Sketch-and-project methods for tensor linear systems

Ling Tang | Yajie Yu | Yanjun Zhang | Hanyu Li

College of Mathematics and Statistics, Chongqing University, Chongqing, China

Correspondence
Hanyu Li, College of Mathematics and Statistics, Chongqing University, Chongqing 401331, China.
Email: liby.hy@gmail.com and hyl@cqu.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Number: 11671060;
Natural Science Foundation of Chongqing, China, Grant/Award Number: cstc2019jcyj-msxmX0267

Abstract
For tensor linear systems with respect to the popular t-product, we first present the sketch-and-project method and its adaptive variants. Their Fourier domain versions are also investigated. Then, considering that the existing sketching tensor or way for sampling has some limitations, we propose two improved strategies. Convergence analyses for the methods mentioned above are provided. We compare our methods with the existing ones using synthetic and real data. Numerical results show that they have quite decent performance in terms of the number of iterations and running time.

KEYWORDS
adaptive sampling, Fourier domain, sketch-and-project, t-product, tensor linear systems

1 | INTRODUCTION

In this article, we aim to solve the following consistent tensor linear system

\[ \mathbf{A} \ast \mathbf{X} = \mathbf{B}, \]

where \( \mathbf{A} \in \mathbb{R}^{m \times n \times l} \), \( \mathbf{X} \in \mathbb{R}^{n \times p \times l} \) and \( \mathbf{B} \in \mathbb{R}^{m \times p \times l} \) are third-order tensors, and the operator \( \ast \) denotes the t-product introduced by Kilmer and Martin. The problem (1) arises in many situations, including image deblurring problems, tensor dictionary learning, and the use of the boundary element method to analyze the electromagnetic and acoustic properties of spherically symmetric objects, and so forth. Naturally, the problem can be transformed into a matrix linear system and solved by matrix methods. However, as stated in References 2-4, tensor-based representations of the data have a distinct advantage over matrix-based counterparts. For example, for a collection of gray-scale images, one can choose to store each image as a vector, yielding a matrix representation of all the images, or one might also store each image as a slice of a third-order tensor, resulting in a third-order tensor representation of all the images. It can be observed that using tensors to represent these images is more natural and can avoid damaging the multilinear structures in the data. In this case, it is natural to consider the tensor linear systems instead of the matrix linear systems. For t-product in (1), it has an advantage that it can reserve the information inherent in the flattening of a tensor and, with it, many properties of numerical linear algebra can be extended to third and high order tensors. Hence, extensive works on t-product have appeared in recent years and have also been applied in many areas such as image and signal processing, computer vision, data denoising, low-rank tensor completion, and so forth. We will review the basic knowledge on t-product in Section 2.

To solve the problem (1), Ma and Molitor extended the matrix randomized Kaczmarz (MRK) method and called it the tensor randomized Kaczmarz (TRK) method. Later, this method was applied to tensor recovery problems. Recently, Du and Sun extended the matrix randomized extended Kaczmarz method to inconsistent tensor recovery problems. As we know, the MRK method is a popular iterative method for solving large-scale matrix linear systems, that is, the case...
for $l = 1$ and $p = 1$ in the problem (1), and it has wide developments; see for example, References 28-34. Most of these methods can be unified into the sketch-and-project (MSP) method and its adaptive variants proposed by Gower et al.\textsuperscript{35,36}

Inspired by the above research, we propose the tensor sketch-and-project (TSP) method and its adaptive variants to solve the problem (1), followed by their theoretical guarantees. Meanwhile, we also present their Fourier domain versions and analyze the corresponding convergence. So, the TRK method and its theoretical analyses\textsuperscript{23} will be the special cases of our results.

Besides the randomized algorithms in Reference 23,26,27 mentioned above, there are some research based on random sketching technique for t-product; see for example, References 15,37,38. In these works, some sketching tensors including the ones extracted from random sampling are formed. However, they have some limitations. For example, the Gaussian random tensor in Reference 37 and 38 is defined as a tensor whose first frontal slice is created by the standard normal distribution and other frontal slices are all zeros; the random sampling tensor in References 15,23,26,27 is formed similarly, that is, its first frontal slice is a sampling matrix but other frontal slices are all zeros. In this way, the transformed tensor by the discrete Fourier transform (DFT) along the third dimension will have the same frontal slices. On the other hand, a tensor problem based on t-product will be transformed into multiple independent matrix subproblems in the Fourier domain. Thus, the above sketching tensors will lead to the sketching matrices or the way for sampling in every matrix subproblem being the same. Taking the TRK method as an example, if we choose an index with the probabilities corresponding to the horizontal slices of $A$, then every subsystem in the Fourier domain uses the same index to update at each iteration. Since these subsystems are independent, choosing different indices for different subsystems may be better.

In Reference 23, the authors also found the above limitation and mentioned that different indices can be selected for different subsystems. However, for real-valued problems in the real field, using this strategy directly is no longer feasible because the final solution will be complex-valued. To the best of our knowledge, there is no work published to solve this problem in the real field. In this paper, we provide two improved strategies for our TSP method and its adaptive variants. The first one is based on an equivalence transformation, and the other is to take the real part of the last iterate directly. For the former, we present its theoretical guarantees. However, it is a little difficult to implement this method when combined with the adaptive sampling idea. For the latter, it has good performance in numerical experiments. However, we cannot provide its thorough theoretical guarantees at present.

The rest of this paper is organized as follows. Section 2 presents the notation and preliminaries. In Section 3, we propose the TSP method and its adaptive variants. The implementation of the proposed methods in the Fourier domain is discussed in Section 4. In Section 5, we devise two improved strategies for the TSP method and its adaptive variants. The numerical results on synthetic and real data are provided in Section 6. Finally, we give the conclusion of the whole paper, and the appendix containing the detailed proofs of the main lemmas and theorems and two additional algorithms.

## 2 Notation and Preliminaries

Throughout this paper, scalars are denoted by lowercase letters, for example, $x$; vectors are denoted by boldface lowercase letters, for example, $\mathbf{x}$; matrices are denoted by capital letters, for example, $X$; higher-order tensors are denoted by Euler script letters, for example, $\mathcal{X}$.

For any square matrix $X$, we use $\lambda_{\text{max}}(X)$, $\lambda_{\text{min}}(X)$, and $\lambda_{\min}^+(X)$ to denote its largest eigenvalue, smallest eigenvalue, and smallest positive eigenvalue, respectively. Meanwhile, $\text{Rank}(X)$, $\text{Tr}(X)$, and $\text{Range}(X)$ stand for the rank, the trace, and the column space of the matrix $X$. In addition, for any matrices $X$ and $Y$, $X > (\geq) Y$ means that $X - Y$ is a positive (semi)definite matrix.

For any third-order tensor $\mathcal{X}$, its $(i,j,k)$th element is represented by $\mathcal{X}_{(i,j,k)}$; its fiber is a one-dimensional array denoted by fixing two indices, for example, $\mathcal{X}_{(\cdot,j,k)}$, $\mathcal{X}_{(i,\cdot,k)}$ and $\mathcal{X}_{(i,j,\cdot)}$ respectively represent the $(j,k)$th column, $(i,k)$th row and $(i,j)$th tube fiber; its slice is a two-dimensional array defined by fixing one index, for example, $\mathcal{X}_{(i,\cdot,\cdot)}$, $\mathcal{X}_{(\cdot,j,\cdot)}$ and $\mathcal{X}_{(\cdot,\cdot,k)}$ respectively represent the $i$th horizontal, $j$th lateral and $k$th frontal slice. For convenience, the frontal slice $\mathcal{X}_{(\cdots,k)}$ is written as $\mathcal{X}^{(k)}$, and $(\mathcal{X}^{(i,\cdots)})^T$, $(\mathcal{X}^{(\cdot,j,\cdots)})^T$ and $(\mathcal{X}^{(\cdot,\cdot,k)})^T$ are simply denoted as $\mathcal{X}^{T(i,\cdots)}$, $\mathcal{X}^{T(\cdot,j,\cdots)}$ and $\mathcal{X}^{T(\cdot,\cdot,k)}$, respectively.

In the following, we review the definition of t-product. Before that, some preparations including the definitions of tubal matrices and several operations over them are presented.

**Definition 1** (see Reference 2). An element $\mathbf{x} \in \mathbb{R}^{1 \times 1 \times l}$ is called a tubal scalar of length $l$ and the set of all tubal scalars of length $l$ is denoted by $\mathbb{K}_l$; an element $\mathcal{X} \in \mathbb{R}^{m \times n \times l}$ is called a vector of tubal scalars of length $l$ with size $m$ and the corresponding set is denoted by $\mathbb{K}_{l(m \times n)}$; an element $\mathcal{X} \in \mathbb{R}^{m \times n \times l}$ is called a matrix of tubal scalars of length $l$ with size $m \times n$ and the corresponding set is denoted by $\mathbb{K}_{l(m \times n)}$. 
Throughout this paper, we will refer to tubal matrix and third-order tensor interchangeably. For a tubal matrix $\mathcal{X} \in \mathbb{K}_l^{m \times n}$, as done in References 1 and 2, define

$$\text{bcirc}(\mathcal{X}) = \begin{bmatrix} \mathcal{X}(1) & \cdots & \mathcal{X}(l) \\ \mathcal{X}(2) & \cdots & \mathcal{X}(l-1) \\ \vdots & \ddots & \vdots \\ \mathcal{X}(l) & \cdots & \mathcal{X}(1) \end{bmatrix}, \quad \text{unfold}(\mathcal{X}) = \begin{bmatrix} \mathcal{X}(1) \\ \vdots \\ \mathcal{X}(l) \end{bmatrix}, \quad \text{fold}(\text{unfold}(\mathcal{X})) = \mathcal{X}, \text{ and } \text{bcirc}^{-1}(\text{bcirc}(\mathcal{X})) = \mathcal{X}.$$

Now, we give the definition of t-product.

**Definition 2** (t-product\(^1\)). Let $\mathcal{X} \in \mathbb{K}_l^{m \times n}$ and $\mathcal{Y} \in \mathbb{K}_l^{m \times p}$. Then the t-product $\mathcal{X} \ast \mathcal{Y} \in \mathbb{K}_l^{m \times p}$ is defined by

$$\mathcal{X} \ast \mathcal{Y} = \text{fold}(\text{bcirc}(\mathcal{X})\text{unfold}(\mathcal{Y})).$$

Note that the matrix bcirc($\mathcal{X}$) can be block diagonalized by the DFT matrix combined with the Kronecker product. Specifically, for a tubal matrix $\mathcal{X} \in \mathbb{K}_l^{m \times n}$ and the unitary DFT matrix $F_l \in \mathbb{C}^{bd}$,

$$(F_l \otimes I_m)\text{bcirc}(\mathcal{X})(F_l^H \otimes I_n) = \begin{bmatrix} \hat{\mathcal{X}}_{(1)} \\ \vdots \\ \hat{\mathcal{X}}_{(l)} \end{bmatrix} = \text{bdiag}((\hat{\mathcal{X}}_{(k)})), \quad \text{def}$$

where $F_l^H$ denotes the conjugate transpose of $F_l$ and the matrices $\hat{\mathcal{X}}_{(k)}$ for $k = 1, 2, \ldots, l$ are the frontal slices of the tubal matrix $\hat{\mathcal{X}}$ which is obtained by applying the DFT on $\mathcal{X}$ along the third dimension. We can use the Matlab function $\hat{\mathcal{X}} = \text{fft}(\mathcal{X}, [], 3)$ to calculate $\hat{\mathcal{X}}$ directly, and use the inverse FFT to calculate $\mathcal{X}$ from $\hat{\mathcal{X}}$, that is, $\mathcal{X} = \text{ifft}(\hat{\mathcal{X}}, [], 3)$. Thus, as noted in Reference 1, the t-product $\mathcal{X} \ast \mathcal{Y}$ can be computed by applying FFT along each tubal fiber of $\mathcal{X}$ and $\mathcal{Y}$ to obtain $\hat{\mathcal{X}} = \text{fft}(\mathcal{X}, [], 3)$ and $\hat{\mathcal{Y}} = \text{fft}(\mathcal{Y}, [], 3)$, multiplying each pair of the frontal slices of $\hat{\mathcal{X}}$ and $\hat{\math{Y}}$ to get the frontal slices of $\hat{\mathcal{Z}}$, and then taking inverse FFT along the third dimension of $\hat{\mathcal{Z}}$ to get the desired result.

Next, we recall other definitions and properties related to t-product; see References 1,2,9,13,14 for more details.

**Definition 3** (Transpose\(^1\)). For $\mathcal{X} \in \mathbb{K}_l^{m \times n}$, the transpose $\mathcal{X}^T$ is defined by taking the transpose of all the frontal slices and reversing the order of frontal slices 2 through $l$.

**Definition 4** (T-symmetric\(^1\)). For $\mathcal{X} \in \mathbb{K}_l^{m \times n}$, it is called T-symmetric if $\mathcal{X} = \mathcal{X}^T$.

**Definition 5** (Identity tubal matrix\(^1\)). The identity tubal matrix $I \in \mathbb{K}_l^{m \times n}$ is the tubal matrix whose first frontal slice is the $n \times n$ identity matrix and other frontal slices are all zeros.

**Definition 6** (Inverse\(^1\)). Let $\mathcal{X} \in \mathbb{K}_l^{m \times n}$. If there exists $\mathcal{Y} \in \mathbb{K}_l^{m \times n}$ such that

$$\mathcal{X} \ast \mathcal{Y} = I \quad \text{and} \quad \mathcal{Y} \ast \mathcal{X} = I,$$

then $\mathcal{X}$ is said to be invertible, and $\mathcal{Y}$ is the inverse of $\mathcal{X}$, which is denoted by $\mathcal{X}^{-1}$.

**Definition 7** (Moore-Penrose inverse\(^9\)). Let $\mathcal{X} \in \mathbb{K}_l^{m \times n}$. If there exists $\mathcal{Y} \in \mathbb{K}_l^{m \times n}$ such that

$$\mathcal{X} \ast \mathcal{Y} \ast \mathcal{X} = \mathcal{X}, \quad \mathcal{Y} \ast \mathcal{X} \ast \mathcal{Y} = \mathcal{Y}, \quad (\mathcal{X} \ast \mathcal{Y})^T = \mathcal{X} \ast \mathcal{Y}, \quad (\mathcal{Y} \ast \mathcal{X})^T = \mathcal{Y} \ast \mathcal{X},$$

then $\mathcal{Y}$ is called the Moore-Penrose inverse of $\mathcal{X}$ and is denoted by $\mathcal{X}^\dagger$.

**Lemma 1** (See Reference 9). The Moore-Penrose inverse of any tubal matrix $\mathcal{X} \in \mathbb{K}_l^{m \times n}$ exists and is unique, and if $\mathcal{X}$ is invertible, then $\mathcal{X}^\dagger = \mathcal{X}^{-1}$.

**Definition 8** (Orthogonal tubal matrix\(^3\)). For $\mathcal{X} \in \mathbb{K}_l^{m \times n}$, it is orthogonal if $\mathcal{X}^T \ast \mathcal{X} = \mathcal{X} \ast \mathcal{X}^T = I$.  

Definition 9 (See Reference 1). For \( \mathcal{X} \in \mathbb{K}^{m \times n}_t \), define

\[
\text{Range}(\mathcal{X}) = \left\{ \mathbf{V} \in \mathbb{K}^m \mid \mathbf{V} = \mathcal{X} \ast \mathbf{v}, \text{ for any } \mathbf{v} \in \mathbb{K}^n \right\}, \quad \text{Null}(\mathcal{X}) = \left\{ \mathbf{V} \in \mathbb{K}^n \mid \mathcal{X} \ast \mathbf{v} = \mathbf{0} \right\},
\]

\[
\text{Colsp}(\mathcal{X}) = \left\{ \mathbf{V} \in \mathbb{K}^{m \times p} \mid \text{ for all } 1 \leq j \leq p, \quad \mathbf{v}_{(:,j)} \in \text{Range}(\mathcal{X}) \right\},
\]

where \( \mathbf{0} \) is the zero tubal vector.

Definition 10 (See Reference 2). For \( P \in \mathbb{K}^{m \times n}_t \), it is a projector if \( P^2 = P \ast P = P \), and is orthogonal projector if \( P^T = P \) also holds.

Note that \( \mathcal{X} \ast (\mathcal{X}^T \ast \mathcal{X})^T \ast \mathcal{X}^T \) is an orthogonal projector onto \( \text{Range}(\mathcal{X}) \).

Definition 11 (T-symmetric T-positive (semi)definite\(^{11} \)). For \( \mathcal{X} \in \mathbb{K}^{m \times n}_t \), it is called T-symmetric T-positive (semi)definite if and only if \( \mathcal{X} \ast \mathbf{v} \) is T-symmetric and \( \langle \mathbf{y}, \mathcal{X} \ast \mathbf{v} \rangle \geq (\geq) 0 \) holds for any nonzero \( \mathbf{y} \in \mathbb{K}^n \) (for any \( \mathbf{y} \in \mathbb{K}^n \)).

Proposition 1 (see References 13,14). For \( \mathcal{X} \in \mathbb{K}^{m \times n}_t \), it is T-symmetric if and only if \( \text{bcirc}(\mathcal{X}) \) is symmetric, is invertible if and only if \( \text{bcirc}(\mathcal{X}) \) is invertible, is orthogonal if and only if \( \text{bcirc}(\mathcal{X}) \) is orthogonal, and is T-symmetric T-positive (semi)definite if and only if \( \text{bcirc}(\mathcal{X}) \) is symmetric positive (semi)definite if and only if \( \mathcal{X}_{(k,j)} = 0 \) for \( k = 1, 2, \ldots, l \) are all Hermitian positive (semi)definite.

The following lemma gives the definition and properties of the square root of a T-symmetric T-positive (semi)definite tubal matrix.

Lemma 2. Assume that \( \mathcal{X} \in \mathbb{K}^{m \times n}_t \) is a T-symmetric T-positive (semi)definite tubal matrix, and define \( \mathcal{X}^\frac{1}{2} = \text{bcirc}^{-1}(\text{bcirc}(\mathcal{X}))^\frac{1}{2} \). Then \( \mathcal{X} = \mathcal{X}^\frac{1}{2} \ast \mathcal{X}^\frac{1}{2} \) and \( \text{bcirc}(\mathcal{X}^\frac{1}{2}) = \text{bcirc}(\mathcal{X})^\frac{1}{2} \).

Proof. Note that \( \text{bcirc}(A \ast B) = \text{bcirc}(A) \ast \text{bcirc}(B) \) holds for any \( A \in \mathbb{K}^{m \times n}_t \) and \( B \in \mathbb{K}^{m \times p}_t \), which can be found in Reference 10. Then, we can obtain

\[
\text{bcirc}^{-1}(\text{bcirc}(A) \ast \text{bcirc}(B)) = \text{bcirc}^{-1}(\text{bcirc}(A \ast B)) = \text{bcirc}^{-1}(\text{bcirc}(A)) \ast \text{bcirc}^{-1}(\text{bcirc}(B)).
\]

Thus, considering \( \mathcal{X}^\frac{1}{2} = \text{bcirc}^{-1}((\text{bcirc}(\mathcal{X}))^\frac{1}{2}) \), we have

\[
\mathcal{X} = \text{bcirc}^{-1}(\text{bcirc}(\mathcal{X})) = \text{bcirc}^{-1}(\text{bcirc}(\mathcal{X})^\frac{1}{2}) \ast \text{bcirc}(\mathcal{X})^\frac{1}{2} = \text{bcirc}^{-1}(\text{bcirc}(\mathcal{X})) \ast \text{bcirc}^{-1}(\text{bcirc}(\mathcal{X})) = \mathcal{X}^\frac{1}{2} \ast \mathcal{X}^\frac{1}{2},
\]

and

\[
\text{bcirc}(\mathcal{X}^\frac{1}{2}) = \text{bcirc}(\text{bcirc}^{-1}(\text{bcirc}(\mathcal{X}))^\frac{1}{2}) = \text{bcirc}(\mathcal{X})^\frac{1}{2}.
\]

Then, the desired results hold.

In the following, we first present the definitions of the weighted norms for vectors and matrices, and then extend them to tubal vectors and tubal matrices, respectively.

Definition 12. Let \( Q \in \mathbb{R}^{m \times n} \) be a symmetric positive definite matrix. For any vectors \( \mathbf{x}, \mathbf{y} \in \mathbb{R}^n \), their weighted inner product and the weighted induced norm are defined as

\[
\langle \mathbf{x}, \mathbf{y} \rangle_Q \overset{\text{def}}{=} \langle Q \mathbf{x}, \mathbf{y} \rangle \quad \text{and} \quad \| \mathbf{x} \|_Q \overset{\text{def}}{=} \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_Q},
\]

respectively. For any matrix \( M \in \mathbb{R}^{m \times p} \), its weighted 2-norm and weighted Frobenius norm are defined as

\[
\| M \|_{2,Q} \overset{\text{def}}{=} \max_{\mathbf{x} \in \mathbb{R}^m, \| \mathbf{x} \|_Q = 1} \| M \mathbf{x} \|_Q \quad \text{and} \quad \| M \|_{F,Q} \overset{\text{def}}{=} \sqrt{\sum_{j=1}^{p} \| M_{(:,j)} \|_Q^2},
\]

respectively.
**Definition 13.** Let $Q \in \mathbb{K}_m^{p \times n}$ be a T-symmetric T-positive definite tubal matrix. For any tubal vectors $\mathbf{x}, \mathbf{y} \in \mathbb{K}_m^p$, their weighted inner product and the weighted induced norm are defined as

$$\langle \mathbf{x}, \mathbf{y} \rangle_Q = \langle Q \ast \mathbf{x}, \mathbf{y} \rangle = \langle \text{bcirc}(Q)\text{unfold}(\mathbf{x}), \text{unfold}(\mathbf{y}) \rangle = \langle \text{unfold}(\mathbf{x}), \text{unfold}(\mathbf{y}) \rangle_{\text{bcirc}(Q)},$$

and

$$\| \mathbf{x} \|_Q = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_Q} = \sqrt{||\text{unfold}(\mathbf{x})||_{\text{bcirc}(Q)}^2} = ||\text{unfold}(\mathbf{x})||_{\text{bcirc}(Q)} = \| \text{bcirc}(Q)\text{unfold}(\mathbf{x}) \|_{\text{bcirc}(Q)} = ||\mathbf{x}||_{\text{bcirc}(Q)},$$

respectively. For any tubal matrix $M \in \mathbb{K}_m^{r \times p}$, its weighted 2-norm and weighted Frobenius norm are defined as

$$\| M \|_2(Q) = \max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_Q = 1} ||M \ast \mathbf{x}||_Q$$

$$\text{and}$$

$$\| M \|_F(Q) = \max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_Q = 1} ||\text{unfold}(M \ast \mathbf{x})||_{\text{bcirc}(Q)}$$

$$\max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_Q = 1} ||\text{unfold}(M \ast \mathbf{x})||_{\text{bcirc}(Q)}$$

$$||M||_2 = \max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_Q = 1} ||M \ast \mathbf{x}||_Q$$

$$\text{and}$$

$$\| M \|_F = \max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_Q = 1} ||\text{unfold}(M \ast \mathbf{x})||_{\text{bcirc}(Q)}$$

**Lemma 3.** Let $Q \in \mathbb{K}_m^{p \times n}$ be a T-symmetric T-positive definite tubal matrix. Then for any tubal matrix $M \in \mathbb{K}_m^{r \times p}$,

$$||M||_F = ||Q \ast M||_F.$$

**Proof.** The result can be concluded by the properties of t-product and the definitions of the weighted norms. Specifically,

$$||M||_F = \sqrt{\sum_{j=1}^{p} ||\text{unfold}(M)_{(:,j)}||_{\text{bcirc}(Q)}^2} = \sqrt{\sum_{j=1}^{p} ||\text{unfold}(M)_{(:,j)}^T \text{bcirc}(Q)\text{unfold}(M)_{(:,j)}||_{\text{bcirc}(Q)}^2}$$

$$= \sqrt{\sum_{j=1}^{p} ||\text{bcirc}(Q)^{\frac{1}{2}}\text{unfold}(M)_{(:,j)}||_{\text{bcirc}(Q)}^4} = \sqrt{\sum_{j=1}^{p} ||\text{bcirc}(Q)^{\frac{1}{2}}\text{unfold}(M)_{(:,j)}||_{\text{bcirc}(Q)}^2} = \|\text{bcirc}(Q)^{\frac{1}{2}}\text{unfold}(M)||_F = ||\text{bcirc}(Q)^{\frac{1}{2}}\text{unfold}(M)||_F = ||Q \ast M||_F.$$  

For a symmetric positive semidefinite matrix $D \in \mathbb{K}_m^{n \times n}$, we write the weighted seminorm induced by $D$ as $\| \mathbf{x} \|_D = \sqrt{\langle \mathbf{x}, D \mathbf{x} \rangle}$. Similarly, we can also define the weighted seminorm induced by a T-symmetric T-positive semidefinite tubal matrix $D \in \mathbb{K}_m^{p \times n}$ as

$$||\mathbf{x}||_D = \sqrt{\langle \mathbf{x}, D \ast \mathbf{x} \rangle} = ||\text{unfold}(\mathbf{x})||_{b,\text{circ}(D)};$$

and, for any tubal matrix $M \in \mathbb{K}_m^{r \times p}$, define

$$||M||_{2(D)} = \max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_D = 1} ||M \ast \mathbf{x}||_D$$

$$\text{and}$$

$$||M||_{F(D)} = \max_{\mathbf{x} \in \mathbb{K}_m^r, \| \mathbf{x} \|_D = 1} ||\text{unfold}(M \ast \mathbf{x})||_{\text{bcirc}(D)}.$$

Moreover, we have $||M||_{F(D)} = ||D^{\frac{1}{2}} \ast M||_F$.  

3 | TSP METHOD AND ITS ADAPTIVE VARIANTS

In this section, we first present the TSP method, and then introduce the adaptive sampling idea into the TSP method.

3.1 | TSP method

Similar to the MSP method, the TSP method is designed to pursue the next iterate \( \mathcal{X}^{t+1} \in \mathbb{K}^{nxp} \) which is the nearest point to \( \mathcal{X}^t \) and at the same time satisfies a sketched version of problem (1), that is

\[
\mathcal{X}^{t+1} = \arg \min_{\mathcal{X} \in \mathbb{K}^{nxp}} \| \mathcal{X} - \mathcal{X}^t \|_F^2 \quad \text{subject to} \quad S^T \cdot \mathcal{A} \cdot \mathcal{X} = S^T \cdot B,
\]

where \( S \in \mathbb{K}^{nxr} \) is a sketching tubal matrix drawn in an i.i.d. fashion from a fixed distribution \( \mathcal{D} \) at each iteration, and \( Q \in \mathbb{K}^{nxn} \) is a T-symmetric T-positive definite tubal matrix. The distribution \( \mathcal{D} \) and tubal matrix \( Q \) are the parameters of the method. Making use of the algebraic properties of t-product, we can get the explicit solution to (3) as

\[
\mathcal{X}^{t+1} = \mathcal{X}^t - Q^{-1} \cdot \mathcal{A}^T \cdot S \cdot (S^T \cdot \mathcal{A} \cdot Q^{-1} \cdot \mathcal{A}^T \cdot S)^\dagger \cdot S^T \cdot (\mathcal{A} \cdot \mathcal{X}^t - B),
\]

and then we obtain the TSP method, that is, Algorithm 1.

Algorithm 1. TSP method

**Input:** \( \mathcal{X}^0 \in \mathbb{K}^{nxp} \), \( \mathcal{A} \in \mathbb{K}^{nxn} \), \( B \in \mathbb{K}^{nxn} \)

**Parameters:** fixed distribution \( \mathcal{D} \) over random tubal matrices from \( \mathbb{K}^{nxr} \), T-symmetric T-positive definite tubal matrix \( Q \in \mathbb{K}^{nxn} \)

for \( t = 0, 1, 2, \cdots \)

Sample an independent copy \( S \sim \mathcal{D} \), and hence the sketching tubal matrix \( S \in \mathbb{K}^{nxr} \)

Compute \( \tilde{G} = S \cdot (S^T \cdot \mathcal{A} \cdot Q^{-1} \cdot \mathcal{A}^T \cdot S)^\dagger \cdot S^T \)

\( \mathcal{X}^{t+1} = \mathcal{X}^t - Q^{-1} \cdot \mathcal{A}^T \cdot \tilde{G} \cdot (\mathcal{A} \cdot \mathcal{X}^t - B) \)

**end for**

**Output:** last iterate \( \mathcal{X}^{t+1} \)

Remark 1. In general, the distribution \( \mathcal{D} \) of the sketching tubal matrix \( S \) can be any continuous or discrete distribution, and the parameter \( Q \) can be any T-symmetric T-positive definite tubal matrix. As a special case, if we set \( S = I_{(i:,\ldots,:)} \in \mathbb{K}^m_i \) with \( i = 1, 2, \ldots, m \) and \( Q = I \in \mathbb{K}^{nxn} \), then it follows from (4) that

\[
\mathcal{X}^{t+1} = \mathcal{X}^t - \mathcal{A}^T_{(i:,\ldots,:)} \cdot \left( \mathcal{A}_{(i:,\ldots,:)} \cdot \mathcal{A}^T_{(i:,\ldots,:)} \right)^\dagger \cdot \left( \mathcal{A}_{(i:,\ldots,:)} \cdot \mathcal{X}^t - B_{(i:,\ldots,:)} \right).
\]
When $i$ is selected uniformly or with the probability proportional to the Frobenius norm of the $i$th horizontal slice of $A^*$, the TSP method will reduce to the TRK method in Reference 23.

Next, we shall give the convergence theorem of the TSP method.

**Theorem 1.** With the notation in Algorithm 1, assume that $E[Z]$ is T-symmetric T-positive definite with probability 1, where $Z = Q^{-\frac{1}{2}} WQ^{-\frac{1}{2}}$ and $W = A^T \ast G \ast A$, $A^*$ satisfies $A \ast A^* = B$, and $A_t$ is the $t$th approximation of $A^*$ with initial iterate $A^0$. Then

$$E \left[ \| \lambda^t - X^* \|^2_{F(Q)} \right] \leq (1 - \lambda_{\min}(E[bcirc(Z)])^t \| \lambda^0 - X^* \|^2_{F(Q)}.$$ 

**Remark 2.** Now, we show that the convergence factor $\rho_{TSP} = 1 - \lambda_{\min}(E[bcirc(Z)])$ is smaller than 1. It is easy to check that $bcirc(Z)$ is an orthogonal projector and hence has eigenvalues 0 or 1. Furthermore, it projects onto a $d$-dimensional subspace

$$\text{Range}(bcirc(Q)^{-\frac{1}{2}} bcirc(A)^T bcirc(S)),$$

where $d \overset{\text{def}}{=} \text{Rank}(bcirc(S)^T bcirc(A)) \leq \min(nl, ml)$. Using Jensen’s inequality, as well as the fact that both $A \mapsto \lambda_{\max}(A)$ and $A \mapsto -\lambda_{\min}(A)$ are convex on the symmetric matrices, we can conclude that the spectrum of $E[bcirc(Z)]$ is contained in $[0, 1]$. Next, we turn to refine the lower and upper bounds of $\lambda_{\min}(E[bcirc(Z)])$. It follows that

$$E[d] = E[\text{Tr}(bcirc(Z))] = \text{Tr}(E[bcirc(Z)]) \geq n\lambda_{\min}(E[bcirc(Z)]),$$

where the inequality holds because the trace of a matrix is equal to the sum of its eigenvalues. Thus, we have $\lambda_{\min}(E[bcirc(Z)]) \leq \frac{E[d]}{nl}$. Furthermore, $E[bcirc(Z)]$ is symmetric positive definite because $E[Z]$ is T-symmetric T-positive definite, which immediately yields $\lambda_{\min}(E[bcirc(Z)]) > 0$. All together, we have the following lower and upper bounds on $\rho_{TSP}$:

$$0 \leq 1 - \frac{E[d]}{nl} \leq \rho_{TSP} < 1.$$ 

So, the convergence factor is indeed smaller than 1 and hence the sequence $\{\lambda^t\}_{t=0}^\infty$ generated by the TSP method can converge to $\lambda^*$. 

**Remark 3.** The convergence guarantee for the TRK method presented in Reference 23 is a special case of Theorem 1. Specifically, setting the sketching tubal matrices $S = I_{(i, \ldots, i)} \in \mathbb{K}_l^{mxn}$ with $i = 1, 2, \ldots, m$ and $Q = I \in \mathbb{K}_l^{mxn}$ in Theorem 1, we can recover the result given in theorem 3.1 in Reference 23.

### 3.2 Three adaptive TSP methods

As shown in Section 3.1, a key step of the TSP method is to choose a sketching tubal matrix $S$ in an i.i.d. fashion from a fixed distribution. In this subsection, we mainly study the adaptive sampling strategies on a finite set of sketching tubal matrices which is selected from a certain distribution in advance (the selection of the finite set is not considered in this paper). That is, letting $S = \{S_i \in \mathbb{K}_l^{nxr}, \text{ for } i = 1, 2, \ldots, q, \ q \in \mathbb{N}\}$ be a finite set of sketching tubal matrices where

**Algorithm 2.** NTSP method

**Input:** $A^0 \in \mathbb{K}_l^{nxp}$, $A \in \mathbb{K}_l^{mxn}$, $B \in \mathbb{K}_l^{mxp}$, and $p \in \Delta_q$

**Parameters** : a set of sketching tubal matrices $S = \{S_1, S_2, \ldots, S_q\}$ with $S_i \in \mathbb{K}_l^{nxr}$, T-symmetric T-positive definite tubal matrix $Q \in \mathbb{K}_l^{mxn}$

**for** $t = 0, 1, 2, \ldots$

$\{i^t\} \sim p$

Compute $G_{i^t} = S_{i^t}^T \ast A \ast Q^{-1} \ast A^T \ast S_{i^t}$

$A^{t+1} = A^t - Q^{-1} \ast A^T \ast G_{i^t} \ast (A \ast A^* - B)$

**end for**

**Output:** last iterate $A^{t+1}$
\( \tau \in \mathbb{N} \) is the sketch size, we want to choose \( \mathcal{S} = S_1 \) from \( \mathcal{S} \) using adaptive sampling strategies. If the sampling probability distribution at each iteration is fixed, we call the corresponding method the nonadaptive TSP (NTSP for short, where \( N \) stands for nonadaptive) method, which is summarized in Algorithm 2.

**Remark 4.** There are some subtle differences between Algorithms 1 and 2. Specifically, the former draws a sketching tubal matrix from a fixed distribution at each iteration, while the latter needs to select a finite set of sketching tubal matrices from a distribution in advance, and then picks one from the finite set with a fixed sampling probability distribution at each iteration. It should be noted that when the TSP method reduces to the TRK method, the two algorithms are the same.

### 3.2.1 Three adaptive sampling strategies

Considering that the fixed sampling strategy in Algorithm 2 may choose a terrible \( S_i \) and hence leads to a bad convergence, we introduce three adaptive sampling strategies which use information about the current iterate. With these adaptive sampling strategies, the algorithms can significantly reduce the number of iterations at the cost of adding a relatively small amount of computation per iteration and hence can reduce the overall computation cost.

Specifically, let

\[
W_\tau = A^T \ast G_\tau \ast A \quad \text{and} \quad Z_\tau = Q^{-\frac{1}{2}} \ast W_\tau \ast Q^{-\frac{1}{2}},
\]

(5)

where \( G_\tau \) is defined in Algorithm 2. Thus, using the facts that \( A \ast \mathcal{X}^* = B \) and \( Z_\tau \) is an orthogonal projector onto \( \text{Range}(Q^{-\frac{1}{2}} \ast A^T \ast S_i) \), we have

\[
\|\mathcal{X}^{t+1} - \mathcal{X}^*\|_{F(Z)}^2 = \|Q^{\frac{1}{2}} \ast (\mathcal{X}^{t+1} - \mathcal{X}^*)\|_F^2 = \|Q^{\frac{1}{2}} \ast (I - Q^{-1} \ast W_\tau) \ast (\mathcal{X}^t - \mathcal{X}^*)\|_F^2
\]

\[
= \|Q^{\frac{1}{2}} \ast (I - Q^{-1} \ast W_\tau) \ast Q^{-\frac{1}{2}} \ast Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_F^2 = \|(I - Z_\tau) \ast Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_F^2
\]

\[
= \|Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_F^2 - \|Z_\tau \ast Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_F^2
\]

\[
= \|\mathcal{X}^t - \mathcal{X}^*\|_{F(Z)}^2 - \|Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_F^2
\]

(6)

which shows that the quantity of the error \( \mathcal{X}^{t+1} - \mathcal{X}^* \) is determined by \( f_\tau(\mathcal{X}^t) \overset{\text{def}}{=} \|Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_{F(Z)}^2 \). Consequently, in order to make the most progress in one step, we should choose \( \tau \) corresponding to the largest sketched loss \( f_\tau(\mathcal{X}^t) \). Since \( \mathcal{X}^* \) is unknown in practice, we first rewrite \( f_\tau(\mathcal{X}^t) \) as

\[
f_\tau(\mathcal{X}^t) = \|Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*)\|_{F(Z)}^2 = \sum_{j=1}^{p} \|\text{unfold}(Q^{\frac{1}{2}} \ast (\mathcal{X}^t - \mathcal{X}^*))_{(:,j)}\|_{bcirc(Z)}^2
\]

\[
= \sum_{j=1}^{p} (\text{unfold}(\mathcal{X}^t - \mathcal{X}^*)_{(:,j)}^T \ast \text{bcirc}(Q_{1,2} \ast \text{bcirc}(Z_{1,2})) \ast \text{bcirc}(Q_{1,2} \ast \text{bcirc}(\mathcal{X}^t - \mathcal{X}^*))_{(:,j)}
\]

\[
= \sum_{j=1}^{p} (\text{unfold}(\mathcal{X}^t - \mathcal{X}^*)_{(:,j)}^T \ast \text{bcirc}(A_{1,2} \ast \text{bcirc}(G_{1,2})) \ast \text{bcirc}(A_{1,2} \ast \text{bcirc}(\mathcal{X}^t - \mathcal{X}^*))_{(:,j)}
\]

\[
= \sum_{j=1}^{p} \|\text{bcirc}(A_{1,2} \ast \text{bcirc}(\mathcal{X}^t - \mathcal{X}^*))_{(:,j)}\|_{bcirc(G_{1,2})}^2 = \|A \ast (\mathcal{X}^t - \mathcal{X}^*)\|_{F(G)}^2 = \|A \ast \mathcal{X}^t - B\|_{F(G)}^2.
\]

Thus, according to (6), we can present the first adaptive sampling strategy as follows

\[
i^* = \arg \max_{i=1,2,\ldots,q} f_i(\mathcal{X}^t) = \arg \max_{i=1,2,\ldots,q} \|A \ast \mathcal{X}^t - B\|_{F(G)}^2,
\]

(7)

which can be called the max-distance selection rule. The corresponding adaptive TSP method is called ATSP-MD method for short, where A and MD stand for adaptive and max-distance, respectively, and the algorithm is described as the case 1 of Algorithm 3.
Now, we consider the expected decrease of the error $\mathcal{X}^{t+1} - \mathcal{X}^*$. Let $p^t \in \Delta_q$ and $i^t \sim p^t$, where $p^t \overset{\text{def}}{=} (p^{t_1}_1, p^{t_2}_2, \ldots, p^{t_q}_q)^T$ with $p^{t_i}_i = \mathbf{P}[S_i = S_i | \mathcal{X}^{t_i}]$ for $i = 1, 2, \ldots, q$, that is, $p^{t_i}_i$ is the probability of $S_i$ being sampled at the $t_i$th iteration. Taking expectation conditioned on $\mathcal{X}^{t}$ in (6), we have

$$E[\mathcal{X}^{t+1} - \mathcal{X}^*] = ||\mathcal{X}^{t} - \mathcal{X}^*||_{F(Q)}^2 - E_{\mathcal{I}}[f_i(\mathcal{X}^{t_i})].$$

which suggests that if we want $E[||\mathcal{X}^{t+1} - \mathcal{X}^*||_{F(Q)}^2]$ to be as small as possible, we should choose adaptive probabilities to make $E_{\mathcal{I}}[f_i(\mathcal{X}^{t_i})]$ as large as possible. Since $E_{\mathcal{I}}[f_i(\mathcal{X}^{t_i})] = \sum_{i=1}^q p^{t_i}_i f_i(\mathcal{X}^{t_i})$, we can achieve the above goal by sampling the indices corresponding to the smaller sketched losses with higher probability. An intuitive way is to choose the probabilities proportional to the sketched losses and we refer to such strategy as adaptive probabilities rule. The corresponding adaptive TSP method is called ATSP-PR method for short, where PR stands for probabilities, and the algorithm is summarized as the case 2 of Algorithm 3.

In addition, there is another effective strategy, which aims to capture the indices corresponding to larger sketched losses as far as possible at each iteration. To this end, it considers removing the indices corresponding to the smaller sketched losses. To be specific, we first define an index set

$$\mathcal{B}_i = \left\{ i | f_i(\mathcal{X}^{t_i}) \geq \theta \max_{j=1,2,\ldots,q} f_j(\mathcal{X}^{t_i}) + (1 - \theta)E_{\mathcal{I} \setminus p}[f_j(\mathcal{X}^{t_j})] \right\},$$

where $p \in \Delta_q$ and $\theta \in [0, 1]$. Then, we choose the probabilities $p^t \in \Delta_q$ such that

$$p^{t_i}_i = \begin{cases} \frac{f_i(\mathcal{X}^{t_i})}{\sum_{i \in \mathcal{B}_i} f_i(\mathcal{X}^{t_i})} & i \in \mathcal{B}_i \\ 0 & i \notin \mathcal{B}_i \end{cases}$$

We call this strategy the capped sampling rule, and the corresponding adaptive TSP method ATSP-CS method for short, where CS stands for capped sampling. The algorithm is summarized as the case 3 of Algorithm 3.

**Algorithm 3. ATSP-(MD/PR/CS) method**

- **Input**: $\mathcal{X}^0 \in \mathbb{F}_q^{mxp}$, $A \in \mathbb{K}_q^{mna}$, $B \in \mathbb{K}_q^{mxp}$, $p \in \Delta_q$, and $\theta \in [0, 1]$  
- **Parameters**: a set of sketching tubal matrices $\mathbf{S} = [S_1, S_2, \ldots, S_q]$ with $S_i \in \mathbb{F}_l^{mxr}$, T-symmetric T-positive definite tubal matrix $Q \in \mathbb{K}_q^{npx}$  
- **for** $t = 0, 1, 2, \cdots$  
  - $f_i(\mathcal{X}^{t_i}) = \|A * \mathcal{X}^{t_i} - B\|_{F(G)}^2$ for $i = 1, 2, \ldots, q$  
  - **switch** adaptive sampling strategies
    - **case** 1 Max-distance selection rule (MD)  
      - $i^t = \arg \max_{i=1,2,\ldots,q} f_i(\mathcal{X}^{t_i})$  
    - **case** 2 Adaptive probabilities rule (PR)  
      - Calculate $p^t \in \Delta_q$ such that $p^{t_i}_i = f_i(\mathcal{X}^{t_i}) / (\sum_{i=1}^q f_i(\mathcal{X}^{t_i}))$ for $i = 1, 2, \ldots, q$  
      - $i^t \sim p^t$  
    - **case** 3 Capped sampling rule (CS)  
      - Determine the index set $\mathcal{B}_i$, which is defined in (9)  
      - Calculate $p^t \in \Delta_q$, which is defined in (10)  
      - $i^t \sim p^t$  
  - Compute $G^{t_i} = S^{t_i}_i * (S^{t_i}_i \ast A \ast Q^{-1} \ast A^T \ast S^{t_i}_i)^T \ast S^{t_i}_i$  
  - $\mathcal{X}^{t+1} = \mathcal{X}^t - Q^{-1} \ast A^T \ast G^{t_i} \ast (A \ast \mathcal{X}^t - B)$  
- **end for**
- **Output**: last iterate $\mathcal{X}^{t+1}$
Remark 5. In the TRK setting, that is, $S_i = I_{(i;1,1)} \in \mathbb{R}_+^m$ for $i = 1, 2, \ldots, m$ and $Q = I \in \mathbb{R}_+^{n \times n}$, the above ATSP-MD, ATSP-PR and ATSP-CS methods are typically referred to as the ATRK-MD, ATRK-PR and ATRK-CS methods, which are the tensor versions of the greedy or adaptive MRK methods given in References 30,32,33.

3.2.2 | Convergence

In this subsection, we discuss the convergence analysis of the nonadaptive and adaptive TSP methods proposed above. Before the formal discussions, we first present two lemmas.

**Lemma 4.** With the notation in the NTSP, ATSP-MD, ATSP-PR, and ATSP-CS methods, let $p \in \Delta_q$ and define

\begin{align}
\delta^2_{\infty}(Q, S) &\overset{\text{def}}{=} \min_{\bar{V} \in \text{Range}(Q^{-1} \ast A^T)} \max_{i=1,2,\ldots,q} \frac{\|Q^{\frac{1}{2}} \ast \bar{V}\|_{L^2}^{2}}{\|V\|_{L^2}^{2}}, \tag{11} \\
\delta^2_{p}(Q, S) &\overset{\text{def}}{=} \min_{\bar{V} \in \text{Range}(Q^{-1} \ast A^T)} \frac{\|Q^{\frac{1}{2}} \ast \bar{V}\|_{L^2}^{2}}{\|V\|_{L^2}^{2}} \tag{12},
\end{align}

where $Z_i$ is the same as $Z_i$ defined in (5) except that $i'$ is replaced by $i$. Let $X^*$ satisfy $A \ast X^* = B$ and $X^i$ be the $i$th approximation of $X^*$ calculated by any nonadaptive and adaptive algorithms with initial iterate $X^0 \in \text{Colsp}(Q^{-1} \ast A^T)$. Then

\begin{align}
\max_{i=1,2,\ldots,q} f_i(X^i) \geq \delta^2_{\infty}(Q, S)\|X^i - X^*\|_{F(Q)}, \tag{13} \\
E_{-p}[f_i(X^i)] \geq \delta^2_{p}(Q, S)\|X^i - X^*\|_{F(Q)}, \tag{14}
\end{align}

**Lemma 5.** Let $p \in \Delta_q$ and the set of sketching tubal matrices $S = [S_1, S_2, \ldots, S_q]$ be such that $E_{-p}[Z_i]$ is $T$-symmetric $T$-positive definite with probability 1. Then

$$0 \leq \lambda_{\min}(E_{-p}[bcirc(Z_i)]) = \delta^2_{p}(Q, S) \leq \delta^2_{\infty}(Q, S) \leq 1. \tag{15}$$

Next, we give the convergence guarantees of the NTSP, ATSP-MD, ATSP-PR and ATSP-CS methods in Theorems 2, 3, 4, and 5, respectively.

**Theorem 2.** Let $X^*$ satisfy $A \ast X^* = B$ and $X^i$ be the $i$th approximation of $X^*$ calculated by the NTSP method, that is, Algorithm 2, with initial iterate $X^0 \in \text{Colsp}(Q^{-1} \ast A^T)$. Then

$$E[\|X^i - X^*\|_{F(Q)}^2 \ast X^0] \leq (1 - \delta^2_{p}(Q, S))\|X^0 - X^*\|_{F(Q)}^2,$$

where $\delta^2_{p}(Q, S)$ is as defined in (12).

**Remark 6.** Since $\delta^2_{p}(Q, S) = \lambda_{\min}(E_{-p}[bcirc(Z_i)])$, the conclusion of Theorem 2 can be rewritten as

$$E[\|X^i - X^*\|_{F(Q)}^2 \ast X^0] \leq (1 - \lambda_{\min}(E_{-p}[bcirc(Z_i)]))\|X^0 - X^*\|_{F(Q)}^2,$$

which is consistent with Theorem 1.

**Theorem 3.** Let $X^*$ satisfy $A \ast X^* = B$ and $X^i$ be the $i$th approximation of $X^*$ calculated by the ATSP-MD method, that is, the first case of Algorithm 3, with initial iterate $X^0 \in \text{Colsp}(Q^{-1} \ast A^T)$. Then

$$\|X^i - X^*\|_{F(Q)}^2 \leq (1 - \delta^2_{\infty}(Q, S))\|X^0 - X^*\|_{F(Q)}^2,$$

where $\delta^2_{\infty}(Q, S)$ is as defined in (11).

**Remark 7.** Since $\delta^2_{p}(Q, S) \leq \delta^2_{\infty}(Q, S)$, the convergence factor for the ATSP-MD method is better than that for the NTSP method.
Theorem 4. Let \( \mathbf{u} = (\frac{1}{q}, \frac{1}{q}, \ldots, \frac{1}{q})^T \in \Delta_q \) and \( \hat{\delta}_0^2(Q, S) \) be as defined in (12). Let \( \lambda^* \) satisfy \( A * \lambda^* = B \) and \( \lambda^i \) be the \( i \)th approximation of \( \lambda^* \) calculated by the ATSP-PR method, that is, the second case of Algorithm 3, with initial iterate \( \lambda^0 \in \text{Colsp}(Q^{-1} * A^T) \). Then, for \( t \geq 1 \),
\[
 E \left[ \|\lambda^{t+1} - \lambda^*\|_{F(Q)}^2 | \lambda^t \right] \leq \left( 1 - \left( 1 + \frac{1}{q} \right) \hat{\delta}_0^2(Q, S) \right) E \left[ \|\lambda^t - \lambda^*\|_{F(Q)}^2 | \lambda^0 \right],
\]
where \( \text{Var}_{\mathbf{u}}[-] \) denotes the variance taken with respect to the uniform distribution \( \mathbf{u} \), that is,
\[
 \text{Var}_{\mathbf{u}}[v_i] = \frac{1}{q} \sum_{i=1}^{q} \left( v_i - \frac{1}{q} \sum_{i=1}^{q} v_i \right)^2, \quad \forall \; v \in \mathbb{R}^q.
\]
Furthermore,
\[
 E \left[ \|\lambda^{t+1} - \lambda^*\|_{F(Q)}^2 | \lambda^t \right] \leq \left( 1 - \left( 1 + \frac{1}{q} \right) \hat{\delta}_0^2(Q, S) \right) E \left[ \|\lambda^t - \lambda^*\|_{F(Q)}^2 | \lambda^0 \right].
\]

Remark 8. The convergence factor for the ATSP-PR method is smaller than that for the NTSP method with respect to uniform sampling, and how much smaller depends on the value of \( 1 + q^2 \text{Var}_{\mathbf{u}}[p_i] \).

Theorem 5. Let \( \lambda^* \) satisfy \( A * \lambda^* = B \) and \( \lambda^i \) be the \( i \)th approximation of \( \lambda^* \) calculated by the ATSP-CS method, that is, the third case of Algorithm 3, with initial iterate \( \lambda^0 \in \text{Colsp}(Q^{-1} * A^T) \). Then
\[
 E[\|\lambda^i - \lambda^*\|_{F(Q)}^2 | \lambda^0] \leq (1 - \theta \hat{\delta}_0^2(Q, S) - (1 - \theta) \hat{\delta}_0^2(Q, S)) \|\lambda^0 - \lambda^*\|_{F(Q)}^2,
\]
where \( \hat{\delta}_0^2(Q, S) \) and \( \hat{\delta}_0^2(Q, S) \) are as defined in (11) and (12), respectively.

Remark 9. The convergence factor of the ATSP-CS method is a convex combination of ones of the NTSP and ATSP-MD methods, and hence we can conclude that the closer \( \theta \) approaches 0, the slower the convergence factor of the ATSP-CS method is. This is because, in this case, the convergence factor will approach the one of the NTSP method, which is a little slower than that of the ATSP-MD method as pointed out in Remark 7.

Remark 10. According to Lemma 5, Theorems 2, 3, 4, and 5, we can conclude that the NTSP, ATSP-MD, ATSP-PR and ATSP-CS methods all converge under the assumption that \( E_{\mathbf{u}}[Z] \) is T-symmetric T-positive definite with probability 1.

Remark 11. In the TRK setting, that is, \( S_i = I(\ldots, 1) \in \mathbb{K}^{m} \) for \( i = 1, 2, \ldots, m \) and \( Q = I \in \mathbb{K}^{p} \), using Theorems 2, 3, 4, and 5, we can get the convergence guarantees for the NTRK, ATRK-MD, ATRK-PR and ATRK-CS methods, respectively.

4 | THE FOURIER VERSION OF THE TSP METHOD

Based on (2) and the discussions following it, we can present an efficient implementation of the TSP method in the Fourier domain, that is, Algorithm 4.

Furthermore, in view of (2), the problem (1) can be reformulated as
\[
\begin{bmatrix}
\hat{A}_{(1)} \\
\hat{A}_{(2)} \\
\vdots \\
\hat{A}_{(l)}
\end{bmatrix}
= 
\begin{bmatrix}
\hat{X}_{(1)} \\
\hat{X}_{(2)} \\
\vdots \\
\hat{X}_{(l)}
\end{bmatrix}
= 
\begin{bmatrix}
\hat{B}_{(1)} \\
\hat{B}_{(2)} \\
\vdots \\
\hat{B}_{(l)}
\end{bmatrix},
\]
where \( \hat{A}_{(k)} \), \( \hat{X}_{(k)} \), and \( \hat{B}_{(k)} \) for \( k = 1, 2, \ldots, l \) are the frontal slices of \( \hat{A} = \text{fft}(A, [1, \ldots, 3]) \), \( \hat{X} = \text{fft}(X, [1, \ldots, 3]) \), and \( \hat{B} = \text{fft}(B, [1, \ldots, 3]) \), respectively. Hence, the original tensor linear system can be transformed into \( l \) matrix subsystems in the Fourier domain, that is, \( \hat{A}_{(k)} \hat{X}_{(k)} = \hat{B}_{(k)} \) for \( k = 1, 2, \ldots, l \), and the update formula in Algorithm 4 is just the same as the one in the MSP method. However, \( \hat{S}_{(k)} \) for \( k = 1, 2, \ldots, l \) are usually not independent. We will discuss this further in the next section.

We now present a theorem that gives the convergence guarantee for Algorithm 4.
**Algorithm 4.** TSP method in Fourier domain

**Input:** $X^0 \in \mathbb{K}_I^{n \times p}$, $A \in \mathbb{K}_I^{m \times n}$, $B \in \mathbb{K}_I^{m \times p}$

**Parameters:** fixed distribution $\mathfrak{D}$ over random tubal matrices from $\mathbb{K}_I^{m \times r}$, T-symmetric T-positive definite tubal matrix $Q \in \mathbb{K}_I^{m \times m}$

\[
\hat{X}^0 \leftarrow \text{fft}(X^0, [\ ] , 3), \hat{A} \leftarrow \text{fft}(A, [\ ] , 3), \hat{B} \leftarrow \text{fft}(B, [\ ] , 3), \hat{Q} \leftarrow \text{fft}(Q, [\ ] , 3),
\]

for $t = 0, 1, 2, \ldots$

Sample an independent copy $S \sim \mathfrak{D}$, and hence the sketching tubal matrix $\tilde{S} \in \mathbb{K}_I^{m \times r}$

\[
\tilde{S} \leftarrow \text{fft}(S, [\ ] , 3)
\]

for $k = 1, 2, \ldots , l$

Compute $\tilde{S}_k(H) = \hat{S}_k \left( \hat{S}_k^H \hat{A}_k \hat{Q}_k^{-1} \hat{A}_k^H \hat{S}_k \right)^\dagger \hat{S}_k^H$

\[
\hat{X}^{t+1} = \hat{X}^t - \tilde{S}_k^{-1} \hat{A}_k \hat{Q}_k \left( \hat{A}_k \hat{X}^t - \tilde{B}_k \right)
\]

end for

end for

$X^{t+1} \leftarrow \text{ifft}(\hat{X}^{t+1}, [\ ] , 3)$

**Output:** last iterate $X^{t+1}$

**Theorem 6.** With the notation in Algorithm 4, assume that $\mathbb{E}[\text{diag}(\hat{Z})]$ is Hermitian positive definite with probability 1, where $\hat{Z} = \text{fft}(Z, [\ ] , 3)$ and $Z$ is as defined in Theorem 1. Let $X^*$ satisfy $A \ast X^* = B$ and $X^t$ be the $t$th approximation of $X^*$ with initial iterate $X^0$. Then

\[
\mathbb{E} \left[ \left\| X^t - X^* \right\|_F^2 \right] \leq \left( 1 - \min_{k=1,2,\ldots,l} \lambda_{\min}(\mathbb{E}[\hat{Z}_k]) \right)^t \left\| X^0 - X^* \right\|_F^2,
\]

(17)

**Remark 12.** Similar to Remark 2, we can obtain that

\[
0 \leq 1 - \min_{k=1,2,\ldots,l} \frac{\mathbb{E}[d_k]}{n} \leq 1 - \min_{k=1,2,\ldots,l} \lambda_{\min}(\mathbb{E}[\hat{Z}_k]) < 1,
\]

where $d_k = \text{Rank}(\hat{S}_k(H) \hat{A}_k)$ for $k = 1, 2, \ldots , l$. So Algorithm 4 is indeed convergent.

Compared with Algorithm 1, Algorithm 4 not only is a practical implementation version but also can motivate us to improve the algorithm. We will discuss it in the next section. Similarly, although Theorem 6 is equivalent to Theorem 1 in essence, with the former, we can more conveniently give the convergence analysis for the TSP method when the random tubal matrix $S$ has a special discrete probability distribution. We will consider it in the following corollary. To this end, we first recall the definition of the complete discrete sampling matrix presented in Reference 35: A sampling matrix $S$ is called a complete discrete sampling matrix if it satisfies three conditions, that is, the random matrix $S$ has a discrete distribution, $S = S_1 \in \mathbb{K}_I^{m \times r}$ with probability $p_1 > 0$ and $S_l^l A$ has full row rank for $i = 1, 2, \ldots , q$, and $S = [S_1, S_2, \ldots , S_q] \in \mathbb{K}_I^{m \times qr}$ is such that $A^T S$ has full row rank.

**Corollary 1.** With the notation in Algorithm 4 and Theorem 6, let $S$ be a discrete sampling tubal matrix satisfying that $\hat{S}_k$ for $k = 1, 2, \ldots , l$ are all complete discrete sampling matrices, where $\hat{S} = \text{fft}(S, [\ ] , 3)$, and $S = S_1 \in \mathbb{K}_I^{m \times r}$ with probability $p_1$ for $i = 1, 2, \ldots , q$. Let $X^*$ satisfy $A \ast X^* = B$ and $X^t$ be the $t$th approximation of $X^*$ with initial iterate $X^0$. Then when

\[
p_i = \frac{\|Q^{-1} A^s S_i^s\|_F^2}{\|Q^{-1} A^T S_i\|_F^2}
\]

with $S = [S_1, S_2, \ldots , S_q]$ for $i = 1, 2, \ldots , q$, we have

\[
\mathbb{E} \left[ \left\| X^t - X^* \right\|_F^2 \right] \leq \left( 1 - \min_{k=1,2,\ldots,l} \frac{\lambda_{\min}(\hat{S}_k(H) \hat{A}_k \hat{Q}_k^{-1} \hat{A}_k^H \hat{S}_k)}{\|Q^{-1} A^T S_i\|_F^2} \right)^t \left\| X^0 - X^* \right\|_F^2,
\]

(18)
when \( p_i = \frac{1}{q} \) for \( i = 1, 2, \ldots, q \), we have

\[
E \left[ \left\| \mathcal{X}^t - \mathcal{X}^* \right\|_{F(Q)}^2 | \mathcal{X}^0 \right] \leq 1 - \min_{k=1,2,\ldots,l} \lambda^+_{\min} \left( \hat{S}^H_{(k)} \hat{A}_{(k)} \hat{S}_{(k)} \right)^{\frac{1}{2}} \left\| \mathcal{X}^0 - \mathcal{X}^* \right\|_{F(Q)}^2. \tag{19}
\]

**Remark 13.** In Corollary 1, choosing \( S_i = I_{(i,:) \cdot} \in \mathbb{R}^m \) for \( i = 1, 2, \ldots, m \) and \( Q = I \in \mathbb{R}^{n \times n} \), and setting the probability \( p_i = \frac{\|A_{(i,:)\cdot}\|_F^2}{\|A\|_F^2} \) (proportional to the magnitude of \( i \)th horizontal slice of \( A \)) and \( p_i = \frac{1}{m} \) (uniform sampling) lead to

\[
E \left[ \left\| \mathcal{X}^t - \mathcal{X}^* \right\|_{F}^2 | \mathcal{X}^0 \right] \leq 1 - \min_{k=1,2,\ldots,l} \lambda^+_{\min} \left( \hat{A}_{(k)} \hat{A}^H_{(k)} \right)^{\frac{1}{2}} \left\| \mathcal{X}^0 - \mathcal{X}^* \right\|_{F}^2 \tag{20}
\]

and

\[
E \left[ \left\| \mathcal{X}^t - \mathcal{X}^* \right\|_{F}^2 | \mathcal{X}^0 \right] \leq 1 - \min_{k=1,2,\ldots,l} \lambda^+_{\min} \left( \hat{A}_{(k)} \hat{A}^H_{(k)} \right) \left\| \mathcal{X}^0 - \mathcal{X}^* \right\|_{F}^2, \tag{21}
\]

respectively, where (21) is just the result for the TRK method given in theorem 4.1 in Reference 23.

**Remark 14.** For the NTSP and three adaptive TSP methods discussed in Section 3.2, we can also implement them in the Fourier domain, and obtain the corresponding convergence guarantees in a similar way. The details are omitted here.

### 5 | TWO IMPROVED STRATEGIES

The sketching tubal matrix \( S \) appearing in the algorithms proposed in Sections 3 and 4 can be formed as done in References 15,23,37,38, that is, be formed similarly to Gaussian random tubal matrix in Definition 14 or random sampling tubal matrix in Definition 15. However, as explained in Section 1, in this case, \( \hat{S}_{(k)} \), for \( k = 1, 2, \ldots, l \), will be the same, and hence the sketching matrices for all the subsystems \( \hat{A}_{(k)} \hat{S}_{(k)} = \hat{B}_{(k)} \), for \( k = 1, 2, \ldots, l \), are the same. For complex-valued problems, Ma and Molitor\(^{23}\) proposed to select different sketching matrices, that is, select different indices, for different subsystems. However, for real-valued problems considered in this paper, using this strategy directly makes the approximate solution \( \mathcal{X}^{t+1} = \text{ifft} (\mathcal{X}^{t+1}, [1, 3]) \) be no longer real-valued. To tackle this problem, we propose two improved strategies. The first one is based on the following equivalence transformation:

\[
\{ \mathcal{X} \in \mathbb{R}^{n \times p \times d} | A * \mathcal{X} = B \} = \left\{ \mathcal{X} \in \mathbb{R}^{n \times p \times d} \left| \begin{bmatrix} \text{Re} (A) \\ \text{Im} (A) \end{bmatrix} \right. * \mathcal{X} = \begin{bmatrix} \text{Re} (B) \\ \text{Im} (B) \end{bmatrix} \right\}. \nonumber
\]

Putting this equivalence transformation into the TSP method, we can get the first improved algorithm, that is, Algorithm 5. We call it the TSP-I method, where \( I \) stands for the first improved strategy. The convergence of the TSP-I method is provided in Theorem 7.

**Theorem 7.** With the notation in Algorithm 5, assume that \( E[\hat{Z}_{(k)}] \) is Hermitian positive definite with probability 1, where \( \hat{Z}_{(k)} = \hat{Q}_{(k)}^{-\frac{1}{2}} \hat{A}_{(k)}^H S_{(k)} \hat{A}_{(k)} \hat{Q}_{(k)}^{-\frac{1}{2}} \hat{A}_{(k)}^{-1} S_{(k)}^{-1} \hat{A}_{(k)}^{-1} \hat{Q}_{(k)}^{-\frac{1}{2}}, \) for \( k = 1, 2, \ldots, l \). Let \( \mathcal{X}^* \) satisfy \( A * \mathcal{X}^* = B \) and \( \mathcal{X}^t \) be the \( t \)th approximation of \( \mathcal{X}^* \) with initial iterate \( \mathcal{X}^0 \). Then

\[
E \left[ \left\| \mathcal{X}^t - \mathcal{X}^* \right\|_{F(Q)}^2 | \mathcal{X}^0 \right] \leq 1 - \min_{k=1,2,\ldots,l} \left( E[\hat{Z}_{(k)}] \right)^{\frac{1}{2}} \left\| \mathcal{X}^0 - \mathcal{X}^* \right\|_{F(Q)}^2.
\]

**Remark 15.** According to Remark 12, we can conclude that Algorithm 5 is convergent.
Algorithm 5. TSP-I method

Input: $X^0 \in \mathbb{R}^{n \times p}$, $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{m \times q}$

Parameters: fixed distribution $D_k$ over random matrices from $\mathbb{C}^{m \times r}$ for $k = 1, 2, \ldots, l$, T-symmetric T-positive definite tubal matrix $Q \in \mathbb{K}^{m \times n}$

$X^0 \leftarrow \text{fft}(X^0, [\cdot, 3])$, $\hat{A} \leftarrow \text{fft}(A, [\cdot, 3])$, $\hat{B} \leftarrow \text{fft}(B, [\cdot, 3])$, $\hat{Q} \leftarrow \text{fft}(Q, [\cdot, 3])$

for $t = 0, 1, 2, \ldots$

for $k = 1, 2, \ldots, l$

Sample an independent copy $S_k \sim D_k$, and hence the sketch matrix $S_k \in \mathbb{C}^{m \times r}$

$S_k = S_k$

$(A_S)_k = S_k^H \hat{A}_k$, $(B_S)_k = S_k^H \hat{B}_k$

end for

$\hat{A}_S = \text{ifft}(A_S, [\cdot, 3])$, $\hat{B}_S = \text{ifft}(B_S, [\cdot, 3])$

$A_S = \begin{bmatrix} \text{Re}(\hat{A}_S) \\ \text{Im}(\hat{A}_S) \end{bmatrix}$, $B_S = \begin{bmatrix} \text{Re}(\hat{B}_S) \\ \text{Im}(\hat{B}_S) \end{bmatrix}$

$\hat{A}_S = \text{fft}(A_S, [\cdot, 3])$, $\hat{B}_S = \text{fft}(B_S, [\cdot, 3])$

for $k = 1, 2, \ldots, l$

$\tilde{X}_t^{k+1} = \tilde{X}_t^k - \hat{Q}^{-1}_k (A_S)_k^H \left( (A_S)_k \hat{Q}^{-1}_k (A_S)_k^H \right)^\dagger \left( (A_S)_k \hat{X}_t^k - (B_S)_k \right)$

end for

$X^{t+1} \leftarrow \text{ifft}(\tilde{X}^{t+1}, [\cdot, 3])$

Output: last iterate $X^{t+1}$

Next, we give a corollary to consider the case where the random matrices $S_k$, for $k = 1, 2, \ldots, l$, are all complete discrete sampling matrices.

Corollary 2. With the notation in Algorithm 5 and Theorem 7, let $S_k$ be a complete discrete sampling matrix for $k = 1, 2, \ldots, l$ and $S_k = S_k \in \mathbb{C}^{m \times 1}$ with probability $p_k$ for $i = 1, 2, \ldots, q$. Let $X^*$ satisfy $A \ast X^* = B$ and $X^t$ be the $t$th approximation of $X^*$ with initial iterate $X^0$. Then when $p_k = \frac{\|\hat{Q}^{-1}_k \tilde{A}_k^H S_k\|_F^2}{\|\hat{Q}^{-1}_k \tilde{A}_k^H S_k\|_F}$ where $S_k = [S_k, S_k, \ldots, S_k]$, we have

$$E \left[ \left\| X^t - X^* \right\|^2_{F(Q)} \middle| X^0 \right] \leq 1 - \frac{\lambda^*_{\min} \left( S_k^H \tilde{A}_k \hat{Q}^{-1}_k \tilde{A}_k^H S_k \right)}{\| \hat{Q}^{-1}_k \tilde{A}_k^H S_k \|_F^2} \left\| X^0 - X^* \right\|^2_{F(Q)};$$

when $p_k = \frac{1}{q}$, we have

$$E \left[ \left\| X^t - X^* \right\|^2_{F(Q)} \middle| X^0 \right] \leq 1 - \frac{\lambda^*_{\min} \left( S_k^H \tilde{A}_k \hat{Q}^{-1}_k \tilde{A}_k^H S_k \right)}{\| \hat{Q}^{-1}_k \tilde{A}_k^H S_k \|_F^2} \left\| X^0 - X^* \right\|^2_{F(Q)};$$

Proof. This proof is similar to that of Corollary 1, so we omit it.

The other improved strategy is to take the real part of the complex approximate solution directly. The specific algorithm is presented in Algorithm 6 and the method is called TSP-II method, where II stands for the second improved strategy. Based on the fact that $\|X^t - X^*\|^2_{F(Q)} \leq \|\tilde{X}^t - X^*\|^2_{F(Q)}$, where $\tilde{X}^t = \text{ifft}(\tilde{X}^t, [\cdot, 3])$ and $X^t = \text{Re}(\tilde{X}^t)$, we can provide its elementary convergence analysis. The obtained results are exactly the same as the ones in Theorem 7 and Corollary 2. So we omitted them here. However, the TSP-II method has good performance confirmed by numerical experiments in Section 6.

Remark 16. Both the two improved strategies can be applied to the NTSP and three adaptive TSP methods, leading to the NTSP-I, ATSP-MD-I, ATSP-PR-I, ATSP-CS-I, NTSP-II, ATSP-MD-II, ATSP-PR-II, and ATSP-CS-II methods. The details of
In this section, we devote ourselves to the numerical studies of the TSP-type methods presented in previous sections. First, we discuss the efficient implementation techniques and corresponding computational complexities of the TSP-type methods in Section 6.1. Then, we numerically verify the correctness and rationality of the convergence analyses in

**Algorithm 6. TSP-II method**

**Input:** \( X^0 \in \mathbb{R}^{n \times p}, A \in \mathbb{K}^{m \times n}, B \in \mathbb{K}^{m \times p} \)

**Parameters:** fixed distribution \( D_k \) over random matrices from \( \mathbb{C}^{m \times r} \) for \( k = 1, 2, \ldots, l \), T-symmetric T-positive definite tubal matrix \( Q \in \mathbb{K}^{m \times n} \)

\[
X^0 \leftarrow \text{fft}(X^0, [], 3), \tilde{X} \leftarrow \text{fft}(A, [ ], 3), \tilde{B} \leftarrow \text{fft}(B, [ ], 3), \tilde{Q} \leftarrow \text{fft}(Q, [ ], 3)
\]

for \( t = 0, 1, 2, \ldots \)

for \( k = 1, 2, \ldots, l \)

- Sample an independent copy \( S_k \sim D_k \), and hence the sketch matrix \( S_k \in \mathbb{C}^{m \times r} \)

- Compute \( \tilde{G}_k = S_k \left( S_k^H \tilde{A}_k \tilde{G}_k \tilde{A}_k^H S_k \right)^+ S_k^H \)

- \( \tilde{X}^t_{k+1} = \tilde{X}^t_{k} - \tilde{G}_k \hat{A}_k \hat{A}_k^H (\hat{A}_k \hat{X}^t_{k} - \tilde{B}_k) \)

end for

\( \tilde{X}^{t+1} \leftarrow \text{Re} \left( \text{ifft} \left( \tilde{X}^{t+1}, [ ], 3 \right) \right) \)

**Output:** last iterate \( \tilde{X}^{t+1} \)

| TSP-type methods | Same or different sketching matrices for the subsystems (16) |
|------------------|-------------------------------------------------|
|                  | Same transformation | Take real part |
|                  | TSP | TSP-I | TSP-II |
| Samping strategies | Nonadaptive (N)  | Fixed sampling probability distribution | NTSP | NTSP-I | NTSP-II |
|                  | Adaptive (A)  | Max-distance selection rule (MD) | ATSP-MD | ATSP-MD-I | ATSP-MD-II |
|                  |      | Adaptive probability rule (PR) | ATSP-PR | ATSP-PR-I | ATSP-PR-II |
|                  |      | Capped sampling rule (CS) | ATSP-CS | ATSP-CS-I | ATSP-CS-II |

these algorithms are omitted here. For understanding and distinguishing these methods more clearly, we summarize all the TSP-type methods in Table 1. It should be noted that since the three adaptive TSP methods are difficult to implement quickly when combined with the first improved strategy, the ATSP-MD-I, ATSP-PR-I, and ATSP-CS-I methods will not be discussed in numerical experiments.

**Remark 17.** For \( k = 1, 2, \ldots, l \), if we choose \( S_k = e_k \in \mathbb{R}^m \) (the unit coordinate vector in \( \mathbb{R}^m \)) with \( i = 1, 2, \ldots, m \) and \( Q = I \in \mathbb{K}^{m \times n} \) in Algorithms 5 and 6, we can obtain two improved TRK (i.e., TRK-I and TRK-II) methods. The convergence guarantee of the TRK-I method can be obtained according to Corollary 2. That is, when selecting \( k_i \) with probability proportional to the magnitude of row \( k_i \) of \( \hat{A}_k \), we have

\[
E \left[ \frac{\| X^t - X^* \|_F^2}{\| X^0 \|_F^2} \right] \leq 1 - \min_{k=1,2,\ldots,l} \lambda^+_{\min} \left( \frac{\lambda^{(k)}_{\hat{A}_k} \hat{A}_k^H}{\| \hat{A}_k \|_F^2} \right) \frac{\| X^0 - X^* \|_F^2}{\| X^0 \|_F^2}.
\]

An identical result can also be obtained for the TRK-II method.

**6 | NUMERICAL EXPERIMENTS**

In this section, we devote ourselves to the numerical studies of the TSP-type methods presented in previous sections. First, we discuss the efficient implementation techniques and corresponding computational complexities of the TSP-type methods in Section 6.1. Then, we numerically verify the correctness and rationality of the convergence analyses in
Section 6.2. It should be pointed out that in order to compare with the existing methods more intuitively, we only consider the relevant numerical experiments on special cases of the TSP-type methods, namely the TRK-type methods. Specifically, we consider ten algorithms including four nonadaptive TRK methods, that is, NTRKU (nonadaptive TRK method with uniform sampling probabilities), ATRKS (nonadaptive TRK method with probabilities proportional to the magnitude of the horizontal slices of A, NTRK-I (nonadaptive TRK-I method with probabilities proportional to the magnitude of the rows of the frontal slices of A) and NTRK-II, as well as six adaptive methods, that is, NTRKR-MD, NTRKR-PR, ATRKS-CS, ATRKR-MD-II, ATRKR-PR-II and ATRKR-CS-II. Next, we compare the performance of the TRK-type methods for solving the tensor linear system (1) in Section 6.3. Moreover, we also illustrate the superiority of the TRK-type methods over the MRK-type methods through an image deblurring problem. Finally, in Section 6.4, we consider the problem of load balancing for the subsystems in (16).

All experiments terminate once the number of iterations exceeds 100,000 or the relative error, defined as

$$e = \frac{\|X^t - X^*\|_F}{\|X^*\|_F},$$

is less than a threshold $\varepsilon^*$. Specifically, we set $\varepsilon^*$ equal to 10$^{-10}$ (Examples 1, 2, and 6), 10$^{-4}$ (Examples 3 and 4) or 0.005 (Example 5). We choose an initial point $X^0$ of all zeros and all results are average on 10 trails. Furthermore, we set the input parameters in the ATRKS-CS and ATRKR-CS-II methods as $p = u = \left(\frac{1}{m}, \frac{1}{m}, \ldots, \frac{1}{m}\right)^T$ and $\theta = 0.5$. All computations were carried out in MATLAB R2018a on a standard MacBook Pro 2019 with an Intel Core i9 processor and 16 GB memory.

6.1 Implementation tricks and computation complexity

In this subsection, we discuss the computation costs at each iteration of some nonadaptive and adaptive TSP methods. Specifically, the nonadaptive methods include the NTSP, NTSP-I and NTSP-II methods, and the adaptive methods include the ATSP-MD, ATSP-PR, ATSP-CS, ATSP-MD-II, ATSP-PR-II, and ATSP-CS-II methods. Similar to Reference 36, we implement these methods except the NTSP-I one in their corresponding fast versions in the following numerical experiments, for example, the fast versions of the ATSP-PR and ATSP-PR-II methods are given in Algorithms 7 and 8 in the appendix, respectively.

We first consider the computation complexities of the NTSP, ATSP-MD, ATSP-PR, and ATSP-CS methods. Since the difference of the fast versions of these methods mainly lies in how to compute the sampling probabilities, we first present the flops of each step of the four algorithms without the step on sampling:

1. Computing the sketched losses $\{f_i(X^k) : i = 1, 2, \ldots, q\}$ requires $2\tau qlq (l > 1)$ or $(2\tau p - 1)q (l = 1)$ flops if the sketched residuals $\{\hat{R}^k_i : i = 1, 2, \ldots, q\}$ are precomputed.

2. Updating $\hat{X}^k_t$ to $\hat{X}^{k+1}_t$ requires $2\tau npq$ flops when

$$\{\hat{Q}^k_{(k)}^{-1}\hat{A}^H_{(k)}(\tilde{S}_t)_{(k)}(\tilde{C}_t)_{(k)} : i = 1, 2, \ldots, q, k = 1, 2, \ldots, l\}$$

are precomputed.

3. Updating $\{\tilde{R}^k_i : i = 1, 2, \ldots, q\}$ to $\{\tilde{R}^{k+1}_i : i = 1, 2, \ldots, q\}$ requires $2\tau^2 pqlq$ flops if

$$\{(\tilde{C}_t)_{(k)}^H(\tilde{S}_t)_{(k)}^H\hat{A}_{(k)}(\hat{S}_t)_{(k)}(\hat{C}_t)_{(k)} : i = 1, 2, \ldots, q, k = 1, 2, \ldots, l\}$$

are precomputed. Note that for the NTSP method, one only needs to compute the single sketched residual $\hat{R}^k_t$, where $\hat{R}^k_t = (\tilde{C}_t)_{(k)}^H(\tilde{S}_t)_{(k)}^H(\hat{A}_{(k)}\hat{X}^k_t - \hat{B}_{(k)})$ for $k = 1, 2, \ldots, l$. If $(\tilde{C}_t)_{(k)}^H(\tilde{S}_t)_{(k)}^H\hat{A}_{(k)}$ and $(\tilde{C}_t)_{(k)}^H(\tilde{S}_t)_{(k)}^H\hat{B}_{(k)}$ are precomputed for $i = 1, 2, \ldots, q, k = 1, 2, \ldots, l$, computing sketched residual $\hat{R}^k_t$ directly from the iterate $X^k$ costs $2\tau npq$ flops. Hence, when $rq > n$, it is cheaper for the NTSP method to compute the sketched residual $\hat{R}^k_t$ directly than using update formula. Therefore, the nonsampling flops of the NTSP method and the adaptive cases (ATSP-MD, ATSP-PR, ATSP-CS) are $2\tau pl\min(n, rq) + 2\tau npq + 2\tau^2 p + 2\tau pqlq + 2\tau npq (l > 1)$ or $(2\tau^2 p + 2\tau p - 1)q + 2\tau npq (l = 1)$, respectively.

Next, we give the cost of computing the sampling probabilities $p^k$ from the sketched losses $\{f_i(X^k) : i = 1, 2, \ldots, q\}$. It depends on the sampling strategy. Specifically, for the NTSP method, it requires $O(1)$ flops; for the ATSP-MD method,
TABLE 2 The computation costs for the NTSP, ATSP-MD, ATSP-PR and ATSP-CS methods, where \( r \) is the sketch size, \( n \) and \( l \) are the dimension of \( A \), and \( p \) is the size of \( B \)

| Method   | Flops per iteration when \( r > 1 \) | Flops per iteration when \( r = 1 \) |
|----------|----------------------------------|----------------------------------|
| NTSP     | \( 2rp_l \min(n, q) + 2rnlp \) | \( 2p_l \min(n, q) + 2np \) |
| ATSP-MD  | \( (2r^2p_l + 2rpl + 1)q + 2rnp \) if \( l > 1 \) | \( 4plq + 2np \) if \( l > 1 \) |
| ATSP-PR  | \( (2r^2p + 2rp + 1)q + 2rnp \) if \( l > 1 \) | \( 4(4p + 1)q + 2np \) if \( l = 1 \) |
| ATSP-CS  | \( (2r^2p + 2rpl + 6)q + 2rnp \) if \( l > 1 \) | \( 4(4p + 6)q + 2np \) if \( l > 1 \) |

TABLE 3 The computation costs of the NTSP-II, ATSP-MD-II, ATSP-PR-II and ATSP-CS-II methods, where \( r \) is the sketch size, \( n \) and \( l \) are the dimension of \( A \), and \( p \) is the size of \( B \)

| Method   | Flops per iteration when \( r > 1 \) | Flops per iteration when \( r = 1 \) |
|----------|----------------------------------|----------------------------------|
| NTSP-II  | \( O(\tau pln) \) | \( O(\tau pln) \) |
| ATSP-MD-II | \( (2r^2p + 2rp)q + 2rnp \) | \( O(\max(q, np)l) \) |
| ATSP-PR-II | \( (2r^2p + 2rp + 1)q + 2rnp \) | \( (4p + 1)ql + 2np \) |
| ATSP-CS-II | \( (2r^2p + 2rpl + 5)q + 2rnp \) | \( (4p + 5)ql + 2np \) |

TABLE 4 Summary of convergence factors for the TSP-type methods

| Method   | Convergence factor | Source |
|----------|-------------------|--------|
| TSP      | \( 1 - \lambda_{\min}(E(\|bcirc(Z)\|)) \) | Theorem 1, Theorem 6 |
| TSP-I    | \( 1 - \min_{k=1,2,...,l} \lambda_{\min}(E(\|\hat{Z}_{(k)}\|)) \) | Theorem 7 |
| NTSP     | \( 1 - \delta_{\min}(Q, S) \) | Theorem 2 |
| NTSP-I   | \( 1 - \min_{k=1,2,...,l} \lambda_{\min}(E_{k-p}(\|\hat{Z}_{(k)}\|)) \) | Theorem 7 |
| ATSP-MD  | \( 1 - \delta_{\min}(Q, S) \) | Theorem 3 |
| ATSP-PR  | \( 1 - (1 + \frac{1}{q}) \delta_{\min}(Q, S) \) | Theorem 4 |
| ATSP-CS  | \( 1 - \theta \delta_{\min}(Q, S) - (1 - \theta) \delta_{\min}(Q, S) \) | Theorem 5 |

it needs \( q \) flops if \( r > 1 \) and \( \Theta(\log(q)) \) flops if \( r = 1 \); the ATSP-PR method requires approximately \( 2q \) flops on average; the ATSP-CS method requires \( 6q \) flops.

Putting all the costs together, we report the total costs per iteration of the above four methods in Table 2.

In a similar way, we can give the computation costs at each iteration of the NTSP-II, ATSP-MD-II, ATSP-PR-II and ATSP-CS-II methods. The details are omitted here, and the total costs per iteration are reported in Table 3.

For the NTSP-II method, it has no fast implement version. We present separately its complexity of each step as follows: 1. Computing \( \hat{A}_S \) and \( \hat{B}_S \) requires \( \Theta(\tau nml + \tau nl \log l) \) and \( \Theta(\tau pln + \tau pl \log l) \) flops, respectively. If \( \{S_k \in \mathbb{C}^{n \times \tau} : k = 1,2,...,l \} \) are random sampling matrices, then computing \( \hat{A}_S \) and \( \hat{B}_S \) requires \( \Theta(\tau nml + \tau pl \log l) \) flops, respectively.

2. Updating \( \hat{A}^t \) to \( \hat{A}^{t+1} \) requires \( \Theta(n^2 \tau l + np \tau l + n \tau^2 l + \tau^3 l) \) flops when \( \{\hat{Q}^{-1}_{(k)} : k = 1,2,...,l \} \) is precomputed. Note that, if \( Q = I \in \mathbb{R}^{n \times n} \), then updating \( \hat{A}^t \) to \( \hat{A}^{t+1} \) requires \( \Theta(\tau npl + n \tau^2 l + \tau^3 l) \) flops.

62 Numerical verification of convergence analyses

In this subsection, we numerically verify the correctness and rationality of the convergence analyses discussed in the previous sections. For clarity, we first summarize the convergence analyses of the TSP-type methods in Table 4.
Theorem 5 and (22)

Let the entries of actual convergence rates of the TRK-type methods, where the convergence rate is defined as

\[
\text{min}_{\alpha \geq 0} \left\{ \alpha \mid \frac{\alpha}{\alpha} < 1 \right\}
\]

Since the term \(\min_{\alpha \geq 0} \left\{ \alpha \mid \frac{\alpha}{\alpha} < 1 \right\}\) cannot be calculated numerically, we cannot plot the convergence factors of the ATRKS-MD and ATRKS-CS methods. However, from Remarks 7 and 9, we know that the convergence factors of these two methods have their own upper bounds, that is, the convergence factors of the NTRKU and NTRKS methods. Therefore, the rationality of the convergence factors of the ATRKS-MD and ATRKS-CS methods can be inferred by the ones of the NTRKU and NTRKS methods.

**Example 1.** Let the entries of \(A \in \mathbb{R}^{m \times n}_{+}\) and \(X \in \mathbb{R}^{n \times p}_{+}\) be drawn i.i.d. from a standard Gaussian distribution, and the right-hand tubal matrix be \(B = A \odot X \in \mathbb{R}^{l \times p}_{+}\). Specifically, we consider a system with \(m = 500\), \(n = 200\), \(p = 50\), and \(l = 50\). From Figure 1, we can find that the convergence factors are all less than 1 and are indeed the tight upper bounds of the actual convergence rates. This means that these methods are convergent and the convergence factors can well characterize the actual convergence rates. However, it should be pointed out that the convergence factors are only the upper bounds of the actual convergence rates, and their value cannot directly indicate the convergence speed of the corresponding methods. For example, the convergence factor of the ATRKS-PR method is larger than that of the NTRKS method, but its actual convergence rate is smaller compared with the NTRKS method. However, the convergence factors are still necessary because they can theoretically show the convergence of the methods. For the
ATRKR-MD-II, ATRKR-PR-II and ATRKR-CS-II methods, their convergence factors can be the same as the ones of the ATRKR-MD, ATRKR-PR and ATRKR-CS methods, respectively, so we do not plot them in the figure. Numerical results on actual convergence rates show that these three methods indeed converge quickly.

6.3 Examples

We use four numerical experiments to illustrate the performance of the TRK-type methods for solving the tensor linear system (1). Note that we do not consider the precomputational cost, but only the costs spent at each iteration. In each figure, we plot the relative error (i.e., Error) on the vertical axis, starting with 1. For the horizontal axis, we use either the number of the iterations (i.e., Iters) or running time (i.e., Time(s)).

Example 2 (Synthetic data). In this experiment, the setting is the same as that of Example 1. Figure 2 shows that the NTRKR-I method has the best performance in terms of CPU time among the ten methods, and has the fewest iteration steps among the four nonadaptive methods. The other three nonadaptive methods, that is, NTRKU, NTRKS and NTRKR-II, perform similarly except that the NTRKR-II method takes a little more time. The number of iteration steps of each of the six adaptive methods is smaller than that of each of the four nonadaptive methods, and, except for NTRKR-I, the time of each of the six adaptive methods is also less than that of each of the other three nonadaptive methods. In addition, compared with the original adaptive methods, that is, ATRKS-MD, ATRKS-PR and ATRKS-CS, the adaptive methods combined with the second improved strategy, that is, ATRKR-MD-II, ATRKR-PR-II and ATRKR-CS-II, vastly reduced the number of iteration steps and CPU time.

Example 3 (CT data). In this experiment, we evaluate the performance of the TRK-type methods on real world CT data set. The underlying signal $\lambda'$ is a tubal matrix of size $512 \times 512 \times 11$, where each frontal slice is a $512 \times 512$ matrix of the C1-vertebrae. The images for the experiment were obtained from the Laboratory of the Human Anatomy and Embryology, University of Brussels (ULB), Belgium. To set up the tensor linear system, we generate randomly a Gaussian tubal matrix $A \in \mathbb{R}^{512 \times 512 \times 11}$ and form the measurement tubal matrix $B$ by $B = A \ast \lambda'$. The numerical results of this experiment are provided in Figure 3, from which we can see that the performance of the four nonadaptive methods is almost the same except that the NTRKR-I method takes less time. Among the ten methods, the six adaptive methods outperform the four nonadaptive methods in terms of the number of iterations. For running time, they perform better than the NTRKU, NTRKS, and NTRKR-II methods. While, the NTRKR-I method spends less running time than the first two adaptive methods, that is, ATRKS-PR and ATRKR-PR-II. In addition, like Example 2, the adaptive methods combined with the second improved strategy perform better than the original adaptive methods.

Example 4 (Video data). The example illustrates the performance of the TRK-type methods on the video data where the frontal slices of the tubal matrix $\lambda'$ are the first 80 frames from the 1929 film “Finding His Voice”. Each video frame
FIGURE 2   Errors versus iterations (left) and CPU time (right) for the TRK-type methods on synthetic data with $m = 500$, $n = 200$, $p = 50$ and $l = 50$.

FIGURE 3   Errors versus iterations (left) and CPU time (right) for the TRK-type methods on real world CT dataset has $480 \times 368$ pixels. Similar to Example 3, we generate randomly a Gaussian tubal matrix $\mathcal{A} \in \mathbb{K}^{1000 \times 480}$ and form the measurement tubal matrix $\mathcal{B}$ by $\mathcal{B} = \mathcal{A} \ast \mathcal{X}$. From Figure 4, we can find that for the four nonadaptive methods, they are similar in the number of iteration steps, however, in terms of CPU time, the NTRKR-I method is the fastest one and the NTRKR-II method is faster than the NTRKU and NTRKS methods. Among the ten methods, the six adaptive methods outperform the four nonadaptive methods in terms of the number of iterations. For running time, they perform better than the NTRKU, NTRKS, and NTRKR-II methods. While, only two adaptive methods, that is, ATRKS-MD-II and ATRKR-CS-II, are faster than the NTRKR-I method. In addition, as in the previous examples, combining with the second improved strategy can indeed improve the adaptive methods in terms of the number of iteration steps and CPU time.

Example 5  (Image deblurring). This example considers an image sequence $\{\hat{X}_j\}_{j=1}^{27}$ from a 3D MRI image data set `mri` in MATLAB, which has 27 slices with dimensions $128 \times 128$. Assume that each image is degraded by a Gaussian convolution kernel $\hat{H}$ of size $5 \times 5$ with standard deviation 2. By the construction, we can obtain the image deblurring problem as follows:

$$H \otimes X_j = Y_j, \quad \text{for } j = 1, 2, \ldots, 27,$$

(23)
where \( \{X_j\}_{j=1}^{27} \) and \( H \) are extended by padding \( \{\hat{X}_j\}_{j=1}^{27} \) and \( \hat{H} \) with the zeros respectively, and they are of size \( 132 \times 132 \); \( Y_j \in \mathbb{R}^{132 \times 132} \), for \( j = 1, 2, \ldots, 27 \), are the observed blurry images; and \( \circ \) is the 2D convolution. Let \( \mathcal{A} \in \mathbb{R}^{132 \times 132 \times 132} \) be a tubal matrix whose \( k \)-th frontal slice is the circulant matrix generated by the \( k \)-th column of \( H \), that is, \( \mathcal{A}_{(k)} = \text{circ} (H(:,k)) \) for \( k = 1, 2, \ldots, 132 \). Thus, we can equivalently convert the problem (23) to

\[
\text{bvec}(\mathcal{A}) \text{vec}(X_j^T) = \text{vec}(Y_j^T), \quad \text{for } j = 1, 2, \ldots, 27,
\]

where \( \text{vec}(\cdot) \) is a vectorization operator. Further, we can represent them as a matrix linear system

\[
\text{bvec}(\mathcal{A}) \begin{bmatrix}
\text{vec}(X_1^T) \\
\text{vec}(X_2^T) \\
\vdots \\
\text{vec}(X_{27}^T)
\end{bmatrix} = \begin{bmatrix}
\text{vec}(Y_1^T) \\
\text{vec}(Y_2^T) \\
\vdots \\
\text{vec}(Y_{27}^T)
\end{bmatrix}.
\]  

(24)

Setting \( X_{(i,j,k)} = (X_j)_{(k,i)} \) and \( B_{(i,j,k)} = (Y_j)_{(k,i)} \) for \( i = 1, 2, \ldots, 132, j = 1, 2, \ldots, 27 \) and \( k = 1, 2, \ldots, 132 \), respectively, the above matrix linear system can be rewritten as

\[
\text{bvec}(\mathcal{A}) \text{unfold}(\mathcal{X}) = \text{unfold}(B).
\]  

(25)

Using the definition of t-product, we finally find that the problem (23) can be equivalently transformed into the following tensor linear system

\[
\mathcal{A} * \mathcal{X} = B.
\]  

(26)

Note that the problem (23) can be written as the matrix linear system (24) or (25). Hence, for this example, we also investigate the performance of the MRK-type methods\(^{35,36} \) and compare them with the TRK-type methods. Specifically, we consider two nonadaptive MRK methods, that is, NMRKU (nonadaptive MRK method with uniform sampling probabilities) and NMRKR (nonadaptive MRK method with probabilities proportional to the magnitude of the rows of coefficient matrix), as well as three adaptive MRK methods, that is, AMRK-MD (adaptive MRK method with max-distance selection rule), AMRK-PR (adaptive MRK method with adaptive probabilities rule) and AMRK-CS (adaptive MRK method with capped sampling rule).

We now compare the empirical performance of the MRK-type and TRK-type methods on linear systems (25) and (26), respectively. As shown in Figure 5, for the MRK-type methods, though the three adaptive methods, that is, AMRK-MD, AMRK-PR and AMRK-CS, converge faster than the two nonadaptive methods, that is, NMRKU and NMRKR, they all converge very slowly actually. For the TRK-type methods, they outperform the MRK-type methods in terms of iterations and CPU time. From the enlarged small graph in Figure 5, for the four nonadaptive TSP methods, the NTRKU, NTRKS, and NTRKR-II methods have similar numerical performance, while the NTRKR-I method is better than the previous
three methods in terms of the number of iteration steps and computing time. Except for the NTRKR-I method which is competitive with the six adaptive TSP methods, the other three nonadaptive TSP methods have considerably larger iteration numbers and more CPU time than the six adaptive TSP methods. For the six adaptive TSP methods, it can be found that the experimental performance is consistent with the previous numerical examples, that is, the combination with the second improved strategy can indeed make the adaptive TSP methods perform better. In addition, the first slice of the clean image sequence, its corresponding blurry observation and the images recovered from the MRK-type and TRK-type methods are shown in Figure 6.
FIGURE 7 Errors versus iterations (left) and CPU time (right) for six adaptive TRK methods with or without the subsystem stopping criteria. Here, “-O” stands for the methods with the subsystem stopping criteria.

6.4 Test of load balancing

Considering that the subsystems in (16) are independent and may converge at very different rates, we consider the problem of load balancing for these subsystems in this subsection. Specifically, we will provide a stopping criterion for each subsystem.

From
\[ \epsilon_k = \frac{\|X^k - X^*\|_F}{\|X^*\|_F} = \left( \frac{1}{l} \sum_{k=1}^{l} \frac{\|\hat{X}^k(k) - \hat{X}^*\|_F^2}{\|\hat{X}^*\|_F^2} \right)^{\frac{1}{2}} \leq \left( \sum_{k=1}^{l} \frac{\|\hat{X}^k(k) - \hat{X}^*\|_F^2}{\|\hat{X}^*\|_F^2} \right)^{\frac{1}{2}} \leq \sum_{k=1}^{l} \frac{\|\hat{X}^k(k) - \hat{X}^*\|_F^2}{\|\hat{X}^*\|_F^2} = \sum_{k=1}^{l} \epsilon_k, \]

where \( \epsilon_k \) for \( k = 1, 2, \ldots, l \) are the relative errors of the solutions of the subsystems, we can see that if all \( \epsilon_k \) are less than \( \frac{1}{l} \epsilon^* \), then \( \epsilon \) is less than the global threshold \( \epsilon^* \). Hence, the stopping criterion for each subsystem can be set to be \( \frac{1}{l} \epsilon^* \), that is, we stop to update the solution of a subsystem once its relative error is less than \( \frac{1}{l} \epsilon^* \). A numerical example is given below to compare the performance of the adaptive TRK methods with or without the subsystem stopping criteria.

In addition, for real world problems, we can use other errors such as \( \frac{\|X^k(k) - X^*\|_F}{\|X^*\|_F} \) and \( \frac{\|X^k(k) - X^*\|_F}{\|X^*\|_F} \) to set the stopping criteria for each subsystem.

Example 6. In this numerical example, \( A \in K^{m \times n} \) is a tabular matrix satisfying \( A = \text{ift}(\hat{A}, [\;], 3) \) and \( \hat{A}(k) \) for \( k = 1, 2, \ldots, l \) have the specified condition numbers. We generate a matrix \( A \in C^{m \times n} \) with the specified condition number \( \kappa \) in the
following way. First, generate an $m \times n$ Gaussian matrix $B$ and compute its SVD $B = U\Sigma V^H$, where $U \in \mathbb{C}^{m \times n}$, $\Sigma \in \mathbb{C}^{n \times n}$, and $V \in \mathbb{C}^{n \times n}$. Then, generate an $n \times n$ diagonal matrix $\Sigma$ such that its diagonal elements are linearly distributed from 1 to $1/\kappa$. Finally, we get the matrix $A$ as $A = U\Sigma V^H$, which has the specified condition number $\kappa$. The entries of $X \in \mathbb{R}^{n \times p \times l}$ are drawn i.i.d. from a standard Gaussian distribution, and the right-hand tubal matrix is set to be $B = A \ast X \in \mathbb{R}^{m \times p \times l}$. Specifically, we consider a system with $m = 80$, $n = 30$, $p = 10$, and $l = 20$, and for $\hat{A}$, we set

$$\hat{A}(k) = \begin{cases} \text{a matrix has the specified condition number } \kappa = 30 & \text{for } k = 1, \ldots, 3, \\ \text{a matrix has the specified condition number } \kappa = 1 & \text{for } k = 4, \ldots, 11, \\ \text{conj} \left( \hat{A}_{(l+2-k)} \right) & \text{for } k = 12, \ldots, 20. \end{cases}$$

Here, conj$(\cdot)$ is the conjugate operator. It can be seen from Figure 7 that, compared with the methods without the subsystem stopping criteria, the number of iterations for the modified methods is nearly unchanged, but the running time is significantly reduced. So, we can conclude that considering the problem of load balancing can indeed reduce the computation cost without damaging the accuracy.

7 | CONCLUSION

In this article, we propose the TSP method and its adaptive variants for tensor linear systems. We also discuss their Fourier domain versions. Two strategies used to improve the sketching or sampling techniques for real tensor linear systems are provided. Extensive numerical results including the ones from the CT signal recovery and image deblurring problems show that the adaptive methods can indeed accelerate the nonadaptive ones and the two improved strategies are indeed effective for real linear systems.

ACKNOWLEDGMENTS

The authors would like to thank the editor and the anonymous reviewers for their detailed comments and helpful suggestions which helped considerably to improve the quality of the paper.

CONFLICT OF INTEREST

This study does not have any conflicts to disclose.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ENDNOTES

*In this case, the distribution $\mathcal{D}$ is a discrete distribution and $\mathbb{P}(S = I(i, :, 1)) = p_i$ with $i = 1, 2, \ldots, m$, where $p_i = \frac{1}{m}$ or $p_i = \frac{||A_{i1}||_1}{L_{A,i1}}$.

†The setting of these six acronyms is similar to what we have done before. Specifically, ATRKS-PR stands for the adaptive TRK method with adaptive probabilities rule, and the probabilities are proportional to the magnitude of the horizontal slices of the sketched residual tensor $R$, where $R = \text{ifft}(\hat{R}, [1, 3])$ and $\hat{R}_{(i, :, :)} = \hat{R}_i$ for $i = 1, 2, \ldots, m$ with $\hat{R}_i$, being as in Section 6.1. ATRKR-PR-II stands for the adaptive TRK-II method with adaptive probabilities rule, and the probabilities are proportional to the magnitude of the rows of the frontal slices of the sketched residuals $\hat{R}_i$. The other four acronyms have similar explanations.

ORCID

Hanyu Li https://orcid.org/0000-0001-5033-578X

REFERENCES

1. Kilmer ME, Martin CD. Factorization strategies for third-order tensors. Linear Algebra Appl. 2011;435(3):641–58.
2. Kilmer ME, Braman K, Hao N, Hoover RC. Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging. SIAM J Matrix Anal Appl. 2013;34(1):148–72.
3. Soltani S, Kilmer ME, Hansen PC. A tensor-based dictionary learning approach to tomographic image reconstruction. BIT. 2016;56(4):1425–54.
4. Newman E, Kilmer ME. Nonnegative tensor patch dictionary approaches for image compression and deblurring applications. SIAM J Imaging Sci. 2020;13(3):1084–112.
5. Ahn CH, Jeong BS, Lee SY. Efficient hybrid finite element-boundary element method for 3-dimensional open-boundary field problems. IEEE Trans Magn. 1991;27:4069–72.
6. Alavikia B, Ramahi QM. Electromagnetic scattering from cylindrical objects above a conductive surface using a hybrid finite-element–surface integral equation method. JOSA A. 2011;28(12):2510–8.
7. Czuprynski KD, Fahline JB, Shontz SM. Parallel boundary element solutions of block circulant linear systems for acoustic radiation problems with rotationally symmetric boundary surfaces. Noise control and acoustics division conference. New York: American Society of Mechanical Engineers; 2012. p. 147–58.
8. Braman K. Third-order tensors as linear operators on a space of matrices. Linear Algebra Appl. 2010;433(7):1241–53.
9. Jin HW, Bai MR, Benitez J, Liu XI. The generalized inverses of tensors and an application to linear models. Comput Math Appl. 2017;74(3):385–97.
10. Lund K. The tensor T-function: a definition for functions of third-order tensors. Numer Linear Algebra Appl. 2020;27(3):e2288.
11. Miao Y, Qi LQ, Wei YM. Generalized tensor function via the tensor singular value decomposition based on the T-product. Linear Algebra Appl. 2020;590:258–303.
12. Miao Y, Qi LQ, Wei YM. T-Jordan canonical form and T-Drazin inverse based on the T-product. Commun Appl Math Comput. 2021;3(2):201–20.
13. Zheng MM, Huang ZH, Wang Y. T-positive semidefiniteness of third-order symmetric tensors and T-semidefinite programming. Comput Optim Appl. 2021;78(1):239–72.
14. Qi LQ, Zhang XZ. T-quadratic forms and spectral analysis of T-symmetric tensors. 2021; arXiv preprint arXiv:2101.10820.
15. Tarzanagh DA, Michailidis G. Fast randomized algorithms for T-product based tensor operations and decompositions with applications to imaging data. SIAM J Imaging Sci. 2018;11(4):2629–64.
16. Xie Y, Tao DC, Zhang WS, Liu Y, Zhang L, Qu YY. On unifying multi-view self-representations for clustering by tensor multi-rank minimization. Int J Comput Vis. 2018;126(11):1157–79.
17. Yin M, Gao JB, Xie SL, Guo Y. Multiview subspace clustering via torsorial T-product representation. IEEE Trans Neural Netw Learn Syst. 2018;30(3):581–64.
18. Zhang CY, Hu WR, Jin TY, Mei ZL. Nonlocal image denoising via adaptive tensor nuclear norm minimization. Neural Comput Appl. 2018;29(1):3–19.
19. Semerci O, Hao N, Kilmer ME, Miller EL. Tensor-based formulation and nuclear norm regularization for multienergy computed tomography. IEEE Trans Image Process. 2014;23(4):1678–93.
20. Zhang ZM, Ely G, Aeron S, Hao N, Kilmer M. Novel methods for multilinear data completion and de-noising based on tensor-SVD. Proc IEEE Conf Computer Vision and Pattern Recognition. Columbus OH: IEEE; 2014. p. 3842–9.
21. Zhang ZM, Aeron S. Exact tensor completion using t-SVD. IEEE Trans Signal Process. 2016;65(6):1511–26.
22. Zhou P, Lu CY, Lin ZC, Zhang C. Tensor factorization for low-rank tensor completion. IEEE Trans Image Process. 2017;27(3):1152–63.
23. Ma A, Molitor D. Randomized Kaczmarz for tensor linear systems. BIT. 2021;62:1–24.
24. Kaczmarz S. Angenäherte Auflösung von systemen linearer glei-chungen. Bull Int Acad Pol Sic Let, Cl Sci Math Nat. 1937;35:355–7.
25. Strohmer T, Vershynin R. A randomized Kaczmarz algorithm with exponential convergence. J Fourier Anal Appl. 2009;15(2):262–78.
26. Chen XM, Qin J. Regularized Kaczmarz algorithms for tensor recovery. SIAM J Imaging Sci. 2021;14(4):1439–71.
27. Du K, Sun XH. Randomized regularized extended Kaczmarz algorithms for tensor recovery. 2021; arXiv preprint, arXiv:2112.08566.
28. Needell D, Tropp JA. Paved with good intentions: analysis of a randomized block Kaczmarz method. Linear Algebra Appl. 2014;441:199–221.
29. Liu J, Wright SJ. An accelerated randomized Kaczmarz algorithm. Math Comput. 2016;85(297):153–78.
30. Bonet and joint CT-scan data. https://isbweb.org/data/vsj/
APPENDIX A. PROOFS FOR THEORETICAL RESULTS

Proof of Theorem 1. Combining (4) and the fact that $A \ast \lambda^\ast = B$, we have

$$\lambda^{t+1} - \lambda^\ast = (I - Q^{-1} \ast W) \ast (\lambda^t - \lambda^\ast).$$  \hspace{1cm} (A1)

Multiplying both sides of (A1) by $Q^\frac{1}{2}$, we obtain

$$Q^\frac{1}{2} \ast (\lambda^{t+1} - \lambda^\ast) = Q^\frac{1}{2} \ast (I - Q^{-1} \ast W) \ast Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast) = (I - Z) \ast Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast).$$

Let $\Gamma^t = Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast)$. Thus, the above equation can be rewritten as $\Gamma^{t+1} = (I - Z) \ast \Gamma^t$. Applying the Frobenius norm to its two sides, we get

$$\|\Gamma^{t+1}\|^2_F = \|I - Z\|_F \ast \|\Gamma^t\|^2_F = \|\Gamma^t\|^2_F - \|Z \ast \Gamma^t\|^2_F,$$

where the second equality is from the Pythagorean theorem. By taking expectation conditioned on $\lambda^t$, we have

$$E[\|\Gamma^{t+1}\|^2_F | \lambda^t] = \|\Gamma^t\|^2_F - E[\|Z \ast \Gamma^t\|^2_F].$$ (A3)

Note that

$$E[\|Z \ast \Gamma^t\|^2_F] = E[\|bcirc(Z)\text{unfold}(\Gamma^t)\|^2_F] = \sum_{j=1}^p \langle E[bcirc(Z)]\text{unfold}(\Gamma^t)_{(:,j)}, \text{unfold}(\Gamma^t_{(:,j)}) \rangle$$

$$\geq \lambda_{\min}(E[bcirc(Z)]) \sum_{j=1}^p \|\text{unfold}(\Gamma^t_{(:,j)})\|_2^2 = \lambda_{\min}(E[bcirc(Z)]) \|\Gamma^t\|^2_F,$$

where the inequality follows from the assumption that $E[Z]$ is T-symmetric T-positive definite with probability 1. Therefore,

$$E[\|\Gamma^{t+1}\|^2_F | \lambda^t] \leq (1 - \lambda_{\min}(E[bcirc(Z)])) \|\Gamma^t\|^2_F,$$

that is

$$E[\|\lambda^{t+1} - \lambda^\ast\|^2_{F(Q)} | \lambda^t] \leq (1 - \lambda_{\min}(E[bcirc(Z)])) \|\lambda^t - \lambda^\ast\|^2_{F(Q)}.$$ (A4)

Taking expectation again and unrolling the recurrence give the desired result. □

Proof of Lemma 4. Since $\lambda^0 \in \text{Colsp}(Q^{-1} \ast A^T)$, we have $\lambda^t - \lambda^\ast \in \text{Colsp}(Q^{-1} \ast A^T)$ and consequently

$$\frac{\max_{i=1,2,\ldots,q} f_i(\lambda^t)}{\|\lambda^t - \lambda^\ast\|^2_{F(Q)}} = \frac{\max_{i=1,2,\ldots,q} \|Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast)\|_{F(Q)}^2}{\|\lambda^t - \lambda^\ast\|^2_{F(Q)}} = \frac{\max_{i=1,2,\ldots,q} \sum_{j=1}^p \|Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast)_{(:,j)}\|_{Z_j}^2}{\|\lambda^t - \lambda^\ast\|^2_{F(Q)}}$$

$$= \frac{\sum_{j=1}^p \|Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast)_{(:,j)}\|_{Z_j}^2}{\sum_{j=1}^p \|\lambda^t - \lambda^\ast\|_{Z_j}^2} \geq \max_{i=1,2,\ldots,q} \min_{j=1,2,\ldots,p} \frac{\|Q^\frac{1}{2} \ast (\lambda^t - \lambda^\ast)_{(:,j)}\|_{Z_j}^2}{\|\lambda^t - \lambda^\ast\|_{Z_j}^2}$$

$$\geq \frac{\min_{V \in \text{Range}(Q^{-1} \ast A^T)} \max_{i=1,2,\ldots,q} \|Q^\frac{1}{2} \ast V\|_{Z_j}^2}{\|V\|_{Q}^2} = \delta_{\lambda^t}(Q, S), \ \forall t.$$
Similarly, we have
\[
\frac{E_{i=p}[f_i(\mathbf{X}^t)]}{\mathbb{E} \left[ \left\| \mathbf{X}^t - \mathbf{X}^* \right\|_F^2 \right]} = \frac{\sum_{j=1}^{p} \sum_{i=1}^{p} \mathbb{E}_{i=p}[f_i(\mathbf{X}^t)]}{\sum_{j=1}^{p} \mathbb{E}_{i=p}[f_i(\mathbf{X}^t)]} \left( \frac{\mathbb{E}_{i=p}[f_i(\mathbf{X}^t)]}{\sum_{j=1}^{p} \mathbb{E}_{i=p}[f_i(\mathbf{X}^t)]} \right)
\]

Then, the desired results hold.

**Proof of Lemma 5.** Using $E_{i=p}[Z_i] = Q^{-\frac{1}{2}} \otimes A^T \star E_{i=p}[G_i] \ast A \ast Q^{-\frac{1}{2}}$, as well as the fact that $E_{i=p}[Z_i]$ is T-symmetric T-positive definite with probability 1, we obtain

\[
\text{Range}(Q^{-\frac{1}{2}} \ast A^T) = \text{Range}(E_{i=p}[Z_i]) = \mathbb{H}_n.
\]

Hence,

\[
\delta_p^2(Q, S) = \min_{\mathbf{V} \in \text{Range}(Q^{-\frac{1}{2}} + A^T)} \frac{\left\| \mathbf{Q}_i^\frac{1}{2} \ast \mathbf{V} \right\|^2_{E_{i=p}[Z_i]}}{\left\| \mathbf{V} \right\|^2_F} = \min_{\mathbf{V} \in \text{Range}(E_{i=p}[Z_i])} \frac{\left\| \mathbf{Q}_i^\frac{1}{2} \ast \mathbf{V} \right\|^2_{E_{i=p}[Z_i]}}{\left\| \mathbf{V} \right\|^2_F} = \lambda_{\min}(\text{birc}(E_{i=p}[Z_i])) = \lambda_{\min}(E_{i=p}[birc(Z_i)]) > 0,
\]

and

\[
\delta_{\infty}^2(Q, S) = \min_{\mathbf{V} \in \text{Range}(Q^{-\frac{1}{2}} + A^T)} \frac{\left\| \mathbf{Q}_i^\frac{1}{2} \ast \mathbf{V} \right\|^2_{E_{i=p}[Z_i]}}{\left\| \mathbf{V} \right\|^2_F} = \min_{\mathbf{V} \in \text{Range}(E_{i=p}[Z_i])} \frac{\left\| \mathbf{Q}_i^\frac{1}{2} \ast \mathbf{V} \right\|^2_{E_{i=p}[Z_i]}}{\left\| \mathbf{V} \right\|^2_F} = \delta_{\infty}^2(Q, S).
\]

Finally, since the tubal matrix $Z_i$ is an orthogonal projector, we have

\[
\delta_{\infty}^2(Q, S) \leq \min_{i=1,2,\ldots,q} \max_{j=1,2,\ldots,q} \frac{\left\| \mathbf{Q}_i^\frac{1}{2} \ast \mathbf{V} \right\|^2}{\left\| \mathbf{V} \right\|^2} = \max_{i=1,2,\ldots,q} \frac{\left\| Z_i \ast Q_i^\frac{1}{2} \ast \mathbf{V} \right\|^2}{\left\| Q_i^\frac{1}{2} \ast \mathbf{V} \right\|^2} \leq \max_{i=1,2,\ldots,q} \frac{\left\| Q_i^\frac{1}{2} \ast \mathbf{V} \right\|^2}{\left\| Q_i^\frac{1}{2} \ast \mathbf{V} \right\|^2} = 1.
\]

Then, the desired results hold.

**Proof of Theorem 2.** From (8) and (14), we have

\[
E[\left\| \mathbf{X}^{t+1} - \mathbf{X}^* \right\|^2_{F(Q)}] = \left\| \mathbf{X}^t - \mathbf{X}^* \right\|^2_{F(Q)} - E_{i=p}[f_i(\mathbf{X}^t)] \leq (1 - \delta_p^2(Q, S)) \left\| \mathbf{X}^t - \mathbf{X}^* \right\|^2_{F(Q)}.
\]

Taking the full expectation and unrolling the recurrence, we arrive at this theorem.

**Proof of Theorem 3.** In view of (7) and (13), we have

\[
\left\| \mathbf{X}^{t+1} - \mathbf{X}^* \right\|^2_{F(Q)} = \left\| \mathbf{X}^t - \mathbf{X}^* \right\|^2_{F(Q)} - \max_{i=1,2,\ldots,q} f_i(\mathbf{X}^t) \leq (1 - \delta_{\infty}^2(Q, S)) \left\| \mathbf{X}^t - \mathbf{X}^* \right\|^2_{F(Q)}.
\]
Unrolling the recurrence gives this theorem.

**Proof of Theorem 4.** First note that, for \( i \sim u \), we have

\[
\operatorname{Var}_{i \sim u} [f_i(\mathbf{X}^t)] = \mathbf{E}_{i \sim u} [f_i(\mathbf{X}^t)^2] - \mathbf{E}_{i \sim u} [f_i(\mathbf{X}^t)]^2 = \frac{1}{q} \sum_{j=1}^{q} (f_j(\mathbf{X}^t))^2 - \frac{1}{q^2} \left( \sum_{j=1}^{q} f_j(\mathbf{X}^t) \right)^2. \tag{A5}
\]

Then from (14), (A5) and the definition of \( p^i \) in the second case of Algorithm 3, we get

\[
\mathbf{E}_{i \sim p^i} [f_i(\mathbf{X}^t)] = \frac{1}{q} \sum_{j=1}^{q} (f_j(\mathbf{X}^t))^2 = \frac{1}{q} \left( \frac{q \operatorname{Var}_{i \sim u} [f_i(\mathbf{X}^t)] + \frac{1}{q} \left( \sum_{j=1}^{q} f_j(\mathbf{X}^t) \right)^2 \right) \\
= \left( 1 + q^2 \operatorname{Var}_{i \sim u} \left[ \frac{f_i(\mathbf{X}^t)}{\sum_{j=1}^{q} f_j(\mathbf{X}^t)} \right] \right) \frac{1}{q} \sum_{j=1}^{q} f_j(\mathbf{X}^t) = (1 + q^2 \operatorname{Var}_{i \sim u} [p^i]) \mathbf{E}_{i \sim u} [f_i(\mathbf{X}^t)] \\
\geq (1 + q^2 \operatorname{Var}_{i \sim u} [p^i]) \delta^2_u(Q, S) \| \mathbf{X}^t - \mathbf{X}^* \|^2_{F(Q)}. \tag{A6}
\]

Thus, substituting (A6) into (8), we obtain

\[
\mathbf{E}[\| \mathbf{X}^{t+1} - \mathbf{X}^* \|^2_{F(Q)} \| \mathbf{X}^t] = \| \mathbf{X}^t - \mathbf{X}^* \|^2_{F(Q)} - \mathbf{E}_{i \sim p^i} [f_i(\mathbf{X}^t)] \\
\leq (1 - (1 + q^2 \operatorname{Var}_{i \sim u} [p^i]) \delta^2_u(Q, S)) \| \mathbf{X}^t - \mathbf{X}^* \|^2_{F(Q)}. \tag{A7}
\]

Next, we further give a lower bound for \( \operatorname{Var}_{i \sim u} [p^i] \). Since

\[
\mathcal{Z}_{\beta} \ast Q^\frac{1}{2} \ast (\mathbf{X}^{t+1} - \mathbf{X}^*) = \mathcal{Z}_{\beta} \ast Q^\frac{1}{2} \ast (\mathbf{X}^t - Q^{-\frac{1}{2}} \ast \mathcal{Z}_{\beta} \ast Q^\frac{1}{2} \ast (\mathbf{X}^t - \mathbf{X}^*)) \\
= \mathcal{Z}_{\beta} \ast Q^\frac{1}{2} \ast (\mathbf{X}^t - \mathbf{X}^*) - \mathcal{Z}_{\beta} \ast Q^\frac{1}{2} \ast (\mathbf{X}^t - \mathbf{X}^*) = O,
\]

where \( O \) is the zero tubal matrix, it follows that

\[
f_v(\mathbf{X}^{t+1}) = \| Q^\frac{1}{2} \ast (\mathbf{X}^{t+1} - \mathbf{X}^*) \|^2_{F(\mathcal{Z}_{\beta})} = \| \text{bcirc}(Q^\frac{1}{2}) \text{unfold}(\mathbf{X}^{t+1} - \mathbf{X}^*) \|^2_{F(\text{bcirc}(\mathcal{Z}_{\beta}))} \\
= \sum_{j=1}^{p} \| \text{bcirc}(Q^\frac{1}{2}) \text{unfold}(\mathbf{X}^{t+1} - \mathbf{X}^*)_{(:,j)} \|^2_{F(\text{bcirc}(\mathcal{Z}_{\beta}))} \\
= \sum_{j=1}^{p} \| \text{unfold}(\mathcal{Z}_{\beta} \ast Q^\frac{1}{2} \ast (\mathbf{X}^{t+1} - \mathbf{X}^*))_{(:,j)} ) \|_{F(\text{bcirc}(\mathcal{Z}_{\beta}))} = 0, \forall t \geq 0,
\]

which implies \( p^i_{t+1} = 0 \), and hence

\[
\operatorname{Var}_{i \sim u} [p^i_{t+1}] = \frac{1}{q} \sum_{j=1}^{q} \left( p^i_{t+1} - \frac{1}{q} \sum_{j=1}^{q} p^j_{t+1} \right)^2 = \frac{1}{q} \sum_{j=1}^{q} \left( p^i_{t+1} - \frac{1}{q} \right)^2 \geq \frac{1}{q} \left( p^i_{t+1} - \frac{1}{q} \right)^2 = \frac{1}{q^3}. \tag{A8}
\]

Therefore, plugging (A8) into (A7), we get

\[
\mathbf{E}[\| \mathbf{X}^{t+1} - \mathbf{X}^* \|^2_{F(Q)} \| \mathbf{X}^t] \leq (1 - (1 + \frac{1}{q}) \delta^2_u(Q, S)) \| \mathbf{X}^t - \mathbf{X}^* \|^2_{F(Q)}.
\]

Taking the expectation and unrolling the recursion give this theorem.

**Proof of Theorem 5.** Due to

\[
\max_{j=1,2,\ldots,q} f_j(\mathbf{X}^t) \geq \mathbf{E}_{j \sim p} [f_j(\mathbf{X}^t)],
\]
we know that $\mathcal{W}$, defined in (9) is not empty and arg max $\gamma_j = 1, 2, \ldots, q f_j(\mathcal{X}) \in \mathcal{W}$. From the definition of $p^*_i$ in (10), we have $p^*_i = 0$ for all $i \notin \mathcal{W}_k$, and thus

$$E[\|\mathcal{X}^{t+1} - \mathcal{X}^*\|^2_{F(Q)} | \mathcal{X}^t] = \|\mathcal{X}^t - \mathcal{X}^*\|^2_{F(Q)} - E_{i \sim p^*} [f_i(\mathcal{X}^t)] = \|\mathcal{X}^t - \mathcal{X}^*\|^2_{F(Q)} - \sum_{i \in \mathcal{W}} p^*_i f_i(\mathcal{X}^t).$$

Note that

$$\sum_{i \in \mathcal{W}} f_i(\mathcal{X}^t)p^*_i \geq \sum_{i \in \mathcal{W}} \left( \theta \max_{j=1, 2, \ldots, q} f_j(\mathcal{X}^t) + (1 - \theta) E_{j \sim p} [f_j(\mathcal{X}^t)] \right) p^*_i = \theta \max_{j=1, 2, \ldots, q} f_j(\mathcal{X}^t) + (1 - \theta) E_{j \sim p} [f_j(\mathcal{X}^t)]$$

$$\geq (\theta \delta_{\infty}(Q, S) + (1 - \theta) \delta_p(Q, S)) \|\mathcal{X}^t - \mathcal{X}^*\|^2_{F(Q)}.$$

Hence,

$$E[\|\mathcal{X}^{t+1} - \mathcal{X}^*\|^2_{F(Q)} | \mathcal{X}^t] \leq (1 - \theta \delta_{\infty}(Q, S) - (1 - \theta) \delta_p(Q, S)) \|\mathcal{X}^t - \mathcal{X}^*\|^2_{F(Q)}.$$

Taking expectation and unrolling the recursion give this theorem.

Proof of Theorem 6. From the assumption that $E[\text{bdia}(\hat{Z})]$ is Hermitian positive definite with probability 1 and Proposition 1, we have that $E[\hat{Z}]$ is T-symmetric T-positive definite with probability 1. Moreover,

$$\lambda_{\text{min}}(E[\text{bcirc}(\hat{Z})]) = \lambda_{\text{min}}((F_i \otimes I_n)E[\text{bcirc}(\hat{Z})](F_i^H \otimes I_n)) = \lambda_{\text{min}}(E[(F_i \otimes I_n)\text{bcirc}(\hat{Z})(F_i^H \otimes I_n)])$$

$$= \lambda_{\text{min}}(E[\text{bdia}(\hat{Z})]) = \min_{k=1, 2, \ldots, J} \lambda_{\text{min}}(E[\hat{Z}_{(k)}]).$$

Then, we conclude from (A4) that

$$E \left[ \|\mathcal{X}^{t+1} - \mathcal{X}^*\|^2_{F(Q)} | \mathcal{X}^t \right] \leq (1 - \lambda_{\text{min}}(E[\text{bcirc}(\hat{Z})])) \|\mathcal{X}^t - \mathcal{X}^*\|^2_{F(Q)}$$

$$= \left( 1 - \min_{k=1, 2, \ldots, J} \lambda_{\text{min}}(E[\hat{Z}_{(k)}]) \right) \|\mathcal{X}^t - \mathcal{X}^*\|^2_{F(Q)}.$$

Taking expectation again and unrolling the recurrence give the result.

Proof of Corollary 1. Since $S$ satisfies that $\hat{S}_{(k)}$, for $k = 1, 2, \ldots, l$ are all complete discrete sampling matrices, we can get that $E[\text{bdia}(\hat{Z})]$ is Hermitian positive definite, which implies that such sketching tubal matrix satisfies the assumptions in Theorem 6. Let

$$D_k = \text{diag} \left( \sqrt{p_1} (\hat{S}_{1(k)}^H \hat{A}_{(k)} \hat{Q}_{1(k)} \hat{A}_{(k)}^H \hat{S}_{1(k)}), \ldots, p_q (\hat{S}_{q(k)}^H \hat{A}_{(k)} \hat{Q}_{q(k)} \hat{A}_{(k)}^H \hat{S}_{q(k)}) \right).$$

Then, $E[\hat{Z}_{(k)}]$ can be expressed as

$$E[\hat{Z}_{(k)}] = \sum_{i=1}^q \left( \hat{Q}_{i(k)}^{-\frac{1}{2}} \hat{A}_{(k)}^H \hat{S}_{i(k)} (\hat{S}_{i(k)}^H \hat{A}_{(k)} \hat{Q}_{i(k)} \hat{A}_{(k)}^H \hat{S}_{i(k)})^{-1} \hat{S}_{i(k)}^H \hat{A}_{(k)} \hat{Q}_{i(k)}^{-\frac{1}{2}} \right) p_i = \hat{Q}_{k(k)}^{-\frac{1}{2}} \hat{A}_{(k)}^H \hat{S}_{k(k)} D_k^2 \hat{S}_{k(k)}^H \hat{A}_{(k)} \hat{Q}_{k(k)}^{-\frac{1}{2}}.$$

Therefore, we obtain

$$\lambda_{\text{min}} \left( E[\hat{Z}_{(k)}] \right) = \lambda_{\text{min}} \left( \hat{Q}_{k(k)}^{-\frac{1}{2}} \hat{A}_{(k)}^H \hat{S}_{k(k)} D_k^2 \hat{S}_{k(k)}^H \hat{A}_{(k)} \hat{Q}_{k(k)}^{-\frac{1}{2}} \right) = \lambda_{\text{min}}^+ \left( \hat{S}_{k(k)}^H \hat{A}_{(k)} \hat{Q}_{k(k)}^{-1} \hat{A}_{(k)}^H \hat{S}_{k(k)} \right) \lambda_{\text{min}} \left( D_k^2 \right).$$

When $p_i = \frac{\|\hat{Q}_{i(k)}^{-\frac{1}{2}} \hat{A}_{(k)} \hat{Q}_{i(k)}^{-\frac{1}{2}} \|^2}{\|\hat{Q}_{i(k)}^{-\frac{1}{2}} \hat{A}_{(k)} \hat{Q}_{i(k)}^{-\frac{1}{2}} + S_i \|^2}$ with $S_i = [S_1, S_2, \ldots, S_q]$ for $i = 1, 2, \ldots, q$, according to the properties of t-product, we have the chain of relations
\[
\|Q^{-\frac{1}{2}} * A^T * S\|_F^2 = \frac{1}{l} \sum_{k=1}^{l} \|\tilde{\mathcal{A}}^H_k (\tilde{S}_l)_k\|_F^2 = \frac{1}{l} \sum_{k=1}^{l} \text{Tr} \left( (\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k) \right) \\
\geq \frac{1}{l} \sum_{k=1}^{l} \lambda_{\text{max}}\left((\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right) = \lambda_{\text{max}}\left((\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right),
\]

which immediately yields
\[
\lambda_{\text{min}}(D_k^2) = \min_{i=1,2,\ldots,q} \left( \frac{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2} \right) = \frac{1}{\lambda_{\text{max}}\left((\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right)} \\
\geq \frac{1}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2} = \frac{1}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2}.
\]

As a consequence,
\[
\lambda_{\text{min}}\left(E(\tilde{Z}_l)\right) \geq \frac{\lambda_{\text{min}}^+\left(\tilde{S}_l^H \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right)}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2}.
\]

When \(p = \frac{1}{q}\) for \(i = 1,2,\ldots,q\), we have
\[
\lambda_{\text{min}}(D_k^2) = \frac{1}{q} \left( \max_{i=1,2,\ldots,q} \lambda_{\text{max}}\left((\tilde{S}_l^H \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right) \right) \geq \frac{1}{q} \max_{i=1,2,\ldots,q} \left\|Q^{-\frac{1}{2}} \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right\|_F^2.
\]

Thus
\[
\lambda_{\text{min}}\left(E(\tilde{Z}_l)\right) \geq \frac{\lambda_{\text{min}}^+\left(\tilde{S}_l^H \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right)}{q \max_{i=1,2,\ldots,q} \left\|Q^{-\frac{1}{2}} \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right\|_F^2}.
\]

Combine (17), (A9) and (A10) to reach the main results (18) and (19).

**Proof of Theorem 7.** The update of Algorithm 5 can be expressed by
\[
X^{l+1} = X^l - Q^{-\frac{1}{2}} * \mathcal{W} * (X^l - X^*),
\]

where
\[
\mathcal{W} = \begin{bmatrix} \text{Re}(S^H * A) \\ \text{Im}(S^H * A) \end{bmatrix}^T * \begin{bmatrix} \text{Re}(S^H * A) \\ \text{Im}(S^H * A) \end{bmatrix} * Q^{-\frac{1}{2}} * \begin{bmatrix} \text{Re}(S^H * A) \\ \text{Im}(S^H * A) \end{bmatrix}^T * \begin{bmatrix} \text{Re}(S^H * A) \\ \text{Im}(S^H * A) \end{bmatrix}
\]

and \(S = \text{ifft}(\tilde{S}, \{1,3\})\) with \(\tilde{S}\) being a random tubal matrix whose frontal slices are \(S_k\) for \(k = 1,2,\ldots,l\). For \(Z = \text{ifft}(\tilde{Z}, \{1,3\})\) and \(\overline{Z} = Q^{-\frac{1}{2}} * \mathcal{W} * Q^{-\frac{1}{2}}\), it is clear that \(\text{bcirc}(Z)\) and \(\text{bcirc}(\overline{Z})\) are both orthogonal projectors, and hence the spectrums of \(\text{bcirc}(Z)\) and \(\text{bcirc}(\overline{Z})\) are contained in \([0,1]\). Let
\[
F = \text{bcirc}(Q^{-\frac{1}{2}}) \begin{bmatrix} \text{Re}(\text{bcirc}(S^H * A)) \\ \text{Im}(\text{bcirc}(S^H * A)) \end{bmatrix}^T \quad \text{and} \quad P = \begin{bmatrix} I & il \end{bmatrix}^T \begin{bmatrix} I & il \end{bmatrix}.
\]

Since
\[
\text{bcirc}(S^H * A) = \text{Re}(\text{bcirc}(S^H * A)) + \text{Im}(\text{bcirc}(S^H * A))il = \begin{bmatrix} I & il \end{bmatrix} \begin{bmatrix} \text{Re}(\text{bcirc}(S^H * A)) \\ \text{Im}(\text{bcirc}(S^H * A)) \end{bmatrix},
\]

\[
\|Q^{-\frac{1}{2}} * A^T * S\|_F^2 = \frac{1}{l} \sum_{k=1}^{l} \|\tilde{\mathcal{A}}^H_k (\tilde{S}_l)_k\|_F^2 = \frac{1}{l} \sum_{k=1}^{l} \text{Tr} \left( (\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k) \right) \\
\geq \frac{1}{l} \sum_{k=1}^{l} \lambda_{\text{max}}\left((\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right) = \lambda_{\text{max}}\left((\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right),
\]

which immediately yields
\[
\lambda_{\text{min}}(D_k^2) = \min_{i=1,2,\ldots,q} \left( \frac{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2} \right) = \frac{1}{\lambda_{\text{max}}\left((\tilde{S}_l^H_k \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right)} \\
\geq \frac{1}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2} = \frac{1}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2}.
\]

As a consequence,
\[
\lambda_{\text{min}}\left(E(\tilde{Z}_l)\right) \geq \frac{\lambda_{\text{min}}^+\left(\tilde{S}_l^H \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right)}{\|Q^{-\frac{1}{2}} * A^T * S\|_F^2}.
\]

When \(p = \frac{1}{q}\) for \(i = 1,2,\ldots,q\), we have
\[
\lambda_{\text{min}}(D_k^2) = \frac{1}{q} \left( \max_{i=1,2,\ldots,q} \lambda_{\text{max}}\left((\tilde{S}_l^H \tilde{\mathcal{A}}_k (\tilde{S}_l)_k)\right) \right) \geq \frac{1}{q} \max_{i=1,2,\ldots,q} \left\|Q^{-\frac{1}{2}} \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right\|_F^2.
\]

Thus
\[
\lambda_{\text{min}}\left(E(\tilde{Z}_l)\right) \geq \frac{\lambda_{\text{min}}^+\left(\tilde{S}_l^H \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right)}{q \max_{i=1,2,\ldots,q} \left\|Q^{-\frac{1}{2}} \tilde{\mathcal{A}}_k (\tilde{S}_l)_k\right\|_F^2}.
\]

Combine (17), (A9) and (A10) to reach the main results (18) and (19).
we can rewritten \( \text{bcirc}(Z) \) and \( \text{bcirc}(Z) \) as \( \text{bcirc}(\hat{Z}) = F(F^TF)^{1/2} \) and \( \text{bcirc}(\hat{Z}) = FP(F^TF)^{1/2} \), respectively. Then we can easily check that
\[
\text{bcirc}(Z)\text{bcirc}(\hat{Z}) = F(F^TF)^{1/2}F^TF(F^TF)^{1/2}PF^T = FP(F^TF)^{1/2}PF^T = \text{bcirc}(\hat{Z}),
\]
and
\[
\text{bcirc}(Z)\text{bcirc}(\hat{Z}) = FP(F^TF)^{1/2}PF^TF(F^TF)^{1/2}F^T = FP(F^TF)^{1/2}PF^T = \text{bcirc}(\hat{Z}).
\]
Therefore, combining with the fact that \( \text{bcirc}(Z) \) and \( \text{bcirc}(\hat{Z}) \) are both orthogonal projectors, we can get that
\[
(bcirc(Z) - bcirc(\hat{Z}))^2 = (bcirc(Z) - bcirc(\hat{Z}))(bcirc(\hat{Z}) - bcirc(Z)) = bcirc(Z)^2 - bcirc(Z)bcirc(\hat{Z}) - bcirc(\hat{Z})bcirc(Z) + bcirc(Z)^2 = bcirc(Z) - bcirc(\hat{Z}) - bcirc(\hat{Z}) + bcirc(Z) = bcirc(Z) - bcirc(\hat{Z}),
\]
which implies that \( \text{bcirc}(Z) - \text{bcirc}(\hat{Z}) \) is also an orthogonal projector and hence its spectrums are contained in \( \{0, 1\} \). Then, \( \lambda_{\min}(\text{bdiag}(\hat{Z})) - \text{bdiag}(\hat{Z}) = \lambda_{\min}((F_l \otimes I_l)(\text{bcirc}(Z) - \text{bcirc}(\hat{Z}))(F_l^{1/2} \otimes I_l)) \geq 0 \), which leads to \( \text{bdiag}(\hat{Z}) - \text{bdiag}(\hat{Z}) \geq 0 \). In addition, the hypothesis that \( E[\hat{Z}(\mathbf{k})] \) is Hermitian positive definite for \( k = 1, 2, \ldots, l \) implies that \( E[\text{bdiag}(\hat{Z})] \) is also Hermitian positive definite. Therefore, \( E[\text{bdiag}(\hat{Z})] \geq E[\text{bdiag}(\hat{Z})] > 0 \), which means that \( \lambda_{\min}(E[\text{bdiag}(\hat{Z})]) \geq \lambda_{\min}(E[\text{bdiag}(\hat{Z})]) > 0 \). Thus, similar to the proof of Theorem 6, we obtain
\[
E \left[ \left\| \Lambda^{\ast} - \Lambda^{\ast} \right\|^2_{F(Q)} \right] \leq \left( 1 - \lambda_{\min}(E[\text{bdiag}(\hat{Z})]) \right) \left\| \Lambda^{0} - \Lambda^{\ast} \right\|^2_{F(Q)} \leq \left( 1 - \lambda_{\min}(E[\text{bdiag}(\hat{Z})]) \right) \left\| \Lambda^{0} - \Lambda^{\ast} \right\|^2_{F(Q)}.
\]

**APPENDIX B. ALGORITHMS**

**Algorithm 7.** Fast ATSP-PR method in Fourier domain

**Input:** \( \Lambda^{0} \in \mathbb{K}_{32}^{n_{s} \times p}, A \in \mathbb{K}_{32}^{n_{s} \times p} \), and \( B \in \mathbb{K}_{32}^{n_{s} \times p} \)

**Parameters:** a set of sketching tubal matrices \( S = [S_1, S_2, \ldots, S_q] \) with \( S_i \in \mathbb{K}_{32}^{n_{s} \times r} \), T-symmetric T-positive definite tubal matrix \( Q \in \mathbb{K}_{32}^{n_{s} \times s} \)

1: \( \hat{Z}^{0} \leftarrow \text{fft}(\Lambda^{0}[, 3]), \hat{A} \leftarrow \text{fft}(A[, 3]), \hat{B} \leftarrow \text{fft}(B[, 3]), \hat{Q} \leftarrow \text{fft}(Q[, 3]), \hat{S}_i \leftarrow \text{fft}(S_i[, 3]) \) for \( i = 1, 2, \ldots, q \)

2: for \( k = 1, 2, \ldots, l \)

3: Compute \( \hat{C}(\mathbf{k})(\mathbf{k}) = \text{cholesky} \left( \left( \hat{S}(\mathbf{k})(\mathbf{k}) \hat{A}(\mathbf{k})(\mathbf{k}) \hat{A}^{-1}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k}) \right)^{1/2} \right) \) for \( i = 1, 2, \ldots, q \)

4: Compute \( \hat{Q}^{-1}(\mathbf{k})(\mathbf{k}) \hat{A}(\mathbf{k})(\mathbf{k}) \hat{A}^{-1}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k})(\mathbf{k}) \) for \( i = 1, 2, \ldots, q \)

5: Compute \( \hat{C}^{-1}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k}) \hat{A}(\mathbf{k})(\mathbf{k}) \hat{A}^{-1}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k})(\mathbf{k}) \) for \( i = 1, 2, \ldots, q \)

6: Initialize \( \hat{R}^{0}(\mathbf{k})(\mathbf{k}) = \hat{C}(\mathbf{k})(\mathbf{k}) \left( \hat{S}(\mathbf{k})(\mathbf{k}) \hat{A}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k})(\mathbf{k}) \right) \) for \( i = 1, 2, \ldots, q \)

7: end for

8: for \( t = 0, 1, 2, \ldots \)

9: \( f_i(\Lambda^{t}) = (1/b) \sum_{k=1}^{l} ||\hat{R}^{t}(\mathbf{k})(\mathbf{k})||^2 \) for \( i = 1, 2, \ldots, q \)

10: Sample \( i \sim \mathbf{p}^{t} \), where \( p_i^{t} = f_i(\Lambda^{t})/(\sum_{i=1}^{q} f_i(\Lambda^{t})) \) for \( i = 1, 2, \ldots, q \)

11: for \( k = 1, 2, \ldots, l \)

12: Update \( \hat{X}^{1+i}(\mathbf{k})(\mathbf{k}) = \hat{X}^{0}(\mathbf{k})(\mathbf{k}) - \left( \hat{Q}^{-1}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k}) \hat{C}(\mathbf{k})(\mathbf{k}) \right) \left( \hat{R}^{t}(\mathbf{k})(\mathbf{k}) \right) \) for \( i = 1, 2, \ldots, q \)

13: Update \( \hat{R}^{t+1}(\mathbf{k})(\mathbf{k}) = \hat{R}^{t}(\mathbf{k})(\mathbf{k}) - \left( \hat{C}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k}) \hat{A}(\mathbf{k})(\mathbf{k}) \hat{S}(\mathbf{k})(\mathbf{k})(\mathbf{k}) \right) \left( \hat{R}^{t}(\mathbf{k})(\mathbf{k}) \right) \) for \( i = 1, 2, \ldots, q \)

14: end for

15: end for

16: \( \Lambda^{t+1} \leftarrow \text{ifft}(\hat{X}^{t+1}[, 3]) \)

**Output:** last iterate \( \Lambda^{t+1} \)
Algorithm 8. Fast ATSP-PR-II method in Fourier domain

\textbf{Input:} $X^0 \in \mathbb{R}_1^{n \times p}$, $A \in \mathbb{K}_i^{m \times n}$, and $B \in \mathbb{K}_i^{m \times p}$

\textbf{Parameters:} \( l \) sets of sketching matrices $S_k = [S_k, S_{k+1}, \ldots, S_{k+l}]$ with $S_k \in \mathbb{C}^{m \times r}$ for $k = 1, 2, \ldots, l$, $T$-symmetric, $T$-positive definite tubal matrix $Q \in \mathbb{K}_i^{m \times m}$

1: $\hat{X}^0 \leftarrow \text{fft}(X^0, [ ] , 3)$, $\hat{A} \leftarrow \text{fft}(A, [ ] , 3)$, $\hat{B} \leftarrow \text{fft}(B, [ ] , 3)$, $\hat{Q} \leftarrow \text{fft}(Q, [ ] , 3)$

2: \textbf{for} $k = 1, 2, \ldots, l$

3: \hspace{1em} Compute $C_k = \text{Cholesky}\left(\left(S_H^k \hat{A}^{-1}(k) \hat{A}^H(k) S_k\right)^{-1}\right)$ for $i = 1, 2, \ldots, q$

4: \hspace{1em} Compute $\hat{C}_k^{-1}(k) \hat{A}^H(k) S_k C_k$ for $i = 1, 2, \ldots, q$

5: \hspace{1em} Compute $C_k^{-1}(k) \hat{A}^{-1}(k) \hat{C}_k C_k$ for $i, j = 1, 2, \ldots, q$

6: \hspace{1em} Initialize $R_0^k = C_k^{-1}(k) \hat{A}^{-1}(k) \hat{C}_k C_k$ for $i = 1, 2, \ldots, q$

7: \textbf{end for}

8: \textbf{for} $t = 0, 1, 2, \ldots$

9: \hspace{1em} \textbf{for} $k = 1, 2, \ldots, l$

10: \hspace{2em} $f_i(\hat{X}_k(t)) = \|R_k^i\|^2_2$ for $i = 1, 2, \ldots, q$

11: \hspace{2em} Sample $k_i \sim p_k$, where $p_k = f_i(\hat{X}_k(t)) / \left(\sum_{i=1}^q f_i(\hat{X}_k(t))\right)$ for $i = 1, 2, \ldots, q$

12: \hspace{2em} Update $\hat{X}_{k+1}^i = \hat{X}_k(t) - \left(\hat{C}_k^{-1}(k) \hat{A}_k^{-1}(k) \hat{C}_k C_k\right) R_k^i$ for $i = 1, 2, \ldots, q$

13: \hspace{2em} Update $R_{k+1}^i = R_k^i - \left(\hat{C}_k C_k \hat{A}_k C_k\right) R_k^i$ for $i = 1, 2, \ldots, q$

14: \hspace{1em} \textbf{end for}

15: \textbf{end for}

16: $X^{t+1} \leftarrow \text{Re}\left(\text{ifft}\left(\hat{X}^{t+1}, [ ] , 3\right)\right)$

\textbf{Output:} last iterate $X^{t+1}$