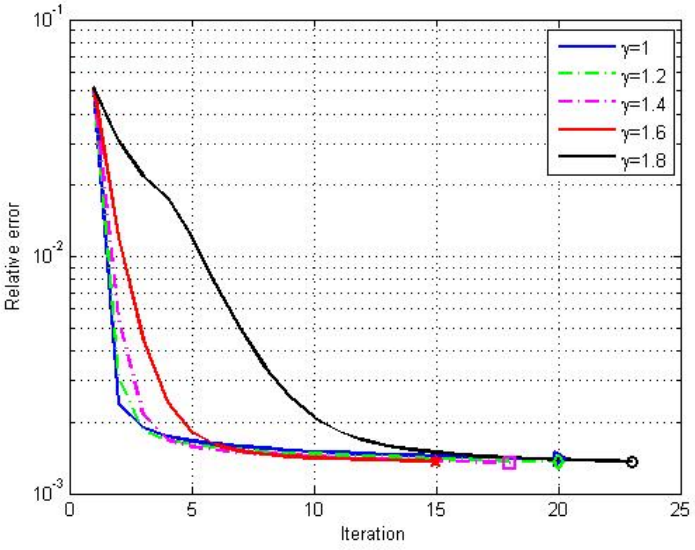

FIG. 3. *Performance of TriD with various* γ.

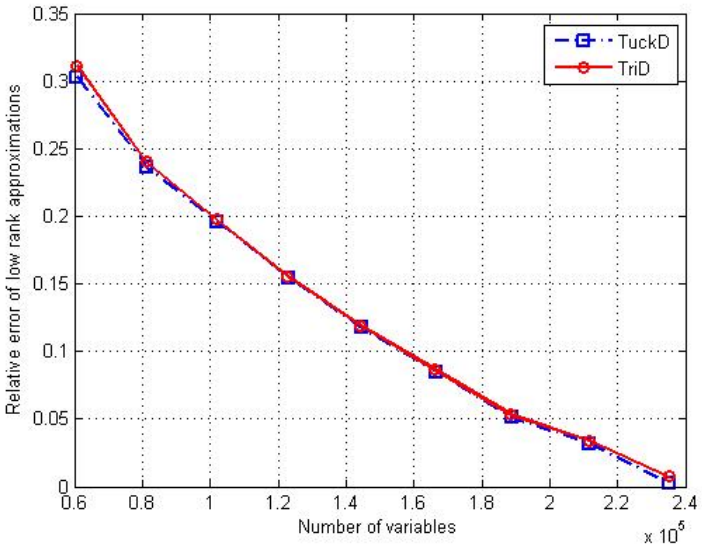

FIG. 4. *Relative errors of low triple rank approximations and low Tucker rank approximations of the* $11 \times 11 \times 2016$ *internet traffic tensor from the Abilene dataset.*

a third order tensor indexed by 121 source-destination pairs, 96 time slots for each day, and 21 days. This is the model used in [17]. We take r from 2 to 20 and set $r_1 = 5r, r_2 = 4r, r_3 = 20$. Figure 5 shows the relative error of the low rank approximations obtained by Algorithm 1 as a function of the number of variables. To meet the relative error level 5%, the low triple rank approximation uses $30,720$ variables, which is less than $69,888$ variables used by low Tucker rank approximation.

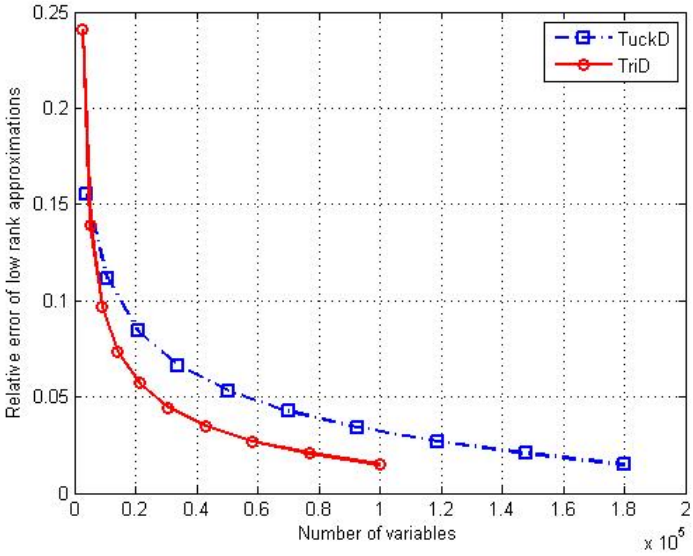

FIG. 5. *Relative errors of low triple rank approximations and low Tucker rank approximations of the* $121 \times 96 \times 21$ *internet traffic tensor from the Abilene dataset.*

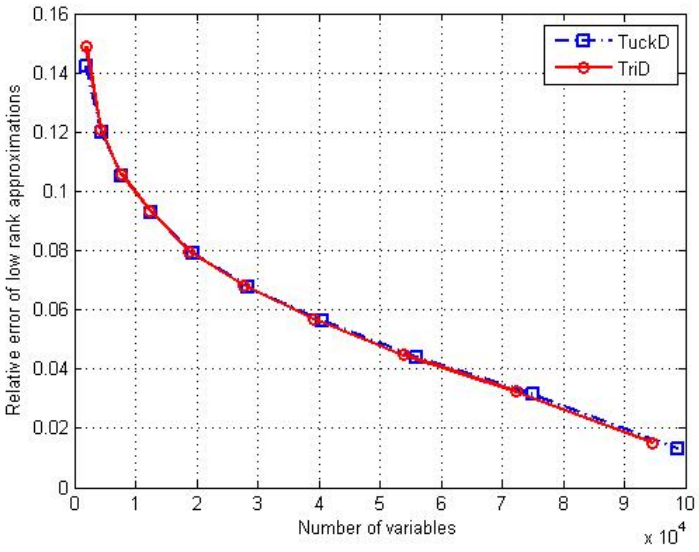

FIG. 6. *Low rank approximations of a third order ORL face data tensor of size* $112 \times 92 \times 10$.

For the CPU time, TuckD and TriD cost totally about 1.36 seconds and 24.04 seconds, respectively. Eigendecomposition for solving $\hat{\mathcal{A}}_{new}, \hat{\mathcal{B}}_{new}$, and $\hat{\mathcal{C}}_{new}$ subproblems costs most of the time of TriD.

5.2. ORL face data. We now investigate the ORL face data from AT&T Laboratories Cambridge [15, 19].

FIG. 7. *Illustration of faces from the ORL dataset. Original images are illustrated in the first line. Approximations with rank* 8, 16, *and* 20 *are shown in lines two, three, and four, respectively.*

The ORL dataset of faces contains images of 40 persons. Each image has 112×92 pixels. For each person, there are 10 images taken at different times, varying the lighting, facial expressions, and facial details. For instance, the first line of Figure 7 illustrates 10 images of a person. Hence, there is a $112 \times 92 \times 10$ tensor $\mathcal{T}_{face}$. Using Algorithm 1, we compute best low triple rank approximations of the tensor $\mathcal{T}_{face}$, where we take $r = 2t, r_1 = 10t, r_2 = 8t, r_3 = t$, and t is an integer from 1 to 10. The relative error of approximations via triple ranks is illustrated in Figure 6. When the triple rank upper bound $r = 8, 16, 20$, the relative error of low triple rank approximations is $9.35\%, 4.47\%, 1.51\%$, respectively. Corresponding images of low triple rank approximations are illustrated in lines 2–4 of Figure 7. For the CPU time, TuckD and TriD cost totally about 0.92 seconds and 23.32 seconds, respectively.

This result clearly shows that the ORL data can be approximated by low rank triple decomposition very well.

6. A tensor recovery method and its convergence analysis. In this section, we consider the tensor recovery problem in a bilevel form:

$$\min \ \|\mathbb{P}([\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W) - \mathbf{d}\|_F^2, \tag{6.1}$$

where $\mathbb{P}$ is a linear operator, $\mathbf{d} \in \Re^m$ is a given vector, and $\hat{\mathcal{A}} \in \Re^{r_1 \times r \times r}$, $\hat{\mathcal{B}} \in \Re^{r \times r_2 \times r}$, $\hat{\mathcal{C}} \in \Re^{r \times r \times r_3}$, $U \in \Re^{n_1 \times r_1}$, $V \in \Re^{n_2 \times r_2}$, and $W \in \Re^{n_3 \times r_3}$ are unknown. To solve (6.1), we introduce a surrogate tensor $\mathcal{X} = (x_{ijt}) \in \Re^{n_1 \times n_2 \times n_3}$ and transform (6.1) into the closely related optimization problem

$$\begin{aligned} &\min \ f(\mathcal{X}, \hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{C}}, U, V, W) := \left\| [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W - \mathcal{X} \right\|_F^2 \\ &\text{s.t.} \ \ \mathbb{P}(\mathcal{X}) = \mathbf{d}. \end{aligned} \tag{6.2}$$

We slightly abuse the notation f for denoting an objective function.

6.1. A tensor recovery method. We propose a modified alternating least squares algorithm for solving the tensor recovery problem (6.2). For fixed positive integers r_1, r_2, r_3, and r, we choose $\mathcal{X}^0 \in \Re^{n_1 \times n_2 \times n_3}$, $\hat{\mathcal{A}}^0 \in \Re^{r_1 \times r \times r}$, $\hat{\mathcal{B}}^0 \in \Re^{r \times r_2 \times r}$,

$\hat{\mathcal{C}}^0 \in \Re^{r\times r\times r_3}$, $U^0 \in \Re^{n_1\times r_1}$, $V^0 \in \Re^{n_2\times r_2}$, $W^0 \in \Re^{n_3\times r_3}$, and set $k \leftarrow 0$. We perform the following steps recursively.

Update $\mathcal{X}^{k+1}$. We solve a subproblem

$$\begin{aligned}\arg\min_{\mathcal{X}\in\Re^{n_1\times n_2\times n_3}} \ & \left\|\mathcal{X} - [\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k\right\|_F^2 + \lambda\|\mathcal{X}-\mathcal{X}^k\|_F^2\\ \text{s.t.}\ \ & \mathbb{P}(\mathcal{X}) = \mathbf{d}.\end{aligned}$$

That is,

$$\begin{aligned}\arg\min_{\mathcal{X}\in\Re^{n_1\times n_2\times n_3}} \ & \left\|\mathcal{X} - \tfrac{1}{1+\lambda}\left([\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k + \lambda\mathcal{X}^k\right)\right\|_F^2\\ \text{s.t.}\ \ & \mathbb{P}(\mathcal{X}) = \mathbf{d}.\end{aligned}$$

Define an operator vec : $\Re^{n_1\times n_2\times n_3} \to \Re^{n_1n_2n_3}$ that maps x_{ijt} to $\hat{x}_\ell$ where $\ell = i + (j-1)n_1 + (t-1)n_1n_2$. Then, the equality constraint $\mathbb{P}(\mathcal{X}) = \mathbf{d}$ may be rewritten as $P\text{vec}(\mathcal{X}) = \mathbf{d}$, where P is the $m\times(n_1n_2n_3)$ matrix corresponding to the application of the operator $\mathbb{P}$ when viewed as a linear transformation from vec$(\mathcal{X})$ to $\mathbf{d}$. Here we assume that PP^T is invertible. Thus, the above optimization problem may be represented as

$$\begin{aligned}\arg\min \ & \left\|\text{vec}(\mathcal{X}) - \tfrac{1}{1+\lambda}\text{vec}\left([\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k + \lambda\mathcal{X}^k\right)\right\|_F^2\\ \text{s.t.}\ \ & P\text{vec}(\mathcal{X}) = \mathbf{d},\end{aligned}$$

which has a closed-form solution

$$\begin{aligned}(6.3)\qquad & \left[I - P^T\left(PP^T\right)^{-1}P\right]\tfrac{1}{1+\lambda}\text{vec}\left([\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k + \lambda\mathcal{X}^k\right)\\ & + P^T\left(PP^T\right)^{-1}\mathbf{d}.\end{aligned}$$

This is defined as vec$(\widetilde{\mathcal{X}}^k)$. Next, we set

$$\mathcal{X}^{k+1} = \gamma\widetilde{\mathcal{X}}^k + (1-\gamma)\mathcal{X}^k.$$

The remaining subproblems on updating $\hat{\mathcal{A}}^{k+1}$, $\hat{\mathcal{B}}^{k+1}$, $\hat{\mathcal{C}}^{k+1}$, U^{k+1}, V^{k+1}, and W^{k+1} are similar to those in TriD; in addition $\mathcal{X}$ is replaced by $\mathcal{X}^{k+1}$. Now, we briefly introduce the process.

Update $\hat{\mathcal{A}}^{k+1}$. Using the approach for (4.3), we solve

$$(6.4)\qquad \arg\min_{\hat{\mathcal{A}}\in\Re^{r_1\times r\times r}} \|[\![\hat{\mathcal{A}}\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k - \mathcal{X}^{k+1}\|_F^2 + \lambda\|\hat{\mathcal{A}} - \hat{\mathcal{A}}^k\|_F^2$$

for $\widetilde{\mathcal{A}}^k$ and then set $\mathcal{A}^{k+1} = \gamma\widetilde{\mathcal{A}}^k + (1-\gamma)\mathcal{A}^k$.

Update U^{k+1}. Using the approach for (4.5), we solve

$$(6.5)\qquad \arg\min_{U\in\Re^{n_1\times r_1}} \|[\![\hat{\mathcal{A}}^{k+1}\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U\times_2 V^k\times_3 W^k - \mathcal{X}^{k+1}\|_F^2 + \lambda\|U - U^k\|_F^2$$

for $\widetilde{U}^k$ and then set $U^{k+1} = \gamma\widetilde{U}^k + (1-\gamma)U^k$.

In a similar approach, we compute $\hat{\mathcal{B}}^{k+1}$, V^{k+1}, $\hat{\mathcal{C}}^{k+1}$, and W^{k+1}. Subsequently, we set $k \leftarrow k+1$ and repeat this process until convergence. The detailed algorithm is stated as Algorithm 2.

Algorithm 2 Triple Recovery Algorithm (TriR).

1: Set $\gamma \in [1,2)$ and $\lambda > 0$. Choose integers r_1, r_2, r_3, r, initial points $\hat{\mathcal{A}}^0 \in \Re^{r_1 \times r \times r}$, $\hat{\mathcal{B}}^0 \in \Re^{r \times r_2 \times r}$, $\hat{\mathcal{C}}^0 \in \Re^{r \times r \times r_3}$, $U^0 \in \Re^{n_1 \times r_1}$, $V^0 \in \Re^{n_2 \times r_2}$, $W^0 \in \Re^{n_3 \times r_3}$, and $\mathcal{X}^0 \in \Re^{n_1 \times n_2 \times n_3}$. Set $k \leftarrow 0$.
2: Compute $\widetilde{\mathcal{X}}^k$ and set $\mathcal{X}^{k+1} = \gamma\widetilde{\mathcal{X}}^k + (1-\gamma)\mathcal{X}^k$.
3: Compute $\widetilde{\mathcal{A}}^k$ and set $\hat{\mathcal{A}}^{k+1} = \gamma\widetilde{\mathcal{A}}^k + (1-\gamma)\hat{\mathcal{A}}^k$.
4: Compute $\widetilde{U}^k$ and set $U^{k+1} = \gamma\widetilde{U}^k + (1-\gamma)U^k$.
5: Compute $\widetilde{\mathcal{B}}^k$ and set $\hat{\mathcal{B}}^{k+1} = \gamma\widetilde{\mathcal{B}}^k + (1-\gamma)\hat{\mathcal{B}}^k$.
6: Compute $\widetilde{V}^k$ and set $U^{k+1} = \gamma\widetilde{V}^k + (1-\gamma)V^k$.
7: Compute $\widetilde{\mathcal{C}}^k$ and set $\hat{\mathcal{C}}^{k+1} = \gamma\widetilde{\mathcal{C}}^k + (1-\gamma)\hat{\mathcal{C}}^k$.
8: Compute $\widetilde{W}^k$ and set $U^{k+1} = \gamma\widetilde{W}^k + (1-\gamma)W^k$.
9: Set $k \leftarrow k+1$ and goto step 2.

When the linear operator $\mathbb{P}$ is cheap, the computational cost of the recovery algorithm TriR is a little bit more expensive than the decomposition algorithm TriD, since TriR has an additional step on $\widetilde{\mathcal{X}}^k$ which costs about $\mathcal{O}(n_1n_2n_3\max\{r_1,r_2,r_3\}+r_1r_2r_3r^2)$.

6.2. Convergence analysis. We now present convergence analysis for this algorithm. For convenience, we collect all variables as an undetermined vector

$$\mathbf{y} := \Big(\mathrm{vec}(\mathcal{X})^T, \mathrm{vec}(\hat{\mathcal{A}}_{(1)})^T, \mathrm{vec}(\hat{\mathcal{B}}_{(2)})^T, \mathrm{vec}(\hat{\mathcal{C}}_{(3)})^T, \mathrm{vec}(U)^T, \mathrm{vec}(V)^T, \mathrm{vec}(W)^T\Big)^T \in \Re^{n_1n_2n_3+(r_1+r_2+r_3)r^2+n_1r_1+n_2r_2+n_3r_3}.$$

The feasible region of $\mathbf{y}$ is defined by

$$\Omega := \{\mathrm{vec}(\mathcal{X}) \in \Re^{n_1n_2n_3} : P\mathrm{vec}(\mathcal{X}) = \mathbf{d}\} \oplus \Re^{r_1r^2} \oplus \Re^{r_2r^2} \oplus \Re^{r_3r^2} \oplus \Re^{n_1r_1} \oplus \Re^{n_2r_2} \oplus \Re^{n_3r_3}.$$

We analyze the convergence of Algorithm 2 for solving an optimization problem

$$\min \ f(\mathbf{y}) := f(\mathcal{X}, \hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{C}}, U, V, W) = \|\mathcal{X} - [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W\|_F^2 \quad \text{s.t. } \mathbf{y} \in \Omega. \tag{6.6}$$

To simplify notation, we use $\mathbf{y} = (\mathcal{X}, \hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{C}}, U, V, W)$ in the following analysis.

By optimization theory, $\mathbf{y}^*$ is a stationary point of (6.6) if and only if the projected negative gradient of f at $\mathbf{y}^*$ vanishes. In the following, we derive the formula of the projected gradient of f. First, let $\mathbf{y} = (\mathcal{X}, \hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{C}}, U, V, W) \in \Omega$. Since

$$f(\mathbf{y}) = \|\mathrm{vec}(\mathcal{X}) - \mathrm{vec}([\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W)\|^2,$$

we have, $\nabla_{\mathrm{vec}(\mathcal{X})} f = 2\mathrm{vec}(\mathcal{X}) - 2\mathrm{vec}([\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W)$. Since the set $\{\mathrm{vec}(\mathcal{X}) \in \Re^{n_1n_2n_3} : P\mathrm{vec}(\mathcal{X}) = \mathbf{d}\}$ is an affine manifold, we obtain the projected gradient of the $\mathcal{X}$-part

$$\begin{aligned}&\Big[I - P^T\left(PP^T\right)^{-1}P\Big]\Big(2\mathrm{vec}(\mathcal{X}) - 2\mathrm{vec}([\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W)\Big)\\ &= 2\Big[I - P^T\left(PP^T\right)^{-1}P\Big]\mathrm{vec}(\mathcal{X} - [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W)\end{aligned}$$

directly.

Next, we rewrite $f(\mathbf{y})$ as

$$\begin{aligned} f(\mathbf{y}) &= \|U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)}\|_F^2 \\ &= \langle \hat{\mathcal{A}}_{(1)}, U^T U\hat{\mathcal{A}}_{(1)}F^T F\rangle - 2\langle \hat{\mathcal{A}}_{(1)}, U^T\mathcal{X}_{(1)}F\rangle + \langle \mathcal{X}_{(1)}, \mathcal{X}_{(1)}\rangle, \end{aligned}$$

where $F \in \Re^{n_2 n_3 \times r^2}$ is a product of $\hat{\mathcal{B}}, \hat{\mathcal{C}}, V$, and W and $\langle\cdot,\cdot\rangle$ stands for the inner product. Hence, the $\hat{\mathcal{A}}_{(1)}$-part of the (projected) gradient is

$$2\left(U^T U\hat{\mathcal{A}}_{(1)}F^T F - U^T\mathcal{X}_{(1)}F\right) = 2U^T(U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)})F.$$

We may write the $\mathrm{vec}(\hat{\mathcal{A}}_{(1)})$-part of the gradient in $2\mathrm{vec}(U^T(U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)})F)$ to correspond to the vector form on $\mathrm{vec}(\hat{\mathcal{A}}_{(1)})$.

By a similar approach, the $\hat{\mathcal{B}}_{(2)}$-part and the $\hat{\mathcal{C}}_{(3)}$-part of the (projected) gradient are

$$2V^T(V\hat{\mathcal{B}}_{(2)}G^T - \mathcal{X}_{(2)})G, \qquad 2W^T(W\hat{\mathcal{C}}_{(3)}H^T - \mathcal{X}_{(3)})H,$$

respectively. Here, G is a $n_3 n_1 \times r^2$ matrix generated by $\hat{\mathcal{A}}, \hat{\mathcal{C}}, U$, and W; and H is a $n_1 n_2 \times r^2$ matrix generated by $\hat{\mathcal{A}}, \hat{\mathcal{B}}, U$, and V.

For the U-part of the (projected) gradient, we rewrite $f(\mathbf{y})$ as

$$\begin{aligned} f(\mathbf{y}) &= \|U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)}\|_F^2 \\ &= \langle U, U\hat{\mathcal{A}}_{(1)}F^T F[\hat{\mathcal{A}}_{(1)}]^T\rangle - 2\langle U, \mathcal{X}_{(1)}F[\hat{\mathcal{A}}_{(1)}]^T\rangle + \langle \mathcal{X}_{(1)}, \mathcal{X}_{(1)}\rangle. \end{aligned}$$

Hence, the U-part of the (projected) gradient is

$$2\left(U\hat{\mathcal{A}}_{(1)}F^T F[\hat{\mathcal{A}}_{(1)}]^T - \mathcal{X}_{(1)}F[\hat{\mathcal{A}}_{(1)}]^T\right) = 2(U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)})F[\hat{\mathcal{A}}_{(1)}]^T.$$

We may write the $\mathrm{vec}(U)$-part of the gradient in $2\mathrm{vec}((U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)})F[\hat{\mathcal{A}}_{(1)}]^T)$ to correspond to the vector form on $\mathrm{vec}(U)$.

Similarly, the V-part and the W-part of the (projected) gradient are

$$2(V\hat{\mathcal{B}}_{(2)}G^T - \mathcal{X}_{(2)})G[\hat{\mathcal{B}}_{(2)}]^T, \qquad 2(W\hat{\mathcal{C}}_{(3)}H^T - \mathcal{X}_{(3)})H[\hat{\mathcal{C}}_{(3)}]^T,$$

respectively.

Therefore, we get the projected gradient of f at $\mathbf{y} = (\mathcal{X}, \hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{C}}, U, V, W) \in \Omega$:

$$\Pi_\Omega\left(\nabla f(\mathbf{y})\right) = 2\begin{pmatrix} [I - P^T(PP^T)^{-1}P]\mathrm{vec}(\mathcal{X} - [\![\hat{\mathcal{A}}\hat{\mathcal{B}}\hat{\mathcal{C}}]\!] \times_1 U \times_2 V \times_3 W) \\ \mathrm{vec}(U^T(U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)})F) \\ \mathrm{vec}(V^T(V\hat{\mathcal{B}}_{(2)}G^T - \mathcal{X}_{(2)})G) \\ \mathrm{vec}(W^T(W\hat{\mathcal{C}}_{(3)}H^T - \mathcal{X}_{(3)})H) \\ \mathrm{vec}((U\hat{\mathcal{A}}_{(1)}F^T - \mathcal{X}_{(1)})F[\hat{\mathcal{A}}_{(1)}]^T) \\ \mathrm{vec}((V\hat{\mathcal{B}}_{(2)}G^T - \mathcal{X}_{(2)})G[\hat{\mathcal{B}}_{(2)}]^T) \\ \mathrm{vec}((W\hat{\mathcal{C}}_{(3)}H^T - \mathcal{X}_{(3)})H[\hat{\mathcal{C}}_{(3)}]^T) \end{pmatrix}, \tag{6.7}$$

where $\Pi_\Omega(\cdot)$ denotes the projection onto the feasible Ω.

The Kurdyka–Łojasiewicz (KL) property [13, 5, 19] with constraint is defined as below. Since $f(\mathbf{y}) + \delta_\Omega(\mathbf{y})$ is a semialgebraic function, where $\delta_\Omega(\cdot)$ is an indicator function defined on the affine manifold Ω, the following KL inequality holds.

DEFINITION 6.1 (Kurdyka–Łojasiewicz (KL) property). *Let $\mathbb{U} \in \Re^n$ be an open set, and let $f : \mathbb{U} \to \Re$ be a semialgebraic function. For every critical point $\mathbf{y}^* \in \mathbb{U}$ of f, there is a neighborhood $\mathbb{V} \subseteq \mathbb{U}$ of $\mathbf{y}^*$, an exponent $\theta \in [\frac{1}{2}, 1)$ and a positive constant μ such that*

$$|f(\mathbf{y}) - f(\mathbf{y}^*)|^\theta \leq \mu \|\Pi_\Omega(\nabla f(\mathbf{y}))\| \qquad \forall\, \mathbf{y} \in \mathbb{V},$$

where $\Pi_\Omega(\nabla f(\mathbf{y}))$ is defined by (6.7).

The main theorem which we will prove is as follows.

THEOREM 6.2. *Suppose that Algorithm* 2 *generates a sequence $\{\mathbf{y}^k\}$.*

If $\mathbf{y}^k = \mathbf{y}^{k+1}$ for some k, then $\mathbf{y}^k$ is a critical point of (6.6). *Otherwise, $\{\mathbf{y}^k\}$ is an infinite sequence. If this sequence is bounded, then it converges to a critical point $\mathbf{y}^*$ of* (6.6), *and the KL inequality holds at $\mathbf{y}^*$.*

(1) *If $\theta = \frac{1}{2}$ in the KL inequality, then there exist $\eta > 0$ and $\nu \in [0, 1)$ such that*

$$\|\mathbf{y}^k - \mathbf{y}^*\| \leq \eta \nu^k,$$

which means that the sequence of iterates converges R-linearly.

(2) *If $\theta \in (\frac{1}{2}, 1)$ in the KL inequality, then there exist $\eta > 0$ such that*

$$\|\mathbf{y}^k - \mathbf{y}^*\| \leq \eta k^{-\frac{1-\theta}{2\theta-1}}.$$

We prove this theorem in several steps. First, we have the following lemma on the optimality condition.

LEMMA 6.3. *Let $\mathbf{y}^* = \left(\mathcal{X}^*, \hat{\mathcal{A}}^*, \hat{\mathcal{B}}^*, \hat{\mathcal{C}}^*, U^*, V^*, W^*\right)^T \in \Omega$ be the optimal solution of optimization problem* (6.6). *Then, the projected negative gradient of f at $\mathbf{y}^*$ vanishes, i.e.,*

$$[I - P^T(PP^T)^{-1}P]\mathrm{vec}(\mathcal{X}^* - [\![\hat{\mathcal{A}}^*\hat{\mathcal{B}}^*\hat{\mathcal{C}}^*]\!] \times_1 U^* \times_2 V^* \times_3 W^*) = 0, \tag{6.8}$$

$$(U^*)^T(U^*\hat{\mathcal{A}}^*_{(1)}(F^*)^T - \mathcal{X}^*_{(1)})F^* = 0, \tag{6.9}$$

$$(V^*)^T(V^*\hat{\mathcal{B}}^*_{(2)}(G^*)^T - \mathcal{X}^*_{(2)})G^* = 0, \tag{6.10}$$

$$(W^*)^T(W^*\hat{\mathcal{C}}^*_{(3)}(H^*)^T - \mathcal{X}^*_{(3)})H^* = 0, \tag{6.11}$$

$$(U^*\hat{\mathcal{A}}^*_{(1)}(F^*)^T - \mathcal{X}^*_{(1)})F^*[\hat{\mathcal{A}}^*_{(1)}]^T = 0, \tag{6.12}$$

$$(V^*\hat{\mathcal{B}}^*_{(2)}(G^*)^T - \mathcal{X}^*_{(2)})G^*[\hat{\mathcal{B}}^*_{(2)}]^T = 0, \tag{6.13}$$

$$(W^*\hat{\mathcal{C}}^*_{(3)}(H^*)^T - \mathcal{X}^*_{(3)})H^*[\hat{\mathcal{C}}^*_{(3)}]^T = 0, \tag{6.14}$$

where $F^* \in \Re^{n_2 n_3 \times r^2}$, $G^* \in \Re^{n_3 n_1 \times r^2}$, and $H^* \in \Re^{n_1 n_2 \times r^2}$ are generated by $\hat{\mathcal{A}}^*$, $\hat{\mathcal{B}}^*$, $\hat{\mathcal{C}}^*$, U^*, V^*, and W^*. That is to say, $\mathbf{y}^* = (\mathcal{X}^*, \hat{\mathcal{A}}^*, \hat{\mathcal{B}}^*, \hat{\mathcal{C}}^*, U^*, V^*, W^*)$ is a stationary point of (6.6).

Now, we consider the case that the sequence generated by Algorithm 2 converges in a finite number of iterations.

LEMMA 6.4. *If there exists an iteration k such that $\mathbf{y}^k = \mathbf{y}^{k+1}$, i.e.,*

$$(\mathcal{X}^k, \hat{\mathcal{A}}^k, \hat{\mathcal{B}}^k, \hat{\mathcal{C}}^k, U^k, V^k, W^k) = (\mathcal{X}^{k+1}, \hat{\mathcal{A}}^{k+1}, \hat{\mathcal{B}}^{k+1}, \hat{\mathcal{C}}^{k+1}, U^{k+1}, V^{k+1}, W^{k+1}),$$

then $(\mathcal{X}^k, \hat{\mathcal{A}}^k, \hat{\mathcal{B}}^k, \hat{\mathcal{C}}^k, U^k, V^k, W^k)$ is a stationary point of (6.6).

Proof. First, for the $\mathcal{X}$-part, since $\widetilde{\mathcal{X}}^k$ is generated by (6.3), we know

$$\begin{aligned}\mathrm{vec}(\widetilde{\mathcal{X}}^k) = \big(I - P^T(PP^T)^{-1}P\big)\tfrac{1}{1+\lambda}\mathrm{vec}(\llbracket \hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k\rrbracket \times_1 U^k \times_2 V^k \times_3 W^k + \lambda\mathcal{X}^k)\\ + P^T(PP^T)^{-1}\mathbf{d}.\end{aligned}$$

In addition, because $\mathcal{X}^k$ satisfies $P\mathrm{vec}(\mathcal{X}^k) = \mathbf{d}$, it yields that

$$\mathrm{vec}(\mathcal{X}^k) = \big(I - P^T(PP^T)^{-1}P\big)\,\mathrm{vec}(\mathcal{X}^k) + P^T(PP^T)^{-1}\mathbf{d}.$$

Combining the above two equations, we have

$$\mathrm{vec}(\widetilde{\mathcal{X}}^k - \mathcal{X}^k) = \big(I - P^T(PP^T)^{-1}P\big)\tfrac{1}{1+\lambda}\mathrm{vec}(\llbracket \hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k\rrbracket \times_1 U^k \times_2 V^k \times_3 W^k - \mathcal{X}^k). \tag{6.15}$$

From $\mathcal{X}^k = \mathcal{X}^{k+1}$, we get $\widetilde{\mathcal{X}}^k - \mathcal{X}^k = 0$. Hence, (6.15) vanishes, i.e., the $\mathcal{X}$-part of the projected negative gradient of f vanishes.

It yields from (6.4) that

$$(U^k)^TU^k\widetilde{\mathcal{A}}^k_{(1)}(F^k)^TF^k - (U^k)^T\mathcal{X}^{k+1}_{(1)}F^k + \lambda\widetilde{\mathcal{A}}^k_{(1)} - \lambda\hat{\mathcal{A}}^k_{(1)} = 0, \tag{6.16}$$

which implies $(U^k)^T(U^k\hat{\mathcal{A}}^k_{(1)}(F^k)^T - \mathcal{X}^k_{(1)})F^k = 0$ by $\mathbf{y}^k = \mathbf{y}^{k+1} = \widetilde{\mathbf{y}}^k$. That is, the $\hat{\mathcal{A}}$-part of the projected negative gradient of f vanishes.

From (6.5), it yields that

$$\widetilde{U}^k\hat{\mathcal{A}}^{k+1}_{(1)}(F^k)^TF^k[\hat{\mathcal{A}}^{k+1}_{(1)}]^T - \mathcal{X}^{k+1}_{(1)}F^k[\hat{\mathcal{A}}^{k+1}_{(1)}]^T + \lambda\widetilde{U}^k - \lambda U^k = 0, \tag{6.17}$$

which implies $(U^k\hat{\mathcal{A}}^k_{(1)}(F^k)^T - \mathcal{X}^k_{(1)})F^k[\hat{\mathcal{A}}^k_{(1)}]^T = 0$ by $\mathbf{y}^k = \mathbf{y}^{k+1} = \widetilde{\mathbf{y}}^k$. That is, the U-part of the projected negative gradient of f vanishes.

Finally, by a similar discussion, we find that the $\mathcal{B}$-part, the $\mathcal{C}$-part, the V-part, and the W-part of the projected negative gradient of f also vanish. Hence, by Lemma 6.3, $\mathbf{y}^k$ is a stationary point of (6.6). ∎

Next, we consider the case that Algorithm 2 generates an infinite sequence of iterates.

LEMMA 6.5. *Let $\{\mathbf{y}^k\}_{k=0,1,2,\ldots}$ be a sequence of iterates generated by Algorithm 2. Then, we have*

$$f(\mathbf{y}^k) - f(\mathbf{y}^{k+1}) \geq \frac{2\lambda}{\gamma}\|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2.$$

Proof. For the $\mathcal{X}$-part, we have

$$
\begin{aligned}
&f(\mathcal{X}^k,\hat{\mathcal{A}}^k,\hat{\mathcal{B}}^k,\hat{\mathcal{C}}^k,U^k,V^k,W^k)-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^k,\hat{\mathcal{B}}^k,\hat{\mathcal{C}}^k,U^k,V^k,W^k)\\
&=\|\mathcal{X}^k-[\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k\|_F^2\\
&\quad-\|\gamma\widetilde{\mathcal{X}}^k+(1-\gamma)\mathcal{X}^k-[\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k\|_F^2\\
&=2\langle[\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k-\mathcal{X}^k,\gamma(\widetilde{\mathcal{X}}^k-\mathcal{X}^k)\rangle-\|\gamma(\widetilde{\mathcal{X}}^k-\mathcal{X}^k)\|_F^2\\
&=2\gamma\langle\mathrm{vec}([\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k-\mathcal{X}^k),\\
&\quad\cdot\tfrac{1}{1+\lambda}\left(I-P^T(PP^T)^{-1}P\right)\mathrm{vec}([\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k-\mathcal{X}^k)\rangle\\
&\quad-\gamma^2\|\widetilde{\mathcal{X}}^k-\mathcal{X}^k\|_F^2\\
&=2\gamma(1+\lambda)\|\tfrac{1}{1+\lambda}\left(I-P^T(PP^T)^{-1}P\right)\mathrm{vec}([\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k-\mathcal{X}^k)\|^2\\
&\quad-\gamma^2\|\widetilde{\mathcal{X}}^k-\mathcal{X}^k\|_F^2\\
&=2\gamma(1+\lambda)\|\mathrm{vec}(\widetilde{\mathcal{X}}^k-\mathcal{X}^k)\|^2-\gamma^2\|\widetilde{\mathcal{X}}^k-\mathcal{X}^k\|_F^2\\
&\geq 2\lambda\gamma\|\widetilde{\mathcal{X}}^k-\mathcal{X}^k\|_F^2\\
&=\frac{2\lambda}{\gamma}\|\mathcal{X}^{k+1}-\mathcal{X}^k\|_F^2,
\end{aligned}
$$

where the third and fifth equalities hold by (6.15), the fourth equality holds because $I-P^T(PP^T)^{-1}P$ is an idempotent matrix, i.e., $(I-P^T(PP^T)^{-1}P)^2=I-P^T(PP^T)^{-1}P$, and the last inequality holds since $2\gamma>\gamma^2$.

For the $\hat{\mathcal{A}}$-part, we could establish

$$
\begin{aligned}
&f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^k,\hat{\mathcal{B}}^k,\hat{\mathcal{C}}^k,U^k,V^k,W^k)-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^k,\hat{\mathcal{C}}^k,U^k,V^k,W^k)\\
&=\left\|[\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k-\mathcal{X}^{k+1}\right\|_F^2\\
&\quad-\left\|[\![(\gamma\widetilde{\mathcal{A}}^k+(1-\gamma)\hat{\mathcal{A}}^k)\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k-\mathcal{X}^{k+1}\right\|_F^2\\
&=2\Big\langle\gamma[\![(\widetilde{\mathcal{A}}^k-\hat{\mathcal{A}}^k)\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k,\mathcal{X}^{k+1}\\
&\quad-[\![\hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k\Big\rangle-\left\|\gamma[\![(\widetilde{\mathcal{A}}^k-\hat{\mathcal{A}}^k)\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k]\!]\times_1 U^k\times_2 V^k\times_3 W^k\right\|_F^2\\
&=2\gamma\left\langle U^k(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)})(F^k)^T,\mathcal{X}^{k+1}_{(1)}-U^k\hat{\mathcal{A}}^k_{(1)}(F^k)^T\right\rangle-\gamma^2\left\|U^k(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)})(F^k)^T\right\|_F^2\\
&=2\gamma\langle(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)}),(U^k)^T\mathcal{X}^{k+1}_{(1)}F_k-(U^k)^TU^k\hat{\mathcal{A}}^k_{(1)}(F^k)^TF^k\rangle\\
&\quad-\gamma^2\|U^k(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)})(F^k)^T\|_F^2\\
&=2\gamma\langle(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)}),(U^k)^TU^k(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)})(F^k)^TF^k+\lambda(\widetilde{\mathcal{A}}^k_{(1)}-\mathcal{A}^k_{(1)})\rangle\\
&\quad-\gamma^2\|U^k(\widetilde{\mathcal{A}}^k_{(1)}-\hat{\mathcal{A}}^k_{(1)})(F^k)^T\|_F^2\\
&\geq 2\lambda\gamma\|\widetilde{\mathcal{A}}^k-\hat{\mathcal{A}}^k\|_F^2\\
&=\frac{2\lambda}{\gamma}\|\hat{\mathcal{A}}^{k+1}-\hat{\mathcal{A}}^k\|_F^2,
\end{aligned}
$$

where the fifth equality holds because of (6.16).

For the U-part, we have

$$
\begin{aligned}
&f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k},\hat{\mathcal{C}}^{k},U^{k},V^{k},W^{k})-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k},\hat{\mathcal{C}}^{k},U^{k+1},V^{k},W^{k})\\
&=\left\|[\![\hat{\mathcal{A}}^{k+1}\hat{\mathcal{B}}^{k}\hat{\mathcal{C}}^{k}]\!]\times_1 U^{k}\times_2 V^{k}\times_3 W^{k}-\mathcal{X}^{k+1}\right\|_F^2\\
&\quad-\left\|[\![\hat{\mathcal{A}}^{k+1}\hat{\mathcal{B}}^{k}\hat{\mathcal{C}}^{k}]\!]\times_1(\gamma\widetilde{U}^{k}+(1-\gamma)U^{k})\times_2 V^{k}\times_3 W^{k}-\mathcal{X}^{k+1}\right\|_F^2\\
&=2\Big\langle\gamma[\![\hat{\mathcal{A}}^{k+1}\hat{\mathcal{B}}^{k}\hat{\mathcal{C}}^{k}]\!]\times_1(\widetilde{U}^{k}-U^{k})\times_2 V^{k}\times_3 W^{k},\mathcal{X}^{k+1}\\
&\quad-[\![\hat{\mathcal{A}}^{k+1}\hat{\mathcal{B}}^{k}\hat{\mathcal{C}}^{k}]\!]\times_1 U^{k}\times_2 V^{k}\times_3 W^{k}\Big\rangle\\
&\quad-\left\|\gamma[\![\hat{\mathcal{A}}^{k+1}\hat{\mathcal{B}}^{k}\hat{\mathcal{C}}^{k}]\!]\times_1(\widetilde{U}^{k}-U^{k})\times_2 V^{k}\times_3 W^{k}\right\|_F^2\\
&=2\gamma\left\langle(\widetilde{U}^{k}-U^{k})\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T},\mathcal{X}_{(1)}^{k+1}-U^{k}\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T}\right\rangle-\gamma^2\left\|(\widetilde{U}^{k}-U^{k})\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T}\right\|_F^2\\
&=2\gamma\langle\widetilde{U}^{k}-U^{k},\mathcal{X}_{(1)}^{k+1}F_k(\hat{\mathcal{A}}_{(1)}^{k+1})^{T}-U^{k}\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T}F^{k}(\hat{\mathcal{A}}_{(1)}^{k+1})^{T}\rangle\\
&\quad-\gamma^2\|(\widetilde{U}^{k}-U^{k})\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T}\|_F^2\\
&=2\gamma\langle\widetilde{U}^{k}-U^{k},(\widetilde{U}^{k}-U^{k})\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T}F^{k}(\hat{\mathcal{A}}_{(1)}^{k+1})^{T}+\lambda(\widetilde{U}^{k}-U^{k})\rangle\\
&\quad-\gamma^2\|(\widetilde{U}^{k}-U^{k})\hat{\mathcal{A}}_{(1)}^{k+1}(F^{k})^{T}\|_F^2\\
&\geq 2\lambda\gamma\|\widetilde{U}^{k}-U^{k}\|_F^2\\
&=\frac{2\lambda}{\gamma}\|U^{k+1}-U^{k}\|_F^2,
\end{aligned}
$$

where the fifth equality holds because of (6.17).

Finally, in a similar way, we obtain

$$
\begin{aligned}
&f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k},\hat{\mathcal{C}}^{k},U^{k+1},V^{k},W^{k})\\
&-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k},U^{k+1},V^{k},W^{k})\\
&\geq\frac{2\lambda}{\gamma}\|\hat{\mathcal{B}}^{k+1}-\hat{\mathcal{B}}^{k}\|_F^2,\\
&f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k},U^{k+1},V^{k},W^{k})\\
&-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k},U^{k+1},V^{k+1},W^{k})\\
&\geq\frac{2\lambda}{\gamma}\|V^{k+1}-V^{k}\|_F^2,\\
&f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k},U^{k+1},V^{k+1},W^{k})\\
&-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k+1},U^{k+1},V^{k+1},W^{k})\\
&\geq\frac{2\lambda}{\gamma}\|\hat{\mathcal{C}}^{k+1}-\hat{\mathcal{C}}^{k}\|_F^2,\\
&f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k+1},U^{k+1},V^{k+1},W^{k})\\
&-f(\mathcal{X}^{k+1},\hat{\mathcal{A}}^{k+1},\hat{\mathcal{B}}^{k+1},\hat{\mathcal{C}}^{k+1},U^{k+1},V^{k+1},W^{k+1})\\
&\geq\frac{2\lambda}{\gamma}\|W^{k+1}-W^{k}\|_F^2.
\end{aligned}
$$

The lemma follows by summing the above seven inequalities. ∎

LEMMA 6.6. *Let* $\{\mathbf{y}^k\}_{k=0,1,2,\ldots}$ *be a sequence of iterates generated by Algorithm 2. Then*

$$\sum_{k=0}^{\infty} \|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2 < \infty$$

and

$$\lim_{k\to\infty} \mathbf{y}^k - \mathbf{y}^{k+1} = 0.$$

Proof. From Lemma 6.5, we know

$$\left\|\mathbf{y}^k - \mathbf{y}^{k+1}\right\|^2 \leq \frac{\gamma}{2\lambda}\left(f(\mathbf{y}^k) - f(\mathbf{y}^{k+1})\right)$$

for $k = 0, 1, 2, \ldots$. By summarizing all k, we have

$$\begin{aligned}\sum_{k=0}^{\infty} \left\|\mathbf{y}^k - \mathbf{y}^{k+1}\right\|^2 &\leq \frac{\gamma}{2\lambda}\sum_{k=0}^{\infty}\left(f(\mathbf{y}^k) - f(\mathbf{y}^{k+1})\right)\\ &\leq \frac{\gamma}{2\lambda} f(\mathbf{y}^0) < \infty,\end{aligned}$$

where the second inequality holds because f is always nonnegative. Hence, $\|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2 \to 0$ and hence $\|\mathbf{y}^k - \mathbf{y}^{k+1}\| \to 0$. The lemma is proved. ∎

THEOREM 6.7. *Suppose that the infinite sequence of iterates* $\{\mathbf{y}^k\}$ *generated by Algorithm 2 is bounded. Then, every limit point of* $\{\mathbf{y}^k\}$ *is a stationary point.*

Proof. Since $\{\mathbf{y}^k\} = \{(\mathcal{X}^k, \hat{\mathcal{A}}^k, \hat{\mathcal{B}}^k, \hat{\mathcal{C}}^k, U^k, V^k, W^k)\}$ is bounded, it must have a subsequence $\{\mathbf{y}^{k_i}\} = \{(\mathcal{X}^{k_i}, \hat{\mathcal{A}}^{k_i}, \hat{\mathcal{B}}^{k_i}, \hat{\mathcal{C}}^{k_i}, U^{k_i}, V^{k_i}, W^{k_i})\}$ that converges to a limit point $\mathbf{y}^\infty = (\mathcal{X}^\infty, \hat{\mathcal{A}}^\infty, \hat{\mathcal{B}}^\infty, \hat{\mathcal{C}}^\infty, U^\infty, V^\infty, W^\infty)$. Furthermore, the subsequence $\{\mathbf{y}^{k_i+1}\}$ $= \{(\mathcal{X}^{k_i+1}, \hat{\mathcal{A}}^{k_i+1}, \hat{\mathcal{B}}^{k_i+1}, \hat{\mathcal{C}}^{k_i+1}, U^{k_i+1}, V^{k_i+1}, W^{k_i+1})\}$ also converges to the limit point $\mathbf{y}^\infty$ by Lemma 6.6.

Because $\mathcal{X}^{k_i} - \mathcal{X}^{k_i+1} \to 0$, we get $\widetilde{\mathcal{X}}^{k_i} - \mathcal{X}^{k_i} = 0$ as $i \to \infty$. It yields from (6.15) that

$$\left(I - P^T(PP^T)^{-1}P\right)\operatorname{vec}\left(\left[\!\left[\hat{\mathcal{A}}^{k_i}\hat{\mathcal{B}}^{k_i}\hat{\mathcal{C}}^{k_i}\right]\!\right] \times_1 U^{k_i} \times_2 V^{k_i} \times_3 W^{k_i} - \mathcal{X}^{k_i}\right) \to 0$$

as $i \to \infty$. That is,

$$\left(I - P^T(PP^T)^{-1}P\right)\operatorname{vec}\left(\left[\!\left[\hat{\mathcal{A}}^{\infty}\hat{\mathcal{B}}^{\infty}\hat{\mathcal{C}}^{\infty}\right]\!\right] \times_1 U^{\infty} \times_2 V^{\infty} \times_3 W^{\infty} - \mathcal{X}^{\infty}\right) = 0.$$

By $\hat{\mathcal{A}}^{k_i} - \hat{\mathcal{A}}^{k_i+1} \to 0$, we have $\widetilde{\mathcal{A}}^{k_i} - \hat{\mathcal{A}}^{k_i} \to 0$. Because $\hat{\mathcal{B}}^{k_i} \to \hat{\mathcal{B}}^\infty$, $\hat{\mathcal{C}}^{k_i} \to \hat{\mathcal{C}}^\infty$, $V^{k_i} \to V^\infty$, and $W^{k_i} \to W^\infty$ as $i \to \infty$, the subsequence $\{F^{k_i}\}$ converges to F^∞ that is bounded above. It yields from (6.16) that

$$\begin{aligned}(U^\infty)^T\left(U^\infty\hat{\mathcal{A}}^\infty_{(1)}(F^\infty)^T - \mathcal{X}^\infty_{(1)}\right)F^\infty &= \lim_{i\to\infty}(U^{k_i})^T(U^{k_i}\widetilde{\mathcal{A}}^{k_i}_{(1)}(F^{k_i})^T - \mathcal{X}^{k_i+1}_{(1)})F^{k_i}\\ &= \lim_{i\to\infty}\lambda(\hat{\mathcal{A}}^{k_i}_{(1)} - \widetilde{\mathcal{A}}^{k_i}_{(1)})\\ &= 0.\end{aligned}$$

By $U^{k_i} - U^{k_i+1} \to 0$, we have $\widetilde{U}^{k_i} - U^{k_i} \to 0$. It yields from (6.17) that

$$\begin{aligned}\Big(U^\infty \hat{\mathcal{A}}^\infty_{(1)}(F^\infty)^T - \mathcal{X}^\infty_{(1)}\Big) F^\infty(\hat{\mathcal{A}}^\infty_{(1)})^T &= \lim_{i\to\infty} (\widetilde{U}^{k_i}\hat{\mathcal{A}}^{k_i+1}_{(1)}(F^{k_i})^T - \mathcal{X}^{k_i+1}_{(1)})F^{k_i}(\hat{\mathcal{A}}^{k_i+1}_{(1)})^T \\ &= \lim_{i\to\infty} \lambda(U^{k_i}_{(1)} - \widetilde{U}^{k_i}_{(1)}) \\ &= 0.\end{aligned}$$

Finally, by a similar discussion we know that $(V^\infty)^T\big(V^\infty \hat{\mathcal{B}}^\infty_{(2)}(G^\infty)^T - \mathcal{X}^\infty_{(2)}\big)G^\infty = 0$, $(W^\infty)^T\big(W^\infty \hat{\mathcal{C}}^\infty_{(3)}(H^\infty)^T - \mathcal{X}^\infty_{(3)}\big)H^\infty = 0$, $\big(V^\infty \hat{\mathcal{B}}^\infty_{(2)}(G^\infty)^T - \mathcal{X}^\infty_{(2)}\big)G^\infty(\hat{\mathcal{B}}^\infty_{(2)})^T = 0$, and $\big(W^\infty \hat{\mathcal{C}}^\infty_{(3)}(H^\infty)^T - \mathcal{X}^\infty_{(3)}\big)H^\infty(\hat{\mathcal{C}}^\infty_{(3)})^T = 0$. Hence, by Lemma 6.3, $\mathbf{y}^\infty = (\mathcal{X}^\infty, \hat{\mathcal{A}}^\infty, \hat{\mathcal{B}}^\infty, \hat{\mathcal{C}}^\infty, U^\infty, V^\infty, W^\infty)$ is a stationary point of (6.6). ∎

Theorem 6.7 shows that every limit point of iterates generated by Algorithm 2 is a stationary point. Next, using the KL property, we prove that the sequence of iterates from Algorithm 2 converges to a stationary point. The analysis in the remainder of this section is based on the outline of [2, 3].

We give a lower bound on the progress made by one iteration.

LEMMA 6.8. *Suppose that the infinite sequence* $\{\mathbf{y}^k\}$ *generated by Algorithm* 2 *is bounded. Then, there is a positive constant* ς *such that*

$$\|\mathbf{y}^k - \mathbf{y}^{k+1}\| \geq \varsigma \|\Pi_\Omega(\nabla f(\mathbf{y}^k))\|.$$

Proof. From (6.15) and $\mathcal{X}^{k+1} - \mathcal{X}^k = \gamma(\widetilde{\mathcal{X}}^k - \mathcal{X}^k)$, it yields that

$$\begin{aligned}&\|2(I - P^T(PP^T)^{-1}P^T)\mathrm{vec}(\llbracket \hat{\mathcal{A}}^k\hat{\mathcal{B}}^k\hat{\mathcal{C}}^k \rrbracket \times_1 U^k \times_2 V^k \times_3 W^k - \mathcal{X}^k)\| \\ &= 2(1+\lambda)\|\mathrm{vec}(\widetilde{\mathcal{X}}^k - \mathcal{X}^k)\| \\ &\leq (2+2\lambda)\|\mathrm{vec}(\mathcal{X}^{k+1} - \mathcal{X}^k)\|.\end{aligned}$$

Owing to the boundedness of $\{\mathbf{y}^k\}$, there exists a compact convex set $\mathbb{Y}$ containing $\{\mathbf{y}^k\}$. Since f is a polynomial, the gradient ∇f is Lipschitz in $\mathbb{Y}$ with a Lipschitz constant L_f, i.e., $\|\nabla f(\mathbf{y}_1) - \nabla f(\mathbf{y}_2)\| \leq L_f\|\mathbf{y}_1 - \mathbf{y}_2\|$ for all $\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{Y}$. The following process is similar for discussions on partial derivatives of f in $\hat{\mathcal{A}}^k$, $\hat{\mathcal{B}}^k$, $\hat{\mathcal{C}}^k$, U^k, V^k, and W^k. Let us take the partial derivatives of f in $\hat{\mathcal{A}}^k$ for example:

$$\begin{aligned}\|\nabla_{\mathrm{vec}(\hat{\mathcal{A}})} f(\mathbf{y}^k)\| &= \|\nabla_{\mathrm{vec}(\hat{\mathcal{A}})} f(\mathbf{y}^k) - \nabla_{\mathrm{vec}(\hat{\mathcal{A}})} f(\mathcal{X}^{k+1}, \widetilde{\mathcal{A}}^k, \hat{\mathcal{B}}^k, \hat{\mathcal{C}}^k, U^k, V^k, W^k)\| \\ &\quad + \|\nabla_{\mathrm{vec}(\hat{\mathcal{A}})} f(\mathcal{X}^{k+1}, \widetilde{\mathcal{A}}^k, \hat{\mathcal{B}}^k, \hat{\mathcal{C}}^k, U^k, V^k, W^k)\| \\ &\leq L_f\|(\mathcal{X}^k - \mathcal{X}^{k+1}, \hat{\mathcal{A}}^k - \widetilde{\mathcal{A}}^k, 0, 0, 0, 0, 0)\| + 2\lambda\|\widetilde{\mathcal{A}}^k - \hat{\mathcal{A}}^k\|_F \\ &\leq (L_f + 2\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|,\end{aligned}$$

where the first inequality holds because of the optimization condition of subproblems (6.4). Similarly, we have

$$\begin{aligned}\|\nabla_{\mathrm{vec}(\hat{\mathcal{B}})} f(\mathbf{y}^k)\| &\leq (L_f + 2\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|, \\ \|\nabla_{\mathrm{vec}(\hat{\mathcal{C}})} f(\mathbf{y}^k)\| &\leq (L_f + 2\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|, \\ \|\nabla_{\mathrm{vec}(U)} f(\mathbf{y}^k)\| &\leq (L_f + 2\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|, \\ \|\nabla_{\mathrm{vec}(V)} f(\mathbf{y}^k)\| &\leq (L_f + 2\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|, \\ \|\nabla_{\mathrm{vec}(W)} f(\mathbf{y}^k)\| &\leq (L_f + 2\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|.\end{aligned}$$

In summation, we have

$$\|\Pi_\Omega(\nabla f(\mathbf{y}^k))\| \le (2 + 6L_f + 14\lambda)\|\mathbf{y}^k - \mathbf{y}^{k+1}\|.$$

This lemma is valid when we set $\varsigma := (2 + 6L_f + 14\lambda)^{-1}$. ∎

LEMMA 6.9. *Let $\mathbf{y}^*$ be one of the limiting points of $\{\mathbf{y}^k\}$. Assume that $\mathbf{y}^0$ satisfies $\mathbf{y}^0 \in \mathbb{B}(\mathbf{y}^*, \rho) \subseteq \mathbb{V}$, where*

$$\rho > \frac{\gamma\mu}{2\lambda\varsigma(1-\theta)}|f(\mathbf{y}^0) - f(\mathbf{y}^*)|^{1-\theta} + \|\mathbf{y}^0 - \mathbf{y}^*\|. \tag{6.18}$$

Then, we have the following assertions:

$$\mathbf{y}^k \in \mathbb{B}(\mathbf{y}^*, \rho) \qquad \forall\, k = 0, 1, \ldots \tag{6.19}$$

and

$$\sum_{k=0}^{\infty} \|\mathbf{y}^k - \mathbf{y}^{k+1}\| \le \frac{\gamma\mu}{2\lambda\varsigma(1-\theta)}|f(\mathbf{y}^0) - f(\mathbf{y}^*)|^{1-\theta}. \tag{6.20}$$

Proof. We prove (6.19) by induction. Obviously, $\mathbf{y}^0 \in \mathbb{B}(\mathbf{y}^*, \rho)$ when $k = 0$. Second, we assume there is an integer K such that

$$\mathbf{y}^k \in \mathbb{B}(\mathbf{y}^*, \rho) \qquad \forall\, 0 \le k \le K,$$

which means that the KL property holds at these points. Now, we are going to prove that $\mathbf{y}^{K+1} \in \mathbb{B}(\mathbf{y}^*, \rho)$.

We define a scalar function

$$\varphi(\alpha) := \frac{1}{1-\theta}|\alpha - f(\mathbf{y}^*)|^{1-\theta}. \tag{6.21}$$

It is easy to see that $\varphi(\cdot)$ is a concave function and $\varphi'(\alpha) = |\alpha - f(\mathbf{y}^*)|^{-\theta}$ if $\alpha \ge f(\mathbf{y}^*)$. Then, for $0 \le k \le K$, it yields that

$$\begin{aligned}
\varphi(f(\mathbf{y}^k)) - \varphi(f(\mathbf{y}^{k+1})) &\ge \varphi'(f(\mathbf{y}^k))(f(\mathbf{y}^k) - f(\mathbf{y}^{k+1})) \\
&\ge \frac{1}{|f(\mathbf{y}^k) - f(\mathbf{y}^*)|^\theta}\frac{2\lambda}{\gamma}\|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2 \qquad \text{[Lemma 6.5]} \\
&\ge \frac{2\lambda}{\gamma\mu}\frac{\|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2}{\|\Pi_\Omega(\nabla f(\mathbf{y}^k))\|} \qquad \text{[KL property]} \\
&\ge \frac{2\lambda\varsigma}{\gamma\mu}\frac{\|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2}{\|\mathbf{y}^k - \mathbf{y}^{k+1}\|} \qquad \text{[Lemma 6.8]} \\
&= \frac{2\lambda\varsigma}{\gamma\mu}\|\mathbf{y}^k - \mathbf{y}^{k+1}\|.
\end{aligned}$$

By summarizing k from 0 to K, we have

$$\begin{aligned}
\sum_{k=0}^{K} \|\mathbf{y}^k - \mathbf{y}^{k+1}\| &\le \frac{\gamma\mu}{2\lambda\varsigma}\sum_{k=0}^{K}[\varphi(f(\mathbf{y}^k)) - \varphi(f(\mathbf{y}^{k+1}))] \\
&= \frac{\gamma\mu}{2\lambda\varsigma}[\varphi(f(\mathbf{y}^0)) - \varphi(f(\mathbf{y}^{K+1}))] \\
&\le \frac{\gamma\mu}{2\lambda\varsigma}\varphi(f(\mathbf{y}^0)).
\end{aligned} \tag{6.22}$$

Hence, it follows from (6.22) and (6.18) that

$$\|\mathbf{y}^{K+1}-\mathbf{y}^*\| \le \sum_{k=0}^{K}\|\mathbf{y}^{k+1}-\mathbf{y}^k\| + \|\mathbf{y}^0-\mathbf{y}^*\| \le \frac{\gamma\mu}{2\lambda\varsigma}\varphi(f(\mathbf{y}^0)) + \|\mathbf{y}^0-\mathbf{y}^*\| < \rho$$

which means (6.19) holds. Moreover, we obtain (6.20) by letting $K \to \infty$ in (6.22) and using (6.21). ∎

THEOREM 6.10. *Assume that Algorithm 2 produces a bounded sequence* $\{\mathbf{y}^k\}$. *Then,*

$$\sum_{k=0}^{\infty}\|\mathbf{y}^{k+1}-\mathbf{y}^k\| < +\infty,$$

which implies that the entire sequence $\{\mathbf{y}^k\}$ *converges.*

Proof. Because $\{\mathbf{y}^k\}$ is bounded, it must have a limit point $\mathbf{y}^*$ and there is an index k_0 such that $\mathbf{y}^{k_0} \in \mathbb{B}(\mathbf{y}^*, \rho)$. If we regard $\mathbf{y}^{k_0}$ as an initial point, Lemma 6.9 holds. The entire sequence $\{\mathbf{y}^k\}$ satisfies (6.20). Hence, this theorem is proved. ∎

We are now able to give the proof of the main theorem in this section.

Proof of Theorem 6.2. If $\mathbf{y}^k = \mathbf{y}^{k+1}$, Lemma 6.4 shows that $\mathbf{y}^k$ is a critical point of (6.6).

Otherwise, $\{\mathbf{y}^k\}$ is an infinite sequence. Assume that this sequence is bounded. Theorems 6.7 and 6.10 show that this sequence converges to a critical point $\mathbf{y}^*$ of (6.6). Finally, by consulting [2], we establish the local convergence rate. Without loss of generality, we assume that $\mathbf{y}^0 \in \mathbb{B}(\mathbf{y}^*, \rho)$. Let

$$\Delta_k := \sum_{i=k}^{\infty}\|\mathbf{y}^i-\mathbf{y}^{i+1}\| \ge \|\mathbf{y}^k-\mathbf{y}^*\|. \tag{6.23}$$

From Lemma 6.9, we have

$$\begin{aligned}
\Delta_k &\le \frac{\gamma\mu}{2\lambda\varsigma(1-\theta)}|f(\mathbf{y}^k)-f(\mathbf{y}^*)|^{1-\theta}\\
&= \frac{\gamma\mu}{2\lambda\varsigma(1-\theta)}\left[|f(\mathbf{y}^k)-f(\mathbf{y}^*)|^{\theta}\right]^{\frac{1-\theta}{\theta}}\\
&\le \frac{\gamma\mu}{2\lambda\varsigma(1-\theta)}\mu^{\frac{1-\theta}{\theta}}\|\Pi_\Omega(\nabla f(\mathbf{y}^k))\|^{\frac{1-\theta}{\theta}} \qquad \text{[KL property]}\\
&\le \frac{\gamma\mu}{2\lambda\varsigma(1-\theta)}\left(\frac{\mu}{\varsigma}\right)^{\frac{1-\theta}{\theta}}\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^{\frac{1-\theta}{\theta}} \qquad \text{[Lemma 6.8]}\\
&= \frac{\gamma\mu^{1/\theta}}{2\lambda\varsigma^{1/\theta}(1-\theta)}\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^{\frac{1-\theta}{\theta}}. \qquad (6.24)
\end{aligned}$$

(1) In the case $\theta = \frac{1}{2}$, we have $\frac{1-\theta}{\theta} = 1$ immediately. Then, the inequality (6.24) gives

$$\Delta_k \le \frac{\gamma\mu^{1/\theta}}{2\lambda\varsigma^{1/\theta}(1-\theta)}(\Delta_k - \Delta_{k+1}),$$

which means that

$$\Delta_{k+1} \le \frac{\gamma\mu^{1/\theta} - 2\lambda\varsigma^{1/\theta}(1-\theta)}{\gamma\mu^{1/\theta}}\Delta_k. \tag{6.25}$$

Let $\nu := \frac{\gamma\mu^{1/\theta}-2\lambda\varsigma^{1/\theta}(1-\theta)}{\gamma\mu^{1/\theta}}$. From (6.23) and (6.25), we know $\|\mathbf{y}^k - \mathbf{y}^*\| \leq \Delta_k \leq \nu\Delta_{k-1} \leq \cdots \leq \nu^k\Delta_0$, where Δ_0 is finite by Theorem 6.10. Hence, assertion (1) is valid by taking $\eta := \Delta_0$.

(2) Let $\chi^{\frac{1-\theta}{\theta}} := \frac{\gamma\mu^{1/\theta}}{2\lambda\varsigma^{1/\theta}(1-\theta)}$. It yields from (6.24) that

$$\Delta_k^{\frac{\theta}{1-\theta}} \leq \chi(\Delta_k - \Delta_{k+1}).$$

We define $h(\alpha) := \alpha^{-\frac{\theta}{1-\theta}}$. Obviously, $h(s)$ is monotonically decreasing. Then,

$$\begin{aligned}\frac{1}{\chi} &\leq h(\Delta_k)(\Delta_k - \Delta_{k+1})\\ &= \int_{\Delta_{k+1}}^{\Delta_k} h(\Delta_k)\mathrm{d}\alpha\\ &\leq \int_{\Delta_{k+1}}^{\Delta_k} h(\alpha)\mathrm{d}\alpha\\ &= -\frac{1-\theta}{2\theta-1}\left(\Delta_k^{-\frac{2\theta-1}{1-\theta}} - \Delta_{k+1}^{-\frac{2\theta-1}{1-\theta}}\right).\end{aligned}$$

We denote $\vartheta := -\frac{2\theta-1}{1-\theta} < 0$ since $\theta \in (\frac{1}{2}, 1)$. Then, it follows from the above inequality that

$$\Delta_{k+1}^{\vartheta} - \Delta_k^{\vartheta} \geq \frac{-\vartheta}{\chi} =: \varpi > 0,$$

which gives

$$\Delta_k \leq [\Delta_0^{\vartheta} + k\varpi]^{\frac{1}{\vartheta}} \leq (k\varpi)^{\frac{1}{\vartheta}}.$$

We obtain the assertion (2) by taking $\eta := \varpi^{\frac{1}{\vartheta}}$. ∎

7. Numerical tests. In this section, we are going to compare the triple decomposition tensor recovery model (6.2) with the CP decomposition tensor recovery model and the Tucker decomposition tensor recovery model. For the triple decomposition tensor recovery model, we use the bilevel form to reduce its complexity. Let $A \in \Re^{n_1\times r}$, $B \in \Re^{n_2\times r}$, and $C \in \Re^{n_3\times r}$. The CP tensor $[[A, B, C]] \in \Re^{n_1\times n_2\times n_3}$ has entries

$$[[A, B, C]]_{ijt} = \sum_{p=1}^{r} a_{ip}b_{jp}c_{tp}.$$

In addition, let $\mathcal{D} \in \Re^{r\times r\times r}$ be a core tensor. The Tucker tensor $[[\mathcal{D}; A, B, C]]$ has entries

$$[[\mathcal{D}; A, B, C]]_{ijt} = \sum_{p,q,s=1}^{r} a_{ip}b_{jq}c_{ts}d_{pqs}.$$

Then, the CP tensor recovery model and the Tucker tensor recovery model could be represented by

$$\begin{aligned}\min\ & \|[[A, B, C]] - \mathcal{X}\|_F^2\\ \text{s.t.}\ & \mathbb{P}(\mathcal{X}) = \mathbf{d}\end{aligned}$$

and

$$\begin{aligned} &\min \ \|[[\mathcal{D}; A, B, C]] - \mathcal{X}\|_F^2 \\ &\text{s.t.} \ \ \mathbb{P}(\mathcal{X}) = \mathbf{d}, \end{aligned}$$

respectively. These two models are solved by variants of TriR in Algorithm 2.

7.1. ORL face data. Next, we apply the triple decomposition tensor recovery method, the CP decomposition tensor recovery method, and the Tucker decomposition tensor recovery method for the ORL dataset of faces. Original images of a person from the ORL dataset are illustrated in the first line of Figure 8. We sample 50 percent of the pixels of these images as shown in the second line of Figure 8.

FIG. 8. *Original images are illustrated in the first row. Samples of* 50 *percent of the pixels are illustrated in the second row. The third to last rows report the recovered images by the proposed method, CP tensor recovery, and Tucker tensor recovery, respectively.*

Once a tensor $\mathcal{T}_{rec}$ is recovered, we calculate the relative error

$$\text{RE} = \frac{\|\mathcal{T}_{rec} - \mathcal{T}_{truth}\|_F}{\|\mathcal{T}_{truth}\|_F}.$$

We set $r_1 = 5t, r_2 = 4t, r_3 = 10$, and t is an integer from 1 to 5. We choose $\text{TriRank} = \lfloor \sqrt{\frac{r_1 r_2 r_3}{r_1 + r_2 + r_3}} \rfloor$ and $\text{CPRank} = \lceil \frac{n_1 r_1 + n_2 r_2 + n_3 r_3 + r_1 r_2 r_3}{n_1 + n_2 + n_3} \rceil$, where $\lfloor \alpha \rfloor$ means the largest integer less than or equal to α and $\lceil \alpha \rceil$ denotes the nearest integer to α. The relative error of recovered tensor by the triple decomposition tensor recovery method, the CP decomposition tensor recovery method, and the Tucker decomposition tensor recovery method is reported in Figure 9. As the rank increases, the relative error of the recovered tensor by each method decreases. It is easy to see that the relative error corresponding to the proposed triple tensor recovery method decreases quickly. The new method performs slightly better than the CP decomposition tensor recovery method and the Tucker decomposition tensor recovery method. We

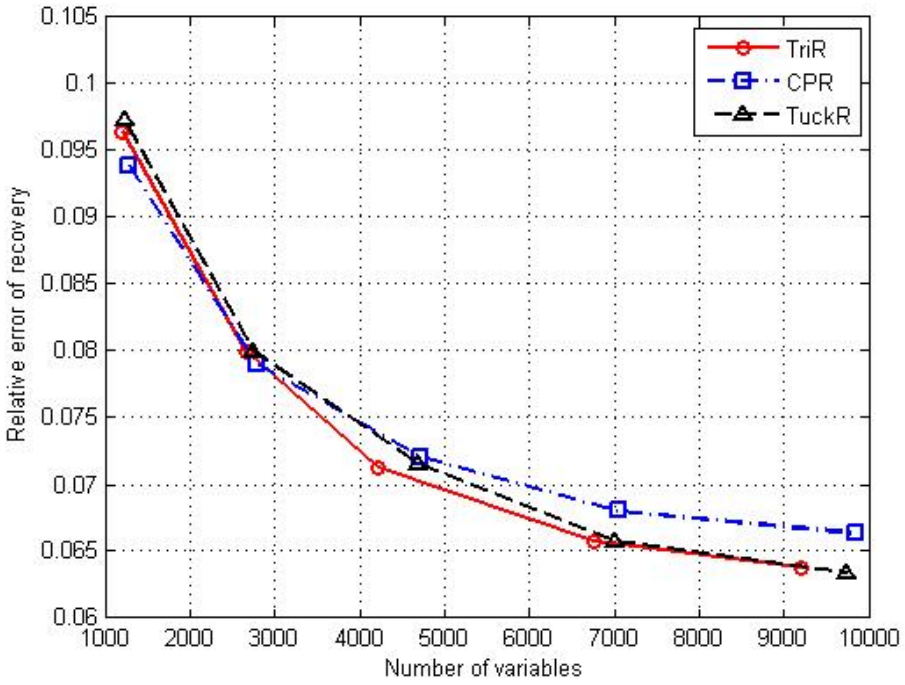

FIG. 9. *Relative error of the recovered tensors.*

TABLE 1
Relative error of estimated images from the McGill Calibrated Color Image Data.

Methods	Number of variables	Relative error of estimated images		
		flower	grape	butterfly
TriR	18672	6.26%	11.18%	14.30%
CPR	19662	6.30%	11.66%	12.64%
TuckR	19593	6.20%	11.23%	15.23%

take rank $t = 5$, for instance. The recovered tensors by the triple tensor recovery method, the CP decomposition tensor recovery method, and the Tucker decomposition tensor recovery method are illustrated in lines 3–5 of Figure 8. Clearly, the quality of images recovered by proposed triple decomposition tensor recovery method is competitive compared with the CP decomposition tensor recovery method and the Tucker decomposition tensor recovery method.

7.2. McGill Calibrated Color Image Data. We now investigate the McGill Calibrated Color Image Data [14]. We choose three color images: flower, grape, and butterfly, which are illustrated in the first column of Figure 10. By resizing the color image, we get a $144 \times 192 \times 3$ tensor. We randomly choose 50 percent entries of the color image tensor. Tensors with missing entries are shown in the second column of Figure 10. By setting $r_1 = 36, r_2 = 48, r_3 = 3$, TriRank = 7, and CPRank = 58, we report the number of variables and relative error of the estimated image of the triple decomposition tensor recovery method, the CP decomposition tensor recovery method, and the Tucker decomposition tensor recovery method in

FIG. 10. *Original images from the McGill dataset are illustrated in the first column. Samples of* 50 *percent of the pixels are illustrated in the second column. The third to last columns report the recovered images by the proposed method, CP tensor recovery, and Tucker tensor recovery, respectively.*

Table 1. Color images estimated by the triple decomposition tensor recovery method, the CP decomposition tensor recovery method, and the Tucker decomposition tensor recovery method are shown in the third, the fourth, and the last columns, respectively. Obviously, the proposed triple tensor recovery method generates unambiguous color images and is competitive when compared with some existing methods.

8. Concluding remarks. In this paper, we introduce triple decomposition for third order tensors. A third order tensor is decomposed to three low rank factor tensors, which inherit the information and features of the three modes of the third order tensor. The lowest rank of the factor tensors is called the triple rank, which is not greater than the middle value of the Tucker rank, and is strictly less than the middle value of the Tucker rank in substantial cases. The number of parameters in the bilevel form of standard triple decomposition is less than the number of parameters of Tucker decomposition in substantial cases. Thus, triple decomposition does not cost more than Tucker decomposition, the three low rank factor tensors inherit the features of the three modes, and the Tucker core is also decomposed simultaneously. These properties of triple decomposition may be useful in tensor recovery problems of transportation and internet, where the three modes of the third order tensor have strong temporal, spatial, and periodic meanings [16, 17, 21]. In this paper, we present a tensor recovery algorithm based upon triple decomposition, and apply it to some video and image problems. Two possible future research topics are as follows: (1) to apply triple decomposition to tensor recovery problems for transportation or internet traffic data and (2) to extend the sparse Tucker decomposition in [18] to sparse triple decomposition as discussed in section 3.

Acknowledgments. We are thankful to Haibin Chen, Ziyan Luo, and three anonymous referees for their helpful comments, and to Qun Wang, who drew Figure 1 for us.

REFERENCES

[1] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, *Scalable tensor factorizations for incomplete data*, Chemom. Intell. Lab. Syst., 106 (2011), pp. 41–56.

[2] H. Attouch and J. Bolte, *On the convergence of the proximal algorithm for nonsmooth functions involving analytic features*, Math. Program., 116 (2009), pp. 5–16.

[3] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, *Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality*, Math. Oper. Res., 35 (2010), pp. 438–457.

[4] K. Batselier, *The Trouble with Tensor Ring Decompositions*, preprint, https://arxiv.org/abs/1811.03813, 2018.

[5] J. Bolte, A. Daniilidis, and A. Lewis, *The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems*, SIAM J. Optim., 17 (2007), pp. 1205–1223, https://doi.org/10.1137/050644641.

[6] J. Chang and Y. Chan, *Essential Uniqueness of Triple Decomposition of Third Order Tensors*, manuscript, 2020.

[7] Y. Chen, W. Sun, M. Xi, and J. Yuan, *A seminorm regularized alternating least squares algorithm for canonical tensor decomposition*, J. Comput. Appl. Math., 347 (2019), pp. 296–313.

[8] L. De Lathauwer, *Decompositions of a higher-order tensor in block terms—Part* II: *Definitions and uniqueness*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1033–1066, https://doi.org/10.1137/070690729.

[9] L. De Lathauwer, D. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278, https://doi.org/10.1137/S0895479896305696.

[10] B. Jiang, F. Yang, and S. Zhang, *Tensor and its Tucker core: The invariance relationships*, Numer. Linear Algebra Appl., 24 (2017), e2086.

[11] M. Kilmer, K. Braman, N. Hao, and R. Hoover, *Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 148–172, https://doi.org/10.1137/110837711.

[12] T. G. Kolda and B. Bader, *Tensor decompositions and applications*, SIAM Rev., 51 (2009), pp. 455–500, https://doi.org/10.1137/07070111X.

[13] S. Łojasiewicz, *Une propriété topologique des sous-ensembles analytiques réels*, Colloques Internationaux du C.N.R.S. 117, Les Équations aux Dérivées Partielles, Éditions du Centre National de la Recherche Scientifique, Paris, 1963, pp. 87–89.

[14] A. Olmos and F. A. Kingdom, *A biologically inspired algorithm for the recovery of shading and reflectance images*, Perception, 33 (2004), pp. 1463–1473.

[15] F. S. Samaria and A. C. Harter, *Parameterisation of a stochastic model for human face identification*, in Proceedings of 1994 IEEE Workshop on Applications of Computer Vision, IEEE, 1994, pp. 138–142.

[16] H. Tan, Z. Yang, G. Feng, W. Wang, and B. Ran, *Correlation analysis for tensor-based traffic data imputation method*, Procedia Soc. Behav. Sci., 96 (2013), pp. 2611–2620.

[17] K. Xie, L. Wang, X. Wang, G. Xie, J. Wen, and G. Zhang, *Accurate recovery of internet traffic data: A tensor completion approach*, in IEEE INFOCOM 2016 – The 35th Annual IEEE International Conference on Computer Communications, San Francisco, 2016, pp. 1–9.

[18] Y. Xu, *Alternating proximal gradient method for sparse nonnegative Tucker decomposition*, Math. Program. Comput., 7 (2015), pp. 39–70.

[19] Y. Xu and W. Yin, *A block coordinate method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion*, SIAM J. Imaging Sci., 6 (2013), pp. 1758–1789, https://doi.org/10.1137/120887795.

[20] Q. Zhao, G. Zhou, S. Xie, L. Zhang, and A. Cichocki, *Tensor Ring Decomposition*, preprint, https://arxiv.org/abs/1606.05535, 2016.

[21] H. Zhou, D. Zhang, K. Xie, and Y. Chen, *Spatio-temporal tensor completion for imputing missing internet traffic data*, in Proceedings of the IEEE 34th International Performance Computing and Communications Conference (IPCCC), 2015, pp. 1–7.

[22] P. Zhou, C. Lu, Z. Lin, and C. Zhang, *Tensor factorization for low-rank tensor completion*, IEEE Trans. Image Process., 27 (2018), pp. 1152–1163.