Research Article Computational Mathematics and Computer Modeling with Applications

Received 10 December 2021
Accepted 31 December 2021

DOI: 10.52547/CMCMA.1.1.1
AMS Subject Classification: 15A18; 15A69; 65F15; 65F10

Tensor LU and QR decompositions and their randomized algorithms

Yuefeng Zhu[a] **and Yimin Wei**[b]

In this paper, we propose two decompositions extended from matrices to tensors, including LU and QR decompositions with their rank-revealing and randomized variations.
We give the growth order analysis of error of the tensor QR (t-QR) and tensor LU (t-LU) decompositions. Growth order of error and running time are shown by numerical examples. We test our methods by compressing and analyzing the image-based data, showing that the performance of tensor randomized QR decomposition is better than the tensor randomized SVD (t-rSVD) in terms of the accuracy, running time and memory.

Keywords: LU decomposition; QR decomposition; rank-revealing algorithm; randomized algorithm; tensor T-product; low-rank approximation.

1. Introduction

As high-dimensional analogues of matrices, tensors are extensions of matrices. The difference is that a matrix entry a_{ij} has two indices i and j, while a tensor entry $a_{i_1 i_2 \dots i_m}$ has m subscripts $i_1, i_2, \dots, i_m$. m is called the order of tensor, if the tensor has m subscripts. Let $\mathbb{C}$ be the complex field and $\mathbb{R}$ be the real field. For a positive integer N, let $[N] = 1, 2, \dots, N$, A tensor is called a real tensor if all its entries are in $\mathbb{R}$ and a tensor is a complex tensor if all its entries are in $\mathbb{C}$.

Recently, the tensor T-product has been proved to be a useful tool in many real applications [1, 2, 7, 8, 6, 5, 13, 21, 18, 23, 24, 28, 29, 30, 31, 32, 33, 34, 37, 46, 48]. Wang et al. [40] investigate the tensor neural network models based on the tensor singular value decomposition (T-SVD). The theory and computations of the tensor can be found in monographs [10, 42].

Several matrix decompositions have been proposed for the low rank approximation, including the rank-revealing LU decomposition (RRLU) [20], rank-revealing QR decomposition (RRQR) [4, 14, 44], randomized LU decomposition (RLU) [35, 26], randomized QR decomposition (RQR) [16] and randomized singular value decomposition (RSVD) [27]. Among these algorithms, rank-revealing algorithms have high accuracy while randomized algorithms feature low running time at the cost of the accuracy.

In [47], RSVD has been applied to t-SVD problem with an efficient implementation. In this paper, we generalize other matrix decompositions to tensor case and present the t-LU, t-QR, t-RRLU, t-RRQR, t-RLU and t-RQR algorithms, respectively.

This paper is organized as follows. Section 2 gives some reviews of the definition of tensor T-product and some algebraic structures of third order tensors via this kind of product. Section 3 provides definition and theoretical round-off analysis of the t-LU and t-QR algorithms. Section 4 presents growth order of error and time cost estimated by numerical experiments. In this section we show how the t-RQR outperforms t-RSVD in experiments of the low-rank decomposition of the image data. We conclude our paper in Section 5.

[a] *School of Mathematical Sciences, Fudan University, Shanghai, P.R. China. E-mail: 17300180026@fudan.edu.cn. This author is supported by the National Natural Science Foundation of China under grant 11771099,*
[b] *School of Mathematical Sciences, Fudan University, Shanghai, P.R. China. E-mail: ymwei@fudan.edu.cn. This author is supported by the Innovation Program of Shanghai Municipal Education Commission.*
[*] *Correspondence to: Y. Wei.*

2. Preliminary

2.1. Notation and index

A multiplication is presented for third-order tensors. Suppose that we have two tensors $\mathcal{A} \in \mathbb{R}^{m\times n\times p}$ and $\mathcal{B} \in \mathbb{R}^{n\times s\times p}$ and denote their frontal faces respectively as $A^{(k)} \in \mathbb{R}^{m\times n}$ and $B^{(k)} \in \mathbb{R}^{n\times s}$, $k = 1, 2, \cdots, p$. The operations bcirc, unfold and fold can be defined as [18, 23, 24],

$$bcirc(\mathcal{A}) := \begin{bmatrix} \mathcal{A}^{(1)} & \mathcal{A}^{(p)} & \cdots & \mathcal{A}^{(2)} \\ \mathcal{A}^{(2)} & \mathcal{A}^{(1)} & \cdots & \mathcal{A}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{A}^{(p)} & \mathcal{A}^{(p-1)} & \cdots & \mathcal{A}^{(1)} \end{bmatrix}, \qquad unfold(\mathcal{A}) := \begin{bmatrix} \mathcal{A}^{(1)} \\ \mathcal{A}^{(2)} \\ \vdots \\ \mathcal{A}^{(p)} \end{bmatrix},$$

and $fold(unfold(\mathcal{A})) := \mathcal{A}$.

2.2. Tensor T-product and tensor norm

The following definitions on the tensor T-product have been developed in [18, 23, 24].

Definition 1 *Let $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$ and $\mathcal{B} \in \mathbb{C}^{n\times s\times p}$ be complex tensors. Then the T-product multiplication $\mathcal{A} * \mathcal{B}$ is defined as an $m \times s \times p$ complex tensor*

$$\mathcal{A} * \mathcal{B} := fold(bcirc(\mathcal{A})unfold(\mathcal{B})).$$

Definition 2 *Suppose that $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$. The transpose $\mathcal{A}^\top$ is defined by transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through p. The conjugate transpose $\mathcal{A}^H$ is defined by conjugate transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through p.*

Definition 3 *The $n \times n \times p$ identity tensor $\mathcal{I}_{nnp}$ is the tensor whose first frontal slice is the $n \times n$ identity matrix I_n, and whose other frontal slices are all zeros.*

It is obvious to find that $\mathcal{A} * \mathcal{I}_{nnp} = \mathcal{I}_{nnp} * \mathcal{A} = \mathcal{A}$ for all $\mathcal{A} \in \mathbb{R}^{m\times n\times p}$.

Definition 4 *A real-valued tensor $\mathcal{P} \in \mathbb{R}^{n\times n\times p}$ is called orthogonal if $\mathcal{P}^\top * \mathcal{P} = \mathcal{P} * \mathcal{P}^\top = \mathcal{I}$. A complex-valued tensor $\mathcal{Q} \in \mathbb{C}^{n\times n\times p}$ is unitary if $\mathcal{Q}^H * \mathcal{Q} = \mathcal{Q} * \mathcal{Q}^H = \mathcal{I}$.*

Definition 5 *Let $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$ be a complex-valued tensor. The tensor norm is deduced as*

$$\|\mathcal{A}\| := \|bcirc(\mathcal{A})\|.$$

The l_∞ norm, the Frobenius norm and the infinity norm are defined, respectively, as

$$\begin{cases} \|\mathcal{A}\|_{l_\infty} := \|bcirc(\mathcal{A})\|_{l_\infty} = \max\limits_{i,j,k} \left|\mathcal{A}^{(k)}(i,j)\right|, \\ \|\mathcal{A}\|_F := \|bcirc(\mathcal{A})\|_F = \left(p \sum\limits_{i,j,k} \left|\mathcal{A}^{(k)}(i,j)\right|^2\right)^{\frac{1}{2}}, \\ \|\mathcal{A}\|_\infty := \|bcirc(\mathcal{A})\|_\infty = \max\limits_{j} \sum\limits_{i,k} \left|\mathcal{A}^{(k)}(i,j)\right|. \end{cases}$$

Kilmer et al. [24] show the fast Fourier transformation (FFT) is useful in the tensor computation under the T-product.

Lemma 1 *Let $\mathcal{A} \in \mathbb{R}^{m\times n\times p}$ be a real tensor. There uniquely exists a tensor $\mathcal{D} \in \mathbb{R}^{m\times n\times p}$, such that*

$$\begin{bmatrix} \mathcal{A}^{(1)} & \mathcal{A}^{(p)} & \cdots & \mathcal{A}^{(2)} \\ \mathcal{A}^{(2)} & \mathcal{A}^{(1)} & \cdots & \mathcal{A}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{A}^{(p)} & \mathcal{A}^{(p-1)} & \cdots & \mathcal{A}^{(1)} \end{bmatrix} = (F_p \otimes I_m) \begin{bmatrix} \mathcal{D}^{(1)} & & & \\ & \mathcal{D}^{(2)} & & \\ & & \ddots & \\ & & & \mathcal{D}^{(p)} \end{bmatrix} (F_p^* \otimes I_n),$$

where F_n is the discrete Fourier matrix of order n, I_m and I_n are identity matrices of order m and n, respectively.

Furthermore, the computation shows that

$$unfold(\mathcal{D}) = \Big((diag(1, \omega, \omega^2, \cdots, \omega^p)F_p^*) \otimes I_m\Big) unfold(\mathcal{A}).$$

In other words, every frontal slice of $\mathcal{D}$ is a weight sum of frontal slices of $\mathcal{A}$, with the weights being some power of $\omega = e^{-2\pi\imath/n}$, ω is the primitive n-th root of unity in which $\imath = \sqrt{-1}$. Therefore, $\mathcal{D}^{(1)}, \mathcal{D}^{(2)}, \cdots, \mathcal{D}^{(p)}$ are lower-triangular (upper-triangular) matrices if and only if $\mathcal{A}^{(1)}, \mathcal{A}^{(2)}, \cdots, \mathcal{A}^{(p)}$ are lower-triangular (upper-triangular) matrices.

2.3. Round-off error of the matrix multiplication and decompositions

When analyzing round-off errors, we denote the floating point presentation of a tensor $\mathcal{A}$ and a matrix A respectively as $\widehat{\mathcal{A}}$ and $\widehat{A}$. Machine accuracy is denoted by u.

Round-off error of the matrix multiplication, the condition numbers of LU and the QR decompositions [25, 38, 45] are well-studied and recorded in textbooks like [17, 19, 22, 36, 43].

Lemma 2 (Matrix multiplication round-off error) *For matrices $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{n\times p}$, it holds*

$$|\widehat{AB} - AB| \leq (2n\mathrm{u} + \mathcal{O}(\mathrm{u}^2))|A||B|.$$

Lemma 3 *Let $A \in \mathbb{C}^{m\times n}$ be a complex matrix. Let $q = \min(m, n)$. Then A has the complete pivoting LU decomposition $PAQ = LU$, where $L \in \mathbb{R}^{m\times q}$ is a unit lower triangular matrix and $U \in \mathbb{R}^{q\times n}$ is an upper triangular matrix, while P, Q are permutation matrices. The floating point operation introduces an error bounded by*

$$|\widehat{P}A\widehat{Q} - \widehat{L}\widehat{U}| \leq (2q\mathrm{u} + \mathcal{O}(\mathrm{u}^2))|\widehat{L}||\widehat{U}|.$$

Furthermore, we have

$$\|\widehat{P}A\widehat{Q} - \widehat{L}\widehat{U}\|_\infty \leq (2q^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\|\widehat{U}\|_\infty \leq (2q^3\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\rho\|A\|_\infty,$$

where $\rho := \max_{i,j}|\widehat{U}(i,j)|/\max_{i,j}|A(i,j)|$ is called the growth factor.

Remark 1 *Naturally we expect a theoretical bound estimation for ρ simply by examining A. Unfortunately, there exist examples of A where $\rho = 2^{q-1}$, which is quite large. Meanwhile, we usually have $\rho \sim \mathcal{O}(q^{\frac{2}{3}})$ in practise. In previous literature it is also mentioned that the factor q^3 is not an operating factor in practise and may be ignored.*

Remark 2 *A complete pivoting LU decomposition requires $2mnq + \frac{1}{3}q^3 - \frac{1}{2}(m+n)q^2$ flops.*

An upper bound of QR decomposition round-off error was given through the proof of Theorem 18.12 in Higham's book [19].

Lemma 4 *Suppose that the modified Gram-Schmidt method is applied to $A \in \mathbb{R}^{m\times n}$ of rank n, yielding computed matrices $\widehat{Q} \in \mathbb{R}^{m\times n}$ and $\widehat{R} \in \mathbb{R}^{n\times n}$. Then*

$$\|A - \widehat{Q}\widehat{R}\|_F \leq 4n^2\mathrm{u}\|A\|_F.$$

Remark 3 *Modified Gram-Schmidt method for the QR decomposition requires $2mn^2$ flops.*

2.4. Numerical rank

It has been pointed out in [17] that matrix rank can be defined equivalently in different ways. A widely used definition is through the singular value decomposition (SVD).

Definition 6 *(Matrix rank) Let $A \in \mathbb{R}^{m\times n}$ and $q = \min(m, n)$. Suppose its singular value decomposition is $A = U\Sigma V^\top$, where $U \in \mathbb{R}^{m\times q}, V \in \mathbb{R}^{q\times r}$ are column orthogonal, $\Sigma = diag(\sigma_1, \sigma_2, \cdots, \sigma_r, \cdots, 0, \cdots, 0)$ is diagonal, and $\sigma_i \neq 0, i = 1, 2, \ldots, r$. Then the rank of matrix A is r.*

Also in [17] it is revealed that, unless remarkable cancellation occurs, none of the computational singular values would be zero because of round-off error. However, we can liberalize the notion of rank by setting small computed singular values to zero and gain an approximation, called numerical rank. When we choose a tolerance δ as the definition of "small", the computed rank $\hat{r}$ is called δ-rank of A.

Definition 7 *(Numerical rank) Let $A \in \mathbb{R}^{m\times n}$ and $q = \min(m, n)$. Suppose its singular value decomposition is $A = U\Sigma V^\top$, where $\hat{\Sigma} = diag(\hat{\sigma}_1, \hat{\sigma}_2, \cdots, \hat{\sigma}_q)$. If the singular values satisfy that $\hat{\sigma}_1 \geq \cdots \geq \hat{\sigma}_r \geq \delta > \hat{\sigma}_{r+1} \geq \ldots \geq \hat{\sigma}_q$, then $\hat{r}$ is called the δ-rank of A.*

For a given δ it is important to stress that the determination of numerical rank is a sensitive computation. If the gap between $\hat{\sigma}_{\hat{r}}$ and $\hat{\sigma}_{\hat{r}+1}$ is small, then A is also close to a matrix with rank $\hat{r} - 1$. Thus, the amount of confidence we have in the correctness of $\hat{r}$ depends on the gap between $\hat{\sigma}_{\hat{r}}$ and $\hat{\sigma}_{\hat{r}+1}$.

2.5. Rank-revealing LU and rank-revealing QR decompositions

Memory and operations of large dense matrices are costly. We can reduce the cost by representing a large matrix as the product of low-rank matrices with an acceptable error, i.e., the matrix low-rank approximation. One way to do low-rank approximation is rank-revealing decompositions.

For an n-order real square matrix A, Businger and Golub [3] presented QR with column pivoting (QRCP). Their idea was to pivot (choose the column with largest norm in lower-right block of R) in a QR decomposition implemented with the Householder transformation, and interchange it with the first column in that block. With the assumption that round-off error has no effect on pivot, the matrix in $k-$th step of the QRCP can be written as

$$\mathrm{fl}(H_k \cdots H_1 A \Pi_1 \cdots \Pi_k) = \widehat{R}^{(k)} = \begin{bmatrix} \widehat{R}_{11}^{(k)} & \widehat{R}_{12}^{(k)} \\ O & \widehat{R}_{22}^{(k)} \end{bmatrix},$$

where $H_i, i = 1, 2, \ldots, k$ are Householder matrices, $\Pi_i, i = 1, 2, \ldots, k$ are permutation matrices that record the pivoting. The block $\widehat{R}_{11}^{(k)}$ is of size $k \times k$.

A natural result of pivoting is that the lower-right block has a small norm: $\|\widehat{R}_{22}^{(k)}(:, i)\|_2 \leq |\widehat{R}_{11}^{(k)}(k, k)|, \forall i = 1, 2, \ldots, n-k$. Since $\sigma_{k+1}(A) \leq \|R_{22}^{(k)}\|_2$, we can conclude that the numerical rank of A is the smallest k satisfying $\|R_{22}^{(k)}\|_2 \leq \delta$.

Chan [4] presents the rank-revealing QR decomposition (RRQR) as a result:

Definition 8 *(Rank-revealing QR decomposition) Let $n-k$ be the numerical rank of $A \in \mathbb{R}^{n\times n}$. Suppose that there are the permutation matrix Π, the orthogonal matrix Q and the block upper triangular matrix R, such that*

$$A\Pi = QR = Q \begin{bmatrix} R_{11} & R_{12} \\ O & R_{22} \end{bmatrix}, \tag{1}$$

where R_{11} is upper triangular with $(n-k)$-th order, and $\|R_{22}\|_2 \leq \delta$ for some small δ in some norm. Then (1) is called the rank-revealing QR factorization of A.

Existence of such decomposition can be easily shown since there must exist zero diagonal element of R in case A is not of full-rank.

This definition can be generalized directly into complex matrices and rectangular matrices.

Remark 4 *Though we take the 2-norm for theoretical convenience, the Frobenius norm is preferred in practice for its high performance.*

Rank-revealing QR decomposition has also been generalized into the LU decomposition [20].

Definition 9 *(Rank-revealing LU decomposition) Let $n-k$ be the numerical rank of $A \in \mathbb{R}^{n\times n}$. Suppose that there are permutation matrices P, Q, a block lower triangular matrix L and a block upper triangular matrix U, such that*

$$PAQ = \begin{bmatrix} L_{11} & O \\ L_{21} & I_k \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ O & U_{22} \end{bmatrix}, \tag{2}$$

where U_{11} is upper triangular with $(n-k)$-th order, and $\|U_{22}\|_2 \leq \delta$ for some small δ in some norm. Then (2) is called the rank-revealing LU factorization of A.

In the rank-revealing algorithms, the lower-right block of U or R has a small norm. This lead to the approximations

$$A\Pi = Q \begin{bmatrix} R_{11} & R_{12} \\ O & R_{22} \end{bmatrix} \approx Q(:, 1:k) \begin{bmatrix} R_{11} & R_{12} \end{bmatrix}, \tag{3}$$

and

$$PAQ = \begin{bmatrix} L_{11} & O \\ L_{21} & I_k \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ O & U_{22} \end{bmatrix} \approx \begin{bmatrix} L_{11} \\ L_{21} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \end{bmatrix}. \tag{4}$$

This means we have two implements of the low-rank approximation through rank-revealing decompositions.

2.6. Randomized LU and randomized QR decompositions

Another approach of low-rank approximation is randomized algorithms. They feature high efficiency and easy parallelization. A well-known example is the randomized singular value decomposition (RSVD) in [27]: Given matrix $A \in \mathbb{R}^{m\times n}$ and expected rank $r \ll \min(m, n)$, compute $U \in \mathbb{R}^{m\times r}, \Sigma = diag(\sigma_1, \ldots, \sigma_r), V \in \mathbb{R}^{n\times r}$ with proper designed random sampling, such that $A \approx U\Sigma V^\top$.

Algorithm 1: Randomized LU decomposition.

Input: A matrix of size $m \times n$ to decompose, k desired rank, $l \geq k$ number of columns to use.
Output: Matrices P, Q, L, U such that $\|PAQ - LU\|_F \leq \mathcal{O}(\sigma_{k+1}(A))$, where P and Q are orthogonal permutation matrices, L and U are the lower and upper triangular matrices, respectively.

1 Create a matrix G of size $n \times l$ whose entries are i.i.d. Gaussian random variables with zero mean and unit standard deviation.
2 $Y \leftarrow AG$.
3 Apply RRLU decomposition to Y such that $PYQ_y = L_y U_y$.
4 Truncate L_y and U_y by choosing the first k columns and the first k rows, respectively, such that $L_y \leftarrow L_y(:, 1:k), U_y \leftarrow U_y(1:k, :)$.
5 $B \leftarrow L_y^{\dagger} PA$.
6 Apply LU decomposition to B with column pivoting $BQ = L_b U_b$.
7 $L \leftarrow L_y L_b$.
8 $U \leftarrow U_b$.

Shabat [35] presented the randomized LU decomposition (RLU) as a generalization. Its input contains the matrix A to decompose and two parameters k and l. The main idea is to randomly sample l columns in A, do (rank-revealing) LU decomposition, and truncate the factors L and U to rank k.

In this paper that the error behavior is closely related with parameters: error decreases exponentially as k grows. For a small $k \approx \mathcal{O}(1)$, the error can be as great as 1e-1.

In consideration of the efficiency, the parameters for $A \in \mathbb{R}^{n\times n}$ are often chosen as $l = \log_2^2(n)$, $k = l - 5$. In case tne accuracy is of more importance, we may double l and take $k = l - 5$.

3. Theoretical analysis of the round-off error

In this section, we generalize complete pivoting LU and QR decompositions to the third-order tensors under the T-product in detail. Both rank-revealing and randomized algorithms of both LU and QR decompositions can be generalized to tensors through the same process. We also show the round-off error of tensor LU and QR decompositions.

3.1. Tensor LU and tensor QR decompositions

We first introduce definition of tensor LU decomposition (t-LU) and tensor QR decomposition (t-QR). A special case of t-LU for third-order tensor with square frontal slices was given in [31] and we simply generalize it. The definition of t-QR was given in [18, Theorem 6.1].

Definition 10 *(Tensor LU decomposition) Let $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$ be a complex tensor. Let $\mathcal{D} \in \mathbb{C}^{m\times n\times p}$ be the unique tensor determined by Lemma 1:*

$$bcirc(\mathcal{A}) = (F_p \otimes I_m) diag(\mathcal{D}^{(1)}, \mathcal{D}^{(2)}, \cdots, \mathcal{D}^{(p)})(F_p^* \otimes I_n).$$

Let $\mathcal{P}_{\mathcal{D}}{}^{(k)}\mathcal{D}^{(k)}\mathcal{Q}_{\mathcal{D}}{}^{(k)} = \mathcal{L}_{\mathcal{D}}{}^{(k)}\mathcal{U}_{\mathcal{D}}{}^{(k)}$, $k = 1, 2, \ldots, p$ be the complete pivoting LU decomposition of corresponding frontal slices $\mathcal{D}^{(k)}$, where $\mathcal{L}_{\mathcal{D}}{}^{(k)} \in \mathbb{R}^{m\times q}, \mathcal{U}_{\mathcal{D}}{}^{(k)} \in \mathbb{R}^{q\times n}$, $q = \min(m, n)$. Then the tensors $\mathcal{P}_{\mathcal{A}}, \mathcal{Q}_{\mathcal{A}}, \mathcal{L}_{\mathcal{A}}, \mathcal{U}_{\mathcal{A}}$ defined by

$$\begin{cases} bcirc(\mathcal{P}_{\mathcal{A}}) = (F_p \otimes I_m) diag(\mathcal{P}_{\mathcal{D}}{}^{(1)}, \mathcal{P}_{\mathcal{D}}{}^{(2)}, \cdots, \mathcal{P}_{\mathcal{D}}{}^{(p)})(F_p^* \otimes I_m), \\ bcirc(\mathcal{Q}_{\mathcal{A}}) = (F_p \otimes I_n) diag(\mathcal{Q}_{\mathcal{D}}{}^{(1)}, \mathcal{Q}_{\mathcal{D}}{}^{(2)}, \cdots, \mathcal{Q}_{\mathcal{D}}{}^{(p)})(F_p^* \otimes I_n), \\ bcirc(\mathcal{L}_{\mathcal{A}}) = (F_p \otimes I_m) diag(\mathcal{L}_{\mathcal{D}}{}^{(1)}, \mathcal{L}_{\mathcal{D}}{}^{(2)}, \cdots, \mathcal{L}_{\mathcal{D}}{}^{(p)})(F_p^* \otimes I_q), \\ bcirc(\mathcal{U}_{\mathcal{A}}) = (F_p \otimes I_q) diag(\mathcal{U}_{\mathcal{D}}{}^{(1)}, \mathcal{U}_{\mathcal{D}}{}^{(2)}, \cdots, \mathcal{U}_{\mathcal{D}}{}^{(p)})(F_p^* \otimes I_n) \end{cases}$$

satisfy that

$$\mathcal{P}_{\mathcal{A}} * \mathcal{A} * \mathcal{Q}_{\mathcal{A}} = \mathcal{L}_{\mathcal{A}} * \mathcal{U}_{\mathcal{A}}. \tag{5}$$

The formula (5) is called the tensor LU decomposition of $\mathcal{A}$.

Definition 11 *(Tensor QR decomposition) Let $\mathcal{A} \in \mathbb{R}^{m\times n\times p}$ be a complex tensor. Let $\mathcal{D} \in \mathbb{R}^{m\times n\times p}$ be the unique tensor determined by Lemma 1:*

$$bcirc(\mathcal{A}) = (F_p \otimes I_m) diag(\mathcal{D}^{(1)}, \mathcal{D}^{(2)}, \cdots, \mathcal{D}^{(p)})(F_p^* \otimes I_n).$$

Let $\mathcal{D}^{(k)} = \mathcal{Q}_{\mathcal{D}}{}^{(k)}\mathcal{R}_{\mathcal{D}}{}^{(k)}$, $k = 1, 2, \ldots, p$ be the QR decomposition of corresponding frontal slices $\mathcal{D}^{(k)}$, where $\mathcal{Q}_{\mathcal{D}}{}^{(k)} \in \mathbb{R}^{m\times n}$ is a orthogonal and normalized column, $\mathcal{R}_{\mathcal{D}}{}^{(k)} \in \mathbb{R}^{n\times n}$ is upper triangular. Then the tensors $\mathcal{Q}_{\mathcal{A}}, \mathcal{R}_{\mathcal{A}}$ defined by

$$\begin{cases} bcirc(\mathcal{Q}_{\mathcal{A}}) = (F_p \otimes I_m) diag(\mathcal{Q}_{\mathcal{D}}{}^{(1)}, \mathcal{Q}_{\mathcal{D}}{}^{(2)}, \cdots, \mathcal{Q}_{\mathcal{D}}{}^{(p)})(F_p^* \otimes I_n), \\ bcirc(\mathcal{R}_{\mathcal{A}}) = (F_p \otimes I_n) diag(\mathcal{R}_{\mathcal{D}}{}^{(1)}, \mathcal{R}_{\mathcal{D}}{}^{(2)}, \cdots, \mathcal{R}_{\mathcal{D}}{}^{(p)})(F_p^* \otimes I_n) \end{cases}$$

satisfy that

$$\mathcal{A} = \mathcal{Q}_{\mathcal{A}} * \mathcal{R}_{\mathcal{A}}. \tag{6}$$

The formula (6) is called the tensor QR decomposition of $\mathcal{A}$.

By the exactly same method of the Fourier transformation and the slice-wise decomposition, we can define the tensor rank-revealing LU and QR decompositions (t-RRLU, t-RRQR) and tensor randomized LU and QR decompositions (t-RLU, t-RQR), respectively. All these methods are tested in numerical experiments.

3.2. Theoretical analysis of the round-off error

In this subsection we discuss the round-off error to explain our confidence for practice: the process of the Fourier transformation, then algorithms in more detail.

The Fourier transform, or a block diagonalization of $bcirc(\mathcal{A})$ process can be viewed as computing the weighted sum of p matrices of size $m \times n$, where the weights are some power of $\omega = e^{\frac{-2\pi i}{p}}$. It can also be considered as doing matrices block production twice, each of which introduces round-off error by at most $2p\mathrm{u}$ times true value:

$$\left|\widehat{\mathcal{D}^{(k)}} - \mathcal{D}^{(k)}\right| \leq (4p\mathrm{u} + \mathcal{O}(\mathrm{u}^2))|\mathcal{D}^{(k)}|.$$

For the t-LU, it deserves mentioning that frontal slices $\mathcal{P}_{\mathcal{D}}{}^{(k)}$, $\mathcal{Q}_{\mathcal{D}}{}^{(k)}$ are permutation matrices, which introduce no round-off error whenever multiplied by another matrix. Therefore the error introduced by block-wise LU decomposition is bounded by

$$\left|\mathcal{P}_{\mathcal{D}}{}^{(k)}\widehat{\mathcal{D}^{(k)}}\mathcal{Q}_{\mathcal{D}}{}^{(k)} - \widehat{\mathcal{L}_{\mathcal{D}}{}^{(k)}}\widehat{\mathcal{U}_{\mathcal{D}}{}^{(k)}}\right| \leq (2q\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\left|\widehat{\mathcal{L}_{\mathcal{D}}{}^{(k)}}\right|\left|\widehat{\mathcal{U}_{\mathcal{D}}{}^{(k)}}\right|, \quad k = 1, 2, \ldots, p.$$

Then for our focus $\mathcal{E}_{LU} = \left|\mathcal{P}_{\mathcal{A}} * \mathcal{A} * \mathcal{Q}_{\mathcal{A}} - \widehat{\mathcal{L}_{\mathcal{A}}} * \widehat{\mathcal{U}_{\mathcal{A}}}\right|$, it holds that

$$\begin{aligned}
bcirc(\mathcal{E}_{LU}) &= \left|bcirc(\mathcal{P}_{\mathcal{A}})bcirc(\mathcal{A})bcirc(\mathcal{Q}_{\mathcal{A}}) - bcirc(\widehat{\mathcal{L}_{\mathcal{A}}})bcirc(\widehat{\mathcal{U}_{\mathcal{A}}})\right| \\
&= \left|(F_p \otimes I_m)[diag((\mathcal{P}_{\mathcal{D}} * \mathcal{D} * \mathcal{Q}_{\mathcal{D}} - \widehat{\mathcal{L}_{\mathcal{D}}} * \widehat{\mathcal{U}_{\mathcal{D}}})^{(1:p)})](F_p^* \otimes I_n)\right| \\
&\leq \left|(F_p \otimes I_m)[diag((\mathcal{P}_{\mathcal{D}} * \widehat{\mathcal{D}} * \mathcal{Q}_{\mathcal{D}} - \widehat{\mathcal{L}_{\mathcal{D}}} * \widehat{\mathcal{U}_{\mathcal{D}}})^{(1:p)})](F_p^* \otimes I_n)\right| \\
&\quad + \left|(F_p \otimes I_m)[diag((\mathcal{P}_{\mathcal{D}} * (\mathcal{D} - \widehat{\mathcal{D}}) * \mathcal{Q}_{\mathcal{D}})^{(1:p)})](F_p^* \otimes I_n)\right|.
\end{aligned}$$

Both terms in the right-hand side are well bounded under the Frobenius norm and the infinity norm as follows

$$\begin{aligned}
&\left\|(F_p \otimes I_m)\left[diag\left((\mathcal{P}_{\mathcal{D}} * \widehat{\mathcal{D}} * \mathcal{Q}_{\mathcal{D}} - \widehat{\mathcal{L}_{\mathcal{D}}} * \widehat{\mathcal{U}_{\mathcal{D}}})^{(1:p)}\right)\right](F_p^* \otimes I_n)\right\|_F \\
\leq\ & \left\|F_p \otimes I_m\right\|_F \left\|F_p \otimes I_n\right\|_F \left(\sum_{k=1}^{p} \left\|(\mathcal{P}_{\mathcal{D}} * \widehat{\mathcal{D}} * \mathcal{Q}_{\mathcal{D}} - \widehat{\mathcal{L}_{\mathcal{D}}} * \widehat{\mathcal{U}_{\mathcal{D}}})^{(k)}\right\|_F^2\right)^{\frac{1}{2}} \\
\leq\ & m^{\frac{1}{2}} n^{\frac{1}{2}} p(2q\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\left(\sum_{k=1}^{p} \left\|\left|\widehat{\mathcal{L}_{\mathcal{D}}{}^{(k)}}\right|\left|\widehat{\mathcal{U}_{\mathcal{D}}{}^{(k)}}\right|\right\|_F^2\right)^{\frac{1}{2}} \\
\leq\ & (2m^{\frac{3}{2}} n^{\frac{1}{2}} pq^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\left\|\widehat{\mathcal{U}_{\mathcal{D}}}\right\|_F,
\end{aligned}$$

and

$$\begin{aligned}
&\left\|(F_p \otimes I_m)[diag((\mathcal{P}_{\mathcal{D}} * \widehat{\mathcal{D}} * \mathcal{Q}_{\mathcal{D}} - \widehat{\mathcal{L}_{\mathcal{D}}} * \widehat{\mathcal{U}_{\mathcal{D}}})^{(1:p)})](F_p^* \otimes I_n)\right\|_\infty \\
\leq\ & \left\|F_p \otimes I_m\right\|_\infty \left\|F_p \otimes I_n\right\|_\infty \left\|diag((\mathcal{P}_{\mathcal{D}} * \widehat{\mathcal{D}} * \mathcal{Q}_{\mathcal{D}} - \widehat{\mathcal{L}_{\mathcal{D}}} * \widehat{\mathcal{U}_{\mathcal{D}}})^{(1:p)})\right\|_\infty \\
\leq\ & (2pq^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\left\|\widehat{\mathcal{U}_{\mathcal{D}}}\right\|_\infty,
\end{aligned}$$

while

$$\begin{aligned}
&\left\|(F_p \otimes I_m)diag((\mathcal{P}_{\mathcal{D}}(\widehat{\mathcal{D}} - \mathcal{D})\mathcal{Q}_{\mathcal{D}})^{(1:p)})(F_p^* \otimes I_n)\right\|_F \\
\leq\ & m^{\frac{1}{2}} n^{\frac{1}{2}} p\left\|diag((\widehat{D} - D)^{(1:p)})\right\|_F \\
\leq\ & (4m^{\frac{1}{2}} n^{\frac{1}{2}} p^2\mathrm{u} + \mathcal{O}(\mathrm{u})^2)\|\mathcal{D}\|_F,
\end{aligned}$$

and

$$\begin{aligned} & \left\|(F_p \otimes I_m) diag((\mathcal{P}_{\mathcal{D}}(\widehat{\mathcal{D}} - \mathcal{D})\mathcal{Q}_{\mathcal{D}})^{(1:p)})(F_p^* \otimes I_n)\right\|_\infty \\ \leq \; & p\left\| diag((\widehat{D} - D)^{(1:p)})\right\|_\infty \\ \leq \; & (4p^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\|\mathcal{D}\|_\infty. \end{aligned}$$

As a result, we get estimations

$$\left\|\mathcal{E}_{LU}\right\|_F \leq (2m^{\frac{3}{2}} n^{\frac{1}{2}} pq^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\left\|\widehat{\mathcal{U}_{\mathcal{D}}}\right\|_F + (4m^{\frac{1}{2}} n^{\frac{1}{2}} p^2\mathrm{u} + \mathcal{O}(\mathrm{u})^2)\|\mathcal{D}\|_F, \tag{7}$$

and

$$\left\|\mathcal{E}_{LU}\right\|_\infty \leq (2pq^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\left\|\widehat{\mathcal{U}_{\mathcal{D}}}\right\|_\infty + (4p^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\|\mathcal{D}\|_\infty.$$

We have a quite similar analysis for the t-QR, as we take the Frobenius norm as the example: for $\mathcal{E}_{QR} = |\mathcal{A} - \widehat{\mathcal{Q}_{\mathcal{A}}} * \widehat{\mathcal{R}_{\mathcal{A}}}|$, we make an equivalent transform.

$$\begin{aligned} bcirc(\mathcal{E}_{QR}) &= |bcirc(\mathcal{A}) - bcirc(\widehat{\mathcal{Q}_{\mathcal{A}}})bcirc(\widehat{\mathcal{R}_{\mathcal{A}}})| \\ &= \left|(F_p \otimes I_m)[diag((\mathcal{D} - \widehat{\mathcal{Q}_{\mathcal{D}}} * \widehat{\mathcal{R}_{\mathcal{D}}})^{(1:p)})](F_p^* \otimes I_n)\right| \\ &\leq \left|(F_p \otimes I_m)[diag((\widehat{\mathcal{D}} - \widehat{\mathcal{Q}_{\mathcal{D}}} * \widehat{\mathcal{R}_{\mathcal{D}}})^{(1:p)})](F_p^* \otimes I_n)\right| \\ &\quad + \left|(F_p \otimes I_m)[diag((\mathcal{D} - \widehat{\mathcal{D}})^{(1:p)}))](F_p^* \otimes I_n)\right|. \end{aligned}$$

Both terms in the right-hand side are well bounded under the Frobenius norm. For the first term, we have

$$\begin{aligned} & \left\|(F_p \otimes I_m)[diag((\widehat{\mathcal{D}} - \widehat{\mathcal{Q}_{\mathcal{D}}} * \widehat{\mathcal{R}_{\mathcal{D}}})^{(1:p)})](F_p^* \otimes I_n)\right\|_F \\ \leq \; & \left\|F_p \otimes I_m\right\|_F \left\|F_p \otimes I_n\right\|_F \Big(\sum_{k=1}^{p} \left\|(\widehat{\mathcal{D}} - \widehat{\mathcal{Q}_{\mathcal{D}}} * \widehat{\mathcal{R}_{\mathcal{D}}})^{(k)}\right\|_F^2\Big)^{\frac{1}{2}} \\ = \; & m^{\frac{1}{2}} n^{\frac{1}{2}} p(4n^2\mathrm{u}) \sum_{k=1}^{p} \left\|\widehat{\mathcal{D}}^{(k)}\right\|_F^2 \\ = \; & 4m^{\frac{1}{2}} n^{\frac{5}{2}} \mathrm{u}\left\|\widehat{\mathcal{D}}\right\|_F, \end{aligned}$$

for the second term, we have

$$\begin{aligned} & \left\|(F_p \otimes I_m)[diag((\mathcal{D} - \widehat{\mathcal{D}})^{(1:p)}))](F_p^* \otimes I_n)\right\|_F \\ \leq \; & \left\|F_p \otimes I_m\right\|_F \left\|F_p \otimes I_n\right\|_F \Big(\sum_{k=1}^{p} \left\|((\mathcal{D} - \widehat{\mathcal{D}}))^{(k)}\right\|_F^2\Big)^{\frac{1}{2}} \\ \leq \; & (4m^{\frac{1}{2}} n^{\frac{1}{2}} p^2\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\|\widehat{\mathcal{D}}^{(k)}\|_F. \end{aligned}$$

As a result, we have an estimation

$$\|\mathcal{E}_{QR}\|_F \leq (4m^{\frac{1}{2}} n^{\frac{1}{2}}(n^2 + p^2)\mathrm{u} + \mathcal{O}(\mathrm{u}^2))\|\widehat{\mathcal{D}}^{(k)}\|_F. \tag{8}$$

In this paper we do not offer the t-RRLU and t-RRQR round-off error for two reasons. First, pivoting is necessary in rank-revealing algorithms so that small entries are moved to lower-right block. Small change in the original matrix or tensor may lead to different permutations in (1). We need the assumption that pivoting is numerically stable, in classical LU or QR algorithms, though it is not theoretically supported. Fortunately, experiments imply similar error bounds. Secondly, revealing the numerical rank of input matrix or tensor is a sensitive problem itself. In practice we need to involve a random matrix in operations to accelerate rank computation, which finally leaves in output and beyond round-off error analysis.

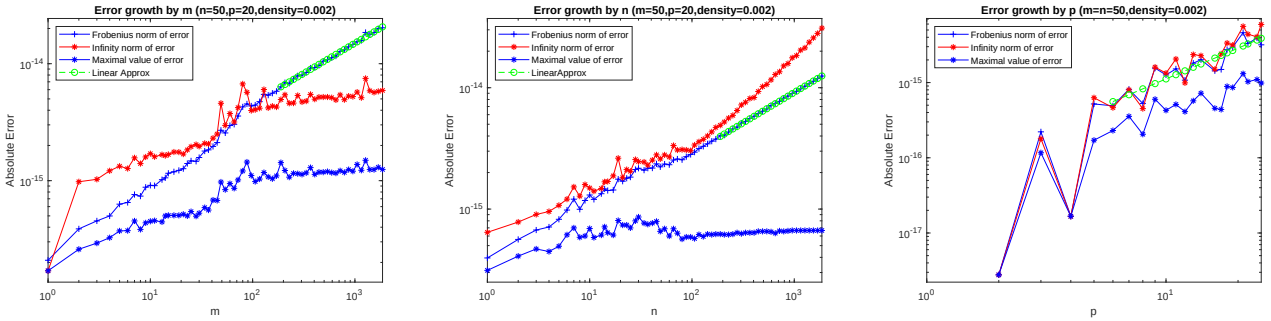
In this paper we do not offer the t-RLU and t-RQR round-off errors, either. The main reason is quite large error involved in random sampling in the very beginning. For instance, when we take recommended parameters in [35], we are actually using an $n \times k = \mathcal{O}(\log_2^2(n))$ Gaussian matrix to sample an $n \times n$ matrix. Since $\log_2^2(n) \ll n$ for large n, such operation cannot avoid a much more loss than round-off error. Figure 5.1.12 in [35] also shows that the overall error could be as much as 1e-4 even for a relatively large k.

4. Numerical examples

In this section, we first apply the t-LU and t-QR on randomly generalized sparse test tensors and record errors and time costs. Then we apply the t-RRLU and t-RLU on randomly generalized dense test tensors, and also record errors and time costs. Finally we use an example of image reconstruction to show the application of our algorithms. We use the laptop with Intel Core (TM) i5-8300H CPU and MATLAB R2018b.

4.1. Error behavior

4.1.1. T-LU We examine the t-LU for random sparse tensors $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$, where absolute value of each component $A^{(k)}(i,j)$ is no more than 1. Growth behavior of absolute error by the size of $\mathcal{A}$ is shown in Figure 1 in the Frobenius norm, infinity norm and l_∞ norm (also called maximal value norm).



(a) (b) (c)

Figure 1. T-LU error growth.

Figure 1(a) shows the absolute error growth when m varies among 64 integers selected nearly logarithmic equidistant from 1 to 1862, while $n = 50$ and $p = 20$ are fixed, with $density = 0.002$ that decides $nnz(\mathcal{A}) \approx density \times mnp$. For each m, we produce 20 random sparse tensors $\mathcal{A}$ to test the decomposition, record their error in three types of norm, and take average respectively as the error corresponding to m. The process of experiments in rest of this paper would be same unless extra notes are given. The figure implies that the error grows by m at the same, slow rate under infinity and l_∞ norm. Meanwhile, its Frobenius norm grows faster. Least squares linear approximation by data in the Frobenius norm shows a growth order of 0.5162.

Figure 1(b) shows the error growth when n varies among 64 integers selected nearly logarithmic equidistant from 1 to 1862, while $m = 50, p = 20$ and $density = 0.002$ are fixed. This figure implies that the error has a polynomial growth by n under both Frobenius norm and infinity norm. LS approximation shows a growth order of 0.5019. Meanwhile, its l_∞ norm keeps almost a constant of machine accuracy.

Figure 1(c) shows the error growth as p varies among 21 integers selected nearly logarithmic equidistant from 2 to 25, while $m = 50, n = 50$ and $density = 0.002$ are fixed. LS approximation shows a growth order of 1.3576, between 1 and 2. This is because of the relatively large coefficient of first order term in (7) for a not very large p. The existence of remarkable error drop when $p = 4, 8, \cdots$ is due to the property of block diagonalization by DFT. When its p is divided by 4, DFT becomes the fast Fourier transform, for which special algorithms are developed with less flops and then less error.

4.1.2. T-QR Test tensors are the same as t-LU. Growth of absolute error is shown in Figure 2 in the Frobenius norm, infinity norm and l_∞ norm (also called maximal value norm). Figure 2 shows the error growth behavior by the size of $\mathcal{A}$ when 3 variants among $m = 50$, $n = 50$, $p = 20$ and $density = 0.002$ are fixed.

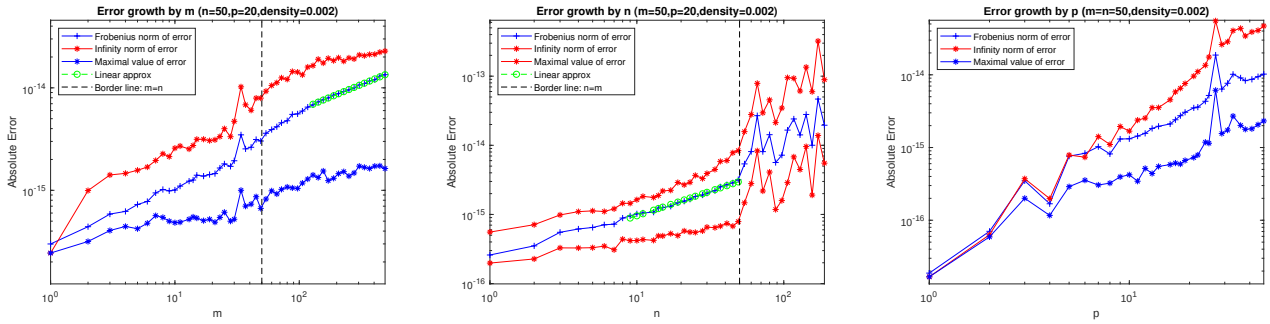
Figure 2(a) shows the case where m grows from 1 to 490. It implies that error has a polynomial growth in the Frobenius norm when m is sufficiently large ($m \geq n$). LS approximation shows a growth order of 0.4970, which satisfies (8).

Figure 2(b) shows the case where n grows from 1 to 189. It implies that error is well-bounded only when $n \leq m$. In this interval, LS approximation shows a growth order of 0.7103. Recall that in (8) we have shown the main term is $\sqrt{n}(n^2 + p^2)$, while the limitation $n \leq m$ puts n^2 and p^2 at the same level. In other similar cases where growth order estimation cannot be easily done, we make no more explanation.

Figure 2(c) shows the case where p grows from 2 to 50. Drops when $p = 4, 8, \ldots$ is due to the FFT.

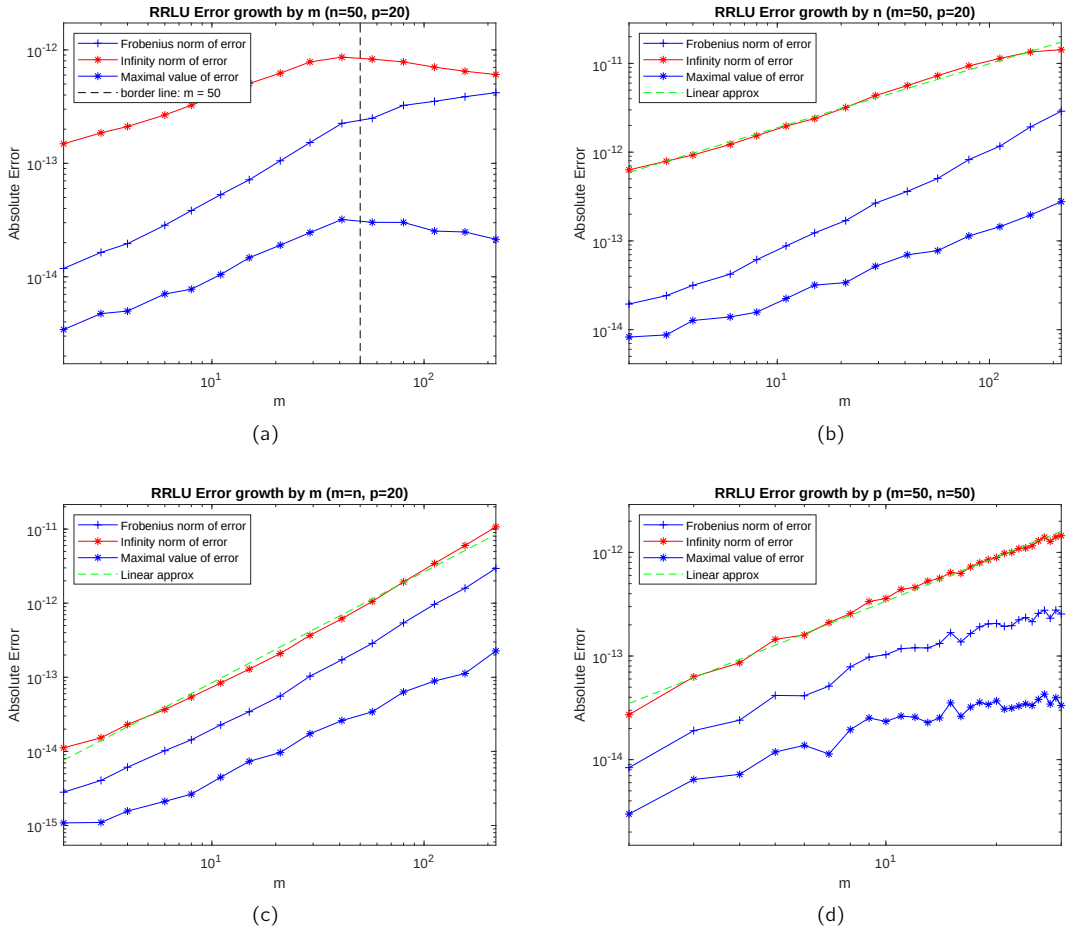
4.1.3. T-RRLU We examine the t-RRLU for random dense tensors $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$, where absolute value of each component $A^{(k)}(i,j)$ is no more than 1. Figure 3 shows error growth by dimensions of $\mathcal{A}$.

Figure 3(a) shows the absolute error growth when m varies among 32 integers selected nearly logarithmic equidistant from 2 to 250, while $n = 50$ and $p = 20$ are fixed. The figure implies that error grows by m when $m \leq n$ and stop increasing when $m \geq n$. Error in Frobenius norm seems to keep increasing due to the increasing size of $\mathcal{A}$.



(a) (b) (c)

Figure 2. T-QR error growth.



(a) (b) (c) (d)

Figure 3. T-RRLU error growth

Figure 3(b) shows the absolute error growth when n varies among 15 integers selected nearly logarithm equidistantly from 2 to 250, while $m = 50$ and $p = 20$ are fixed. The figure shows a polynomial error growth in $l_\infty-$ norm. LS approximation shows a growth order of 0.72.

Figure 3(c) shows a special case where $m = n$ with $p = 20$ fixed. Let m vary among 15 integers selected nearly logarithm equidistantly from 2 to 250. The figure shows a polynomial error growth in $l_\infty-$ norm. LS approximation shows a growth order of 1.5.

Figure 3(d) shows the absolute error growth by p from 2 to 30 when $m = 50$ and $n = 50$ are fixed. The figure shows a polynomial error growth in $l_\infty-$ norm. LS approximation shows a growth order of 1.4.

4.1.4. T-RLU Test tensors are the same as the t-RRLU. Figure 4 shows the error growth by $m = n$ when $p = 20$ is fixed. It implies that the error is bounded by constant in $l_\infty-$ norm. LS for data in the Frobenius norm shows a growth order of 0.3353.

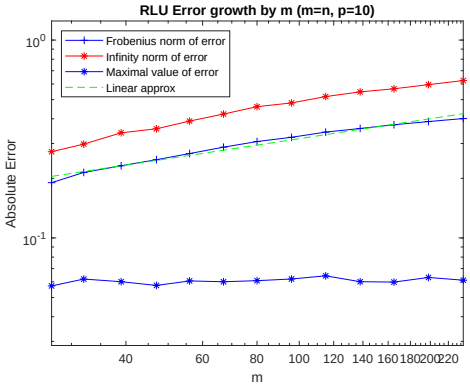
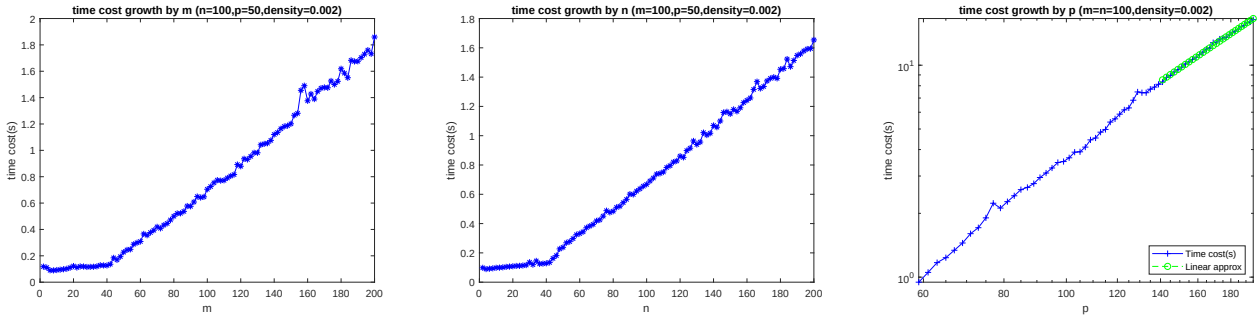


Figure 4. T-RLU error growth.

It deserves mentioning that the error growth starts high (at level 1e-1). This was demonstrated and explained in [35] through its Figure 4, where error was about 1e0 when expected rank $k = \mathcal{O}(1)$ and 1e-3 when $k \approx 600$. It is also mentioned in this paper that randomized algorithms pursue speed rather than accuracy and give the parameter recommendation $l = \log_2^2(n)$ and $k = l - 5$.

4.2. Computational time behavior

4.2.1. T-LU We record time cost of t-LU applied to third-order sparse random tensor $\mathcal{A} \in \mathbb{R}^{m\times n\times p}$ and show it in Figure 5.



(a) (b) (c)

Figure 5. T-LU time cost growth.

4.2.2. T-QR We do the same through the t-QR and show the result in Figure 6.

4.2.3. T-RRLU We examine the t-RRLU for random dense tensors $\mathcal{A} \in \mathbb{C}^{m\times n\times p}$, where absolute value of each component $A^{(k)}(i, j)$ is no more than 1. Figure 7 shows error growth by dimensions of $\mathcal{A}$.

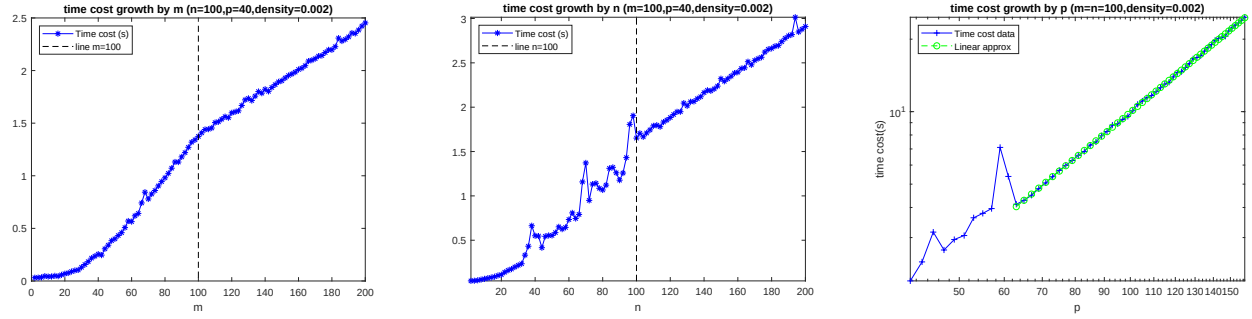
All these figures show polynomial growth of time cost of the t-RRLU.

4.2.4. T-RLU Test tensors are the same as t-RRLU.

Figure 8 shows the time cost growth by $m = n$ from 27 to 238 when $p = 10$ is fixed. LS approximation shows a growth order of 2.1589. For a relatively large test tensor ($m = n \approx 200$), it costs less than 0.1 seconds. This reminds us the high performance on time again.

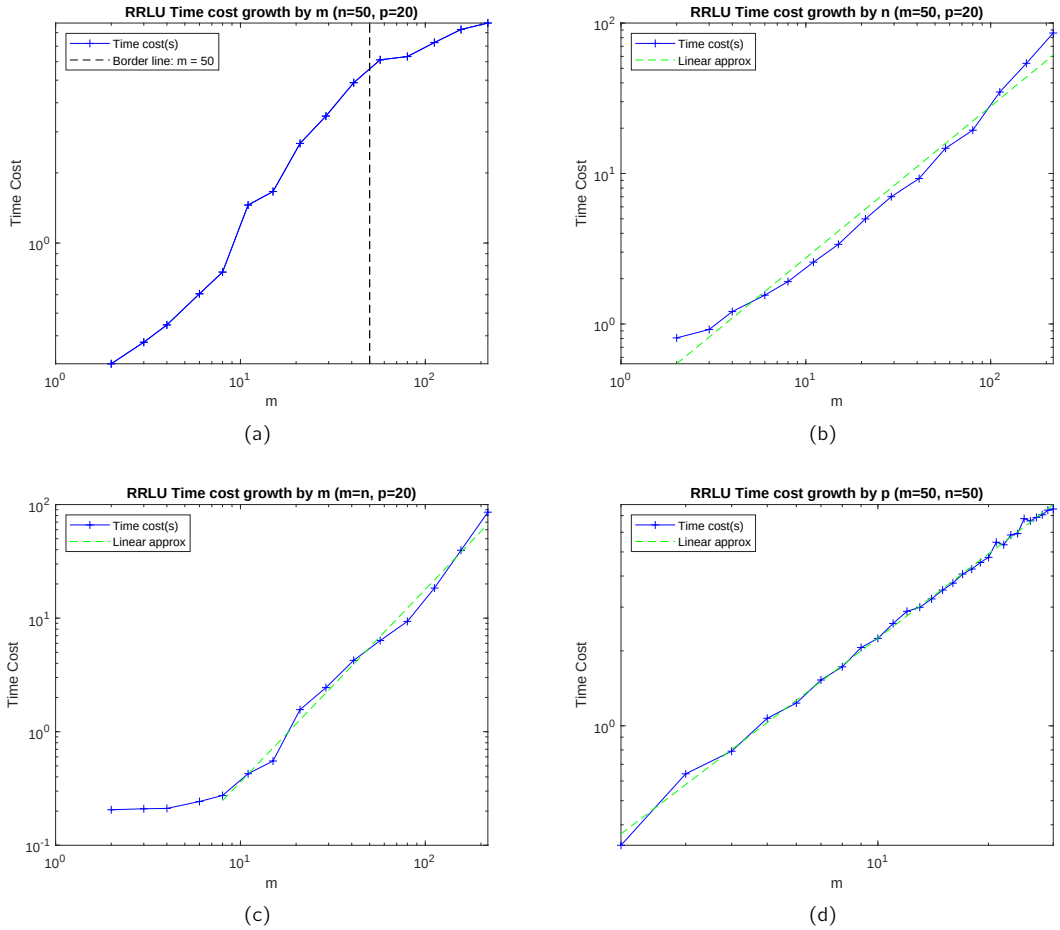
4.3. Applications

We develop previous algorithms with the purpose of having practical application in the low-rank approximation. A common example is the image processing. A color image of size $m \times n$ can be viewed as a third-order tensor $\mathcal{A} \in [0, 1]^{m\times n\times 3} \subseteq \mathbb{R}^{m\times n\times 3}$, whose three frontal slices represent red, green and blue respectively. Then we can do the low-rank tensor decomposition of $\mathcal{A}$ to



(a) (b) (c)

Figure 6. T-QR time cost growth.



(a) (b) (c) (d)

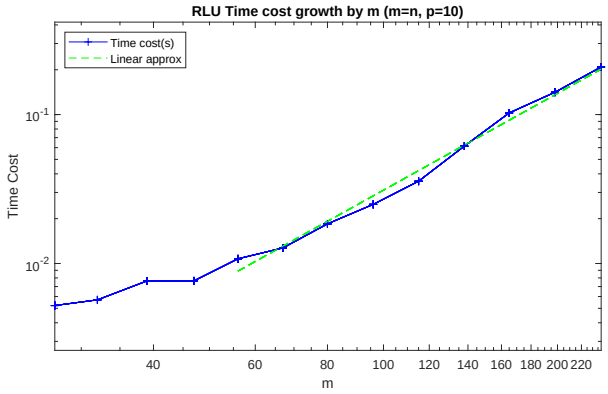
Figure 7. T-RRLU time cost growth.

realize a low memory storage. We call this problem image reconstruction. We use a human face photograph as the test image and apply the t-RRLU and t-RLU, as shown in Figure 9.

Figure 9(a) is the original image of size 512×512.

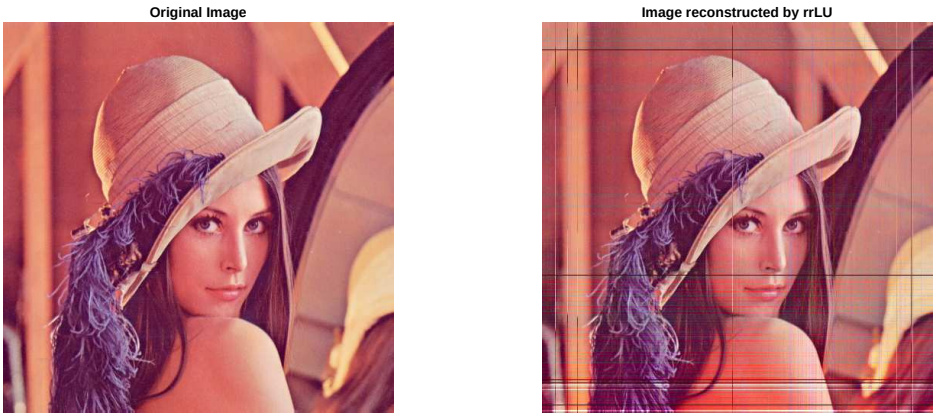
Figure 9(b) is the image reconstructed by t-RRLU. Computational costs 8.0725 seconds with numerical rank $\hat{r} = 464 \approx m$. Thus the storage reduced from $mnp = 512 \times 512 \times 3 = 7.86432e5$ to $(m + n - \hat{r})\hat{r}p + (m + n)p = 7.82592e5$, less than 1%. Meanwhile, vertical and horizontal stripes can be easily observed, which is an unacceptable defect. An occurrence of such stripes is due to lack of orthogonalization steps in RRLU. In comparison, all QR algorithms and randomized algorithms involve at least one orthogonalization step.

Figure 9(c) is the image reconstructed by t-RLU with lower rank k. Computations cost 0.1722 seconds with numerical rank

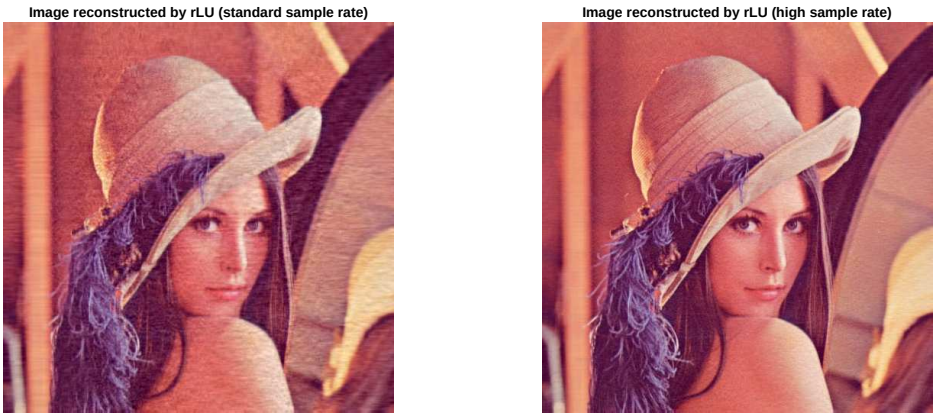


(a)

Figure 8. T-RLU time cost growth.



(a) (b)



(c) (d)

Figure 9. Image reconstruction of human face.

$k = 76$. Thus the storage reduced from $mnp = 512 \times 512 \times 3 = 7.86432e5$ to $2kmp + 2mp = 2.36544e5$ by about 69%. There is obvious loss in image quality, due to having a low rank.

Figure 9(d) is the image reconstructed by t-RLU with higher rank k. Computations cost 0.2687 seconds with numerical rank

$k = 157$. Thus the storage reduced from $mnp = 512 \times 512 \times 3 = 7.86432e5$ to $2kmp + 2mp = 4.85376e5$ by about 38%. No obvious loss is observed.

The same experiment is also taken on a fruit image.

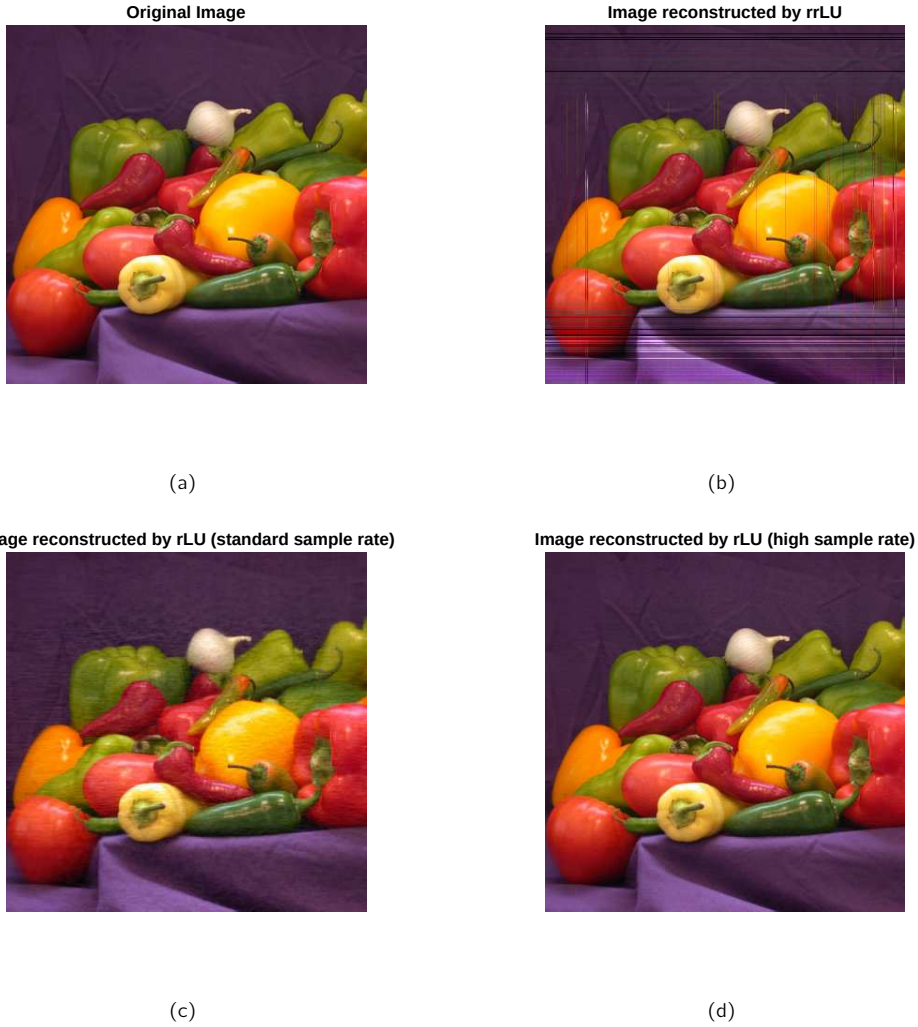


Figure 10. Image reconstruction of fruit.

Figure 10(a) is original image of size 384×384.

Figure 10(b) is the image reconstructed by t-RRLU. Computations cost 2.0912 seconds with numerical rank $\hat{r} = 332 \approx m$. Thus the storage reduced from $mnp = 384 \times 384 \times 3 = 5.89824e5$ to $(m + n - \hat{r})\hat{r}p + (m + n)p = 5.64432e5$ by about 4%. Vertical and horizontal stripes can also be easily observed.

Figure 10(c) is the image reconstructed by t-RLU with lower rank k. Computations cost 0.0897 seconds with numerical rank $k = 69$. Thus the storage reduced from $mnp = 384 \times 384 \times 3 = 5.89824e5$ to $2kmp + 2mp = 1.6128e5$ by about 73%. There is obvious loss in image quality, due to quite low rank.

Figure 10(d) is the image reconstructed by t-RLU with higher rank k. Computations cost 0.1625 seconds with numerical rank $k = 143$. Thus the storage reduced from $mnp = 384 \times 384 \times 3 = 5.89824e5$ to $2kmp + 2mp = 3.31776e5$ by about 44%. No obvious loss is observed.

These two experiments lead to conclusion that t-RRLU is not a competitive solution.

Finally we compare all method we develop for this problem. We perform the tensor randomized singular value decomposition (t-RSVD) in [47] as benchmark.

The first column is original image of a baboon and its reconstruction by t-RRLU. The second through fifth columns are respectively images reconstructed by t-RRQR, t-RSVD, t-RLU and t-RQR. The first row involves quite low rank $k = \log_2^2(n) - 5$ while the second row involves $k = 2\log_2^2(n) - 5$ for higher image quality.

These figures show that methods t-RRQR, t-RSVD, t-RLU and t-RQR are comparable. We show key performance indices of high quality reconstructions in Table 1. Methods with best performance are emphasized in red.

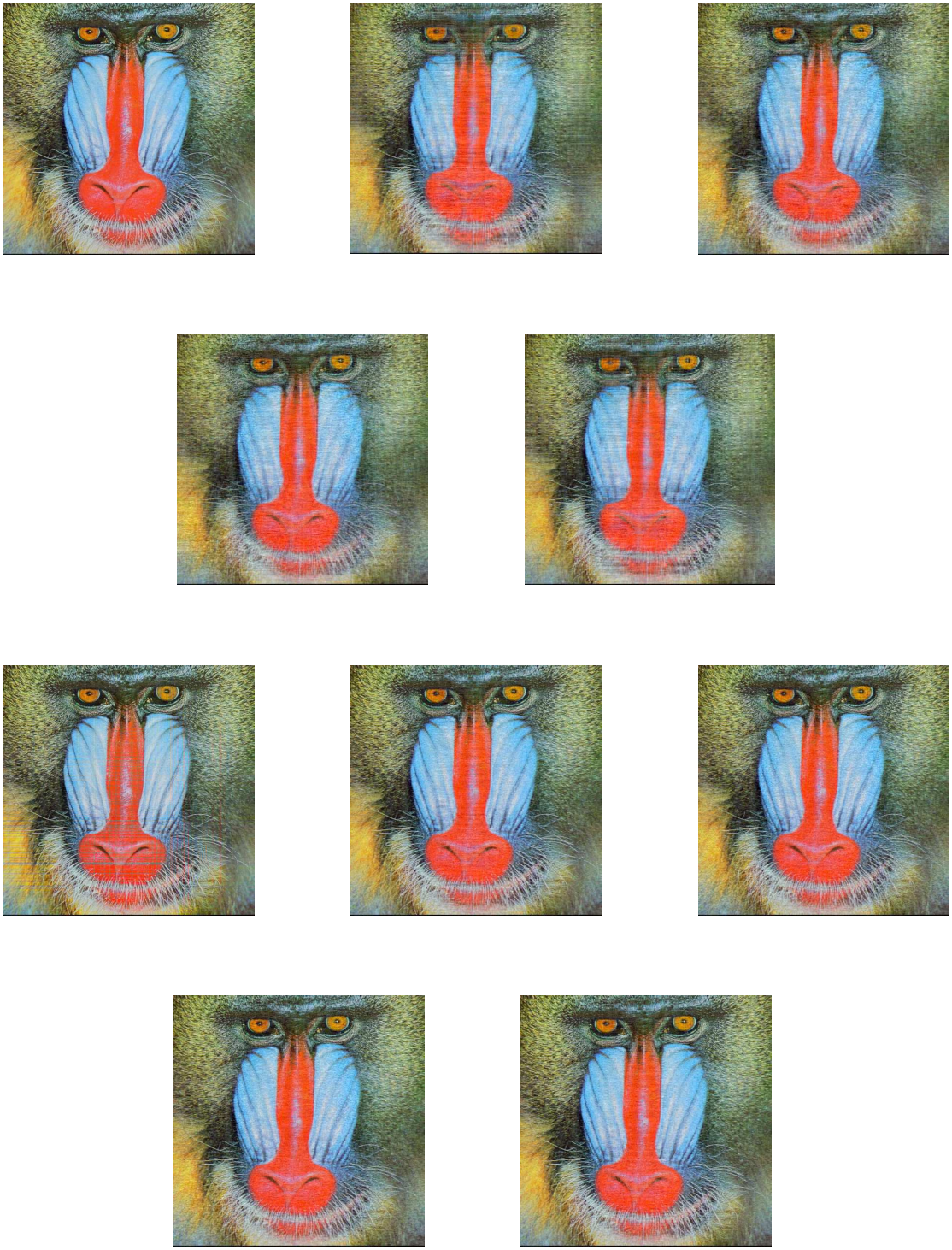


Figure 11. Image reconstruction by different methods.

Method	t-RRQR	t-RSVD	t-RLU	t-RQR
Time cost(s)	0.2302	0.1805	0.1462	0.1713
Error(F-norm)	43.45	47.66	48.62	46.55
Storage save rate	43.28%	38.61%	38.28%	43.28%

Table 1. Performance of different methods for high quality reconstructions.

This table shows that in comparison with our benchmark (t-RSVD), t-RLU is more preferable in time cost, t-RRQR has is more preferable in accuracy and memory storage, while t-RQR is more preferable in all three indices.

5. Conclusion and future work

In this paper, we discussed the advantages and limitations of the tensor rank-revealing and randomized algorithms [9, 11, 12]. The randomized algorithms we designed, extend the dimensionality reduction advantage from matrices to tensors. We provided the growth order estimation of both error and the time cost by the dimension of the tensor with numerical examples. We also performed numerical experiments on two commonly used publicly available data sets. Results show our algorithms have advantages over the benchmark, in a part of indices or all indices.

For future research, there are several potential research directions. In this paper, we extended the most basic matrix decompositions and their improved variants to tensors. Parallel implementation is not discussed, though parallel algorithms for LU and QR are well-developed. One possible direction is to extend other practical tensor algorithms including parallelization. In this paper, our randomization choice for the dimensionality reduction is the Gaussian random tensors. A potential improvement is to investigate other structured random tensors. Our current methods can only process data completely gained in the very beginning and fail on streaming data. More practical application would be available if the up-dating and down-dating tensor algorithms for the tensor equations [15, 39, 41] are developed.

Acknowledgements

We would like to thank the handling editor and the reviewer for their very detailed comments.

References

1. R. Behera, J. Sahoo, R. Mohapatra, and M. Nashed. Computation of generalized inverses of tensors via t-product. *Numer. Linear Algebra Appl.*, 29:e2416, 2022.
2. F. Beik, A. Ichi, K. Jbilou, and R. Sadaka. Tensor extrapolation methods with applications. *Numer Algorithm*, 87:1421–1444, 2021.
3. P. Businger and G. H. Golub. Linear least square solutions by householder transformations. *Numer. Math.*, 7:269–276, 1965.
4. T. F. Chan. Rank revealing QR factorizations. *Linear Algebra Appl.*, 88-89:67–82, 1987.
5. S. Chang and Y. Wei. General tail bounds for random tensors summation: majorization approach. *arXiv: 2105.06078*, 2021.
6. S. Chang and Y. Wei. Generalized T-product tensor bernstein bounds. *arXiv: 2109.10880*, 2021.
7. S. Chang and Y. Wei. T-product tensors part I: inequalities. *arXiv: 2107.06285v2*, 2021.
8. S. Chang and Y. Wei. T-product tensors part II: tail bounds for sums of random t product tensors. *arXiv: 2107.06224v2*, 2021.
9. M. Che and Y. Wei. Randomized algorithms for the approximations of Tucker and the tensor train decompositions. *Adv. Comput. Math.*, 45:395–428, 2019.
10. M. Che and Y. Wei. *Theory and Computation of Complex Tensors and its Applications*. Springer, Singapore, 2020.
11. M. Che, Y. Wei, and H. Yan. The computation of low multilinear rank approximations of tensors via power scheme and random projection. *SIAM J. Matrix Anal. Appl.*, 41:605–636, 2020.
12. M. Che, Y. Wei, and H. Yan. Randomized algorithms for the low multilinear rank approximations of tensors. *J. Comput. Appl. Math.*, 390:e113380, 2021.
13. Y. Cui and H. Ma. The perturbation bound for the T-Drazin inverse of tensor and its application. *Filomat*, 35:1565–1587, 2021.
14. J. Demmel, L. Grigori, M. Gu, and H. Xiang. Communication avoiding rank revealing QR factorization with column pivoting. *SIAM J. Matrix Anal. Appl.*, 36:55–89, 2015.
15. W. Ding and Y. Wei. Solving multi-linear systems with M-tensors. *J. Sci. Comput.*, 68:689–715, 2016.
16. J. A. Duersch and M. Gu. Randomized QR with column pivoting. *SIAM J. Sci. Comput.*, 39(4):263–291, 2017.
17. G. Golub and C. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 2012.
18. N. Hao, M. Kilmer, K. Braman, and R. Hoover. Facial recognition using tensor-tensor decompositions. *SIAM J. Imaging Sci.*, 6:437–463, 2013.
19. N. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, 1996.
20. T.-M. Hwang, W.-W. Lin, and E. K. Yang. Rank revealing LU factorization. *Linear Algebra Appl.*, 175:115–141, 1992.
21. A. E. Ichi, K. Jbilou, and R. Sadaka. On tensor tubal-Krylov subspace methods. *Linear Multilinear Algebra*, page https://doi.org/10.1080/03081087.2021.1999381., 2021.
22. X. Jin, Y. Wei, and Z. Zhao. *Numerical Linear Algebra and Its Applications, 2nd edition*. Science Press, Beijing, 2015.
23. 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.*, 34:148–172, 2013.
24. M. Kilmer and C. Martin. Factorization strategies for third-order tensors. *Linear Algebra Appl.*, 435:641–658, 2011.
25. H. Li and Y. Wei. Improved rigorous perturbation bounds for the LU and QR factorizations. *Numer. Linear Algebra Appl.*, 22:1115–1130, 2015.
26. H. Li and S. Yin. Single-pass randomized algorithms for LU decomposition. *Linear Algebra Appl*, 595:101–122, 2020.
27. E. Liberty, F. Woolfe, P.-G. Martinsson, V. Rokhlin, and M. Tygert. Randomized algorithms for the low-rank approximation of matrices. *Proceedings of the National Academy of Sciences*, 104:20167–20172, 2007.

28. C. Ling, J. Liu, C. Ouyang, and L. Qi. ST-SVD factorization and S-diagonal tensors. *arXiv: 2104.05329*, 2021.
29. C. Martin, R. Shafer, and B. Larue. An order-p tensor factorization with applications in imaging. *SIAM J. Sci. Comput.*, 35(1):A474–A490, 2013.
30. Y. Miao, L. Qi, and Y. Wei. Generalized tensor function via the tensor singular value decomposition based on the T-product. *Linear Algebra Appl.*, 590:258–303, 2020.
31. Y. Miao, L. Qi, and Y. Wei. T-Jordan canonical form and T-Drazin inverse based on the T-product. *Commun. Appl. Math. Comput.*, 3:201–220, 2021.
32. L. Qi, C. Ling, J. Liu, and C. Ouyang. An orthogonal equivalence theorem for third order tensors. *arXiv:2103.17259*, 2021.
33. L. Reichel and U. Ugwu. Tensor Arnoldi-Tikhonov and GMRES-type methods for ill-posed problems with a t-product structure. *J Sci. Comput.*, 90:Article 59, 2022.
34. L. Reichel and U. Ugwu. The tensor Golub-Kahan-Tikhonov method applied to the solution of ill-posed problems with a t-product structure. *Numer. Linear Algebra Appl.*, 29:e2412, 2022.
35. G. Shabat, Y. Shmueli, Y. Aizenbud, and A. Averbuch. Randomized LU decomposition. *Appl. Comput. Harmon. Anal.*, 44:246–272, 2018.
36. G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Boston, Academic Press, 1990.
37. D. Tarzanagh and G. Michailidis. Fast randomized algorithms for t-product based tensor operations and decompositions with applications to imaging data. *SIAM J. Imaging Sci.*, 11:2629–2664, 2018.
38. W. Wang and Y. Wei. Mixed and componentwise condition numbers for matrix decompositions. *Theoret. Comput. Sci.*, 681:199–216, 2017.
39. X. Wang, M. Che, and Y. Wei. Neural network approach for solving nonsingular multi-linear tensor systems. *J. Comput. Appl. Math.*, 368:e112569, 2020.
40. X. Wang, M. Che, and Y. Wei. Tensor neural network models for tensor singular value decompositions. *Computat. Optim. Appl.*, 75:753–777, 2020.
41. X. Wang, C. Mo, S. Qiao, and Y. Wei. Predefined-time convergent neural networks for solving the time-varying nonsingular multi-linear tensor equations. *Neurocomputing*, 472:68–84, 2022.
42. Y. Wei and W. Ding. *Theory and Computation of Tensors: Multi-Dimensional Arrays*. Elsevier/Academic Press, London, 2016.
43. J. H. Wilkinson. *Rounding Errors in Algebraic Processes*. Her Majestys Stationery Office, London, 1963.
44. N. Wu and H. Xiang. Randomized QLP decomposition. *Linear Algebra Appl.*, 599:18–35, 2020.
45. H. Xiang and Y. Wei. Structured mixed and componentwise condition numbers of some structured matrices. *J. Comput. Appl. Math.*, 202:217–229, 2007.
46. M. Yin, J. Gao, S. Xie, and Y. Guo. Multiview subspace clustering via tensorial t-product representation. *IEEE Trans. Neural Netw. Learn. Syst.*, 30(3):851–864, 2019.
47. J. Zhang, A. K. Saibaba, M. E. Kilmer, and S. Aeron. A randomized tensor singular value decomposition based on the t-product. *Numer Linear Algebra Appl.*, 25:e2179, 2018.
48. Z. Zhang and S. Aeron. Exact tensor completion using t-SVD. *IEEE Trans. Signal Process*, 65:1511–1526, 2017.

