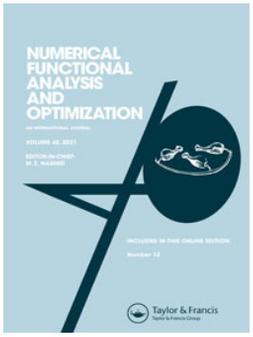


Numerical Functional Analysis and Optimization

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/lnfa20

Tensor CUR Decomposition under T-Product and Its Perturbation

Juefei Chen, Yimin Wei & Yanwei Xu

To cite this article: Juefei Chen, Yimin Wei & Yanwei Xu (2022): Tensor CUR Decomposition under T-Product and Its Perturbation, Numerical Functional Analysis and Optimization, DOI: 10.1080/01630563.2022.2056198

To link to this article: https://doi.org/10.1080/01630563.2022.2056198

NUMERICAL FUNCTIONAL ANALYSIS AND OPTIMIZATION
https://doi.org/10.1080/01630563.2022.2056198

Tensor CUR Decomposition under T-Product and Its Perturbation

Juefei Chen[a], Yimin Wei[b], and Yanwei Xu[c]

[a]School of Mathematical Sciences, Fudan University, Shanghai, P. R. of China; [b]School of Mathematical Sciences and Shanghai Key Laboratory of Contemporary Applied Mathematics, Fudan University, Shanghai, P. R. of China; [c]Theory Lab, 2012 Labs Huawei Tech. Investment Co., Ltd, Shatin, New Territory, P. R. of China Hong Kong.

ABSTRACT

In order to process the large-scale data, a useful tool in dimensionality reduction of a matrix, *the CUR decomposition* has been developed, which can compress the huge matrix with its original elements. Tensor-tensor decompositions have become prevalent and a new multiplication of a tensor based on the T-product has been presented for the tensor computation. Using the T-product, we propose a dimensionality reduction tool of three-order tensor called the *T-product CUR decomposition* (t-CUR decomposition for short) and analyze its stability of the perturbation. The t-CUR decomposition can reduce the size of a large-scale tensor with its original entries, its perturbation error bound is refined in the first order of the noise tensor under the spectrum norm. Numerical tests are provided to verify the results of our theoretical error analysis as well.

ARTICLE HISTORY
Received 4 May 2021
Revised 16 March 2022
Accepted 16 March 2022

KEYWORDS
Tensor CUR decomposition; T-product; perturbation; dimensionality reduction

AMS CLASSIFICATION
15A18; 15A69; 65F15; 65F10

1. Introduction

In the era of the big data, we are faced with the need of processing large-scale data. These data are mostly in the three or even higher order dimensions, whose orders of magnitude can reach billions. Such a large amount of data is a great challenge for the computer storage and computational complexity. Therefore, simplifying data becomes necessary. In mathematics, one usually uses the matrix or tensor decompositions or the low-rank approximations to capture the essential information of the initial data, reduce its size or simplify the computation [1, 47–50, 52]. Using tensor methods can greatly save the data storage and decrease computational costs in processing data in the scientific computing and engineering.

In numerical linear algebra, there are various matrix decompositions [2, 51], such as the LU decomposition, the QR decomposition, the Schur

CONTACT Y. Wei ymwei@fudan.edu.cn School of Mathematical Sciences and Shanghai Key Laboratory of Contemporary Applied Mathematics, Fudan University, Shanghai, 200433, P. R. of China.

decomposition and the singular value decomposition (SVD), etc. Randomized algorithms for matrix approximations of the Tucker decomposition and the tensor train decompositions have been developed by Che and Wei [3]. The matrix CUR decomposition [4–11, 45, 46] have been proposed and widely investigated in recent years, since it can utilize the self-expression of the matrix effectively, using representative columns and rows to reduce the size of the initial matrix. As a result, compared with other methods, the CUR decomposition does not change the value of matrix elements, which has more interpretability.

When we process tensors, such as images and videos, tensor decompositions or approximation methods are quite different from matrix methods due to the extra dimensions. Several exact decompositions of tensor have been proposed, like the CP decomposition, the Tucker decomposition, the tensor train decomposition and the tensor ring decomposition [1]. The recent results on the tensor equations, the tensor complementarity problems, and tensor low-rank approximations can be found in [12–24]. The CUR-type tensor decomposition theory based on mode-k-product [9] have also been designed, though the mode-k-product is more or less complicated to compute.

The tensor T-product and tensor operations under the T-product [25, 26] have been presented, which builds a connection between the tensor-tensor decomposition and the matrix case. These tools inspire us to tackle the tensor by the matrix approach. For example, the matrix SVD can be extended to the tensor case under the T-product as the t-SVD [27, 28, 53]; the function of matrices is extended to the tensor t-function in [25]. Tarzanagh and Michailidis proposed the t-CX tensor low-rank approximation and t-CUR approximation for the special case in [29], which use subtensors to express the original data tensor approximately.

In this paper, due to the strong interpretability of the matrix CUR decomposition, we extend the matrix CUR decomposition to the tensor CUR under the T-product framework and propose an exact t-CUR decomposition. We will show that the tensor $\mathcal{A}$ can be factorized as $\mathcal{A} = \mathcal{C} * \mathcal{U}^{\dagger} * \mathcal{R}$ under specific tensor rank condition. Its perturbation analysis and theoretical error bound are established as well, including the basic error bound and the rank-forced error bound. In the last part of this paper, the results of our perturbation analysis are simulated and tested with several examples.

This paper is organized as follows. We review the concept of the tensor T-product and the matrix CUR decomposition in Section 2. We establish the t-CUR decomposition and carry out the perturbation analysis in Sections 3 and 4. We perform numerical simulations in Section 5.

2. Preliminaries

There have been rich results on the matrix and tensor CUR decompositions [4–11, 29, 30], in view of their frequent references in the later, we will review them in this section.

2.1. Notations

In this paper, capital italics $A \in \mathbb{K}^{n_1 \times n_2}$ denotes matrix A with the order $n_1 \times n_2$, and capital cursive italics $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ represents the tensor $\mathcal{A}$ with size $n_1 \times n_2 \times n_3$. $\mathbb{K}$ is field $\mathbb{R}$ or $\mathbb{C}$. We use lowercase bold italics to denote vectors, such as $\boldsymbol{v}$.

For convenience, this paper will adopt the notation similar to that of MATLAB for the matrix and tensor, that is, $\{1:n\}$ denotes integer set $\{1, 2, ..., n\}$, $A(I, J)$ and $\mathcal{A}(I, J, K)$ denote the submatrix and subtensor, respectively, where $I \subset \{1:n_1\}$, $J \subset \{1:n_2\}$, $K \subset \{1:n_3\}$. In particular, If $I = \{1:n_1\}$, then $\mathcal{A}(I, J, K) = \mathcal{A}(:, J, K)$. Besides, $A(i, j)$ and $\mathcal{A}(i, j, k)$ represent elements in matrix and tensor. $|S|$ stands for the number of elements in set S.

$A^{\dagger}$ and A^{*} denote the Moore-Penrose inverse [31, 32] and conjugate transpose of A, respectively; $\|A\|_2$ and $\|A\|_F$ stand for the spectrum norm and the Frobenius norm of A; $A \otimes B$ is the Kronecker product of A and B. F_n is an $n \times n$ the discrete Fourier transform (DFT) matrix, that is,

$$F_n = \left(f_{jk}\right), f_{jk} = \omega_n^{jk}, \omega_n = \mathrm{e}^{-\frac{2\pi\imath}{n}}, \imath = \sqrt{-1}.$$

We also use functions in MATLAB to denote $\hat{\mathcal{A}} \triangleq \mathtt{fft}(\mathcal{A}, [], 3)$ and $\mathcal{A} \triangleq \mathtt{ifft}(\hat{\mathcal{A}}, [], 3)$ for the DFT and the inverse DFT along the third dimension of the tensor $\mathcal{A}$,

$$\mathtt{diag}(\boldsymbol{v}) = \begin{pmatrix} v_1 & & & \\ & v_2 & & \\ & & \ddots & \\ & & & v_n \end{pmatrix}, \quad \mathtt{diag}(A) = \begin{pmatrix} A_1 & & & \\ & A_2 & & \\ & & \ddots & \\ & & & A_n \end{pmatrix}$$

denote the diagonal matrices of a vector $\boldsymbol{v} = (v_1, v_2, ..., v_n)$ and a matrix of $A = (A_1 \quad A_2 \quad \cdots \quad A_n)$.

If $\mathrm{rank}(A) = r$, then the matrix SVD [33] can be represented as

$$A = W\Sigma V^{*},$$

where $\Sigma = \mathtt{diag}(\sigma_1(A), \sigma_2(A), ..., \sigma_r(A), 0, ..., 0)$, $\sigma_i(A)$ is the ith singular value and satisfies $\sigma_1(A) \geqslant \sigma_2(A) \geqslant ... \geqslant \sigma_r(A) > 0$. W and V are called the left and right singular vectors of A, respectively. Then, the s-order truncated SVD of A becomes

$$A_s = W_s \Sigma_s V_s^*,$$

where $W_s = W(:, 1:s), V_s = V(:, 1:s), \Sigma_s = \Sigma(1:s, 1:s)$.

Moreover, A_s is the best rank-s approximation of A under the Frobenius norm [33]. We can threshold the singular values of a matrix A with a parameter $\tau > 0$, and will denote by

$$[A]_\tau = W[\Sigma]_\tau V^*,$$

where the diagonal matrix $[\Sigma]_\tau(i,i) = \sigma_i(A)$ if $\sigma_i(A) \geqslant \tau$, and is 0 otherwise.

Based on the SVD, we also obtain the Moore-Penrose inverse [31, 32]

$$A^\dagger = V\Sigma^\dagger W^*,$$

where $\Sigma^\dagger = \mathtt{diag}(\frac{1}{\sigma_1(A)}, \frac{1}{\sigma_2(A)}, \ldots, \frac{1}{\sigma_r(A)}, 0, \ldots, 0)$.

The Moore-Penrose inverse of tensors via the Einstein product [34] and its perturbation analysis can be found in [35, 36] respectively.

2.2. Matrix CUR decomposition

For the matrix CUR decomposition, the related theorem and its proof are provided in [4, 6], there is a recall without the proof.

Theorem 2.1. *Let the matrix* $A \in \mathbb{K}^{n_1 \times n_2}$ *have* $\mathrm{rank}(A) = r$, $I \subset \{1:n_1\}$ *and* $J \subset \{1:n_2\}$ *satisfying* $|I|, |J| \geqslant r$. *Let matrices* $C = A(:, J)$, $R = A(I, :)$, $U = A(I, J)$, *if* $\mathrm{rank}(U) = \mathrm{rank}(A)$, *then*

$$A = CU^\dagger R. \tag{2.1}$$

Theorem 2.1 reveals that matrices C, U and R are composed of several selected rows and columns of A. Under certain conditions, A equals to the product of these submatrices. As shown in Figure 1 [6], R and C are the row and column submatrices of A, respectively, and U is the intersection of C and R.

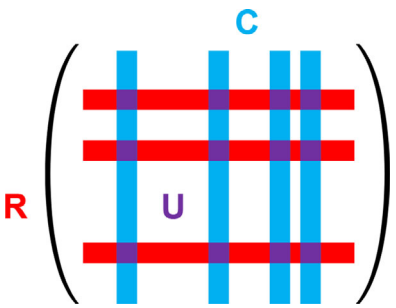


Figure 1. Illustration of the matrix CUR decomposition [4, 6].

Example 2.1. Let a 4×4 matrix $A=\begin{pmatrix}1&2&3&4\\5&6&7&8\\9&10&11&12\\13&14&15&16\end{pmatrix}$ have rank 2.

We first select the full column rank matrix C_1 and full row rank matrix R_1,

$$C_1=\begin{pmatrix}1&2\\5&6\\9&10\\13&14\end{pmatrix},\quad R_1=\begin{pmatrix}1&2&3&4\\5&6&7&8\end{pmatrix}.$$

It is easy to see that

$$U_1=\begin{pmatrix}1&2\\5&6\end{pmatrix}^{-1}=\begin{pmatrix}-1.5&0.5\\1.25&-0.25\end{pmatrix}$$

and the exact CUR decomposition $A=C_1U_1R_1$.

Next we choose the rank-deficient matrices C_2 and R_2 with the rank two,

$$C_2=\begin{pmatrix}1&2&3\\5&6&7\\9&10&11\\13&14&15\end{pmatrix},\quad R_2=\begin{pmatrix}1&2&3&4\\5&6&7&8\\9&10&11&12\end{pmatrix},$$

It is direct to verify that the rank-two-matrix

$$U_2=\begin{pmatrix}1&2&3\\5&6&7\\9&10&11\end{pmatrix}^{\dagger}=\begin{pmatrix}-0.5833&-0.1667&0.2500\\-0.0417&-0.0000&0.0417\\0.5000&0.1667&-0.1667\end{pmatrix}.$$

The exact CUR decomposition is $A=C_2U_2R_2$.

2.3. Tensor T-product

3rd tensor is a three-dimensional matrix, which can be regarded as a series of matrices of the same size. As shown in Figure 2, the left-hand side is a tensor $\mathcal{A}\in\mathbb{K}^{n_1\times n_2\times n_3}$. Following the concept from [27], the vector formed by fixing two coordinates of $\mathcal{A}$ is called the fiber. In Figure 2, (a) is row

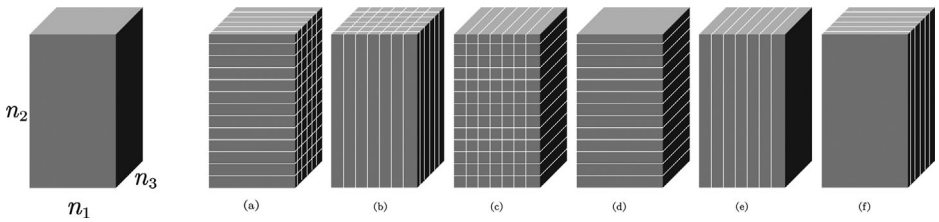


Figure 2. Illustration of a tensor.

fiber, (b) is column fiber, (c) is tube fiber, the matrix formed by fixing one coordinate is called slice, (d) is horizontal slice, (e) is lateral slice, and (f) is frontal slice.

The operations of tensors can also be defined.

Definition 2.1. ([27]) Let tensors $\mathcal{A}$ and $\mathcal{B} \in \mathbb{K}^{n_1\times n_2\times n_3}$. Then

$$(\mathcal{A}\pm\mathcal{B})(i,j,k) \triangleq \mathcal{A}(i,j,k)\pm\mathcal{B}(i,j,k), \quad i \in \{1:n_1\}, j \in \{1:n_2\}, k \in \{1:n_3\}. \tag{2.2}$$

Definition 2.2. ([27]) Let the tensor $\mathcal{A} \in \mathbb{K}^{n_1\times n_2\times n_3}$, $A_k = \mathcal{A}(:,:,k), k \in \{1:n_3\}$. Then

$$\text{bcirc}(\mathcal{A}) \triangleq \begin{pmatrix} A_1 & A_{n_3} & \cdots & A_2 \\ A_2 & A_1 & \cdots & A_3 \\ \vdots & \vdots & \ddots & \vdots \\ A_{n_3} & A_{n_3-1} & \cdots & A_1 \end{pmatrix} \tag{2.3}$$

is called the block circulant matrix [37] generated by $\mathcal{A}$.

Definition 2.3. ([27]) With the same notations of Definition 2.2, we define

$$\text{unfold}(\mathcal{A}) \triangleq \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_{n_3} \end{pmatrix}, \quad \text{fold}(\text{unfold}(\mathcal{A})) \triangleq \mathcal{A}. \tag{2.4}$$

Remark 2.1. If $\mathcal{I} \in \mathbb{K}^{n_2\times n_2\times n_3}$ satisfies $\text{unfold}(\mathcal{I}) = (I_{n_2\times n_2} \quad O \quad \cdots \quad O)^\top$, then $\text{unfold}(\mathcal{I})$ is called standard block unit vectors [25], and for convenience, is denoted by $E_1 \in \mathbb{K}^{n_2n_3\times n_2}$. Moreover, any E_1 mentioned in this paper refers to the standard block unit vectors.

Definition 2.4. ([27]) Let tensors $\mathcal{A} \in \mathbb{K}^{n_1\times n_2\times n_3}$ and $\mathcal{B} \in \mathbb{K}^{n_2\times n_4\times n_3}$, then the T-product of $\mathcal{A}$ and $\mathcal{B}$ is denoted as $\mathcal{A} * \mathcal{B}$, which is a tensor with size $n_1 \times n_4 \times n_3$ and

$$\mathcal{A} * \mathcal{B} \triangleq \text{fold}(\text{bcirc}(\mathcal{A})\text{unfold}(\mathcal{B})). \tag{2.5}$$

To simplify (2.5), some properties of the T-product need to be introduced without the proof, which have been proved completely in [25, 27].

Proposition 2.1. *Let tensors $\mathcal{A} \in \mathbb{K}^{n_1\times n_2\times n_3}$ and $\mathcal{B} \in \mathbb{K}^{n_2\times n_4\times n_3}$. Then*

(a) $\text{unfold}(\mathcal{A}) = \text{bcirc}(\mathcal{A})E_1$,
(b) $\text{bcirc}(\text{fold}(\text{bcirc}(\mathcal{A})E_1)) = \text{bcirc}(\mathcal{A})$,
(c) $\text{bcirc}(\mathcal{A} * \mathcal{B}) = \text{bcirc}(\mathcal{A})\text{bcirc}(\mathcal{B})$.

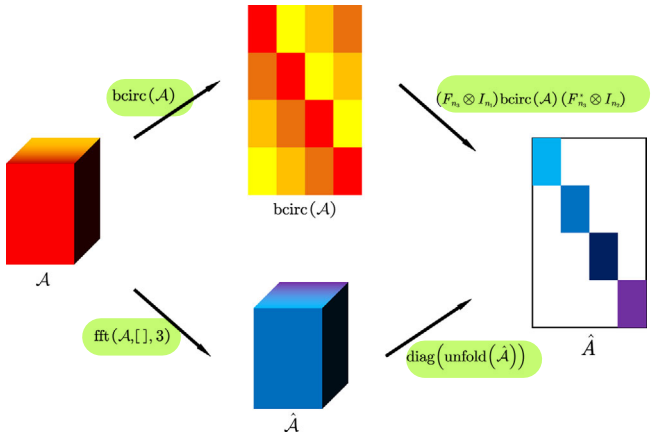


Figure 3. The relationship between the tensor operation and the tensor DFT.

The T-product can be computed by means of the DFT. Firstly, a block circulant matrix [37] can be block-diagonalized by the DFT, i.e., $(F_{n_3} \otimes I_{n_1})\texttt{bcirc}(\mathcal{A})(F_{n_3}^* \otimes I_{n_2}) \triangleq \hat{A}$ is a block diagonal matrix, and the tensor $\texttt{fold}((\hat{A}_1 \quad \hat{A}_2 \quad \cdots \quad \hat{A}_{n_3})^\top) = \hat{\mathcal{A}}$, where $k \in \{1 : n_3\}$, $\hat{A}_k$ is a diagonal block of $\hat{A}$. $\hat{\mathcal{A}}$ is also the result of applying the DFT along the third dimension of $\mathcal{A}$ [27], which is shown in Figure 3.

We can conduct another mathematical expression of the T-product.

$$\begin{aligned}
\mathcal{A} * \mathcal{B} &= \texttt{fold}(\texttt{bcirc}(\mathcal{A})\texttt{bcirc}(\mathcal{B})E_1) \\
&= \texttt{fold}\big((F_{n_3}^* \otimes I_{n_1})(F_{n_3} \otimes I_{n_1})\texttt{bcirc}(\mathcal{A})(F_{n_3}^* \otimes I_{n_2}) \\
&\qquad (F_{n_3} \otimes I_{n_2})\texttt{bcirc}(\mathcal{B})E_1\big) \\
&= \texttt{fold}\Big((F_{n_3}^* \otimes I_{n_1})\hat{A}(F_{n_3} \otimes I_{n_2})\texttt{bcirc}(\mathcal{B})(F_{n_3}^* \otimes I_{n_1})(F_{n_3} \otimes I_{n_1})E_1\Big) \\
&= \texttt{fold}\Big((F_{n_3}^* \otimes I_{n_1})\hat{A}\hat{B}(F_{n_3} \otimes I_{n_1})E_1\Big).
\end{aligned}$$

It indicates that the T-product can be computed under the DFT, as Algorithm T-MULT in [38] and let $p = 3$.

As a consequence, some useful tools used to establish our results under the T-product framework [27].

Definition 2.5. ([27]) Let the tensor $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$, then the conjugate transpose of $\mathcal{A}$ is $\mathcal{A}^* \in \mathbb{K}^{n_2 \times n_1 \times n_3}$, $\mathcal{A}^*(:, :, k) = \mathcal{A}(:, :, n_3 - k + 2)^*$ and $A^*(:, :, 1) = A(:, :, 1)^*$.

Definition 2.6. ([27]) The tensor $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ is called orthogonal, if $\mathcal{A}^* * \mathcal{A} = \mathcal{A} * \mathcal{A}^* = \mathcal{I}$, where $\mathcal{I} \in \mathbb{K}^{n \times n \times n_3}$, except for $\mathcal{I}(:, :, 1) = I_n$, the other elements in $\mathcal{I}$ are all 0.

Definition 2.7. ([27]) The tensor $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ is called the f-diagonal, if $\mathcal{A}(:, :, k)$ is a diagonal matrix, $k \in \{1 : n_3\}$.

The t-SVD can been developed with Definition 2.4 and Proposition 2.1, which is Theorem 2.2.

Theorem 2.2. ([27]) *Let the tensor $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$, then $\mathcal{A}$ can be decomposed by*

$$\mathcal{A} = \mathcal{W} * \ \Sigma * \mathcal{V}^*, \tag{2.6}$$

where $\mathcal{W}$ and $\mathcal{V}$ are orthogonal and Σ is f-diagonal, the s-order truncated t-SVD of $\mathcal{A}$ is

$$\mathcal{A}_s = \mathcal{W}_s * \ \Sigma_s * \mathcal{V}_s^*,$$

where $\Sigma_s = \Sigma(1 : s, 1 : s, :)$, $\mathcal{W}_s = \mathcal{W}(:, 1 : s, :)$ and $\mathcal{V}_s = \mathcal{V}(:, 1 : s, :)$.

Remark 2.2. We can also threshold singular values of a tensor $\mathcal{A}$ with parameter τ, which is denoted by

$$[\mathcal{A}]_\tau = \mathcal{W} * [\Sigma]_\tau * \mathcal{V}^*,$$

where $[\Sigma]_\tau = \texttt{ifft}([\hat{\Sigma}]_\tau, [], 3)$, $[\hat{\Sigma}]_\tau(i, i, k) = \sigma_i(\hat{A}_k)$ if $\sigma_i(\hat{A}_k) \geqslant \tau$, and $[\hat{\Sigma}]_\tau(i, i, k) = 0$ otherwise.

Before we define the tensor rank, there are other definitions of tensor rank in [39, 40]. Now we will recall a definition under the T-product framework. The matrix rank can be defined by the SVD, the tensor rank can be defined by the t-SVD similarly. Since the t-SVD can be computed under the DFT, we define the tensor rank with $\texttt{fft}(\mathcal{A}, [], 3)$.

Definition 2.8. ([39, 40]) Let the tensor $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ satisfy $k \in \{1 : n_3\}$, $\text{rank}(\hat{A}_k) = r_k$, then vector $\boldsymbol{r} = (r_1, r_2, \ldots, r_{n_3})$ is called the multirank of $\mathcal{A}$ and denoted as $\text{rank}_m(\mathcal{A})$. The tubalrank of $\mathcal{A}$, denoted as $\text{rank}_t(\mathcal{A})$, is defined as the number of nonzero singular values tubes of Σ. That is

$$\text{rank}_t(\mathcal{A}) = \left|\left\{i | \Sigma(i, i, :) \neq 0\right\}\right|,$$

where Σ satisfies t-SVD and $\mathcal{A} = \mathcal{W} * \ \Sigma * \mathcal{V}^*$.

Remark 2.3. It is easy to verify that $\text{rank}_t(\mathcal{A}) = \max\{\boldsymbol{r}\}$, where $\text{rank}_m(\mathcal{A}) = \boldsymbol{r}$. We will mainly adopt this definition in this paper.

These two tensor ranks are very important in this paper, where the multirank guarantees our t-CUR decomposition in the main theorem; the tubalrank is usually used in practical sampling of the t-CUR decomposition. We analyze the role of tubalrank in the perturbation analysis. These results will be shown in next few sections.

Definition 2.9. ([28, 41]) The tensor $\mathcal{A}^{\dagger} \in \mathbb{K}^{n_2\times n_1\times n_3}$ is called the Moore-Penrose inverse of $\mathcal{A} \in \mathbb{K}^{n_1\times n_2\times n_3}$, if $\mathcal{A}^{\dagger}$ satisfies the following four equations,

$$\mathcal{A} * \mathcal{A}^{\dagger} * \mathcal{A} = \mathcal{A}, \quad \mathcal{A}^{\dagger} * \mathcal{A} * \mathcal{A}^{\dagger} = \mathcal{A}^{\dagger}, \quad (\mathcal{A} * \mathcal{A}^{\dagger})^{*} = \mathcal{A} * \mathcal{A}^{\dagger},$$
$$(\mathcal{A}^{\dagger} * \mathcal{A})^{*} = \mathcal{A}^{\dagger} * \mathcal{A}.$$

Theorem 2.3. *Given a tensor $\mathcal{A} \in \mathbb{K}^{n_1\times n_2\times n_3}$, then*

$$\mathcal{A}^{\dagger} = \text{fold}\Big((F_{n_3}^{*} \otimes I_{n_2})\hat{A}^{\dagger}(F_{n_3} \otimes I_{n_1})E_1\Big), \tag{2.7}$$

where $\hat{A}^{\dagger} = \text{diag}(\hat{A}_1^{\dagger} \quad \hat{A}_2^{\dagger} \quad \cdots \quad \hat{A}_{n_3}^{\dagger})$.

It indicates that the Moore-Penrose inverse of $\mathcal{A} \in \mathbb{K}^{n_1\times n_2\times n_3}$ [31, 32] is the inverse DFT of $\hat{A}^{\dagger}$.

Proof. It is easy to prove that $\mathcal{A}^{\dagger}$ in (2.7) satisfies four equations in Definition 2.9. We only prove the first one and the rest parts are similar.

By Proposition 2.1, we have

$$\begin{cases} \texttt{bcirc}(\mathcal{A}^{\dagger}) = \texttt{bcirc}\Big(\texttt{fold}\Big((F_{n_3}^{*} \otimes I_{n_2})\hat{A}^{\dagger}(F_{n_3} \otimes I_{n_1})E_1\Big)\Big) \\ \qquad\qquad = (F_{n_3}^{*} \otimes I_{n_2})\hat{A}^{\dagger}(F_{n_3} \otimes I_{n_1}), \\ \mathcal{A} * \mathcal{A}^{\dagger} * \mathcal{A} = \texttt{fold}\big(\texttt{bcirc}(\mathcal{A})\texttt{bcirc}(\mathcal{A}^{\dagger})\texttt{bcirc}(\mathcal{A})E_1\big) \\ \qquad\qquad = \texttt{fold}\Big((F_{n_3}^{*} \otimes I)\hat{A}\hat{A}^{\dagger}\hat{A}(F_{n_3} \otimes I)E_1\Big) \\ \qquad\qquad = \texttt{fold}(\texttt{bcirc}(\mathcal{A})E_1) = \mathcal{A}. \end{cases}$$

□

Moreover, the perturbation analysis in Section 4 needs some norms of the tensor [27, 29].

Definition 2.10. ([27]) $\|\mathcal{A}\|_F$ is called the Frobenius norm of tensor $\mathcal{A} \in \mathbb{K}^{n_2\times n_1\times n_3}$, where

$$\|\mathcal{A}\|_F \triangleq \sqrt{\sum_k \|\mathcal{A}(:,:,k)\|_F^2} = \sqrt{\sum_{i,j,k} |\mathcal{A}(i,j,k)|^2}. \tag{2.8}$$

Remark 2.4. The s-order truncated t-SVD $\mathcal{A}_s$ is the best tubalrank– s approximation of $\mathcal{A}$ under the Frobenius norm [27].

Definition 2.11. ([29]) $\|\mathcal{A}\|_2$ is called the spectrum norm of tensor $\mathcal{A} \in \mathbb{K}^{n_2\times n_1\times n_3}$,

$$\|\mathcal{A}\|_2 \triangleq \|\hat{A}\|_2, \tag{2.9}$$

where $\|\hat{A}\|_2$ is the spectrum norm of $\hat{A}$.

Remark 2.5. According to matrix spectrum norm $\|\hat{A}\|_2 = \sigma_1(\hat{A})$, we have $\|\mathcal{A}\|_2 = \|\hat{A}\|_2 = \max_k \|\hat{A}_k\|_2$.

3. t-CUR decomposition

In the T-product framework, the matrix CUR decomposition [4, 6] can be extended to the tensor t-CUR decomposition.

Theorem 3.1. *Let the tensor* $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ *have* $\text{rank}_m(\mathcal{A}) = \boldsymbol{r}$ *and* $\text{rank}_t(\mathcal{A}) = r$. *Index sets* $I \subset \{1 : n_1\}$ *and* $J \subset \{1 : n_2\}$ *satisfy* $|I|, |J| \geqslant r$. $\mathcal{C} = \mathcal{A}(:, J, :)$, $\mathcal{R} = \mathcal{A}(I, :, :)$ *and* $\mathcal{U} = \mathcal{A}(I, J, :)$. *If* $\text{rank}_m(\mathcal{U}) = \text{rank}_m(\mathcal{A}) = \boldsymbol{r}$, *then*

$$\mathcal{A} = \mathcal{C} * \mathcal{U}^{\dagger} * \mathcal{R}. \tag{3.1}$$

Proof. Using the DFT, we get the block diagonal matrix $\hat{A}$. Similarly, the same operation can be applied to $\mathcal{C}, \mathcal{U}, \mathcal{R}$ to obtain $\hat{C}, \hat{U}$, and $\hat{R}$, respectively.

Since $\text{rank}_m(\mathcal{U}) = \text{rank}_m(\mathcal{A}) = \boldsymbol{r}$, we have $\text{rank}(\hat{A}_k) = \text{rank}(\hat{U}_k)$. For $k \in \{1 : n_3\}$, according to Theorem 2.1, $\hat{A}_k = \hat{C}_k \hat{U}_k^{\dagger} \hat{R}_k$, then $\hat{A}$ can be decomposed as

$$\begin{pmatrix} \hat{A}_1 & & & \\ & \hat{A}_2 & & \\ & & \ddots & \\ & & & \hat{A}_{n_3} \end{pmatrix} = \begin{pmatrix} \hat{C}_1 & & & \\ & \hat{C}_2 & & \\ & & \ddots & \\ & & & \hat{C}_{n_3} \end{pmatrix} \begin{pmatrix} \hat{U}_1^{\dagger} & & & \\ & \hat{U}_2^{\dagger} & & \\ & & \ddots & \\ & & & \hat{U}_{n_3}^{\dagger} \end{pmatrix} \begin{pmatrix} \hat{R}_1 & & & \\ & \hat{R}_2 & & \\ & & \ddots & \\ & & & \hat{R}_3 \end{pmatrix}. \tag{3.2}$$

Multiplying $F_{n_3}^* \otimes I$ to the left-hand side and $F_{n_3} \otimes I$ to the right-hand side of each block diagonal matrix on both sides of (3.2), the result reduces to

$$\texttt{bcirc}(\mathcal{A}) = \texttt{bcirc}(\mathcal{C})\texttt{bcirc}(\mathcal{U}^{\dagger})\texttt{bcirc}(\mathcal{R}).$$

Using Proposition 2.1, we have

$$\begin{cases} \texttt{bcirc}(\mathcal{A})E_1 = \texttt{bcirc}(\mathcal{C})\texttt{bcirc}(\mathcal{U}^\dagger)\texttt{bcirc}(\mathcal{R})E_1 \\ \texttt{unfold}(\mathcal{A}) = \texttt{bcirc}(\mathcal{C})\texttt{bcirc}(\mathcal{U}^\dagger)\texttt{unfold}(\mathcal{R}) \\ \mathcal{A} = \texttt{fold}\big(\texttt{bcirc}(\mathcal{C})\texttt{unfold}\big(\texttt{fold}\big(\texttt{bcirc}(\mathcal{U}^\dagger)\texttt{unfold}(\mathcal{R})\big)\big)\big) \\ \quad = \texttt{fold}\big(\texttt{bcirc}(\mathcal{C})\texttt{unfold}\big(\mathcal{U}^\dagger * \mathcal{R}\big)\big) \\ \quad = \mathcal{C} * \mathcal{U}^\dagger * \mathcal{R}. \end{cases}$$

□

Remark 3.1. The assumption $\text{rank}_m(\mathcal{U}) = \text{rank}_m(\mathcal{A}) = \boldsymbol{r}$ in Theorem 3.1 indicates that the tensor rank of $\mathcal{A}$ should be low, or subtensor $\mathcal{U} = \mathcal{A}$. Besides, the assumption of the tensor rank is multirank, rather than tubalrank, because the tubalrank can not satisfy $\text{rank}(\hat{A}_k) = \text{rank}(\hat{U}_k)$. If the assumption is $\text{rank}_t(\mathcal{U}) = \text{rank}_t(\mathcal{A}) = r$, according to the definition of tubalrank, then we have some k_0, such that $\text{rank}(\hat{A}_{k_0}) > \text{rank}(\hat{U}_{k_0})$.

Here is a counterexample. Let $\mathcal{A} \in \mathbb{R}^{4\times4\times4}$,

$$\mathcal{A}(:,:,1) = \mathcal{A}(:,:,3) = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 0 & \\ & & & 1 \end{pmatrix},$$

$$\mathcal{A}(:,:,2) = \mathcal{A}(:,:,4) = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 0 & \\ & & & -1 \end{pmatrix},$$

then we have

$$\hat{\mathcal{A}}(:,:,1) = \begin{pmatrix} 4 & & & \\ & 4 & & \\ & & 0 & \\ & & & 0 \end{pmatrix}, \hat{\mathcal{A}}(:,:,3) = \begin{pmatrix} 0 & & & \\ & 0 & & \\ & & 0 & \\ & & & 4 \end{pmatrix},$$

$$\hat{\mathcal{A}}(:,:,2) = \hat{\mathcal{A}}(:,:,4) = O_{4\times4}.$$

Let $\mathcal{U} = \mathcal{A}(1:2, 1:2, :)$, then

$$\hat{\mathcal{U}}(:,:,1) = \begin{pmatrix} 4 & \\ & 4 \end{pmatrix}, \hat{\mathcal{U}}(:,:,2) = \hat{\mathcal{U}}(:,:,3) = \hat{\mathcal{U}}(:,:,4) = O_{2\times2}.$$

It is obvious that $\text{rank}_t(\mathcal{U}) = \text{rank}_t(\mathcal{A}) = 2$, but $\text{rank}(\hat{A}_3) > \text{rank}(\hat{U}_3)$, which means $\hat{A}_3 \neq \hat{C}_3\hat{U}_3^\dagger\hat{R}_3$ and then $\mathcal{A} \neq \mathcal{C} * \mathcal{U}^\dagger * \mathcal{R}$.

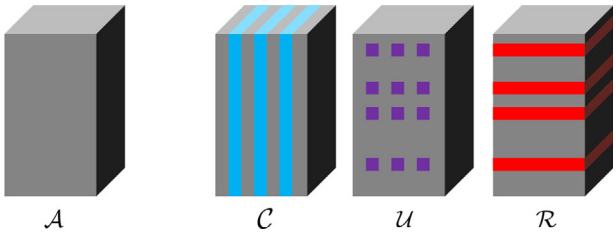


Figure 4. Illustration of the t-CUR decomposition.

The t-CUR decomposition is given in Figure 4. Compared with the matrix CUR decomposition, the t-CUR decomposition can be seen as the matrix CUR decomposition for each frontal slice with the same rows and columns chosen.

However, the assumption $\text{rank}_m(\mathcal{U}) = \text{rank}_m(\mathcal{A})$ in Theorem 3.1 is too strict. It is difficult to sample a subtensor $\mathcal{U}$ to satisfy $\hat{A}_k = \hat{C}_k \hat{U}_k^{\dagger} \hat{R}_k$, $k \in \{1 : n_3\}$ in practical use. But the CUR sampling theorem in matrix form [4, Theorem 6.1] indicates that slightly oversampling rows and columns randomly is successful to obtain $\text{rank}(U_k) = \text{rank}(A_k)$ with a high probability. Therefore, oversampling rows and columns of each frontal slice yields $\hat{A}_k = \hat{C}_k \hat{U}_k^{\dagger} \hat{R}_k$ with a high probability, which inspires us to oversample the horizontal slices and lateral slices of $\mathcal{A}$.

In fact, applying [4, Theorem 6.1] to each frontal slice, the number of horizontal and lateral slices oversampled is related to each $\text{rank}(A_k)$. We have the fact that $\text{rank}(A_k) \leqslant \text{rank}_t(\mathcal{A})$, then it can be only related with $\text{rank}_t(\mathcal{A})$, which relaxes the assumption in the t-CUR decomposition to $\text{rank}_t(\mathcal{U}) = \text{rank}_t(\mathcal{A})$.

We can conduct an error estimation of tubalrank forced t-CUR decomposition in Section 4.2. In Section 5, we will only force the tubalrank instead of multirank of a full-rank random tensor.

4. Perturbation of the t-CUR decomposition

In the practical use, tensor data are often perturbed, such as the noise in pictures and videos, which may cause huge errors. In this section, we analyze the perturbation of the t-CUR decomposition.

4.1. Error estimation of the t-CUR decomposition

Before analyzing the perturbations, we will adopt some notations firstly. Let the tensor $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ with $\text{rank}_t(\mathcal{A}) = r$ have t-CUR decomposition $\mathcal{A} = \mathcal{C} * \mathcal{U}^{\dagger} * \mathcal{R}$, where $\mathcal{C} = \mathcal{A}(:, J, :)$, $\mathcal{R} = \mathcal{A}(I, :, :)$ and $\mathcal{U} = \mathcal{A}(I, J, :)$ for index sets I, J. Let $\mathcal{E} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ be a noise tensor and $\mathcal{A}$ be perturbed to $\mathcal{A}' = \mathcal{A} + \mathcal{E} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$, then we write

$$\mathcal{C}' = \mathcal{A}'(:,J,:), \mathcal{R}' = \mathcal{A}'(I,:,:), \mathcal{U}' = \mathcal{A}'(I,J,:) \tag{4.1}$$

and $\mathcal{E}_I = \mathcal{E}(I,:,:)$, $\mathcal{E}_J = \mathcal{E}(:,J,:)$, $\mathcal{E}_{IJ} = \mathcal{E}(I,J,:)$.

Let the tensor $\mathcal{A}$ have r-order truncated t-SVD of $\mathcal{A}_r = \mathcal{W}_r * \Sigma_r * \mathcal{V}_r^*$ as in Theorem 2.2, then we write

$$\mathcal{W}_{r,I}^\dagger = \mathcal{W}_r(I,:,:)^\dagger, \mathcal{V}_{r,J}^\dagger = \mathcal{V}_r(:,J,:)^\dagger. \tag{4.2}$$

for index sets I, J.

We will first give an estimation by thresholding the singular values of $\mathcal{U}'$ with a parameter $\tau > 0$:

Theorem 4.1. *Using notations in (4.1) and (4.2), and let $\tau \geqslant 0$, then*

$$\begin{aligned}
&\|\mathcal{A} - \mathcal{C}' * [\mathcal{U}']_\tau^\dagger * \mathcal{R}'\|_2 \\
&\leqslant \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\big(2\|\mathcal{E}_{IJ}\|_2 + \|[\mathcal{U}']_\tau - \mathcal{U}\|_2\big) \\
&\quad + \|[\mathcal{U}']_\tau^\dagger\|_2\Big[\Big(\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 \\
&\quad + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2\Big)\|\mathcal{E}_{IJ}\|_2 + \|\mathcal{E}_I\|_2\|\mathcal{E}_J\|_2\Big].
\end{aligned}$$

Proof. With the help of the perturbation analysis of the matrix CUR decomposition [7, Theorem 4.2] and Definition 2.11, we have

$$\begin{aligned}
\|\mathcal{A} - \mathcal{C}' * [\mathcal{U}']_\tau^\dagger * \mathcal{R}'\|_2 &= \|\hat{A} - \hat{C}'\big[\hat{U}'\big]_\tau^\dagger \hat{R}'\|_2 \\
&= \max_k \|\hat{A}_k - \hat{C}_k'\big[\hat{U}'\big]_{\tau,k}^\dagger \hat{R}_k'\|_2 \\
&\leqslant \max_k \Big\{\|\hat{W}_{r,I,k}^\dagger\|_2\|\hat{E}_{I,k}\|_2 + \|\hat{V}_{r,J,k}^\dagger\|_2\|\hat{E}_{J,k}\|_2 \\
&\quad + \|\hat{W}_{r,I,k}^\dagger\|_2\|\hat{V}_{r,J,k}^\dagger\|_2\Big(2\|\hat{E}_{IJ,k}\|_2 + \Big\|\Big[\hat{U}_k'\Big]_\tau - \hat{U}_k\Big\|_2\Big) \\
&\quad + \Big\|\Big[\hat{U}_k'\Big]_\tau^\dagger\Big\|_2\Big[\Big(\|\hat{W}_{r,I,k}^\dagger\|_2\|\hat{E}_{I,k}\|_2 + \|\hat{V}_{r,J,k}^\dagger\|_2\|\hat{E}_{J,k}\|_2 \\
&\quad + \|\hat{W}_{r,I,k}^\dagger\|_2\|\hat{V}_{r,J,k}^\dagger\|_2\|\hat{E}_{IJ,k}\|_2\Big)\|\hat{E}_{IJ,k}\|_2 + \|\hat{E}_{I,k}\|_2\|\hat{E}_{J,k}\|_2\Big]\Big\}.
\end{aligned}$$

By Definition 2.11, we maximize each term of the above equation, then

$$\begin{aligned}
&\|\mathcal{A} - \mathcal{C}' * [\mathcal{U}']_\tau^\dagger * \mathcal{R}'\|_2 \\
&\leqslant \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\big(2\|\mathcal{E}_{IJ}\|_2 + \|[\mathcal{U}']_\tau - \mathcal{U}\|_2\big) \\
&\quad + \|[\mathcal{U}']_\tau^\dagger\|_2\Big[\Big(\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 \\
&\quad + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2\Big)\|\mathcal{E}_{IJ}\|_2 + \|\mathcal{E}_I\|_2\|\mathcal{E}_J\|_2\Big].
\end{aligned}$$

□

According to Remark 2.2, if $\text{rank}_t(\mathcal{U}') = r$ and $0 \leqslant \tau \leqslant \sigma_r(\hat{U}'_k)$, $k \in \{1 : n_3\}$, then $\mathcal{C}' * \mathcal{U}'^{\dagger} * \mathcal{R}' = \mathcal{C}' * [\mathcal{U}']_\tau^{\dagger} * \mathcal{R}'$. Especially when $\tau = 0$, we can obtain the following corollary.

Corollary 4.1. *If $\tau = 0$, then we have*

$$\|\mathcal{A} - \mathcal{C}' * \mathcal{U}'^{\dagger} * \mathcal{R}'\|_2 \leqslant \left(\|\mathcal{W}_{r,I}^{\dagger}\|_2 + \|\mathcal{V}_{r,J}^{\dagger}\|_2 + 3\|\mathcal{W}_{r,I}^{\dagger}\|_2\|\mathcal{V}_{r,J}^{\dagger}\|_2\right)\|\mathcal{E}\|_2 + \|\mathcal{U}'^{\dagger}\|_2\left(\|\mathcal{W}_{r,I}^{\dagger}\|_2 + \|\mathcal{V}_{r,J}^{\dagger}\|_2 + \|\mathcal{W}_{r,I}^{\dagger}\|_2\|\mathcal{V}_{r,J}^{\dagger}\|_2 + 1\right)\|\mathcal{E}\|_2^2.$$

Proof. It is obvious that if $\tau = 0$, $[\mathcal{U}']_\tau = \mathcal{U}'$, and $\|\mathcal{E}_I\|_2, \|\mathcal{E}_J\|_2, \|\mathcal{E}_{IJ}\|_2$ are all less than or equal to $\|\mathcal{E}\|_2$ [42]. Using the fact that $\|[\mathcal{U}']_\tau - \mathcal{U}\|_2 = \|\mathcal{U} - \mathcal{U}'\|_2 = \|\hat{U} - \hat{U}'\|_2 \leqslant \|\hat{E}\|_2 = \|\mathcal{E}\|_2$, then we collect terms. □

4.2. Error estimation of the tubalrank forced t-CUR decomposition

Let the original tensor have $\text{rank}_t(\mathcal{A}) = r \ll \min\{n_1, n_2\}$. However, the noise tensor $\mathcal{E}$ usually has large or even full tubalrank, which result in the tubalrank of a perturbed tensor, $\text{rank}_t(\mathcal{A}')$, being large. If we oversample, that is, sample more than r slices in choosing $\mathcal{U}'$, we will obtain $\text{rank}_t(\mathcal{U}') > \text{rank}_t(\mathcal{A})$, then causing larger error. Therefore, forcing the tubalrank of $\mathcal{U}'$ to be r after sampling is a good way to reduce error. We will present the refined estimates under tubalrank forced t-CUR decomposition.

Proposition 4.1. *Let $\mathcal{U}'_r$ be the best tubalrank-r approximation of $\mathcal{U}'$, $\mathcal{W}_{r,I}$ and $\mathcal{V}_{r,J}$ defined as in (4.2), then*

$$\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^{\dagger} * \mathcal{R}'\|_2 \leqslant \left(\|\mathcal{W}_{r,I}^{\dagger}\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^{\dagger}\|_2\|\mathcal{E}_J\|_2 + 4\|\mathcal{W}_{r,I}^{\dagger}\|_2\|\mathcal{V}_{r,J}^{\dagger}\|_2\|\mathcal{E}_{IJ}\|_2\right) + \|\mathcal{U}_r'^{\dagger}\|_2 \left[\left(\|\mathcal{W}_{r,I}^{\dagger}\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^{\dagger}\|_2\|\mathcal{E}_J\|_2 + \|\mathcal{W}_{r,I}^{\dagger}\|_2\|\mathcal{V}_{r,J}^{\dagger}\|_2\|\mathcal{E}_{IJ}\|_2\right)\|\mathcal{E}_{IJ}\|_2 + \|\mathcal{E}_I\|_2\|\mathcal{E}_J\|_2\right].$$

Proof. Using the rank-forced perturbation analysis of the matrix CUR decomposition [7, Proposition 8.4] and Definition 2.11, we can easily prove the result. □

We want to reveal the relationship between the error bound and the noise tensor $\mathcal{E}$, but the right-hand side of the inequality in Proposition 4.1 has terms including $\|\mathcal{U}_r'^{\dagger}\|_2$, which is different from the form we need. We apply the following lemma to eliminate $\|\mathcal{U}_r'^{\dagger}\|_2$.

Lemma 4.1. *With notations and assumptions of Proposition 4.1, if* $\|\mathcal{E}_{IJ}\|_2\|\mathcal{U}^\dagger\|_2 < \frac{1}{2\mu}$, *then*

$$\|\mathcal{U}_r'^\dagger\|_2 \leqslant \frac{\|\mathcal{U}^\dagger\|_2}{1-2\mu\|\mathcal{U}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2},$$

where $\mu = \frac{1+\sqrt{5}}{2}$.

Proof. Note that $\|\mathcal{U}^\dagger\|_2 = \|\hat{U}^\dagger\|_2 = \frac{1}{\sigma_{\min}(\hat{U})}$, where $\sigma_{\min}(\hat{U})$ is the smallest nonzero singular value of $\hat{U}$ [31]. It follows from [7, Lemma 8.5] and Definition 2.11 that

$$\begin{aligned}
\|\mathcal{U}_r'^\dagger\|_2 &= \|\hat{U}_r'^\dagger\|_2 = \max_k \|\hat{U}_{r,k}'^\dagger\|_2 \\
&\leqslant \max_k \frac{\|\hat{U}_k^\dagger\|_2}{1-2\mu\|\hat{U}_k^\dagger\|_2\|\hat{E}_{IJ,k}\|_2} \\
&\leqslant \frac{\max_k \|\hat{U}_k^\dagger\|_2}{1-2\mu\max_k\|\hat{U}_k^\dagger\|_2\max_k\|\hat{E}_{IJ}\|_2} \\
&= \frac{\|\hat{U}^\dagger\|_2}{1-2\mu\|\hat{U}^\dagger\|_2\|\hat{E}_{IJ}\|_2} \\
&= \frac{\|\mathcal{U}^\dagger\|_2}{1-2\mu\|\mathcal{U}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2}.
\end{aligned}$$

□

Theorem 4.2. *With notations and assumptions of Lemma 4.1, we have*

$$\begin{aligned}
&\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^\dagger*\mathcal{R}'\|_2 \\
&\leqslant \left(\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 + 4\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2\right) + \frac{\|\mathcal{U}^\dagger\|_2}{1-2\mu\|\mathcal{U}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2} \\
&\left[\left(\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2\right)\|\mathcal{E}_{IJ}\|_2 + \|\mathcal{E}_I\|_2\|\mathcal{E}_J\|_2\right].
\end{aligned}$$

Proof. Combining Proposition 4.1 and Lemma 4.1 and we get the result. □

Corollary 4.2. *With notations of Theorem 4.2, if* $\|\mathcal{E}\|_2\|\mathcal{U}^\dagger\|_2 < \frac{1}{4\mu}$, *then*

$$\begin{aligned}
&\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^\dagger*\mathcal{R}'\|_2 \\
&\leqslant \left[\left(1+\frac{1}{2\mu}\right)\left(\|\mathcal{W}_{r,I}^\dagger\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\right) + \left(4+\frac{1}{2\mu}\right)\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2 + \frac{1}{2\mu}\right]\|\mathcal{E}\|_2.
\end{aligned}$$

Proof. If $\|\mathcal{E}\|_2\|\mathcal{U}^\dagger\|_2 < \frac{1}{4\mu}$, we have $\frac{\|\mathcal{U}^\dagger\|_2\|\mathcal{E}\|_2}{1-2\mu\|\mathcal{U}^\dagger\|_2\|\mathcal{E}\|_2} < \frac{1}{2\mu}$, then

$$\begin{aligned}
&\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^\dagger * \mathcal{R}'\|_2 \\
&\leqslant \left(\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 + 4\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2\right) + \frac{\|\mathcal{U}^\dagger\|_2}{1-2\mu\|\mathcal{U}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2} \\
&\quad \left[\left(\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{E}_I\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_J\|_2 + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\|\mathcal{E}_{IJ}\|_2\right)\|\mathcal{E}_{IJ}\|_2 + \|\mathcal{E}_I\|_2\|\mathcal{E}_J\|_2\right] \\
&\leqslant \left(\|\mathcal{W}_{r,I}^\dagger\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2 + 4\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2\right)\|\mathcal{E}\|_2 \\
&\quad + \frac{\|\mathcal{U}^\dagger\|_2\|\mathcal{E}\|_2}{1-2\mu\|\mathcal{U}^\dagger\|_2\|\mathcal{E}\|_2}\left(\|\mathcal{W}_{r,I}^\dagger\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2 + \|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2 + 1\right)\|\mathcal{E}\|_2 \\
&\leqslant \left[\left(1+\frac{1}{2\mu}\right)\left(\|\mathcal{W}_{r,I}^\dagger\|_2 + \|\mathcal{V}_{r,J}^\dagger\|_2\right) + \left(4+\frac{1}{2\mu}\right)\|\mathcal{W}_{r,I}^\dagger\|_2\|\mathcal{V}_{r,J}^\dagger\|_2 + \frac{1}{2\mu}\right]\|\mathcal{E}\|_2.
\end{aligned}$$

□

The estimation in Corollary 4.2 is pretty accurate, where the error bound is the first order of $\|\mathcal{E}\|_2$. However, the tensors $\mathcal{W}$ and $\mathcal{V}$ need some computations by the t-SVD, they rely on the choice of the slice. We will introduce some propositions to eliminate those problems.

Proposition 4.2. *Let $\mathcal{W}_{r,I}$ and $\mathcal{V}_{r,J}$ be subtensors of $\mathcal{W}_r$ and $\mathcal{V}_r$, $\hat{W}_{r,I,k_1}^\dagger$ and $\hat{V}_{r,J,k_2}^\dagger$ satisfy $\|\mathcal{W}_{r,I}^\dagger\|_2 = \|\hat{W}_{r,I,k_1}^\dagger\|_2$ and $\|\mathcal{V}_{r,J}^\dagger\|_2 = \|\hat{V}_{r,J,k_2}^\dagger\|_2$. If $\hat{W}_{r,I,k_1}$ and $\hat{V}_{r,J,k_2}$ are the maximal volume submatrices of $\hat{W}_{r,k_1}$ and $\hat{V}_{r,k_2}$, then*

(i) $\|\mathcal{W}_{r,I}^\dagger\|_2 \leqslant \sqrt{1+\frac{r(n_1-|I|)}{|I|-r+1}} \triangleq g(r,n_1,|I|)$,

(ii) $\|\mathcal{V}_{r,J}^\dagger\|_2 \leqslant \sqrt{1+\frac{r(n_2-|J|)}{|J|-r+1}} \triangleq g(r,n_2,|J|)$,

(iii) $\|\mathcal{U}^\dagger\|_2 \leqslant g(r,n_1,|I|)g(r,n_2,|J|)\|\mathcal{A}^\dagger\|_2$.

Proof. The result of matrix case have been developed in [30, Lemma 1], i.e.,

$$\begin{aligned}
&\|W_{r,I}^\dagger\|_2 \leqslant \sqrt{1+\frac{r(n_1-|I|)}{|I|-r+1}} = g(r,n_1,|I|), \\
&\|V_{r,J}^\dagger\|_2 \leqslant \sqrt{1+\frac{r(n_2-|J|)}{|J|-r+1}} = g(r,n_2,|J|), \\
&\|U^\dagger\|_2 \leqslant \sqrt{1+\frac{r(n_1-|I|)}{|I|-r+1}}\sqrt{1+\frac{r(n_2-|J|)}{|J|-r+1}}\|A^\dagger\|_2 = g(r,n_1,|I|)g(r,n_2,|J|)\|A^\dagger\|_2,
\end{aligned} \tag{4.3}$$

where $W_r\Sigma_r V_r^*$ is the truncated SVD of $A \in \mathbb{K}^{n_1\times n_2}$, $W_{r,I}$ and $V_{r,J}$ are the maximal volume submatrices of W_r and V_r, respectively, and $U = A(I,J)$.

Since $\hat{U}^{\dagger} = (F_{n_3} \otimes I_{n_1})\texttt{bcirc}(\mathcal{U}^{\dagger})(F_{n_3}^{*} \otimes I_{n_1})$, we see that $\hat{U}^{\dagger}$ is a block diagonal matrix, each block of $\hat{U}^{\dagger}$ satisfies (4.2). Besides, $\hat{W}_{r,I,k_1}$ and $\hat{V}_{r,J,k_2}$ are the maximal volume submatrices of $\hat{W}_{r,k_1}$ and $\hat{V}_{r,k_2}$, then

$$\begin{cases} \|\mathcal{W}_{r,I}^{\dagger}\|_2 &= \max\limits_k \left\{\|\hat{W}_{r,I,k}^{\dagger}\|_2\right\} = \|\hat{W}_{r,I,k_1}^{\dagger}\|_2 \leqslant g(r,n_1,|I|) \\ \|\mathcal{V}_{r,J}^{\dagger}\|_2 &= \max\limits_k \left\{\|\hat{V}_{r,J,k}^{\dagger}\|_2\right\} = \|\hat{V}_{r,J,k_2}^{\dagger}\|_2 \leqslant g(r,n_2,|J|) \\ \|\mathcal{U}^{\dagger}\|_2 &= \max\limits_k \left\{\|\hat{U}_k^{\dagger}\|_2\right\} \leqslant \max\limits_k \left\{g(r,n_1,|I|)g(r,n_2,|J|)\|\hat{A}_k^{\dagger}\|_2\right\} \\ &= g(r,n_1,|I|)g(r,n_2,|J|)\|\mathcal{A}^{\dagger}\|_2. \end{cases}$$

□

Corollary 4.3. *With notations and assumptions of Proposition 4.2, if* $\|\mathcal{E}\|_2\|\mathcal{A}^{\dagger}\|_2 < \frac{1}{4\mu g_1 g_2}$, *then we have*

$$\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^{\dagger} * \mathcal{R}'\|_2 \leqslant \left[\frac{1}{2\mu} + \left(1 + \frac{1}{2\mu}\right)(g_1 + g_2) + \left(4 + \frac{1}{2\mu}\right)g_1 g_2\right]\|\mathcal{E}\|_2,$$

where $g_1 = g(r,n_1,|I|)$, $g_2 = g(r,n_2,|J|)$.

Proof. Combining Corollary 4.2 and Proposition 4.2, we obtain the result. □

Then we can write the error bound in the following form. Let $f(g_1,g_2)$ be a function of g_1, g_2, then

$$\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^{\dagger} * \mathcal{R}'\|_2 \leqslant f(g_1,g_2)O(\|\mathcal{E}\|_2). \tag{4.4}$$

If g_1 and g_2 are kept constants, then the error bound will be less than the first order of the spectrum norm of the noise tensor, it will not rely on the choice of the slice. Based on Proposition 4.2, g_1 and g_2 in (4.3) rely on the tubalrank, the tensor size and number of samples. We will verify it in the following section through simulations.

5. Numerical tests

In this section, we will verify the error estimation through numerical simulations. For computation of the T-product, we perform Algorithm T-MULT in [38], for the t-CUR decomposition, we follow Algorithm 1. All the numerical tests in this section are performed with MATLAB, where the MATLAB function of the T-product comes from [43, 44].

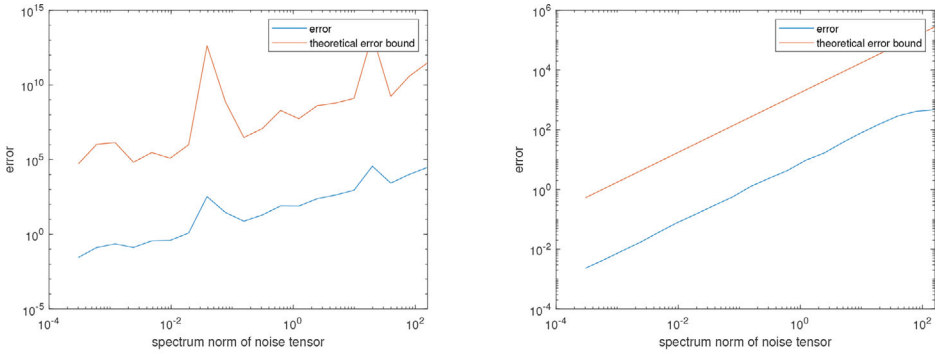
Algorithm 1. t-CUR decomposition

input $\mathcal{A} \in \mathbb{K}^{n_1 \times n_2 \times n_3}$; sample sizes d_1, d_2
$\hat{\mathcal{A}} = \texttt{fft}(\mathcal{A}, [], 3)$
Sample row index I and column index J, where $|I| = d_1$ and $|J| = d_2$
for $k = 1 : n_3$ **do**
  $\hat{\mathcal{C}}(:, :, k) = \hat{\mathcal{A}}(I, :, k)$
  $\hat{\mathcal{R}}(:, :, k) = \hat{\mathcal{A}}(:, J, k)$
  $\hat{\mathcal{U}}(:, :, k) = \hat{\mathcal{A}}(I, J, k)$
  such that $\hat{\mathcal{A}}(:, :, k) \approx \hat{\mathcal{C}}(:, :, k)\hat{\mathcal{U}}(:, :, k)^{\dagger}\hat{\mathcal{R}}(:, :, k)$
end for
$\mathcal{C} = \texttt{ifft}(\hat{\mathcal{C}}, [], 3)$
$\mathcal{R} = \texttt{ifft}(\hat{\mathcal{R}}, [], 3)$
$\mathcal{U} = \texttt{ifft}(\hat{\mathcal{A}}, [], 3)$
return $\mathcal{C}, \mathcal{U}, \mathcal{R}$ such that $\mathcal{A} \approx \mathcal{C} * \mathcal{U}^{\dagger} * \mathcal{R}$

Since our t-CUR decomposition in Theorem 3.1 requires that the subtensor $\mathcal{U}$ have the same multirank with the original data tensor $\mathcal{A}$, which means that each $\hat{A}_k$ should not have large rank $\text{rank}_m(\mathcal{A})$ and $\text{rank}_t(\mathcal{A})$ should not be large. Thus, in this section, we will use t-SVD to generate low-rank random tensor $\mathcal{A}$. We will also test tubalrank forced t-CUR decomposition in this section, i.e., $\mathcal{A} \approx \mathcal{C} * \mathcal{U}_r^{\dagger} * \mathcal{R}$, bacause the result in Section 4.2 shows a better error bound if we force the tubalrank of $\mathcal{U}$, where $\mathcal{U}_r$ is also generated by t-SVD from $\mathcal{U}$.

Example 5.1. This example verifies the error estimation in Corollaries 4.1 and 4.3. We generate a standard Gaussian random tensor $\mathcal{A} \in \mathbb{R}^{500 \times 500 \times 50}$ and force its tubalrank r to 50. Then we generate Gaussian random noise tensors with the same size of $\mathcal{A}$ with different standard deviation from 10^{-1} to 10^{-6}, we get the perturbed tensor $\mathcal{A}'$. Applying Algorithm 1 to $\mathcal{A}'$ and sampling 100 horizontal and lateral slices. We also force the tubalrank of $\mathcal{U}'$ to be r and compute the error $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}'^{\dagger} * \mathcal{R}'\|_2$ and $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^{\dagger} * \mathcal{R}'\|_2$. The result is shown in Figure 5.

Figure 5 shows that the error $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}'^{\dagger} * \mathcal{R}'\|_2$ and $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^{\dagger} * \mathcal{R}'\|_2$ increases almost linearly when $\|\mathcal{E}\|_2$ increases, yet the error in Figure 5(a) increases with fluctuations. In fact, $\mathcal{U}'$ is sampled from the perturbed tensor $\mathcal{A}'$ who usually has full tubalrank, which makes the tubalrank of $\mathcal{U}'$ and $\mathcal{C}' * \mathcal{U}'^{\dagger} * \mathcal{R}'$ larger than r (especially when many slices sampled), then it causes large deviation. If we force the tubalrank of $\mathcal{U}'$ to be r, as shown in Figure 5(b), then $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r'^{\dagger} * \mathcal{R}'\|_2$ increases linearly as $\|\mathcal{E}\|_2$ increases, which verifies the conclusion of Corollary 4.1 and (4.2).



(a) $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}'^{\dagger}*\mathcal{R}'\|_2$ vs. $\|\mathcal{E}\|_2$ (b) $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^{\dagger}*\mathcal{R}'\|_2$ vs. $\|\mathcal{E}\|_2$

Figure 5. Perturbation error bounds.

Example 5.2. Example 5.1 indicates that forcing the tubalrank can decrease the error, this example reveals the relationship between $\text{rank}_t(\mathcal{U})$ and relative error $\frac{\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^{\dagger}*\mathcal{R}'\|_2}{\|\mathcal{A}\|_2}$. We generate a standard Gaussian random tensor $\mathcal{A}\in\mathbb{R}^{500\times500\times50}$ and force its tubalrank r to 10. Then we generate the Gaussian random noise tensor $\mathcal{E}$ with the same size of $\mathcal{A}$, whose standard deviation is 10^{-5}, to get $\mathcal{A}'$. Applying the t-CUR decomposition to $\mathcal{A}'$ with 60 horizontal and lateral slices sampled, and forcing the tubalrank of $\mathcal{U}$ from $\text{rank}_t(\mathcal{A})$ to $\text{rank}_t(\mathcal{U}')$. Finally we compute the relative error. The result is displayed in Figure 6.

Figure 6 tells us that with the increasing of $\text{rank}_t(\mathcal{U})$, the error increases rapidly because of the perturbation. On the other hand, when tubalrank of $\mathcal{U}$ is close to that of $\mathcal{A}$, the relative error is rather small as expected.

Proposition 4.2 indicates that g_1 and g_2 rely on the sampled slice number $|I|,|J|$, the size n and $\text{rank}_t(\mathcal{A})$. Thus, the following examples will analyze how these factors will affect the error $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^{\dagger}*\mathcal{R}'\|_2$ respectively.

Example 5.3. To test the relationship between $|I|,|J|$ and the error, we generate a standard Gaussian random tensor $\mathcal{A}\in\mathbb{R}^{500\times500\times50}$, and force its tubalrank r to 50. Then we generate noise tensor $\mathcal{E}$ with standard deviation 10^{-5} as well, and get $\mathcal{A}'$. We apply the t-CUR decomposition to $\mathcal{A}'$ with $|I|=|J|=d$ horizontal and lateral slices sampled and force tubalrank of $\mathcal{U}'$ to r, where d varies from 60 to 300. For different d, we compute the error $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^{\dagger}*\mathcal{R}'\|_2$. The result is presented in Figure 7.

Example 5.4. To test the relationship between $\text{rank}_t(\mathcal{U})$ and the error, we generate a same standard Gaussian random tensor $\mathcal{A}\in\mathbb{R}^{500\times500\times50}$ with tubalrank r from 10 to 50. Then we generate a same noise tensor $\mathcal{E}$ with standard deviation 10^{-5} and get $\mathcal{A}'$. We apply the t-CUR decomposition to

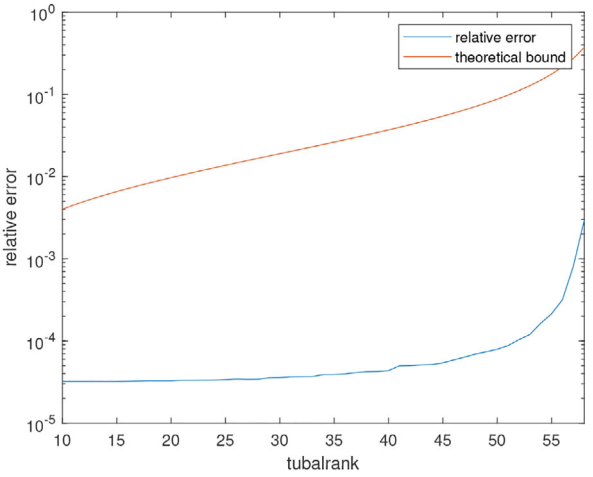


Figure 6. $\frac{\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'\dagger*\mathcal{R}'\|_2}{\|\mathcal{A}\|_2}$ vs. $\text{rank}_t(\mathcal{U})$.

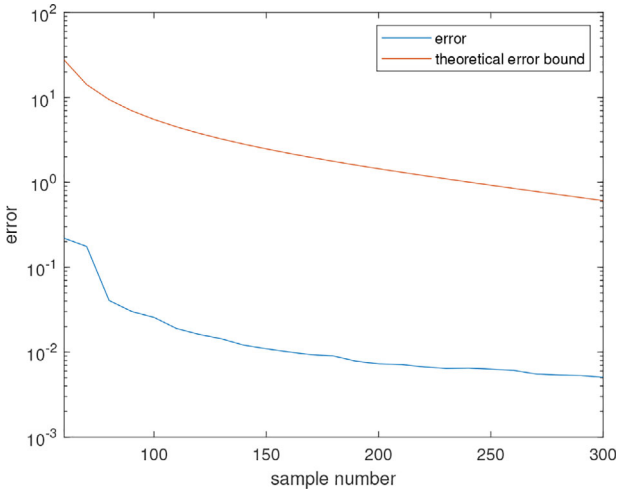


Figure 7. $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'\dagger*\mathcal{R}'\|_2$ vs. d.

$\mathcal{A}'$ with 100 horizontal and lateral slices sampled and force $\text{rank}_t(\mathcal{U})$ to r. For different $\text{rank}_t(\mathcal{A})$, we compute the error $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^{\dagger}*\mathcal{R}'\|_2$. The result is given in Figure 8.

Example 5.5. To test the relationship between size of $\mathcal{A}$ and the error, we generate a standard Gaussian random tensor $\mathcal{A}$ and Gaussian random perturbation tensor $\mathcal{E}$ with size of $n\times n\times 50$ as well, where $\mathcal{E}$ has standard deviation of 10^{-5}, n varies from 100 to 500. Forcing the tubalrank of $\mathcal{A}$ be 20, we can get $\mathcal{A}'$. We apply the t-CUR decomposition to $\mathcal{A}'$ with 60 horizontal and lateral slices sampled, and force $\text{rank}_t(\mathcal{U})$ to 20. Finally we compute the error $\|\mathcal{A}-\mathcal{C}'*\mathcal{U}_r'^{\dagger}*\mathcal{R}'\|_2$ for different n. The result is proposed in Figure 9.

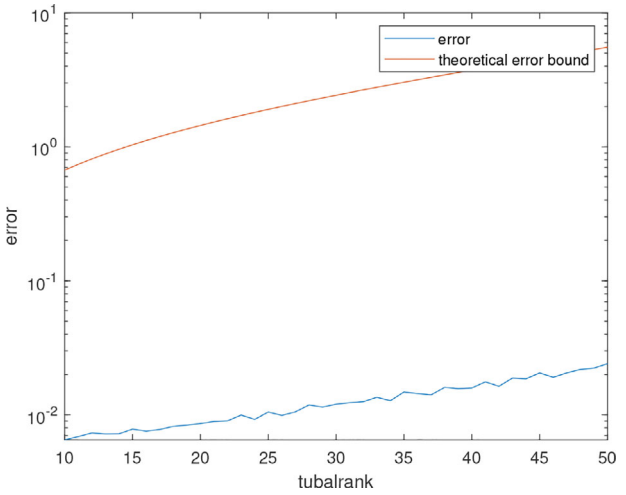


Figure 8. $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r' \dagger * \mathcal{R}'\|_2$ Vs. $\text{rank}_t(\mathcal{A})$.

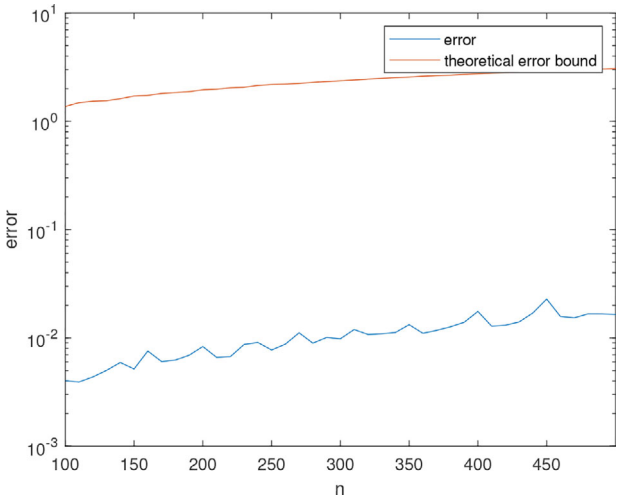


Figure 9. $\|\mathcal{A} - \mathcal{C}' * \mathcal{U}_r' \dagger * \mathcal{R}'\|_2$ Vs. Size of $\mathcal{A}$.

Figures 7–9 show that, as d, r and n increase, g_1 and g_2 increase, then the error will increases accordingly, which coincides with result in (4.4).

Finally, we list the advantage of t-CUR decomposition over t-SVD.

Example 5.6. We still generate the standard Gaussian random tensor $\mathcal{A} \in \mathbb{R}^{500\times500\times50}$ with $\text{rank}_t(\mathcal{A}) = 50$ forced and a same-size Gaussian random noise tensor $\mathcal{E}$, where $\mathcal{E}$ has standard deviation of 0.1. Then we get $\mathcal{A}'$. We apply the t-CUR decomposition to $\mathcal{A}'$ with 200 horizontal and lateral slices sampled, and apply the t-SVD with code in [43]. We repeat the experiment for 10 times and compute the average error and running time for two methods, where for t-CUR decomposition, we only compute the runtime of tubalrank forced t-CUR decomposition. The result is listed in Table 1.

Table 1. t-CUR decomposition vs. t-SVD.

Freq	$\|\mathcal{A}-\mathcal{C}*\mathcal{U}^{\dagger}*\mathcal{R}\|_2$	$\|\mathcal{A}-\mathcal{C}*\mathcal{U}_r^{\dagger}*\mathcal{R}\|_2$	$\|\mathcal{A}-\mathcal{W}*\mathcal{S}*\mathcal{V}^{*}\|_2$	t-CUR time	t-SVD time
1	2.7864e + 03	79.0350	31.5945	0.3267s	1.9410s
2	1.5022e + 03	79.2166	31.5945	0.3617s	1.8005s
3	1.1114e + 03	83.1840	31.5945	0.3235s	1.8279s
4	0.7534e + 03	77.5527	31.5945	0.3184s	1.7648s
5	1.4088e + 03	77.7718	31.5945	0.3371s	1.7724s
6	1.4758e + 03	80.5532	31.5945	0.3181s	1.7948s
7	0.6995e + 03	75.8949	31.5945	0.3519s	1.7913s
8	1.4142e + 04	86.2563	31.5945	0.3124s	1.7427s
9	1.2350e + 03	78.0261	31.5945	0.3239s	1.8382s
10	1.4000e + 03	78.2813	31.5945	0.3481s	2.4088s
Avg	2.6514e + 03	79.5772	31.5945	0.3320s	1.8682s

Table 1 reveals that forcing the tubalrank of the subtensor $\mathcal{U}$ can significantly reduce errors, although it is still larger than t-SVD. However, the runtime of t-CUR decomposition is significant lower than t-SVD, because t-CUR decomposition only needs random sampling the slice of the original tensor without tensor-tensor computation. Even if we still use t-SVD to force the tubalrank of $\mathcal{U}$, the overall running time is smaller than t-SVD as well, because the sampled subtensor has relative smaller size than the original tensor, which can save much time in the computing.

6. Conclusion

In order to overcome the storage difficulty and reduce the computational complexity of the large-scale tensor, we propose a tensor decomposition method to compress the massive data by extracting its key data. Under the T-product framework, we extend the matrix CUR decomposition to tensor case, the t-CUR decomposition. We make perturbation analysis of this decomposition and draw a conclusion that the error in the tensor spectral norm is correlated with the noise tensor in the first order.

The t-CUR decomposition requires the original data tensor to be low tubalrank, and the sample sizes are related with the rank, it should be prior estimated by t-SVD. In this paper, we restrict the original data to low-rank by t-SVD a priori. However, t-CUR decomposition has two advantages over t-SVD. First of all, it has good interpretability: the singular tensor generated by t-SVD span the slice space of the original tensor, it destroys the original data, while the sub-tensor generated by t-CUR decomposition preserves the original data and has clear physical meaning. Secondly, in terms of computing efficiency, t-CUR decomposition only needs random sampling the slice of the original tensor, and the final subtensor generated is small-scale in storage.

This paper provides a new idea for tensor data compression, and it has advantages of reducing the data scale without destroying the original data, which has more interpretable results.

Acknowledgments

We would like to thank the handling editor and two reviewers for their detailed comments.

Funding

This project is supported by the Basic Theory and Algorithm Research between Fudan University and Huawei Technology Investment Co. under grant YBN 19095097.

J. Chen is supported by the Innovation Program of Shanghai Municipal Education Committee and the National Natural Science Foundation of China under grant 11771099.

Y. Wei is supported by the National Natural Science Foundation of China under Shanghai Municipal Science and Technology Commission under grant 22WZ2501900.

ORCID

Yimin Wei http://orcid.org/0000-0001-6192-0546

References

[1] Kolda, T., Bader, B. (2009). Tensor decompositions and applications. *SIAM Rev*. 51(3):455–500. DOI: 10.1137/07070111X.

[2] Stewart, G. (1998). Matrix Algorithms: Volume 1: Basic Decompositions, SIAM.

[3] Che, M., Wei, Y. (2019). Randomized algorithms for the approximations of Tucker and the tensor train decompositions. *Adv. Comput. Math*. 45(1):395–428. DOI: 10.1007/s10444-018-9622-8.

[4] Cai, H., Hamm, K., Huang, L., Li, J., Wang, T. (2021). Rapid robust principal component analysis: CUR accelerated inexact low rank estimation. *IEEE Signal Process. Lett*. 28:116–120. DOI: 10.1109/LSP.2020.3044130.

[5] Goreinov, S., Tyrtyshnikov, E., Zamarashkin, N. (1997). A theory of pseudoskeleton approximations. *Linear Algebra Appl*. 261(1-3):1–21. DOI: 10.1016/S0024-3795(96)00301-1.

[6] Hamm, K., Huang, L. (2020). Perspectives on CUR decompositions. *Appl. Comput. Harmon. Anal*. 48(3):1088–1099. DOI: 10.1016/j.acha.2019.08.006.

[7] Hamm, K., Huang, L. (2021). Perturbations of CUR Decompositions. *SIAM J. Matrix Anal. Appl*. 42(1):351–375. DOI: 10.1137/19M128394X.

[8] Mahoney, M., Drineas, P. (2009). CUR matrix decompositions for improved data analysis. *Proc. Natl. Acad. Sci. U S A*. 106(3):697–702. DOI: 10.1073/pnas.0803205106.

[9] Mahoney, M., Maggioni, M., Drineas, P. (2008). Tensor-CUR decompositions for tensor-based data. *SIAM J. Matrix Anal. Appl*. 30(3):957–987. DOI: 10.1137/060665336.

[10] Sorensen, D., Embree, M. (2016). A DEIM induced CUR factorization. *SIAM J. Sci. Comput*. 38(3):A1454–A1482. DOI: 10.1137/140978430.

[11] Wang, S., Zhang, Z. (2013). Improving CUR matrix decomposition and the Nystrom approximation via adaptive sampling. *J. Mach. Learn. Res*. 14:2729–2769.

[12] Che, M., Qi, L., Wei, Y. (2016). Positive-definite tensors to nonlinear complementarity problems. *J. Optim. Theory Appl*. 168(2):475–487. DOI: 10.1007/s10957-015-0773-1.

[13] Che, M., Wei, Y. (2020). *Theory and Computation of Complex Tensors and Its Applications*, Singapore: Springer.

[14] Che, M., Wei, Y., Yan, H. (2020). The computation of low multilinear rank approximations of tensors via power scheme and random projection. *SIAM J. Matrix Anal. Appl*. 41(2):605–636. DOI: 10.1137/19M1237016.

[15] Ding, W., Wei, Y. (2016). Solving multi-linear systems with $\mathcal{M}$-tensors. *J. Sci. Comput*. 68:689–715.
[16] Du, S., Che, M., Wei, Y. (2020). Stochastic structured tensors to stochastic complementarity problems. *Comput. Optim. Appl*. 75(3):649–668. DOI: 10.1007/s10589-019-00144-3.
[17] Du, S., Ding, W., Wei, Y. (2021). Acceptable solutions and backward errors for tensor complementarity problems. *J. Optim. Theory Appl*. 188(1):260–276. DOI: 10.1007/s10957-020-01774-y.
[18] Ling, C., Liu, J., Ouyang, C., Qi, L. (2022). ST-SVD factorization and s-diagonal tensors. *Communi. Math. Sci*. 20:597–610.
[19] Qi, L., Yu, G. (2021). T-singular values and T-Sketching for third order tensors. *arXiv:2013.00976*.
[20] Wang, X., Che, M., Qi, L., Wei, Y. (2020). Modified gradient dynamic approach to the tensor complementarity problem. *Optim. Methods Softw*. 35(2):394–415. DOI: 10.1080/10556788.2019.1578766.
[21] Wang, X., Che, M., Wei, Y. (2018). Best rank-one approximation of fourth-order partially symmetric tensors by neural network. *NMTMA*. 11(4):673–700. DOI: 10.4208/nmtma.2018.s01.
[22] Wang, X., Che, M., Wei, Y. (2020). Tensor neural network models for tensor singular value decompositions. *Comput. Optim. Appl*. 75(3):753–777. DOI: 10.1007/s10589-020-00167-1.
[23] Wang, X., Che, M., Wei, Y. (2020). Neural network approach for solving nonsingular multi-linear tensor systems. *J. Comput. Appl. Math*. 368:e112569. DOI: 10.1016/j.cam.2019.112569.
[24] Wang, X., Mo, C., Qiao, S., Wei, Y. (2022). Predefined-time convergent neural networks for solving the time-varying nonsingular multi-linear tensor equations. *Neurocomputing*, 472:68–84.
[25] Lund, K. (2020). The tensor t-function: a definition for functions of third-order tensors. *Numer. Linear Algebra Appl*. 27(3):e2288. DOI: 10.1002/nla.2288.
[26] Miao, Y., Qi, L., Wei, Y. (2021). T-Jordan canonical form and T-Drazin inverse based on the T-product. *Commun. Appl. Math. Comput*. 3(2):201–220. DOI: 10.1007/s42967-019-00055-4.
[27] Kilmer, M., Martin, C. (2011). Factorization strategies for third-order tensors. *Linear Algebra Appl*. 435(3):641–658. DOI: 10.1016/j.laa.2010.09.020.
[28] Miao, Y., Qi, L., Wei, Y. (2020). Generalized tensor function via the tensor singular value decomposition based on the T-product. *Linear Algebra Appl*. 590:258–303. DOI: 10.1016/j.laa.2019.12.035.
[29] Tarzanagh, D., Michailidis, G. (2018). Fast randomized algorithms for T-product based tensor operations and decompositions with applications to imaging data. *SIAM J. Imaging Sci*. 11(4):2629–2664. DOI: 10.1137/17M1159932.
[30] Osinsky, A., Zamarashkin, N. (2018). Pseudo-skeleton approximations with better accuracy estimates. *Linear Algebra Appl*. 537:221–249. DOI: 10.1016/j.laa.2017.09.032.
[31] Penrose, R. (1955). A generalized inverse for matrices. *Math. Proc. Cambridge Philos. Soc*. 51(3):406–413. DOI: 10.1017/S0305004100030401.
[32] Wang, G., Wei, Y., Qiao, S. (2018). *Generalized Inverses: Theory and Computations*, 2nd ed. Beijing: Springer, Singapore and Science Press.
[33] Golub, G., Van Loan, C. (2013). *Matrix Computations*, 4th ed. Baltimore, MD: Johns Hopkins University Press.
[34] Sun, L., Zheng, B., Bu, C., Wei, Y. (2016). Moore-Penrose inverse of tensors via Einstein product. *Linear and Multilinear Algebra*. 64(4):686–698. DOI: 10.1080/03081087.2015.1083933.

[35] Cong, Z., Ma, H. (2022). Acute perturbation for Moore-Penrose inverses of tensors via the T-product. *J. Appl. Math. Comput*. DOI: 10.1007/s12190-021-01687-7.
[36] Ma, H., Li, N., Stanimirović, P., Katsikis, V. (2019). Perturbation theory for Moore-Penrose inverse of tensor via Einstein product. *Comput. Appl. Math*. 38(111) Paper
[37] Jin, X., Wei, Y., Zhao, Z. (2015). *Numerical Linear Algebra and Its Applications*, 2nd ed. Beijing: Science Press.
[38] Martin, C., Shafer, R., LaRue, B. (2013). An order-*p* tensor factorization with applications in imaging. *SIAM J. Sci. Comput*. 35(1):A474–A490. DOI: 10.1137/110841229.
[39] Bader, B., Kolda, T. (2021). Matlab tensor toolbox version Version 3.2.1, www.tensortoolbox.org., April 5.
[40] Tucker, L. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika*. 31(3):279–311. DOI: 10.1007/BF02289464.
[41] Behera, R., Sahoo, J., Mohapatra, R., Zuhair Nashed, M. (2022). Computation of generalized inverses of tensors via t-product. *Numer. Linear Algebra App*. 29(2): e2416. DOI: 10.1002/nla.2416.
[42] Horn, R., Johnson, C. (1991). *Topics in Matrix Analysis*. Cambridge: Cambridge University Press,
[43] Lu, C. Tensor-tensor product toolbox. Carnegie Mellon University, June 2018. preprint arXiv:1806.07247. https://github.com/canyilu/tproduct.
[44] Lu, C., Feng, J., Chen, Y., Liu, W., Lin, Z., Yan, S. (2020). Tensor robust principal component analysis with a new tensor nuclear norm. *IEEE Trans. Pattern Anal. Mach. Intell*. 42(4):925–938. DOI: 10.1109/TPAMI.2019.2891760.
[45] Boutsidis, C., Woodruff, D. (2017). Optimal CUR matrix decompositions. *SIAM J. Comput*. 46(2):543–589. DOI: 10.1137/140977898.
[46] Drineas, P., Mahoney, M., Muthukrishnan, S. (2008). Relative-error CUR matrix decompositions. *SIAM J. Matrix Anal. Appl*. 30(2):844–881. DOI: 10.1137/07070471X.
[47] Li, L., Huang, W., Gu, I., Tian, Q. (2004). Statistical modeling of complex backgrounds for foreground object detection. *IEEE Trans. Image Process*. 13(11): 1459–1472. DOI: 10.1109/tip.2004.836169.
[48] Lin, X., Ng, M., Zhao, X. (2020). Tensor factorization with total variation and Tikhonov regularization for low-rank tensor completion in imaging data. *J. Math. Imaging Vis*. 62(6–7):900–918. DOI: 10.1007/s10851-019-00933-9.
[49] Martin, C., Fowlkes, C., Tal, D., Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV, Vol. 2. IEEE, pp. 416–423.
[50] Mickelin, O., Karaman, S. (2020). Multi-resolution low-rank tensor formats. *SIAM J. Matrix Anal. Appl*. 41(3):1086–1114. DOI: 10.1137/19M1284579.
[51] Tropp, J. (2009). Column subset selection, matrix factorization, and eigenvalue optimization. In: Proc. 2009 ACM-SIAM Symp. Discrete Algorithms (SODA), New York, NY, Jan 2009, pp. 978–986.
[52] Wei, Y., Ding, W. (2016). *Theory and Computation of Tensors: Multi-Dimensional Arrays*, London: Elsevier/Academic Press.
[53] Zhang, J., Saibaba, A., Kilmer, M., Aeron, S. (2018). A randomized tensor singular value decomposition based on the T-product. *Numer. Linear Algebra Appl*. 25(5): e2179. DOI: 10.1002/nla.2179.