Low-Tubal-Rank Tensor Completion Using Alternating Minimization

Xiao-Yang Liu®, Student Member, IEEE, Shuchin Aeron®, Senior Member, IEEE,
Vaneet Aggarwal®, Senior Member, IEEE, and Xiaodong Wang, Fellow, IEEE

Abstract—The low-tubal-rank tensor model has been recently proposed for real-world multidimensional data. In this paper, we study the low-tubal-rank tensor completion problem, i.e., to recover a third-order tensor by observing a subset of its elements selected uniformly at random. We propose a fast iterative algorithm, called Tubal-AltMin, that is inspired by a similar approach for low-rank matrix completion. The unknown low-tubal-rank tensor is represented as the product of two much smaller tensors with the low-tubal-rank property being automatically incorporated, and Tubal-AltMin alternates between estimating those two tensors using tensor least squares minimization. First, we note that tensor least squares minimization is different from its matrix counterpart and nontrivial as the circular convolution operator of the low-tubal-rank tensor model is intertwined with the sub-sampling operator. Secondly, the theoretical performance guarantee is challenging since Tubal-AltMin is iterative and nonconvex. We prove that 1) Tubal-AltMin generates a recovery guarantee up to any predefined accuracy ε at an exponential rate, and 2) for an \( n \times n \times k \) tensor \( M \) with tubal-rank \( r \ll n \), the required sampling complexity is \( O((n^2r^2k)/||M||_F^2 \log^3 n)/\sigma_{rk} \), where \( \sigma_{rk} \) is the \( rk \)-th singular value of the block diagonal matrix representation of \( M \) in the frequency domain, and the computational complexity is \( O(n^2r^2k^3 \log n \log(n/\varepsilon)) \). Finally, on both synthetic data and real-world video data, evaluation results show that compared with tensor-nuclear norm minimization using alternating direction method of multipliers (TNN-ADMM), Tubal-AltMin-Simple (a simplified implementation of Tubal-AltMin) improves the recovery error by several orders of magnitude. In experiments, Tubal-AltMin-Simple is faster than TNN-ADMM by a factor of 5 for a 200 × 200 × 20 tensor.

Index Terms—Low-tubal-rank tensor completion, alternating minimization, tensor least squares minimization, sampling complexity, circulant algebra.

I. INTRODUCTION

The big data era calls for efficient algorithms to analyze the enormous amount of data generated by high-resolution sensors, mobile devices, online merchants, and social networks [2]. Such real-world data/signals\(^1\) are naturally represented as multidimensional data arrays [3], namely, vectors, matrices, high-order tensors or tensor networks. Signal recovery from partial measurements [4] by exploiting the redundancy property modeled as sparse or low-rank has received wide attention in various research and engineering communities. We are interested in fast algorithms for multidimensional data completion [5] where the measurement procedure is modeled as a uniformly random element-wise sampling pattern (essentially a down-sampling operation). Exemplar applications include MRI imaging [4], signal processing [3], big data analysis with missing entries [6], data privacy [7], network engineering [8]–[11], Internet of Things [12]–[14], machine learning [15], computer vision [16], [17], recommendation system [18], and system identification [19].

Such diverse applications motivate and justify the developments of compressive sensing (vector case) [4], [20], matrix completion/sensing [21], [22], higher-order tensor decompositions [23]–[25], tensor completion [5] and tensor sensing [26]. Compressive sensing [4], [20] advocated relaxing the original NP-hard problem to its convex surrogate, i.e., replacing the \( \ell_0 \)-norm with \( \ell_1 \)-norm. Similarly, researchers introduced nuclear norm [27] and tensor-nuclear norm [28] to approximate the combinatorial rank function for the low-rank matrix and tensor completion problem\(^2\), respectively. Those two relaxation approaches achieve optimal results with high computational cost, mainly because of the time-consuming SVD (singular value decomposition) or tensor-SVD operations [28], [29].

Alternating minimization approaches have been analyzed for matrix completion [33]–[35]. First, it is both computation- and storage-efficient in implementation. The unknown low-rank matrix \( M \in \mathbb{R}^{m \times n} \) is factorized into two much smaller matrices \( X \) and \( Y \) of size \( m \times r \) and \( n \times r \), respectively, i.e., \( M = XY^H \), and rank \( r \ll \min(m, n) \) implies \( (m + n)r \ll mn \), thus requiring much less memory to optimize. Secondly, this factorization approach enables easier modeling. Besides the low-rank property, this factorization approach allows one to impose extra constraints on the target matrix \( M \) or factors \( X, Y \). For example, Sparse PCA [32] seeks a low-rank \( M \) that is the product of sparse \( X \) and \( Y \). Thirdly, theoretic guarantees become available recently [33]–[35].

However, extending existing alternating minimization algorithms (originally designed for the matrix case and enjoyed

\(^{1}\)In the following, we use the words “signal” and “data” interchangeably.

\(^{2}\)A vector is a first-order tensor while a matrix is a second-order tensor.
empirical successes) [9], [12], [33]–[35] to higher-order tensors is impeded by three major challenges: 1) there exist different definitions for tensor operators that lead to different low-rank tensor models, i.e., the CP-rank tensor [23], the Tucker-rank tensor [24], tensor-train/tensor-ring decomposition [30], [31], and the low-tubal-rank tensor [24], [25]; 2) existing approaches would be rather inefficient for higher-order tensors due to the curse of dimensionality; and 3) those algorithms do not guarantee good theoretical performance.

In this paper, we address these challenges for the third-order tensor completion problem. More specifically, we are interested in the low-tubal-rank tensor model that shares a similar algebraic structure with the low-rank matrix model. Our goal is to design a fast algorithm under the alternating minimization framework, and theoretically assess its performance. We believe this approach would benefit the computations of higher-order tensors due to the following three perspectives:

• The low-tubal-rank tensor model [24], [25] is proposed for modeling multi-dimensional data. The convolution operator in this model is superior for capturing a “spatial-shifting” property in the data, and we believe such a property is ubiquitous in real-world applications, such as WiFi fingerprints [8], RF tomographic imaging [26], videos [29], seismic data [37], and machine learning [36].

• Although being iterative and nonconvex, the alternating minimization approach is usually much faster than convex relaxations of the tensor completion problem. The potential computational efficiency comes from the fact that it automatically incorporates the low-rank property, resulting in massive dimension reduction. Note that computational efficiency is critical for processing big data.

• It has been proved that alternating minimization generates an approximate global optima (best $r$-rank approximation) up to any predefined accuracy at an exponential convergence rate for matrix completion [33]–[35]. According to similar algebra laws, such performance guarantees are expected to hold for higher-order tensors.

First, we propose a fast alternating minimization algorithm, Tubal-AltMin, for the low-tubal-rank tensor completion problem. A key novelty is solving a least squares minimization for tensors by defining a new set of operators, which may be of independent interest. Tensor least squares minimization is different from the standard least squares minimization because the circular convolution operator of the low-tubal-rank tensor model is intertwined with the sub-sampling operator. Therefore, the tensor completion problem is essentially different from matrix completion, implying that existing alternating minimization algorithms [33]–[35] cannot be extended straightforwardly to our problem.

Secondly, the proposed alternating minimization-based approach is much faster than the tensor-nuclear norm minimization with alternating direction method of multipliers (TNN-ADMM) [28], [29]. We prove that 1) the proposed algorithm generates a best rank-$r$ approximate up to any predefined accuracy $\epsilon$ at an exponential rate, which is much faster than TNN-ADMM; and 2) for an $n \times n \times k$ tensor $\mathcal{M}$ with tubal-rank $r \ll n$, the required sampling complexity is $O((nr^2k\|\mathcal{M}\|^2_2 \log^3 n)/\sigma_{rk})$ (where $\sigma_{rk}$ is the $r$-th singular value of the block diagonal matrix representation of $\mathcal{M}$ in the frequency domain) and the computational complexity is $O(n^2r^2k^3 \log n \log (n/\epsilon))$. Please note that there is no constraint on the size of the third-dimension. The proof is based on exploiting an injective mapping between the circulant algebra and the circular matrix space.

Finally, we evaluate a simplified implementation of Tubal-AltMin, namely Tubal-AltMin-Simple, on both synthetic data and real-world video data. The performances are measured in terms of recovery error and running time. Compared with the convex relaxation-based algorithm TNN-ADMM [28], [29], Tubal-AltMin-Simple improves the recovery error by one order of magnitude at sampling rate 50% for synthetic data, and three orders of magnitude for the video data. In experiments, Tubal-AltMin-Simple is faster than TNN-ADMM by a factor of 5 for a $200 \times 200 \times 20$ tensor.

The remainder of the paper is organized as follows. In Section II, we present the low-tubal-rank tensor model and preliminaries of the circulant algebra. Section III describes the low-tubal-rank tensor completion problem and the proposed Tubal-AltMin algorithm, including a novel routine to solve the key subproblem: tensor least squares minimization. Section IV provides theoretical performance guarantees of the Tubal-AltMin algorithm, while detailed proofs are given in the Appendix. In Section V, we evaluate a simplified variant of the proposed scheme on both synthetic data and real-world video data. The conclusions and future works are given in Section VI. This work was presented in part in [1].

II. NOTATIONS AND PRELIMINARIES

We begin by first outlining the notations, the low-tubal-rank tensor model and some useful results [24], [25], [38], [40]. Lemma 3 in this section is new and critical in our proof. Section II-A presents the low-tubal-rank tensor model, used for problem formulation and algorithm design in Section III. Section II-B presents preliminaries of the circulant algebra, used for performance analysis in Section IV and the Appendix.

For computational and sampling complexity, the following standard asymptotic notations are used throughout this paper. Given two non-negative functions $f(n)$ and $g(n)$:

- $f(n) = o(g(n))$ means $\lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$,
- $f(n) = O(g(n))$ means $\lim_{n \to \infty} \frac{f(n)}{g(n)} < \infty$,
- $f(n) = \omega(g(n))$ means $\lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$,
- $f(n) = \Omega(g(n))$ means $\lim_{n \to \infty} \frac{f(n)}{g(n)} > 0$,
- $f(n) = \Theta(g(n))$ means $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

A. Low-Tubal-Rank Tensor Model

Throughout the paper, we will focus on real valued third-order tensors in the space $\mathbb{R}^{m \times n \times k}$. We use $m$, $n$, $k$, $r$ for tensor dimensions, $x$, $y \in \mathbb{R}$ for scalar variables, $x$, $y \in \mathbb{R}^n$ for vectors, and $X$, $Y \in \mathbb{R}^{m \times n}$ for matrices. Tensors

Note that $\Theta$ and $\Omega$ are re-used later, whose meanings will be clear from the context.
are denoted by calligraphic letters and their corresponding circular matrices (defined in Section II-B) are tagged with the superscript \(^{c}\), i.e., \(T \in \mathbb{R}^{m \times n \times k}\), \(X \in \mathbb{R}^{m \times r \times k}\), \(Y \in \mathbb{R}^{n \times r \times k}\), and \(T_c \in \mathbb{R}^{m \times k \times n}, X_c \in \mathbb{R}^{n \times k \times r}, Y_c \in \mathbb{R}^{m \times n \times k}\).

Let \(X^H\) denote the (Hermitian) transpose of matrix \(X\). We use \(i, j, \kappa\) to index the first, second, and third dimension of a tensor, and \(s, t\) for temporary indexing. \([n]\) denotes the set \(\{1, 2, \ldots, n\}\). Usually, \(i \in [m], j \in [n], \kappa \in [k]\) unless otherwise specified. For tensor \(T \in \mathbb{R}^{m \times n \times k}\), the \((i, j, \kappa)\)-th entry is \(T(i, j, \kappa)\), or concisely represented as \(T_{i,j,}\). The \(\ell_2\)-norm of a vector is defined as \(||x||_2 = \sqrt{\sum_{i=1}^{m} |x_i|^2}\), while the Frobenius norm of a matrix \(X\) is \(||X||_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |X_{ij}|^2}\). and of a tensor is \(||T||_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{k} |T_{ij,k}|^2}\).

Tubes/fibers, and slices of a tensor: A tube (also called a fiber) is a 1-D section defined by fixing all indices but one, while a slice is a 2-D section defined by fixing all but two indices. We use \(T(:,:,\kappa)\), \(T(:,i,\kappa)\), \(T(i,:,\kappa)\) to denote the mode-1, mode-2, mode-3 tubes, which are vectors, and \(T(:,:,\kappa)\), \(T(:,i,:\kappa)\), \(T(i,:,\kappa)\) to denote the frontal, lateral, horizontal slices, which are matrices. For easy representation, we use \(T^{(\kappa)} = T(:,:,\kappa)\).

Tensor transpose and frequency domain representation: The tensor transpose \(T^\dagger \in \mathbb{R}^{m \times n \times k}\) is defined by transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through \(k\), i.e., \(T^\dagger(:,:,1) = (T(:,:,1))^H\), and for \(2 \leq \kappa \leq k\), \(T^\dagger(:,:,\kappa) = (T(:,:,k+2-\kappa))^H\) (the transpose of matrix \(T(:,:,k+2-\kappa)\)). For reasons to be clear soon, we define tensor \(T^\dagger \in \mathbb{C}^{m \times n \times k}\) as the frequency domain representation of \(T\), which is obtained by taking the Fourier transform along the third mode of \(T\), i.e., \(T(:,:,\kappa) = \text{fft}(T(i,:,\kappa))\). In MATLAB notation, \(T = \text{fft}(T(:,:,\kappa),3)\), and one can also compute \(T\) from \(T^\dagger\) using \(T = \text{ifft}(T^\dagger(:,:,\kappa),3)\).

We now summarize the algebraic development [25] for the low-tubal-rank tensor model. It rests on defining a tensor-tensor product between two 3-D tensors, referred to as the t-product in Def. 1. For two tubes (vectors) of the same size, i.e., \(a, b \in \mathbb{R}^k\), let \(a \ast b\) denote the circular convolution between these two tubes, which preserves the size. Next, we give the definition of tensor product and other related definitions.

**Definition 1.** [24], [25] \(t\)-product. The tensor-product \(C = A \ast B\) of \(A \in \mathbb{R}^{n_1 \times n_2 \times k}\) and \(B \in \mathbb{R}^{n_2 \times n_3 \times k}\) is a tensor of size \(n_1 \times n_3 \times k\), where \(C(i,j,:) = \sum_{s=1}^{n_2} A(i,s,:) \ast B(s,j,:),\) for \(i \in [n_1]\) and \(j \in [n_3]\).

A 3-D tensor of size \(n_1 \times n_2 \times k\) can be viewed as an \(n_1 \times n_2\) matrix of tubes that lie in the third-dimension. The t-product of two tensors can be regarded as a matrix-matrix multiplication, except that the operation between scalars is replaced by the circular convolution between two tubes. Therefore, the two operators element-wise addition and the t-product, and the space \(\mathbb{R}^{n \times n \times k}\) together define a non-commutative group [40].

**Definition 2.** [24], [25] Identity tensor. The identity tensor \(I \in \mathbb{R}^{n \times n \times k}\) is a tensor whose first frontal slice \(I(:,:,1)\) is the \(n \times n\) identity matrix and all other frontal slices \(I^{(\kappa)}\), \(\kappa = 2, \ldots, k\), are zero matrices.

**Definition 3.** [24], [25] Orthogonal tensor. A tensor \(Q \in \mathbb{R}^{n \times n \times k}\) is orthogonal if it satisfies \(Q^\ast Q = Q \ast Q^\ast = I\).

**Definition 4.** [24], [25] Inverse. The inverse of a tensor \(T \in \mathbb{R}^{n \times n \times k}\) is written as \(T^{-1} \in \mathbb{R}^{n \times n \times k}\) and satisfies \(T^{-1} \ast T = T \ast T^{-1} = I\).

**Definition 5.** [24], [25] Block diagonal form of third-order tensor. Let \(\tilde{A}\) denote the block-diagonal matrix representation of the tensor \(A\) in the frequency domain, i.e.,

\[
\tilde{A} = \text{blkdiag}(\tilde{A}) \triangleq \begin{bmatrix}
\tilde{A}^{(1)} & 0 & \cdots & 0 \\
0 & \tilde{A}^{(2)} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \tilde{A}^{(k)}
\end{bmatrix} \in \mathbb{C}^{m \times n \times k},
\]

where \(\mathbb{C}\) denotes the set of complex numbers. It is easy to verify that \(\tilde{A}^\dagger = \tilde{A}^H\).

**Remark 1.** [24], [25] The following fact will be used throughout the paper for calculating t-product and tensor inverse. For tensors \(A \in \mathbb{R}^{n_1 \times n_2 \times k}\) and \(B \in \mathbb{R}^{n_2 \times n_3 \times k}\), we have

\[
A \ast B = C \iff \tilde{A} \tilde{B} = \tilde{C}.
\]
Algorithm 1 t-SVD [24, 25]

Input: $T \in \mathbb{C}^{m \times n \times k}$

$T \leftarrow \text{fft}(T, [:], :); \quad \% \text{Take DFT along the 3rd dimension}$

for $\kappa = 1$ to $k$ do

$[\hat{U}, \hat{S}, \hat{V}] = \text{SVD}(\hat{T}(::, :\kappa));$

$U^{(\kappa)} = \hat{U}; \quad S^{(\kappa)} = \hat{S}; \quad V^{(\kappa)} = \hat{V};$

end for

$U \leftarrow \text{ifft}(U, [:], :); \quad S \leftarrow \text{ifft}(S, [:], :); \quad V \leftarrow \text{ifft}(V, [:], :).$

and $V \in \mathbb{R}^{n \times r \times k}$ satisfying $U^t U = I$, $V^t V = I$, and $\Theta$ is an $f$-diagonal tensor of size $r \times r \times k$, as illustrated in Fig. 1. This reduced version of t-SVD will be used throughout the paper unless otherwise noted.

Lemma 1. [24, 25] Best rank-$r$ approximation. Let the t-SVD of $T \in \mathbb{R}^{m \times n \times k}$ be $T = U \Theta V^t$. For a given positive integer $r$, define $T_r = \sum_{s=1}^{r} \mathcal{U}(::, s; :)$, where $\mathcal{U}$ is the $s$-tube in $T$. Then $T_r = \arg \min_{\mathcal{R} \in \mathbb{R}^{m \times n \times k}} ||T - \mathcal{R}||_F$, where $T = \{X \ast Y | X \in \mathbb{R}^{m \times n \times k}, Y \in \mathbb{R}^{n \times r \times k}\}$.

Note that $\Theta$ in t-SVD is organized in a decreasing order, i.e., $||\Theta(1, 1, :)||_2 \geq ||\Theta(2, 2, :)||_2 \geq ...$, which is implicitly defined in [25]. Therefore, the best rank-$r$ approximation of tensors is similar to matrix PCA (principal component analysis).

We define the incoherence of third-order tensors which is a condition to guarantee unique solutions. The concept of incoherence is first introduced by [21], [22] for matrices and is a standard assumption for low-rank matrix/tensor completion problems.

Definition 9. [28] Tensor Incoherence Condition. Given the t-SVD of a tensor $T = U \Theta V^t$ with tubal-rank $r$, $T$ is said to satisfy the tensor incoherent condition, if there exists $\mu_0 > 0$ such that for $\kappa \in [k]$

\[
\text{Tensor-column)} \quad (\mu(U) \triangleq \max_{r \in [m]} \max_{j \in [n]} \|U^t \ast \hat{e}_j\|_F^2 \leq \mu_0, }
\]

\[
\text{Tensor-row)} \quad (\mu(V) \triangleq \max_{r \in [m]} \max_{j \in [n]} \|V^t \ast \hat{e}_j\|_F^2 \leq \mu_0,
\]

(3)

where $\hat{e}_i$ is the $m \times 1 \times k$ column basis with $\hat{e}_{111} = 1$ and $\hat{e}_j$ is the $n \times 1 \times k$ column basis with $\hat{e}_{11j} = 1$, redwich is formally given in Def. 13.

Remark 3. Intuitively, incoherence gives a way of ensuring that the target third-order tensor is not near sparse (here, sparse refers to the number of nonzero tubes is small), by ensuring that no standard (tensor-column) basis can be accurately approximated in the tensor-row or -column subspace. The smallest $\mu_0$ is 1 which is achieved by the case when each tensor-column subspace $U(:, i, :)$, $i \in \mathbb{I}$, has elements with magnitude $1 / \sqrt{nk}$, or each tensor-column subspace $U(:, j, :)$, $j \in \mathbb{I}$, has elements with magnitude $1 / \sqrt{mk}$. The largest possible value of $\mu_0$ is $\min(m, n) / r$ when one of the tensor columns of $U$ is equal to the standard tensor column basis $\hat{e}_i$.

With low $\mu_0$, each element of $T$ is supposed to play a similar role in recovering $T$.

B. Circulant Algebra

The circulant algebra is introduced to generalize matrix algebra to the third-order tensor case. We borrow some notations and operations from [38], [40], meanwhile we propose several new definitions to facilitate our analysis in the Appendix.

Throughout the paper, circulants are denoted by underlined letters. We define tubal scalar, tubal vector, and tubal matrix in the following. Note that they are one-dimension higher than their counterparts in traditional linear algebra. In circulant algebra, a tubal scalar is a vector of length $k$. $\mathbb{K}$ denotes the space of length-$k$ tubal scalars, $\mathbb{K}^n$ denotes the space of tubal vectors where each element is a tubal scalar in $\mathbb{K}$, and $\mathbb{K}^{m \times n}$ denotes the space of $m \times n$ tubal matrices where each element is a tubal scalar in $\mathbb{K}$. We use $\alpha, \beta \in \mathbb{K}$ for tubal scalars, $x, y \in \mathbb{K}^n$ for tubal vectors, and $\underline{A}, \underline{B} \in \mathbb{K}^{m \times n}$ for tubal matrices. Their corresponding circular matrices are tagged with the superscript $c$, i.e., $\alpha^c, \beta^c, x^c, y^c, A^c$.

Gleich et al. [38] introduced the operator $\text{circ}(\cdot)$ to map circulants to their corresponding circular matrix representations. For tubal scalar $\alpha = \{a_1, a_2, \ldots, a_k\} \in \mathbb{K}$, tubal vector $\underline{g} \in \mathbb{K}^n$, and tubal matrix $\underline{A} \in \mathbb{K}^{m \times n}$, we use the notation $\alpha \leftrightarrow \alpha^c$ to denote this mapping as follows:

\[
\alpha \leftrightarrow \alpha^c = \text{circ}(\alpha) = \begin{bmatrix}
\alpha_1 & \alpha_k & \cdots & \alpha_2 \\
\alpha_2 & \alpha_1 & \cdots & \vdots \\
\vdots & \vdots & \ddots & \alpha_k \\
\alpha_k & \alpha_{k-1} & \cdots & \alpha_1
\end{bmatrix},
\]

\[
x \leftrightarrow x^c = \text{circ}(x) = \begin{bmatrix}
\text{circ}(x_1) \\
\vdots \\
\text{circ}(x_m)
\end{bmatrix},
\]

\[
A \leftrightarrow A^c = \text{circ}(A) = \begin{bmatrix}
\text{circ}(A_{1,1}) & \cdots & \text{circ}(A_{1,n}) \\
\vdots & \ddots & \vdots \\
\text{circ}(A_{m,1}) & \cdots & \text{circ}(A_{m,n})
\end{bmatrix}.
\]

(4)

Lemma 2. [40] $(\mathbb{K}^n, +, \ast)$ is a commutative ring with the multiplicative identity $1 = \{1, 0, 0 \ldots \}$, where $+$ and $\ast$ denote the addition and circular convolution, respectively. We have:

\[
\underline{\alpha} + \underline{\beta} \leftrightarrow \alpha^c + \beta^c = \text{circ}(\alpha) \ast \text{circ}(\beta),
\]

\[
\underline{\alpha} \ast \underline{\beta} \leftrightarrow \alpha^c \ast \beta^c = \text{circ}(\alpha) \ast \text{circ}(\beta),
\]

\[
\underline{x} \ast \underline{\alpha} \leftrightarrow x^c \ast \alpha^c = \text{circ}(x) \ast \text{circ}(\alpha),
\]

\[
\underline{A} \ast \underline{\alpha} \leftrightarrow A^c \ast \alpha^c = \text{circ}(A) \ast \text{circ}(\alpha).
\]

(5)

Throughout this paper, we view a tensor in the space $\mathbb{R}^{m \times n \times k}$ as a tubal matrix in the space $\mathbb{K}^{m \times n}$. The tensors $T, X, Y$ have circulant representations $\underline{T}, \underline{X}, \underline{Y}$ and circular matrix representations $T^c, X^c, Y^c$. We define the Frobenius norm of a circulant as follows.
Lemma 3. If $T = X \ast Y$, according to (4), the following equality holds:

$$
||T||_F = ||X \ast Y||_F = \frac{1}{\sqrt{k}} ||X \otimes Y||_F.
$$

(6)

Definition 10. Tubal-wise transpose, circulant transpose.
Let $X^T \in \mathbb{R}^{m \times n \times k}$ denote the tube-wise transpose of $X \in \mathbb{R}^{m \times n \times k}$, i.e., $X^T(i, j, :) = X(j, i, :)$. Similarly, let $X^r$ denote the circulant transpose of $X$, i.e., $X^r(i, j) = X(j, i)$, which can be viewed as the transpose of a matrix of vectors.

III. PROBLEM STATEMENT AND PROPOSED ALGORITHM

We first describe the low-tubal-rank tensor completion problem. Then, we present our Tubal-AltMin algorithm followed by detailed descriptions of its key components. Finally, we provide a procedure to implement the tensor least squares minimization.

A. Problem Statement

We consider the problem of completing a 3-D tensor under the assumption that the 3-D tensor has low-tubal-rank. Specifically, assume that the data tensor $T \in \mathbb{R}^{m \times n \times k}$ has tubal-rank $r \ll \min(m, n)$. By observing a set $\Omega \subseteq [m] \times [n] \times [k]$ of $T$’s elements, our aim is to recover the elements outside of $\Omega$ almost accurately as possible.

Let $P_{\Omega}(\cdot)$ denote the projection of a tensor onto the observed set $\Omega$, such that

$$
[P_{\Omega}(T)]_{ij\kappa} = \begin{cases} 
T_{ij\kappa}, & \text{if } (i, j, \kappa) \in \Omega, \\
0, & \text{otherwise}.
\end{cases}
$$

Since $T$ is known to be a low-tubal-rank tensor and the estimated $\hat{T}$ should be close to $T$ on the observation set $\Omega$, the low-tubal-rank tensor completion problem is formulated as the following optimization problem:

$$
\hat{T} = \arg \min_{Z \in \mathbb{R}^{m \times n \times k}} ||P_{\Omega}(Z) - P_{\Omega}(T)||_F \quad \text{s.t. } \text{rank}(Z) \leq r,
$$

(7)

where $Z \in \mathbb{R}^{m \times n \times k}$ is the decision variable, and the function rank(·) refers to the tensor tubal-rank. Note that the noisy case is inherently included since the tensor least squares minimization deals with noise. Problem (7) is NP-hard since the constraint rank(·) allows combinatorially many feasible solutions [49, 50]. Existing works [28, 29] seek to relax the rank function to its convex surrogate, namely, the tensor-nuclear norm. In [28], it was shown that tensor-nuclear norm minimization results in exact recovery under random sampling if the tensors satisfy certain tensor incoherence conditions (3).

However, the computational cost of [25] [28] comes from three aspects: (i) thresholding the eigenvalues in the frequency domain for minimizing the tensor nuclear norm, while transform back to the time domain for element-wise equality in the constraint; (ii) calculating the SVDs of $n_3$ blocks; (iii) the iterations are jointly carried out in both time and frequency domains, thus involving frequent and large number of Fourier and inverse Fourier transforms. Therefore, we will propose an alternating minimization algorithm to solve (7), inspired by the alternating minimization approach’s empirical and theoretical successes in low-rank matrix completion [33–35].

B. The Alternating Minimization Algorithm for Low-Tubal-Rank Tensor Completion

We decompose the target symmetric tensor $\hat{T} \in \mathbb{R}^{n \times n \times k}$ as $\hat{T} = \hat{X} \ast \hat{X}^\dagger$, $\hat{X} \in \mathbb{R}^{n \times r \times k}$, and $r$ is the target tubal-rank. With this decomposition, the problem (7) reduces to

$$
\hat{T} = \arg \min_{\hat{X} \in \mathbb{R}^{n \times r \times k}} ||P_{\Omega}(T) - P_{\Omega}(\hat{X} \ast \hat{X}^\dagger)||_F^2,
$$

(6)

which finds a target tensor $\hat{T} = X \ast X^\dagger$. If problem (7) has a unique solution, namely there exists a best tubal-rank-$r$ tensor that approximate $T$, according to Lemma 1 we know that such a solution is also the best solution that lies in the product space $T$, which means that (6) is equivalent to problem (7). Note that the uniqueness is guaranteed together by the incoherence condition and the random sampling assumption [28, 29].

For an alternating minimization algorithm, there are two key steps. The first is the initialization, while the second is to estimate the best $\hat{X}$ iteratively. Each alternating optimization in isolation is essentially a tensor least squares update which is convex and tractable. For the analysis of the algorithm, we propose a variant of the alternating minimization algorithm, which is a smoothed alternating minimization, as shown in Alg. 2. This algorithm has the following framework: 1) randomizing $X_0$ (line 3) as the initial input for the iterative loop, and 2) fixing $X_{\ell-1}$ and optimizing $X_\ell$ (line 4–6). Different from the standard alternating minimization that alternatively performs a least squares minimization by fixing one and another, Alg. 2 introduces a median operation over the least squares minimization, and a SmoothQR process. This median
operation enables us to get tighter concentration bounds in the proof, while the SmoothQR process bounds the incoherence of $X_t$ along the iterations.

The general flow of Alg. 2 is as follows.

- Line 1-2: Throughout the algorithm, the samples in $\Omega$ are utilized for two purposes: to initialize $X_0$ as a “good” starting point, and to update $X$ in the iterations. The Split function first splits $\Omega$ into two same-sized subsets $\Omega_0, \Omega_+$ and then splits $\Omega_+$ into $L$ subsets of roughly equal size, while preserving the distributional assumption that our theorem uses.
- Line 3: Using the samples in $\Omega_0$, the Initialize procedure generates a good starting point that is relatively close to the optimal solution. This is required for analysis purpose, since from this starting point we are able to prove convergence.
- Line 4-6: The MedianLS-minimization procedure relies on the tensor least squares minimization as described in Section III-D. The median process is introduced to derive concentration bounds, while the SmoothQR function guarantees that in each iteration $X_t$ satisfies the tensor incoherence condition defined in (3).

In Section IV, we will prove that separating the nonconvex minimization (8) into two consecutive convex minimization subproblems will also yield the optimal solution of (7).

**Remark 4.** The framework of Alg. 2 follows a similar approach analyzed in [35] for matrix completion. Note that Alg. 2 differs from that of [35] in three major aspects: 1) the low-tubal-rank tensor completion problem is essentially different from the matrix completion problem as shown in Section IV-A, which indicates that matrixing a tensor that leads to a matrix completion problem is not appropriate for (7); 2) the key routines of Alg. 2 in Section III-C has new forms, namely, the Initialize procedure in Alg. 3 and the tensor least squares minimization in Alg. 5; and 3) the implementation of tensor least squares minimization is newly proposed in Section III-D.

**C. Key Routines**

The key routines of Alg. 2 include the Split function, the Initialize procedure in Alg. 3, the median least squares minimization in Alg. 4, the tensor least squares minimization in Alg. 5, and the SmoothQR factorization in Alg. 6. In the following, we describe each one in detail.

1) Splitting the Samples: We use a Split function that is similar to the matrix case [33]–[35] since essentially they are both set operations. Reference [35] (in Appendix C) proved that there exists a procedure Split$(\Omega, t)$ such that if $\Omega$ is sampled by including each element independently with probability $p$ (sampling with replacement), then Split$(\Omega, t)$ outputs independent random variables $\Omega_1, \ldots, \Omega_t$, such that each set $\Omega_i$ includes each element independently with probability $p_i = p/t$.

2) Finding a Good Starting Point: Alg. 3 describes the procedure for finding a good starting point. Since it is unclear how well the least squares minimization will converge from a random initial tensor, we start with an initial tensor that has bounded distance from the optimal result as shown in Appendix C. The algorithm serves as a fast initialization procedure for our main algorithm. It computes the top-$r$ eigenslices of $P_{\Omega}(T)$, and truncates them in order to ensure incoherence. Note that the truncation for our tensor problem scales the coefficients of a tube, which is different from the matrix case [35] that truncates an element. We use a random orthonormal transformation to spread out the tubes of the eigenslices before truncation.

3) Tensor Least Squares Minimization: We describe the MedianLS iteration of $Y$. Although the iteration for $X$ is different from $Y$, essentially it can be computed in a similar way as shown in Section III-D. Each MedianLS minimization relies on the basic tensor least squares minimization. Partition $\Omega_+$ into $t = O(\log n)$ subsets, then performing the least squares minimization on each subset and then take the median of the returned tensors. The median operation is performed in an element-wise manner.

4) Smooth QR: For the main theorem to hold, it is required that each iterate $X_t$ has coherence less than $\mu$. To achieve this, we adopt the smooth operation, as shown in Alg. 6. Note the $\text{QR}(\cdot)$ and $\text{GS}(\cdot)$ operations for third-order tensors are defined in [25]. For the tensor versions of $\text{QR}(\cdot)$ and $\text{GS}(\cdot)$, it is defined in the following manner: (i) taking DFT along the third dimension of the input tensor to obtain the block diagonal representation in the frequency domain, (ii) performing matrix $\text{QR}(\cdot)$ and $\text{GS}(\cdot)$ on each frontal slice, and (iii) taking inverse DFT to obtain the result tensor. The $\text{QR}(\cdot)$ process returns the orthogonal projector while $\text{GS}(\cdot)$ make it to be orthonormal tensor-column space. The Gaussian perturbation $H\epsilon$ is added to $Y_t$ to ensure small coherence. In our context, the tensor $X_t$ is the outcome of a noisy operation $X_t = A * X_{t-1} + G\epsilon$ (in Appendix E), and so there is no harm in actually adding

**Algorithm 3 Initialization Algorithm: Initialize($P_{\Omega_0}, \Omega_0, r, \mu$)**

**Input:** observation set $\Omega_0 \subseteq [n] \times [n] \times [k]$ and elements $P_{\Omega_0}(T)$, target dimension $r$, coherence parameter $\mu \in \mathbb{R}$.

Compute the first $r$ eigenslices $A \in \mathbb{R}^{n \times r \times k}$ of $P_{\Omega_0}(T)$, $Z \leftarrow A*O$ where $O \in \mathbb{R}^{r \times r \times k}$ is a random orthonormal tensor, $Z' \leftarrow \text{Truncate}_{\mu'}(Z)$ with $\mu' = \sqrt{8\mu \log n/n}$, where Truncate$_{\mu'}$ scales tubes $Z(i,j,:)\| ||Z(i,j,:)||_F > \mu' \|$ by $Z(i,j,:)\|_F/\mu'$, $X_0' \leftarrow \text{QR}(Z')$, where QR$(\cdot)$ returns an orthonormal tensor.

**Output:** Orthonormal tensor $X_0' \in \mathbb{R}^{n \times r \times k}$.

**Algorithm 4 Median Least Squares:** MedianLS-$Y(P_{\Omega}(T), \Omega, X', r)$

**Input:** target tubal-rank $r$, observation set $\Omega \subseteq [n] \times [n] \times [k]$ and elements $P_{\Omega}(T)$, orthonormal tensor $X' \in \mathbb{R}^{n \times r \times k}$, $(\Omega_1, \ldots, \Omega_t) \leftarrow \text{Split}(\Omega, t)$ for $t = 3\log_2 n$, $Y_i = \text{LS}(P_{\Omega_i}(T), \Omega_i, X', r)$ for $i \in [t]$.

**Output:** median$(Y_1, \ldots, Y_t)$. 

---

**Algorithm 2**

- **Input:** observation set $\Omega_0 \subseteq [n] \times [n] \times [k]$ and elements $P_{\Omega_0}(T)$, target dimension $r$, coherence parameter $\mu \in \mathbb{R}$.
- **Output:** $X_r$.

...
minimization subproblems:
\[ \hat{\mathcal{Y}}(\cdot, j, \cdot) = \arg \min_{\mathcal{Y} \in \mathbb{R}^{n \times s \times k}} ||\mathcal{P}_\Omega(T - \mathcal{X}^\dagger \mathcal{Y})||_F^2, \]
where each subproblem corresponds to estimating a lateral slice \( \hat{\mathcal{Y}}(\cdot, j, \cdot), j \in [n] \). One can solve it by performing the following steps:

1) A lateral slice, \( \hat{T}_\Omega(\cdot, j, \cdot) \) of size \( n \times 1 \times k \), is organized into a vector \( \mathbf{b} \) of size \( nk \times 1 \) in the following way:
\[ \mathbf{b} = [T_\Omega(\cdot, j, 1); T_\Omega(\cdot, j, 2); \ldots; T_\Omega(\cdot, j, k)] \in \mathbb{C}^{nk \times 1}. \]

2) \( \hat{\mathcal{X}} \) is transformed into a block diagonal matrix of size \( nk \times nk \), namely \( \hat{\mathcal{X}} \) as in (5).

3) The \( j \)-th lateral slice \( \hat{\mathcal{P}}_\Omega(\cdot, j, \cdot) \) is transformed into a tensor \( \mathcal{A} \) of size \( k \times k \times n \) first, and then into a matrix \( \mathcal{A} \) of size \( nk \times nk \).
\[ \mathcal{A}(\cdot, \cdot, \cdot) = \text{circ}(\hat{\mathcal{P}}_\Omega(\cdot, j, \cdot)), \quad \mathcal{A}(\cdot, \cdot, \cdot) \in [n], \]

\[ \mathcal{A} = \begin{bmatrix}
\text{diag}(\mathcal{A}(1, 1, \cdot)) & \text{diag}(\mathcal{A}(1, 2, \cdot)) & \cdots & \text{diag}(\mathcal{A}(1, k, \cdot)) \\
\text{diag}(\mathcal{A}(2, 1, \cdot)) & \text{diag}(\mathcal{A}(2, 2, \cdot)) & \cdots & \vdots \\
\vdots & \vdots & \ddots & \vdots \\
\text{diag}(\mathcal{A}(k, 1, \cdot)) & \cdots & \cdots & \text{diag}(\mathcal{A}(k, k, \cdot))
\end{bmatrix}. \]

where the operator \( \text{diag}(\cdot) \) transform a tube into a diagonal matrix by putting the elements in the diagonal.

Therefore, the problem in (10) becomes the following standard LS problem:
\[ \hat{x} = \arg \min_{x \in \mathbb{R}^{nk \times 1}} ||\mathbf{b} - \mathcal{A} \mathcal{X} \hat{\mathcal{Y}}||_F^2. \]

Then organizing \( \hat{x} \) back to \( \hat{\mathcal{Y}}(\cdot, j, \cdot), j \in [n] \).

**IV. Performance Analysis of the Proposed Algorithm**

We first describe a counter example to show that the low-tubal-rank tensor completion problem is essentially different from the conventional matrix completion problem. Then, we present the analytical results for the performance guarantees.

**A. Why is Low-Tubal-Rank Tensor Completion Different From Matrix Completion**

One would ask if the tensor completion is essentially equivalent to the matrix completion problem. This concern is reasonable due to the following two aspects, which we would like to clarify. On one hand, in Section II-B we introduce the operation \( \text{circ}(\cdot) \) to establish a mapping between tensor product and matrix multiplication. However, such a mapping is injective, and we use it for easier understanding and it does not mean equivalence. On the other hand, in Appendix D we express the least squares update step as \( \mathcal{Y} = \mathcal{T} \ast (\mathcal{X} + \mathcal{G}) \) and then transform tensor operations to matrix operations on the corresponding circular matrices. Since the mapping is
injective, this transformation is for analysis purpose, which holds only if the operations in the tensor forms hold. In other words, it is a necessary condition (not sufficient condition) resulting from the injective mapping between tensor product and its circular matrix multiplication.

Next, we show that the low-tubal-rank tensor completion problem cannot be solved by converting to a matrix completion problem.

The projection of the multiplication between two third-order tensors, i.e., \( P_{\Omega}(T) = P_{\Omega} \otimes T \). We define a corresponding projection \( P_{\Omega}(T) \) as follows: \( \Omega = \text{circ}(\Omega) \), \( P_{\Omega} = \text{circ}(P_{\Omega}) \), i.e., \( P_{\Omega}(T) = P_{\Omega} \otimes T \).

For the least squares update \( Y = \arg \min_{Y \in \mathbb{R}^{n \times k}} \left\{ ||P_{\Omega}(T - X \ast Y^{\dagger})||_F^2 \right\} \), its circular form would be \( Y = \arg \min_{Y \in \mathbb{R}^{n \times k}} \left\{ ||P_{\Omega}(T^{\circ} - X^{\circ}Y^{\dagger})||_F^2 \right\} \). If \( Y \) is circular we can transform \( Y \) back to a tensor, then these two problems are equivalent. Therefore, the original question becomes: will this circular least squares minimization output a circular matrix \( Y \)? In the following, we give a negative answer by presenting a counterexample.

Assume that \( \Omega = [n] \times [n] \times [k] \), then the circular least squares minimization is equivalent to the following optimization problem:

\[
\min_Y \|G\|_F^2, \\
\text{s.t. } P_{\Omega}^{\circ}(T^{\circ} - X^{\circ}Y + G),
\]  

where \( G \) is a noise term. Without loss of generality, considering the following example with \( \Omega \) being the whole set:

\[
\min \|G\|_F^2, \\
\text{s.t. } T_1 T_2^{\dagger} + T_2 T_1^{\dagger} = X_1 X_2 Y_1 Y_2 + G_{11} G_{12} + G_{21} G_{22},
\]  

where \( T, X, Y \in \mathbb{R}^{1 \times 2} \times \mathbb{R}^{2 \times 2} \), and \( G \in \mathbb{R}^{2 \times 2} \). The constraint in (16) can be transformed to the following four linear equations:

\[
T_1 = X_1 Y_1, T_2 = X_2 Y_2, T_3 = X_1 Y_2, T_4 = X_2 Y_1.
\]  

Considering the first two equations (17) and (18), if \( G_{11} = G_{22} \) (and also \( G_{12} = G_{21} \) in (19) and (20)), then the solution \( Y \) is a circular matrix. Given \( G_{11} = -G_{22} \) that ensuring \( G_{11}^{\dagger} = G_{22}^{\dagger} \), then the solution \( Y \) is not a circular matrix. Therefore, the problem in (15) does not guarantee to output a circular matrix.

**B. Sampling Complexity and Computational Complexity**

1) **Recovery Complexity**

We assume that the unknown noisy tensor \( T \in \mathbb{R}^{n \times n \times n} \) is approximately low-tubal-rank in Frobenius norm. According to Lemma 1, \( T \) has the form \( T = \mathcal{M} + \mathcal{N} \) where \( \mathcal{M} = U \ast \Theta \ast V^{\dagger} \) has tubal-rank \( r \) and eigentubes \( \Theta(1, 1, :) \), \( \Theta(2, 2, :) \), ..., \( \Theta(r, r, :) \) such that

\[
||\Theta(1, 1, :)||_F \geq ||\Theta(2, 2, :)||_F \geq ... \geq ||\Theta(r, r, :)||_F, U \in \mathbb{R}^{n \times r \times k}, V \in \mathbb{R}^{n \times r \times k}, \Theta \in \mathbb{R}^{r \times r \times k}, \text{ and } \mathcal{N} = (I - U \ast U^{\dagger}) \ast T \text{ is the part of } T \text{ not captured by the tensor-column subspace } U \text{ and the dominant eigentubes. Here, we assume that } \mathcal{N} \text{ can be an arbitrary deterministic tensor that satisfies the constraints:}
\]

\[
\max_{i \in [n]} \|e_i^* \ast N\|_F^2 \leq \frac{\mu_N}{n} \sigma_{rk}, \quad \text{and} \quad \max_{i,j,k} |N_{ijk}| \leq \frac{\mu_N}{\max(m, n)} \|T\|_F.
\]

where \( e_i^* \) denotes the \( i \)-th tensor-column basis so that \( \|e_i^* \ast N\|_F \) is the Frobenius norm of the \( i \)-th horizontal slice of \( N \), and \( \sigma_{rk} \) is the \( rk \)-th singular value of the block diagonal matrix \( T \). The block diagonal matrix \( T \) has (approximately) tubal-rank \( rk \) and we denote those \( rk \) singular values as \( \sigma_{1}, \sigma_{2}, ..., \sigma_{rk} \) such that \( \sigma_{1} \geq \sigma_{2} \geq ... \geq \sigma_{rk} \), and \( \sigma_{rk+1} > 0 \), where \( \sigma_{rk} \) is the smallest singular value of \( M \) and \( \sigma_{rk+1} \) is the largest singular value of \( \mathcal{N} \). These constraints state that no element and no horizontal slice of \( \mathcal{N} \) should be too large compared with the Frobenius norm of \( \mathcal{N} \). One can think of the parameter \( \mu_N \) as an analog to the coherence parameter \( \mu(U) \) in (3).

Let \( \mu^* = \max\{\mu(U), \mu(\mathcal{N}), \log k \max(m, n)\} \) and \( \gamma_{rk} = 1 - \sigma_{rk+1}/\sigma_{rk} \), then we have the following theorem.

We first present the theoretical result for the symmetric tensor, and then give a corollary for the asymmetric tensor following the argument in Remark 5.

**Theorem 1.** Suppose that we have a sample set \( \Omega \) of size \( O(n^3k) \) with each element randomly drawn from an unknown symmetric square tensor \( T \) of size \( n \times n \times k \), where \( T = M + \mathcal{N}, M = U \ast \Theta \ast U^{\dagger} \) has tubal-rank \( r \), and \( \mathcal{N} = (I - U \ast U^{\dagger}) \ast T \) satisfies condition (21). Let \( \gamma_{rk} = 1 - \sigma_{rk+1}/\sigma_{rk} \) where \( \sigma_{rk} \) is the smallest singular value of \( M \) and \( \sigma_{rk+1} \) is the largest singular value of \( \mathcal{N} \) (Note that \( \sigma_{rk+1} = 1 \) for the exact tensor completion problem). Then, there exist parameters \( C > 0, \mu = \Theta(\gamma_{rk}^2 (\mu^* + \log nk)) \) and \( L = \Theta(\gamma_{rk}^5 \log(n/e)) \) such that Alg. 2 will output \( (\hat{X}, \hat{Y}) \) with probability at least \( 1 - \Theta \left( \frac{\gamma_{rk}^{-4} \log(n/e)}{nk} \right) \). \( \hat{X} \) is an orthogonal \( n \times r \times k \) tensor that approximates the tensor-column subspace \( U \) as \( ||(I - U \ast U^{\dagger}) \ast \hat{X}||_2^2 \leq \epsilon, \) provided that

\[
p \geq C \gamma_{rk}^7 \mu^* (\|M\|_F^2 + \|N\|_F^2/\epsilon^2) \log(n/e) \log^2 nk \gamma_{rk}^5 \sigma_{rk}/\sigma_{rk+1}^2 k.
\]

The detailed proof is presented in Appendix B. Before the proof, we state the following remark on the reconstruction error in the Frobenius norm, and also the way of extending the above theoretical results from the symmetric square tensor case to general tensor cases.

**Corollary 1.** Under the assumptions of Theorem 1, the output \( (\hat{X}, \hat{Y}) \) of Alg. 2 satisfies \( ||M - \hat{X} \ast \hat{Y}^{\dagger}||_F \leq \frac{\epsilon}{2} ||T||_F \).

**Proof:** Let \( (\hat{X}, \hat{Y}) \) be the tensors output by Alg. 2 when invoked with error parameter \( \epsilon \). By Theorem 1 and Definition 15, we have \( \|U \ast U^{\dagger} \ast \hat{X}^{\dagger}\hat{Y} = ||(I - U \ast U^{\dagger}) \ast \hat{X}||_2^2 \leq \epsilon. \) Lemma 13 shows that \( \hat{Y} = T \ast \hat{X} + G \) (note that \( \hat{X} = X_L \), and
in the proof of Theorem 1 we will verify that \( \{(X_r, G)\}_{r=1}^{L} \) is \((\epsilon/4)\)-tensor-admissible (Definition 18). Therefore we have
\[
\|G\|_F \leq \sqrt{r \sigma_{r,k}/2} \quad \text{since} \quad G \in \mathbb{R}^{r \times k} \quad \text{with rank}(G) = r
\]
and
\[
\|G\|_F \leq \sqrt{r \sigma_{r,k}} \|W^\dagger \ast X_L\|_F + \frac{\sqrt{e/4}}{32} \gamma_{r,k} \sigma_{r,k} \leq \frac{\sqrt{e/4}}{32} \gamma_{r,k} \sigma_{r,k}/2,
\]
where \( \|W^\dagger \ast X_L\|_F \leq \epsilon/4 \) (plugging \( \epsilon/4 \) into (77)), and \( \gamma_{r,k} \leq 1 \).

Therefore, we have
\[
\|M - \hat{X} \ast \hat{Y}\|_F = \|M - \hat{X} \ast \hat{X}^\dagger \ast T - \hat{X} \ast G\|_F \\
\leq \|U \ast U^\dagger \ast T - \hat{X} \ast \hat{X}^\dagger \ast T\|_F + \|\hat{X} \ast G\|_F \\
\leq \|U \ast U^\dagger - \hat{X} \ast \hat{X}^\dagger\|_F \|T\|_F + \|G\|_F \\
\leq \epsilon\|T\|_F + \frac{\sqrt{e/4}}{2} \gamma_{r,k} \sigma_{r,k}, \tag{24}
\]
where in the second inequality, we use the relationship that for all tensors \( P \) and \( Q \), we have \( \|P \ast Q\|_F \leq \|P\|_F \|Q\|_F \) (which can be derived from the definition of the spectrum norm in Definition 15).

**Remark 5.** The result in Theorem 1 for symmetric square tensors can be directly extended to general tensors as follows. For a general tensors \( T \in \mathbb{R}^{m \times n \times k} \), we can construct a symmetric square tensor
\[
T' = \begin{bmatrix} 0 & T \\ T^\dagger & 0 \end{bmatrix} \in \mathbb{R}^{(m+n) \times (m+n) \times k}. \tag{25}
\]

This new tensor \( T' \) has the following property: \( T' \) has tubal-rank \( 2r \) and each two columns \( \Theta(T) \) \((1, 1), \ldots, \Theta(T)(r, r) \) occurs twice for \( T' \). The eigenslices corresponding to a eigentube \( \Theta(T)(i, i) \) spanned by the eigenslices \( \{(U(\cdot, i, 0)_{r \times k}), (U(\cdot, i, 0)_{m \times k}, V(\cdot, i)), \} \).

Therefore, an algorithm outputs a tubal-rank \( 2r \) estimate to \( T' \), which is also a tubal-rank \( 2r \) estimate to \( T \) with the same recovery error level. Moreover, let \( U' \) denote the tensor-column subspace spanned by the top \( 2r \) eigenslices of \( T' \) and let \( U, V \) denote the tensor-column, tensor-row subspaces spanned by the top \( r \) left, right eigenslices of \( T \). Then, we have
\[
\mu(U') \leq \frac{m+n}{2k} \left( \frac{\mu(U)}{m} + \frac{\mu(V)}{n} \right) \leq \frac{n+m}{\max(m, n)} \max\{\mu(U), \mu(V)\}.
\]

Therefore, to recover a tubal-rank \( r \) tensor \( T \), one can invoke Alg. 2 with parameters \( r/2 \),
\[
\frac{n+m}{\max(m, n)} \mu_0 \text{ for this induced symmetric tensor } T' \text{ and keep the other parameters unchanged. Moreover, in the order of sampling complexity, one should change } n \text{ to } \max(m, n) \text{ correspondingly.}
\]

**Theorem 2.** Given a sample set \( \Omega \) of size \( O(pmnk) \) with each element randomly drawn from an unknown \( m \times n \times k \) tensor \( T = M + N \), where \( M = U \ast \Theta \ast \hat{Y} \) has tubal-rank \( r \) and \( N = (I - U \ast U^\dagger) \ast T \) satisfies condition (21), then with probability at least \( 1 - \Theta(\gamma_{r,k}^{-1} \log(m, n)/\epsilon) \), Alg. 2 outputs \((\hat{X}, \hat{Y})\) such that
\[
\|\hat{X} \ast \hat{Y} - M\|_F \leq \frac{\epsilon}{2r} \|T\|_F, \quad \text{provided that} \quad p \geq C r^2 \mu^2.
\]

\[
\left( \|M\|_F^2 + \|N\|_F^2/\epsilon^2 \right) \log(\max(m, n)/\epsilon) \log^2(k \max(m, n)) \gamma_{r,k}^{-3} \sigma_{r,k}^{-3} \min(m, n), \tag{26}
\]
where \( C > 0 \) is some constant.

**Proof:** This theorem is a direct result of Theorem 1. Here, we briefly discuss the high-level proof structure for easy understanding.

Initialization is analyzed in Appendix C. We prove in Lemma 8 that initializing \( X_r \) as the top-\( r \) left orthogonal eigenslices of \( \frac{1}{r} P_{T^1}(T) \) will result in bounded distance to the optimum tensor \( T \), i.e., \( \| (I - U \ast U^\dagger) \ast X_0 \| \leq 1/4 \) with probability at least \( 1 - 1/\max(m, n)^2 \). This is necessary because the key step in Alg. 2, iterating the tensor least squares minimization, does not provide information about how well it converges from a random initial tensor.

With the “good” initialization, our analysis proceeds as follows:

- Express the least squares update step as an update step of the noisy tensor-column subspace iteration which has the form \( \hat{Y}_t = T \ast \hat{X} + \hat{G} \).
- Analyze the local convergence of Alg. 5 by proving that the noisy tensor-column subspace iteration will converge at an exponential rate in Appendix E.
- Prove that the Frobenius norm of the error term \( G \), i.e., \( \|G\|_F \) in spectrum norm, is bounded in Lemma 15.

The smoothed alternating tensor least squares minimization is analyzed in Appendix D. The key step of Alg. 2 is a tensor least squares minimization \( \hat{Y}_t = \arg\min_{\hat{Y} \in \mathbb{R}^{r \times n \times k}} \|P_{T^1}(T - X_{t-1}) \ast \hat{Y}\|_F^2 \). We will show that the solution to this has the form \( \hat{Y}_t = T \ast X_{t-1} + \hat{G}_t \). The error term \( \|\hat{G}_t\|_F \) in spectrum norm depends on the quantity \( \| (I - U \ast U^\dagger) \ast X_{t-1} \| \) which coincides with the sine of the largest principal angle between \( U \) and \( X_{t-1} \). As the algorithm converges, the spectrum norm of the error term diminishes.

**Remark 6.** The low-tubal-rank tensor completion problem generalizes the low-rank matrix completion problem. One can verify that the result in Theorem 2 reduces to Theorem I.2 in [35] when setting \( k = 1 \).

**Remark 7.** As compared to the TNN-ADMM’s [28] sampling complexity of \( O((m, n)k \log \log (m, n)) \), Tubal-Alt-Min requires an extra \( r \) factor and a term \( \|M\|_F^2/\sigma_{r,k}^2 \geq r \). Even though the theoretical results predict these two extra factors, the numerical results in Section V depict lower errors using fewer number of samples.

2) Computational Complexity:

**Theorem 3.** The Tubal-Alt-Min algorithm in Alg. 2 has a computational complexity \( O(mn^2k^3 \log n \log(n)/\epsilon) \), if \( k \geq \sqrt{\frac{n}{r \log n \log(n/r)}} \), and \( O(mn^2k) \) otherwise.

**Proof:** We characterize the computational complexity by counting the multiplications involved in Alg. 2, which comes from the initialization process (line 1-3) and the for-loop (line 4-6).

The initialization includes two steps: the t-SVD for computing the first \( r \) eigenslices and the DFT along the third dimension. The t-SVD in Alg. 1 has complexity \( O(mn^2k) \), while taking the DFT over \( mn \) tubes of size \( k \) has complexity \( O(mnk \log k) \). Therefore, the initialization process has
total complexity $O(mn^2k + mnk \log k) = O(mn^2k)$ since $\log k \ll n$.

Note that in Alg. 2 there are $L = \Theta(\log(n/e))$ iterations, and each median operation is taken over $t = O(\log n)$ tensor least squares minimizations. The complexity of the tensor least squares minimization is dominated by (14), which has complexity $O(nrk^2)$. More specifically, calculating $A\overline{X}$ of (14) has complexity $O(rk)$ exploiting the sparseness of both $A$ and $\overline{X}$ (since $A$ has diagonal block structure and $\overline{X}$ has block diagonal structure), and the least squares minimization (14) requires $O(nrk^2) = O(nk \cdot (rk)^2)$ multiplications in MATLAB [51]. Note that each lateral slice is treated separately as shown in (10), thus a tensor least squares minimization has complexity $O(mnrk^2k^3)$.

Note that the MedianLS process dominates the computational complexity. For calculating the tensor Gram-Schmidt process, there are $k$ parallel Gram-Schmidt operations over matrices of size $n \times r$ (each has complexity $O(nr^2)$) [Ref.] Gene H. Golub and Charles F. Van Loan, “Matrix Computations.” JHU Press, 2012.), leading to a total computational complexity $O(k \cdot n^2)$. Comparing with the MedianLS in Alg. 2 line 5 (taking median over $t = O(\log n)$ tensor least squares), with computational complexity of $O(mnrk^2 \log n)$, therefore the computational complexity is dominated by the MedianLS process.

Therefore, the total complexity of Alg. 2 is $O(Lmnk^2k^3t + mnk^2k)$ where $L = \Theta(\log(n/e))$ and $t = O(\log n)$. In particular, the complexity is $O(mnrk^2k^3 \log n \log(n/e))$, if $k \geq \sqrt{\frac{n^2 \log n \log(n/e)}{r}}$, and if $k < \sqrt{\frac{n^2 \log n \log(n/e)}{r}}$, the total complexity is $O(mn^2k^2)$, dominated by the t-SVD.

**Corollary 2.** The recovery error of the algorithm $||\hat{X} \ast \hat{Y} - \hat{M}||_F$ decreases exponentially with the number of iterations $L$. However, the sampling complexity increases by a constant factor with $L$.

**Proof:** In $L = \Theta(\log(n/e))$ iterations, $||\hat{X} \ast \hat{Y} - \hat{M}||_F \leq \epsilon \frac{||T||_F}{\epsilon}$. Since $L = \Theta(\log(n/e))$, $\epsilon = \Theta(\exp(-CL))$ for some constant $C$ thus showing that the error $||\hat{X} \ast \hat{Y} - \hat{M}||_F$ decreases exponentially with $L$.

**Remark 8.** The number of update steps $(L = \Theta(\log(n/e)))$ enters the sample complexity since we assume (as in [35]) that fresh samples are used in each step.

**V. Evaluation**

We evaluate a simplified variant of the proposed alternating minimization algorithm (Tubal-AltMin-Simple) on both synthetic and real video data. The synthetic data, generated according to our low-tubal-rank tensor model, serves as well-controlled inputs for testing and understanding Tubal-AltMin-Simple’s performance over the convex algorithm NN-ADMM. The real video data tests the applicability of our low-tubal-rank tensor model, compared with other tensor models.

Please note that for actual implementation, we simplify Alg. 2 by removing the Split operation (we use the whole observation set $\Omega$ in all subroutines of Alg. 2), the median operation, and the SmoothQR function. These three techniques are introduced only for the purpose of obtaining theoretical performance guarantees. The detailed steps of the simplified algorithm are in Alg 7. Note that random initialization is sufficient in practice as we observed almost the same performance with Alg. 3 in all of our testing cases. We choose to report the simplified Alg. 7 with two considerations: (i) the computational advantages of alternating minimization is simply alternatively performing least squares minimization, and Alg. 7 is a proof of idea that the tensor least squares minimization given in Section III D is practically amenable; (ii) we believe Alg. 7 will be more useful to engineers as in the case of matrix completion using alternating minimization.

**Algorithm 7** Simplified Tubal Alternating Minimization: Tubal-AltMin-Simple($P_\Omega(T), \Omega, L, r$)

**Input:** Observation set $\Omega \subseteq [m] \times [n] \times [k]$ and the corresponding elements $P_\Omega(T)$, number of iterations $L$, target tubal-rank $r$.

1: $X_0 \leftarrow \text{Initialize}(P_\Omega(T), \Omega, r)$,
2: For $t = 1$ to $L$
3: $Y_t \leftarrow \text{LS}(P_\Omega(T), \Omega, X_{t-1}, r)$,
4: $X_t \leftarrow \text{LS}(P_\Omega(T), \Omega, Y_t, r)$,

**Output:** Pair of tensors $(X_L, Y_L)$.

**A. Experiment Setup**

We use Matlab installed on a server with Linux operating system. The parameters of the server is: Intel® Xeon® Processor E5-2650 v3, 2.3 GHz clock speed, 2 CPU each having 10 physical cores, virtually maximum 40 threads, 25 MB cache, and 64 GB memory.

For synthetic data, we compare our Tubal-AltMin-Simple algorithm with tensor-nuclear norm (NN-ADMM) [29], since both are designed for low-tubal-rank tensors, to show the advantages of our non-convex approach over its convex counterpart. We conduct experiments to recover third-order tensors of different sizes $m \times n \times k$ and tubal-ranks $r$, from observed elements in the subset $\Omega$. Three metrics are adopted for comparison, e.g., the recovery error, the running time, and the convergence speed.

- For recovery error, we adopt the relative square error metric, defined as RSE=$||\hat{T} - T||_F/||T||_F$.
- For running time, varying the tensor size and fixing other parameters, we measure CPU time in seconds.
- For convergence speed, we measure the decreasing rate of the RSE across the iterations by linearly fitting the measured RSEs (in log scale). We include those plots due to three reasons: 1) both algorithms are iterative; 2) our theoretical analysis predicts exponential convergence; and 3) the decreasing speed of the RSE provides explanations for the observed performance of the recovery error and the running time.

For real dataset, we choose an NBA basketball video of size $144 \times 256 \times 40$ (source: YouTube, as used in [29]), with a non-stationary panning camera moving from left to right.
Fig. 2. Recovery error RSE in log-scale for different sampling rates.

Fig. 3. Convergence speed for the two algorithms on synthetic data.

Fig. 4. Comparison of running time for varying tensor size.

horizontally following the running players. Besides the low-tubal-rank tensor model, there are two widely used tensor models: low CP-rank tensor [23] and low Tuker-rank tensor [17]. The compared algorithms are briefly described as follows:

- **TNN-ADMM**: [29] introduced a convex norm, tensor-nuclear norm (TNN) [28], to approximate the tubal-rank and proposed an Alternating Direction Method of Multipliers (ADMM) algorithm. It is a convex approach and shares the alternating and iterative features. We include TNN-ADMM [29] to show that the predicted advantages on the synthetic data in Section V-B also hold for real datasets. Note that we set the maximum iteration number to be 200 and the Lagrangian multiplier to be 0.01.

- The alternating minimization algorithm (CP-Altmin) [43] under CANDECOMP/PARAFAC decomposition: The CP decomposition models a tensor as the outer product of \( r \) (the CP-rank) vector components. The CP-Altmin algorithm [43] alternates among the least squares minimization subproblems for each component.

- The gradient-type algorithm Tuker-Gradient under Tuker decomposition: Tuker decomposition matricizes a tensor from \( n \) modes, while the Tuker-Gradient algorithm [44] iteratively estimate each matricized component using the conjugate gradient decent. Essentially,
Tuker-Gradient does not fall into the alternating minimization approach. However, we feel it is necessary to compare with it since both Tubal-AltMin-Simple and Tuker-Gradient iteratively estimate each factor and are nonconvex.

B. Synthetic Data

For recovery error, our input is a low-tubal-rank tensor of size $200 \times 200 \times 50$ and tubal-rank 10. We first generate two Guassian random tensors of sizes $200 \times 10 \times 50$ and $10 \times 200 \times 50$ and then perform tensor product in Definition 1 to get the input tensor of size $200 \times 200 \times 50$. We set the maximum iteration number to be 10 for Tubal-AltMin-Simple and 500 for TNN-ADMM, respectively. Varying the sampling rate as 5%, 10%, ..., 95% by uniformly selecting entries, we test each sampling rate 5 times and then plot the average results.

Fig. 2 shows the recovery error performance (RSE in log scale) of Tubal-AltMin-Simple and TNN-ADMM for varying sampling rates. For a clear comparison, we draw two plots for sampling rate $\leq 50\%$ and sampling rates 5% $\sim$ 95%, respectively. When the sampling rate is higher than 50%, the RSE of Tubal-AltMin-Simple is orders of magnitude lower than that of TNN-ADMM. For sampling rates 15% $\sim$ 45%, the RSE of Tubal-AltMin-Simple is approximately half of or one order lower than that of TNN-ADMM except for an abnormal case at sampling rate 25%. However, for sampling rates 5% $\sim$ 10%, for Tubal-AltMin-Simple behaves badly since with insufficient samples, Tubal-AltMin-Simple switches between the factor tensors that are both not well determined. The possible reason that TNN-ADMM has much higher RSE than Tubal-AltMin-Simple is that TNN-ADMM does not have the exact tubal-rank value $r$ and keeps a higher tubal-rank value to avoid the risk of experiencing higher RSE.

For convergence speed, the input is a low-tubal-rank tensor of size $200 \times 200 \times 50$ and tubal-rank 10. We set the maximum iteration number to be 10 for Tubal-AltMin-Simple and 500 for TNN-ADMM. We fix the sampling rate to be 50% and record the RSE in each iteration. TNN-ADMM terminates at the 134-th iteration because the algorithm detected that the decrease of RSE is lower than a preset threshold.

Fig. 3 shows the decreasing RSEs across the iterations for Tubal-AltMin-Simple and TNN-ADMM. For a clear comparison, we draw two plots with 30 and 140 iterations for TNN-ADMM, respectively. Clearly, Tubal-AltMin-Simple decreases much faster than TNN-ADMM. Interestingly, TNN-ADMM behaves very badly during the first 11 iterations.

For running time, we measure the CPU time in seconds. We vary the tensor size from $25 \times 25 \times 20$ to $200 \times 200 \times 20$. Note that the tubal-rank tensor model views a tensor as a matrix of vectors, the third-demention $k$ acts as a linear scaling factor, therefore, to test large tensor cases we set $k = 20$ that is smaller than $m$ and $n$ to avoid the ERR: out of memory. The tubal-rank is set to be 5 for all cases, the sampling rate is 50%, while the target RSE less than $10^{-5}$. For fairness, we preset a threshold $10^{-5}$ and measure the running time that the algorithms need to reach an RSE less than this threshold.

Fig. 4 shows the running time curves of Tubal-AltMin-Simple and TNN-ADMM. For tensors larger than $75 \times 75 \times 20$, our Tubal-AltMin-Simple algorithm beats TNN-ADMM. The acceleration ratio is about 2 times for tensors of size $125 \times 125 \times 20$ and 5 times for tensors of size $200 \times 200 \times 20$. However, since our implementation of Tubal-AltMin-Simple in Section III-D introduces intermediate matrices of size $nk \times nk \times n$, we encounter the ERR: out of memory and thus we are not able to test larger tensor sizes.

C. Real Data

For the NBA video, we only compare the recovery error. One can visually see the improvements of Tubal-AltMin-Simple by downloading the video results [47]. Since we are interested in fast algorithms and we think it is not fair to compare the running time and convergence speed of our Tubal-AltMin-Simple algorithm with the Tuker-Gradient [44] that has much higher recovery error.

Fig. 5 shows the recovery error performance (RSE in log scale) of Tubal-AltMin-Simple, TNN-ADMM and Tuker-Gradient for varying sampling rates. For a clear comparison, we draw two plots for sampling rate $\leq 50\%$ and sampling rates 5% $\sim$ 95%, respectively. Tubal-AltMin-Simple achieves relative lower error by orders of magnitude for sampling rates higher than 35%, while the RSE does not decrease much more. The Tuker-Gradient [44] implementation tested tensor of size 10,000 $\times$ 10,000 $\times$ 10,000, which is not possible in our implementation due to the ERR: out of memory for our implementation in Section III-D.
for TNN-ADMM. Here, Tubal-AltMin-Simple behaves worse than that in Fig. 2 for sampling rates $5\% \sim 30\%$, and much better for sampling rates $35\% \sim 95\%$. Comparing with Fig. 2, Tubal-AltMin-Simple has similar recovery error for both synthetic data and real video data, while TNN-ADMM’s performance for video data is worse than that for synthetic data. The reasons are: 1) the synthetic input tensor has tubal-rank 10, while the NBA video data has larger tubal-rank (approximately 30) with the residual error modelled as noise, and 2) Tubal-AltMin-Simple is more capable of dealing with noise in the real-world video data. Note that in log-scale, the performance of TNN-ADMM and Tucker-Gradient are in the same order and thus indistinguishable.

VI. CONCLUSIONS

Alternating minimization provides an empirically appealing approach to solve low-rank matrix and tensor completion problems. We provide the first theoretical guarantees on the global optimality for the low-tubal-rank tensor completion problem. In particular, this paper proposes a fast iterative algorithm for the low-tubal-rank tensor completion problem, called Tubal-AltMin, which is based on the alternating minimization approach. The unknown low-tubal-rank tensor is represented as the product of two much smaller tensors with the low-tubal-rank property being automatically incorporated, and Tubal-AltMin alternates between estimating those two tensors using tensor least squares minimization. We have obtained theoretical performance guarantees on the success of the proposed algorithm under the tensor incoherency conditions. Based on extensive evaluations and comparisons on both synthetic and real-world data, the proposed algorithm is seen to be fast and accurate.

We note that the guarantees for tensors for low CP-rank tensors has been provided in [43], and this paper provides the results for low tubal-rank tensors. Having such results for matrix product states, where such algorithms have been shown to be good empirically [52], is an open problem.

APPENDIX

Fig. 6 shows the flow chart of our proof for Theorem 1, which analyzes the iterative algorithm Alg. 2 with a two-stage approach. In the first stage, we prove in Theorem 4 that Alg. 3 (the initialization step of Alg. 2) returns a good initial point that is relatively close to the global optima, while in the second stage, Theorem 5 states that Alg. 4 (the iteration steps of Alg. 2) converges locally. Also, we would like to explicitly point out that our proof flow and organization are inspired by Hardt’s work in [35]

Theorem 4 in Appendix C states that $X_0$ approximates the tensor-column subspace $\mathcal{U}$ well such that $||((I - U^T U) \ast X_0)||_F \leq 1/4$, if the elements are included in $\Omega_0$ with probability at least $1 - 1/n^2$, and $\gamma_k = 1 - \sigma_k/\sigma_{k+1}$ where $\sigma_k$ denotes the singular value of the block diagonal matrix $\mathcal{T}$, and $\gamma_k = 1 - \sigma_k/\sigma_{k+1}$.
The convergence of Alg. 2 relies on the fact that alternately calling Alg. 4 will lead to local convergence. First, we express each iteration as a noisy tensor-column subspace iteration in Lemma 13 as \( Y = T \ast X + G \) where \( G \) is the noise/error term. Secondly, we show that if the error term \( G \) is bounded as in Lemma 15, then Alg. 2 converges at an exponential rate as in Lemma 17. Thirdly, to guarantee that \( G \) is bounded in Frobenius norm, we obtain a requirement that the elements are included in \( \Omega_+ \) with probability \( p_+ = O \left( \frac{\text{poly}(n)}{n^2} \right) \) in Lemma 14.

Therefore, the sampling complexity is the sum of two terms: the samples required by the initialization step, and the samples required to bound the error term in the tensor least squares minimization steps, i.e., \( p = p_0 + p_+ \). Theorem 1 shows that if the sampling complexity is larger than the sum of these two terms, Alg. 2 converges to the true unknown tensor rapidly.

A. Additional Definitions and Lemmas

For easy description of the performance analysis, we consider a symmetric square tensor for analysis, i.e., \( T = X \ast X^\dagger \in \mathbb{R}^{n \times n \times n} \).

Definition 12. Square tensor, rectangular tensor, symmetric square tensor. A tensor \( T \in \mathbb{R}^{n \times n \times n} \) is a square tensor, and a tensor \( X \in \mathbb{R}^{n \times n \times n} \) is a rectangular tensor. A symmetric square tensor is the t-product of a tensor and its transpose, i.e., \( T = X \ast X^\dagger \in \mathbb{R}^{n \times n \times n} \).

Therefore, a symmetric square tensor \( T \) has the t-SVD decomposition \( T = U \ast \Theta \ast U^\dagger \). A tensor-column subspace of \( T \) is the space spanned by the lateral slices of \( U \) under the t-product, i.e., t-span(\( U \)) = \{ \( X = \sum_{i=1}^r U(:, s, :) \ast e_s \in \mathbb{R}^{n \times 1 \times k}, e_s \in \mathbb{R}^{1 \times 1 \times k} \) \}, where \( r \) denotes the tensor tubal-rank.

Definition 13. Tensor basis and the corresponding decomposition [28]. We introduce two tensor bases. The first one is called column basis \( e_i \), of size \( n \times 1 \times k \) with only non-zero entry equal to 1 and the rest equal to 0. Note that this nonzero entry 1 will only appear at the \( i \)-th entry of the first frontal slice of \( e_i \). Naturally, its transpose \( e_i^\dagger \) is called row basis. The second tensor basis is called tubal basis \( e_i \), of size \( 1 \times 1 \times k \) with one entry equal to 1 and rest equal to 0. Fig. 7 illustrates these two bases.

With the above two bases, one can obtain a unit tensor \( E \) with only non-zero entry \( E_{ijc} = 1 \) as follows:

\[
E = e_i \ast e_k \ast e_j^\dagger.
\] (27)

Given any third-order tensor \( T \in \mathbb{R}^{m \times n \times k} \), we have the following decomposition

\[
X = \sum_{i=1}^m \sum_{j=1}^n \sum_{c=1}^k X_{ijc} e_i \ast e_k \ast e_j^\dagger.
\] (28)

Definition 14. [28] \( \ell_2^\ast, \ell_{\infty, 2}^\ast \)-norm of tensor. Let \( X \) be an \( m \times n \times k \) tensor. We define the \( \ell_2^\ast \)-norm of its \( j \)-th lateral slice \( \hat{X}(\cdot, j, \cdot) \), and the \( \ell_{\infty, 2}^\ast \)-norm as follows

\[
\|X(\cdot, j, \cdot)\|_{2^\ast} = \sqrt{\sum_{i=1}^m \sum_{k=1}^n X_{ijk}^2},
\]

\[
\|X(\cdot, j, \cdot)\|_{\infty, 2^\ast} = \max_{i \in [n]} \|X(\cdot, j, \cdot)\|_{2^\ast}.
\] (29)

Moreover, we have the following relationship between the \( \ell_{2^\ast} \)-norm of \( \hat{X}(\cdot, j, \cdot) \) and its FFT along the third dimension \( \hat{X}(\cdot, j, \cdot) \),

\[
\|X(\cdot, j, \cdot)\|_{2^\ast} = \frac{1}{\sqrt{k}} \|\hat{X}(\cdot, j, \cdot)\|_{2^\ast},
\] (30)

which is a direct result of Parseval’s Theorem.

Definition 15. [28] Tensor spectral norm. The tensor spectral norm \( ||X|| \) of a third-order tensor \( X \in \mathbb{R}^{m \times n \times k} \) is defined as the largest singular value of \( X \)

\[
||X|| = \sup_{\Lambda \in \mathbb{R}^{m \times n \times k}, ||\Lambda||_{F} \leq 1} ||X \ast \Lambda||_{F}.
\] (31)

Lemma 4. The tensor spectral norm of \( X \) equals to the spectral norm of the circular matrix \( X^e \) and equals to the spectral norm of the block diagonal matrix \( \Gamma \), i.e.,

\[
||X|| = ||X^e|| = ||\Gamma||.
\] (32)

Proof: According to Lemma 3, we expand (31) as follows

\[
||X|| = \sup_{\Lambda \in \mathbb{R}^{m \times n \times k}, ||\Lambda||_{F} \leq 1} ||X \ast \Lambda||_{F}
\]

\[
= \sup_{\Lambda \in \mathbb{R}^{m \times n \times k}, ||\Lambda||_{F} \leq \sqrt{k}} \frac{1}{\sqrt{k}} ||X^e \Lambda^e||_{F}
\]

\[
= \sup_{Y \in \mathbb{R}^{m \times n \times k}, ||Y||_{F} \leq 1} \frac{1}{\sqrt{k}} \|X^e Y\|_{F}
\]

\[
= \sup_{Y \in \mathbb{R}^{m \times n \times k}, ||Y||_{F} \leq 1} ||X^e Y||_{F} = ||X^e||,
\] (33)

where the last equality is the definition of matrix spectrum norm. Further, we provide an argument for the third and fourth equalities to hold, namely, \( Y \) should be a circular matrix to achieve the maximum value of \( ||X^e Y||_{F} \). Consider two matrices \( A \in \mathbb{R}^{m \times n} \) and \( B \in \mathbb{R}^{n \times k} \) with the SVD decomposition \( A = U_A S_A V_A^H \) and \( B = U_B S_B V_B^H \), we have

\[
\max_{||B||_{F} \leq 1} \|AB\|_{F} = \max_{||B||_{F} \leq 1} ||S_A V_A^H U_B S_B||_{F}^2,
\] (34)
which achieves the maximum when \( V_A^H U_B = I_n \). Let \( A \) be a circular matrix, then \( V_A^H \) is a DFT matrix \( F^n \) and the condition \( V_A^H U_B = I_n \) will require \( U_B \) to be also a DFT matrix \( F \). Therefore, we conclude that \( Y \) should be a circular matrix to achieve the maximum value of \( \|X^c Y\|_F \).

Now note that since the circular matrices are diagonalized by the Fourier transform, it follows that \( \|X^c\| = \|X\| \). Formally, we provide the following proof. According to (2) and Parseval’s Theorem as used in (30), we expand (31) as follows:

\[
\|X\| = \sup_{\mathcal{L} \in \mathbb{R}^{n \times n \times k}, \|\mathcal{L}\|_F \leq 1} \|\mathcal{X} \ast \mathcal{L}\|_F = \sup_{\mathcal{L} \in \mathbb{R}^{n \times n \times k}, \|\mathcal{L}\|_F \leq \sqrt{k}} \sqrt{k} \|\mathcal{X} \ast \mathcal{L}\|_F = \|\mathcal{X}\|_F. \tag{35}
\]

**Definition 16. Tensor infinity norm.** The tensor infinity norm \( \|A\|_\infty \) is defined as the largest absolute value of any of its entry, i.e.,

\[
\|A\|_\infty = \max_{i,j,k} |A_{ijk}|. \tag{36}
\]

Note that the tensor-column subspace iteration in Alg. 8 relies on definitions of inverse, angle function, inner products, norm, conjugate, and also the circulant Fourier transform [38]. We describe the inverse and the circulant Fourier transform while omit the rest since they are implicitly used in our paper. The inverse of \( \alpha \in \mathbb{K} \) is \( \alpha^{-1} \) \( \rightarrow \) circ(\( \alpha^{-1} \)), where circ(\( \alpha^{-1} \)) is also a circulant. The circulant Fourier transforms, cft(\( \alpha \)) \( \rightarrow \mathbb{K}^k \times k \) and its inverse icft(\( \alpha \)) \( \rightarrow \mathbb{K} \), are defined as follows:

\[
cft(\alpha) = \begin{bmatrix} \hat{\alpha}_1 \\ \vdots \\ \hat{\alpha}_k \end{bmatrix} \rightarrow F^* \text{circ}(\alpha) F, \\
\text{icft} \left( \begin{bmatrix} \hat{\alpha}_1 \\ \vdots \\ \hat{\alpha}_k \end{bmatrix} \right) = \alpha \rightarrow F \text{icft}(\alpha) F^*, \tag{37}
\]

where \( \hat{\alpha}_k \) are the eigenvalues of \( \text{circ}(\alpha) \) as produced in the Fourier transform order, \( F \) is the \( k \times k \) discrete Fourier transform matrix, and \( F^* \) denotes the (circulant) conjugate [38].

**Eigentubes and Eigenslices:** As in [38], we describe the eigentubes and eigenslices. The existence of an eigentube \( \lambda \in \mathbb{K}_k \) implies the existence of a corresponding eigenslice \( \lambda \in \mathbb{K}^{n \times n} \), satisfying \( A \ast \lambda = X^c \lambda \). The corresponding Fourier transforms cft(\( \lambda \)), cft(\( \lambda \)), cft(\( \lambda \)), cft(\( \lambda \)) satisfy

\[
cft(\lambda) \ast cft(\lambda) = cft(\lambda) \ast cft(\lambda). \tag{38}
\]

From here on we will always assume that the symmetric square tensor \( T \) has the decomposition \( T = \mathcal{U} \ast \Theta \ast \mathcal{U}^\dagger + \mathcal{W} \ast \Theta W \ast \mathcal{W}^\dagger = \mathcal{M} + \mathcal{N} \), where \( \mathcal{M} = \mathcal{U} \ast \Theta \ast \mathcal{U}^\dagger \) and \( \mathcal{N} = \mathcal{W} \ast \Theta W \ast \mathcal{W}^\dagger \). Therefore, Lemma 15 states that with probability \( 1 - 1/n^2 \), the upper bound \( \|G_t\|_F \leq \sqrt{\gamma_{rk}} \|\mathcal{N}\|_F \), satisfying the \( \epsilon/4 \)-admissible condition. This results in the probability \( p_+ \) satisfying:

\[
p_+ = \sum_{t=1}^{L} p_t = O \left( \frac{\gamma_{rk} \log(n/\epsilon) \log(n) \log(nk)}{\gamma_{rk} \log(n)} \right), \tag{40}
\]

where \( L = \Theta(\gamma_{rk} \log(n/\epsilon)) \).
The remaining two lemmas follow from Theorem 6. We will apply this theorem to $\mathcal{P}^* + \epsilon$ with $\nu \in \sigma_{rk}(\|W^1 + \mathcal{X}_{t-1}\| + \epsilon)$ and $\tau$ as above. Since $|N^s * \mathcal{X}_{t-1}| \leq \sigma_{rk}(\|W^1 + \mathcal{X}_t\|)$, it holds that $\nu \geq \max(\|\mathcal{G}\|, |N^s * \mathcal{X}_{t-1}|)$ as required by Theorem 6, and it also requires a lower bound $\mu$. To satisfy the lower bound, we combine Lemma 5 states that with probability $1 - 1/n^2$, we have $\frac{1}{2}(\rho(\mathcal{G}) + \rho(N^s * \mathcal{X}_t)) \leq 10\mu^*$. The SmoothQR process produces with probability at least $1 - 1/n^2$ a tensor $\mathcal{H}_\ell$ such that $||\mathcal{H}_\ell||| \leq \frac{\tau \nu}{\gamma rk} \leq \frac{\gamma rk}{\tau \nu} \leq \frac{1}{4} (\frac{\gamma rk}{\tau \nu})$, satisfying the $\epsilon/k$-admissible condition. Therefore, the second and third claim hold.

Note that all error probabilities that incurred were less than $1/n^2$, then we sum up the error probabilities over all $L = \Theta(\gamma^{-1} log(n/\epsilon))$ steps, getting a probability at least $1 - \Theta(\gamma^{-1} log(n/\epsilon))$.

The resulting probability $p = p_0 + p_+$ would be

$$p_0 \geq \frac{6444r^2\mu(\mathcal{U})(||T||F/\gamma rk \sigma_{rk})^2 log n}{n} + \frac{64^{3/2}\mu(\mathcal{U})(||T||F/\gamma rk \sigma_{rk}) log n}{n},$$

$$p_+ = O\left(\frac{\nu^2 \mu^*(||\mathcal{M}||F^2 + ||N||F^2/\epsilon^2)(log(n/\epsilon)) log^2 nk}{\gamma^5 rk \sigma_{rk}}\right),$$

$$p = O\left(\frac{\nu^2 \mu^*(||\mathcal{M}||F^2 + ||N||F^2/\epsilon^2)(log(n/\epsilon)) log^2 nk}{\gamma^5 rk \sigma_{rk}}\right),$$

where $\mu^* = \max\{\mu(\mathcal{U}), \mu_N, log nk\}$.

The proof of Theorem 3 concludes.

**Lemma 5.** Under the assumptions of Theorem 1, we have for every $\ell \in [L]$ and $\nu = \sigma_{rk}(\|W^1 + \mathcal{X}_{t-1}\| + \epsilon)$ with probability $1 - 1/n^2$, $\frac{1}{2}(\rho(\mathcal{G}) + \rho(N^s \mathcal{X}_t)) \leq \mu^*$.

**Proof:** Given the lower bound on $p$ in Theorem 1, we apply Lemma 15 to conclude that $\|\nu_i^1 * G_i^M||F \leq \sqrt{\mu(\mathcal{U})/n} v$ and $\|\nu_i^1 * G_i^N||F \leq \sqrt{\mu^*/n} v$. Thus, $\rho(\mathcal{G})/v^2 \leq \mu^*$. Further, we claim that $\|\nu_i^1 * N \mathcal{X}_t||F \leq (\mu^*/n) \sigma_{rk}(\|W^1 + \mathcal{X}_t\|)$ for all $i \in [n]$ since

$$||\nu_i^1 * N \mathcal{X}_t||F \leq ||\nu_i^1 * W \Theta W||F ||W^1 + \mathcal{X}_t|| = ||\nu_i^1 * N||F ||W^1 + \mathcal{X}_t||,$$

where we used the following fact that

$$||\nu_i^1 * N \mathcal{X}_t||F^2 = ||\nu_i^1 * N \mathcal{X}_t||F^2 + ||\nu_i^1 * N \mathcal{X}_t||F^2 = ||\nu_i^1 * N \mathcal{X}_t||F^2.$$

Combining (21), this shows that $\rho(N^s * \mathcal{X}_{t-1})/v^2 \leq \mu^*$, and conclude the proof.

**C. Initialization**

Alg. 3 computes the top-$r$ eigenslopes of $\mathcal{P}_{\Omega_0}(T)$, and truncates them in a “scaling manner” to ensure the incoherence. This initialization procedure serves as an acceleration of our main algorithm Alg. 2. We analyze Alg. 3 and derive the required sampling probability $p_0$ to get a good starting point in Theorem 4. The corresponding proofs relies mainly on the matrix Bernstein inequality in Lemma 6 and the Davis-Kahan sin-$\theta$-theorem [45] in Lemma 8.

**Theorem 4.** Let $T \in \mathbb{R}^{n \times n \times k}$ be a symmetric square tensor with tubal-rank $r$. Assume that each element is included in $\Omega$ independently with probability

$$p_0 \geq \frac{644r^2\mu(\mathcal{U})(||T||F/\gamma rk \sigma_{rk})^2 log n}{n} + \frac{64^{3/2}\mu(\mathcal{U})(||T||F/\gamma rk \sigma_{rk}) log n}{n}$$

where $\sigma_{rk}$ denotes the $rk$-th singular value of the block diagonal matrix $\mathcal{T}$, and $\gamma rk = 1 - \frac{\gamma rk}{\gamma rk+1}. \sigma_{rk}$ Then, Alg. 3 returns an orthonormal tensor $\mathcal{X}_0 \in \mathbb{R}^{n \times n \times k}$ such that with probability at least $1 - 1/n^2$, we have

$$||W^1 * \mathcal{X}_0||F \leq 1/4, \quad \mu(\mathcal{X}_0) \leq 32\mu(\mathcal{U}) log n.$$

**Proof:** The proof follows directly from Lemma 7, Lemma 8, Lemma 9, and Lemma 10.

**Lemma 6.** [48] (Matrix Bernstein Inequality) Consider a finite sequence $\{\mathcal{Z}_i\}$ of independent random matrices with dimensions $d_1 \times d_2$. Assume that each random matrix satisfies $E \mathcal{Z}_i = 0$ and $||\mathcal{Z}_i|| \leq R$ almost surely. Define $\zeta^2 = \max\{||\sum_{i} E \mathcal{Z}_i \mathcal{Z}_i^H||, ||\sum_{i} E \mathcal{Z}_i^H \mathcal{Z}_i||\}$. Then, for all $u \geq 0$,

$$\mathbb{P}\left\{\left|\sum_{i} \mathcal{Z}_i\right| \geq u\right\} \leq (d_1 + d_2) \exp \left\{-\frac{u^2/2}{\zeta^2 + \frac{R^2}{3}}\right\}.$$
and \[ |E_{ijr}| \leq \frac{1}{p_0} ||T||_\infty. \]

\[
\mathbb{E} \left[ \sum_{i,j} \mathbf{e}_{ij}^\dagger \mathbf{e}_{ij} \right] = \mathbb{E} \left[ \sum_{i,j} \mathbf{e}_{ij}^\dagger \mathbf{e}_{ij} \right] = \left[ \sum_{i,j} \mathbf{T}_{ij}^2 \mathbf{e}_{ij} \mathbf{e}_{ij}^\dagger \right] = \left[ \sum_{i,j} \mathbf{T}_{ij}^2 \mathbf{e}_{ij} \mathbf{e}_{ij}^\dagger \right] = \left[ \frac{1}{p_0} \sum_{i,j} \mathbf{T}_{ij}^2 \mathbf{e}_{ij} \mathbf{e}_{ij}^\dagger \right]
\]

(48)

since \( \mathbf{e}_{ij} \) will return a zero tensor except for \((j,i,1)-th\) entry equaling 1, we have

\[
\mathbb{E} \left[ \sum_{i,j} \mathbf{e}_{ij}^\dagger \mathbf{e}_{ij} \right] = \frac{1 - \frac{1}{p_0}}{p_0} \max_j \left\{ \sum_{i,j} \mathbf{T}_{ij}^2 \right\} \leq \frac{1}{p_0} ||T||_{\infty,2^*},
\]

(49)

And similarly, \( \mathbb{E} \left[ \sum_{i,j} \mathbf{e}_{ij}^\dagger \mathbf{e}_{ij} \right] \leq \frac{1}{p_0} ||T||_{\infty,2^*} \) which is bounded. Then plugging into Lemma 6 concludes the proof.

Lemma 8. If \( p_0 \) satisfies that

\[
p_0 \geq \frac{6144r^2\mu(\mathcal{U})(||T||_F/\sqrt{r} ||T||_{rk})^2 \log n}{n} + \frac{64r^3/2\mu(\mathcal{U})(||T||_F/\sqrt{r} ||T||_{rk}) \log n}{n}
\]

where \( \sqrt{r} ||T||_{rk} \) denotes the \( rk \)-th singular value of the block diagonal matrix \( \mathbf{T} \) and \( \sqrt{r} \|T\|_{rk} = 1 - \sqrt{r} \|T\|_{rk+1} \), we obtain that \( \|P\mathcal{T}(T) - T\| \leq \frac{\sqrt{r} ||T||_{rk}}{32\sqrt{r}} \) holds with high probability at least 1 - \( \frac{1}{n^2} \). Then, we have

\[
P \left\{ \|W^* \cdot A\| \leq \frac{1}{10\sqrt{r}} \right\} > 1 - \frac{1}{n^2},
\]

(51)

where \( A \) is the tensor in Alg. 3.

Proof: We have that

\[
||T||_{\infty,2^*} = \frac{\mu(\mathcal{U})}{n} ||T||_F,
\]

\[
||T||_{\infty} = \frac{\mu(\mathcal{U})}{n} ||T||_F.
\]

(52)

Then \( \zeta^2 \) in Lemma 7 becomes

\[
\zeta^2 = \frac{1}{p_0} \max \left\{ \frac{r\mu(\mathcal{U})}{n} ||T||_F^2, \frac{r\mu(\mathcal{U})}{n} ||T||_F \right\} = \frac{r\mu(\mathcal{U})}{p_0n} ||T||_F^2.
\]

(53)

This can be re-arranged to get

\[
p_0 \geq \frac{2Cr\mu(\mathcal{U}) \log n}{n} ||T||_F^2 + \frac{2Cr\mu(\mathcal{U}) \log n}{n} ||T||_F.
\]

(55)

Set \( u = \frac{\gamma_{rk}^c (\gamma_{rk}^c + \sigma_{rk}^c)}{32 \sqrt{r}} \) for reasons to be clear in (57)), leading to the condition that \( p_0 \geq \frac{2048r^2\mu(\mathcal{U})(||T||_F/\sqrt{r} \gamma_{rk}^c \sigma_{rk}^c \log n)}{3n} + \frac{64r^3/2\mu(\mathcal{U})(||T||_F/\sqrt{r} \gamma_{rk}^c \sigma_{rk}^c \log n)}{3n}. \)

Let \( C = 3 \), plug in the above parameters into Lemma 7, we get:

\[
P \left\{ \|P\mathcal{T}(T) - T\| > \frac{\gamma_{rk}^c \sigma_{rk}^c}{32 \sqrt{r}} \right\} \leq \frac{1}{n^2}.
\]

(56)

Let \( W \) be the top \( r \) eigenspaces of \( \mathcal{T}(T) \), \( \sigma_{rk}^c \) denote the \( rk \)-th singular value of the circular matrix \( T^c \) and define \( \gamma_{rk}^c = 1 - \sigma_{rk+1}^c/\sigma_{rk}^c \). Now let us assume that \( ||T - P\mathcal{T}(T)|| \leq u \), then \( ||T^c - P\mathcal{T}(T^c)|| \leq u \), thus \( \gamma_{rk}^c (P\mathcal{T}(T^c)) \leq \sigma_{rk}^c \). \( \gamma_{rk}^c \leq \sigma_{rk}^c \sigma_{rk}^c - 2 \), and \( \sigma_{rk}^c (P\mathcal{T}(T^c)) - \sigma_{rk+1}^c (T^c) = \sigma_{rk}^c (P\mathcal{T}(T^c)) - \sigma_{rk}^c \sigma_{rk}^c + \gamma_{rk}^c \sigma_{rk}^c > \gamma_{rk}^c \sigma_{rk}^c / 2 \).

By Davis-Kahan sin \( \theta \)-theorem [45] and combining Definition 15, we have that

\[
\|W^* \cdot A\| = \|W^* \cdot A\| = \sin \theta_{rk}(U^c, A^c) \leq \|T^c - P\mathcal{T}(T^c)||\leq u = \frac{\gamma_{rk}^c \sigma_{rk}^c}{32 \sqrt{r}} \]

(57)

Note that the \( rk \)-th singular value \( \sigma_{rk}^c \) of the circular matrix \( T^c \) equals to that (i.e., \( \sigma_{rk}^c \)) of the block diagonal matrix \( \mathbf{T} \). Then the probability formula becomes \( p_0 \geq \frac{6144r^2\mu(\mathcal{U})(||T||_F/\sqrt{r} ||T||_{rk})^2 \log n}{n} + \frac{64r^3/2\mu(\mathcal{U})(||T||_F/\sqrt{r} ||T||_{rk}) \log n}{n}. \)

Lemma 9. Let \( \mathcal{A} \in \mathbb{R}^{n \times r \times k} \) be any orthonormal basis with \( \mathcal{A} \leq \mu \). Then, for a random orthonormal tensor \( \mathcal{O} \in \mathbb{R}^{n \times r \times k} \), we have \( \mathbb{P} \left\{ \max_{i,j} ||A \cdot O(i,j,:)||_F > \sqrt{8\mu} \log n/n \right\} \leq \frac{1}{n^2} \).

Proof: Consider a tube \( Z = [A \cdot O(i,j,:)] \). Note that \( ||Z||_F \) is distributed like a coordinate of a random vector \( \mathcal{F}^r \) of norm at most \( \sqrt{\mu r/n} \). By measure concentration, we obtain:

\[
P \left\{ ||Z||_F > \epsilon \sqrt{\mu r/n} \right\} \leq 4 \exp\left\{-2\epsilon^2 r/2 \right\}. \]

(58)

This follows from Levy’s Lemma [46], as in [35]. Set \( \epsilon = \sqrt{8\log n/r}, \) then

\[
P \left\{ ||Z||_F > \sqrt{8\mu} \log n/n \right\} \leq 4 \exp\left\{-4 \log n \right\} = 4n^{-4}. \]

(59)

Taking a union bound over all \( nr \leq n^2/4 \) tubes of \( W^* \cdot \mathcal{A}, \) we have that with probability \( 1 - 1/n^2, \)

\[
\max_{i,j} ||A \cdot O(i,j,:)||_F \leq \sqrt{8\mu} \log n/n. \]

(60)

Lemma 10. Assume that \( \|W^* \cdot A\| \leq \frac{1}{10\sqrt{r}} \). Then with probability \( 1 - 1/n^2, \) we have \( ||W^* \cdot \mathcal{X}||_F \leq 1/4 \) and \( \mu(\mathcal{X}) \leq 32\mu(\mathcal{U}) \log n. \)
Proof: Assume that \( \|W^\dagger \ast A\| \leq \frac{1}{16\tau} \), then there exists an orthonormal transformation \( Q \in \mathbb{R}^{r \times r \times k} \) such that \( \|U \ast Q - A\|_F \leq \frac{1}{16} \). Because of the following three facts: \( \|U \ast Q - A\|_F \leq \left( \sum_{i=1}^{r} \|\sigma_i\|^2 \right)^{1/2} \) where \( \sigma_i \) denotes the \( i \)-th eigentube of \( (U \ast Q - A) \), \( \|\sigma_i\|_F \geq \|\sigma_2\|_F \geq \ldots \geq \|\sigma_r\|_F \), and \( \|\sigma_i\|_F = \|W^\dagger \ast A\| \leq \frac{1}{16\tau} \).

Since \( \mu(U \ast Q) = \mu(U) \leq \mu \) (the orthonormal transformation \( Q \) does not change the incoherence of \( U \)). Therefore, \( A \) is close in Frobenius norm to an orthonormal basis of small coherence. However, it is possible that some tubes of \( U \ast Q \) have Frobenius norm as large as \( \mu \). Rotating \( U \ast Q \) by a random rotation, Lemma 9 asserts that with probability \( 1 - 1/n^2 \), \( \|U \ast Q \ast O(i, j, :)|_F \leq \sqrt{8\mu log n}/n \), for all \( i, j \). Moreover, because a rotation does not increase Frobenius norm, then we have \( \|U \ast Q \ast O - A \ast O\|_F \leq \frac{1}{16} \). Truncating the tubes of \( A \ast O \) that have Frobenius norm larger than \( \mu \) can therefore only decrease the coherence in Frobenius norm to \( U \ast Q \ast O \), hence, \( \|U \ast Q \ast O - Z'\|_F \leq \frac{1}{16} \).

Since truncation is a projection onto the set \( \{B : \|B(i, j, :)\|_F \leq \mu'\} \) with respect to Frobenius norm, we have:

\[
\|A \ast O - Z'\|_F \leq \|U \ast Q \ast O - Z'\|_F \leq \frac{1}{16}.
\]

We can write \( X' = Z' \ast R^{-1} \) where \( R \) is an invertible linear transformation with the same eigentubes as \( Z' \) and thus satisfies

\[
\|R^{-1}\| = \frac{1}{\|\sigma_1(Z')\|_F} \leq \frac{1}{\|\sigma_1(A \ast O - Z')\|_F} \leq \frac{1}{1 - 1/16} \leq 2.
\]

Therefore,

\[
\|e^i_j \ast X\| = \|e^i_j \ast T \ast R^{-1}\| \leq 2\|e^i_j \ast T\| \leq 2\sqrt{8\mu\mu U} log n/n.
\]

Hence,

\[
\mu(X) \leq \frac{n32\mu(U)}{8} \leq \frac{32\mu(U)}{n}.
\]

Therefore,

\[
\|W^\dagger \ast A\|_F \leq 2\|W^\dagger \ast T \ast R^{-1}\|_F \leq 2\|W^\dagger \ast T\|_F \leq 2\|W^\dagger \ast A \ast O\|_F + 2\|A \ast O - T\|_F
\]

\[
\leq 2\|W^\dagger \ast A\| + 1 \leq \frac{1}{2}.
\]

Thus, we can use the tensor least squares minimization update step, specialized to the case of a symmetric square tensor. Our goal in this section is to express the tensor least squares minimization update step as \( Y = T \ast X \ast G \), then we will be able to apply our convergence analysis of the noisy tensor-column subspace iteration in Appendix E. This syntactic transformation is given in Appendix D that is followed by a bound on the norm of the noise term \( G \) in Appendix D.2. With this, we prove in Appendix D.3 that the element-wise median process (in Alg. 4) on \( t = O(\log n) \) tensor least squares minimizations will result in a much tighter bound of the noise term \( G \) that is the average of those \( t = O(\log n) \) copies of \( G \).

1) From Alternating Least Squares to Noisy Tensor-Column Subspace Iteration: We first show that the tensor completion can be analyzed in its circular form in Lemma 11, then give an optimality condition in the circular form that the optimizer \( Y \) satisfies a set of linear equations in Lemma 12. With these constraints, we express the tensor least squares minimization update step as \( Y = T \ast X \ast G \) in Lemma 13.

**Lemma 11.** The function \( f(Y) = \|P_{[\Omega]}(T \ast X \ast Y^\dagger)\|_F^2 \) has an injective mapping to the circular form \( f(Y^c) = \frac{1}{|\kappa|}\|P_{[\Omega]}(T^c \ast X^c Y^c)^\dagger\|_F^2 \). (Note that in Section IV-A we show that those two objective functions are different.)

**Proof:** For the least squares update \( Y = \arg\min_{\hat{Y} \in \mathbb{R}^{r \times r \times k}} \|P_{[\Omega]}(T \ast X \ast \hat{Y})\|_F^2 \), it is equivalent to the following optimization problem:

\[
\min_{\hat{Y}} \|\hat{Y}\|_F^2, \quad \text{s.t.} \quad \hat{P}_{[\Omega]}(T \ast \hat{X} \ast \hat{Y}^\dagger) \}
\]

where \( G \) is a noise term. The circular form of (66) is:

\[
\min_{\hat{Y}} \frac{1}{|\kappa|}\|\hat{G}\|_F^2, \quad \text{s.t.} \quad \hat{P}_{[\Omega]}(T \ast \hat{X} \ast \hat{Y}^\dagger) \}
\]

Since the addition/subtraction and inverse operations are closed in the circulant algebra [38], then \( Y^c = (P^c T)^{-1}(P^c T \ast P^c G) \) is also circular.

We can see that \( f(Y) \) implies \( f(Y^c) \) while the opposite direction does not hold as shown in Section IV-A. Therefore, this mapping is injective.

**Lemma 12.** (Optimality Condition). Let \( P_i : \mathbb{R}^{rk} \rightarrow \mathbb{R}^{rk} \) be the linear projection onto the coordinates in \( \Omega_i = \{j : (i,j) \in \Omega\} \) scaled by \( p^{-1} = (nk)^2/(|\Omega|) \), i.e., \( P_i = p^{-1}\sum_{j \in \Omega_i} e^i_j e^i_j \) where \( e^i_j \) is the standard vector bases and \( e^i_j \) is the corresponding row basis. Further, define the matrix \( B_i \in \mathbb{R}^{rk \times rk} \) as \( B_i = X^c P_i X^c \) (note that \( B_i \) is invertible as shown in [35]). Then, for every \( i \in [nk] \), the \( i \)-th row of \( Y^c \) satisfies \( e^i Y^c = e^i T^c P_i X^c B_i^{-1} \).

**Proof:** By Lemma 11, we consider the circular objective function \( f(Y^c) = \frac{1}{|\kappa|}\|P^c T^c \ast X^c Y^c\|_F^2 \). For every \( i \in [nk] \), \( j \in [rk] \), we have \( \frac{\mu}{|\kappa|} \sum_{e^i_j} \sum_{\Omega_i} T_{i e^i_j} X^c_{e^i_j} X^c_{e^i_j} \) + \( \frac{2}{|\kappa|} \sum_{s=1}^{r k} Y^c_{i s} \sum_{s \in \Omega_i} X^c_{e^i_s} X^c_{e^i_s} \). Therefore, we know that the optimal \( Y^c \) must satisfy \( e^i T^c P_i X^c = e^i Y^c X^c P_i X^c = e^i Y^c B_i \), hence, \( e^i Y^c = e^i T^c P_i X^c B_i^{-1} \).

**Lemma 13.** Let \( E^c = (I^c - X^c X^c)^\dagger U^c \), and assume that \( T \) is a noisy tensor (approximately \( r \)-tubal-rank) that is the superposition of an exact \( r \)-tubal-rank tensor \( M \) and a noisy tensor \( N \), i.e., \( T = M + N \). We express the least squares update as \( Y = T \ast X \ast G \) where \( G = G_M + G_N \) and their circular matrices \( G_M \) and \( G_N \) satisfy that each row \( i \in [nk] \), we have the following expressions:

\[
e^i \ast G_M = e^i (T^c P_i X^c B_i^{-1} - N^c X^c).
\]

\[
e^i \ast G_N = e^i (N^c P_i X^c B_i^{-1} - N^c X^c).
\]
Proof: Since $T = \mathcal{M} + \mathcal{N}$, we have $\mathcal{N} = (\mathcal{I} - \mathcal{U} \ast \mathcal{U}^T) \ast T$. By Lemma 12, $e_i^\dagger Y^c = e_i^\dagger T^c P_i X^c B_i^{-1} = e_i^\dagger M^c P_i X^c B_i^{-1}$. Let $C_i = U_i^c P_i X^c$ and $D_i = U_i^c X^c$. Let $C_i = U_i^c P_i X^c$ and $D_i = U_i^c X^c$. We have:

$$e_i^\dagger M^c P_i X^c B_i^{-1} = -e_i^\dagger U_i^c \Delta U_i^c C_i B_i^{-1} = e_i^\dagger \left( U_i^c \Delta U_i^c D_i - U_i^c \Delta U_i^c (DB_i - C_i) B_i^{-1} \right) = e_i^\dagger \left( M_i^c X^c - e_i^\dagger U_i^c \Delta U_i^c (DB_i - C_i) B_i^{-1} \right),$$

$$C_i = U_i^c P_i X^c = (X^c)^T U_i^c e_i^c P_i X^c = (X^c)^T e_i^c P_i X^c = DB_i + E_i^c X^c.$$  

(69)

Then, we have $e_i^\dagger M^c P_i X^c B_i^{-1} = e_i^\dagger M_i^c X^c + e_i^\dagger U_i^c \Delta U_i^c E_i^c P_i X^c B_i^{-1}$. From (68) we know that $e_i^\dagger N_i^c P_i X^c B_i^{-1} = e_i^\dagger N_i^c X^c + e_i^\dagger G_i \Delta X^c$. Putting all together, we have $Y^c = M_i^c X^c + G_i \Delta X^c + G_i \Delta X^c = T_i^c X^c + G_i \Delta X^c + G_i \Delta X^c$, therefore, transforming it to the tensor form we have $Y = T_i^c X^c + G_i \Delta X^c$.

Let $\delta \in (0, 1)$. Assume that each entry is included in $\Omega$ independently with probability $p_{\mathcal{X}} \geq \frac{\mu(\mathcal{X}) \log nk}{\delta^2 n}$, (70) then, for all $i \in [n]$, $\mathbb{P}(\{||e_i^c \ast G_i||_F > \delta \left(||e_i^c \ast \mathcal{M}_i||_F \cdot ||W_i^c \ast X^c|| + ||e_i^c \ast N_i||_F\right) \}} \leq \frac{1}{\delta}$.  

(71)

Proof: The set $\Omega'$ is exactly $k$ replicates of $\Omega$, and the probability $p_{\mathcal{X}}$ of $\Omega'$ corresponds to one replica of each $\Omega_i$ replicates. According to Lemma 4, $p_{\mathcal{X}} \geq \frac{\mu(\mathcal{X}) \log nk}{\delta^2 n}$ for each replica in $\Omega'$, then we have $\mathbb{P}(\{||e_i^c \ast G_i||_F > \delta(||e_i^c \ast M^c|_F \cdot ||W_i^c \ast X^c|| + ||e_i^c \ast N_i||_F\}) \leq \frac{1}{\delta}$, corresponding to $\mathbb{P}(\{||e_i^c \ast G_i||_F > \delta(||e_i^c \ast \mathcal{M}_i||_F \cdot ||W_i^c \ast X^c|| + ||e_i^c \ast N_i||_F\}) \leq \frac{1}{\delta}$.  

(72)

Lemma 4. Note that $\mu(X^c) = k \mu(X)$, we have $p_{\mathcal{X}} \geq \frac{\mu(X^c) \log nk}{\delta^2 n}$.

3) Median Tensor Least Squares Minimization: Here, we further analyze the element-wise median process in Alg. 4. Given the previous error bound of $\mathcal{G}$ in Lemma 14, we can further derive a stronger concentration bound by taking the element-wise median of multiple independent samples of the error term.

Lemma 15. Let $\Omega$ be a sample set in which each element is included independently with probability $p_{\mathcal{X}}$. Let $\mathcal{G}_1, \mathcal{G}_2, ..., \mathcal{G}_t$ be i.i.d. copies of $\mathcal{G}$, and $\mathcal{G} = \text{median}(\mathcal{G}_1, \mathcal{G}_2, ..., \mathcal{G}_t)$ be the element-wise median, and assume $p_{\mathcal{X}}$ satisfy (70). Then, for every $i \in [n]$,

$$\mathbb{P}(\{||e_i^c \ast \mathcal{G}||_F > \delta \left(||e_i^c \ast \mathcal{M}_i||_F \cdot ||W_i^c \ast X^c|| + ||e_i^c \ast N_i||_F\right) \} \leq \exp(-\Omega(t)),$$

(73)

where $\Omega(t)$ denotes some polynomial of $t$.

Proof: For each $i \in [n]$, let $g_1, g_2, ..., g_t \in \mathbb{R}^{1 \times r \times k}$ denote the $i$-th horizontal slice of $\mathcal{G}_1, \mathcal{G}_2, ..., \mathcal{G}_t$. Let $S = \{j \in [t] : ||g_j||_F \leq B\}$ where $B = \frac{\delta}{4} \left(||e_i^c \ast \mathcal{M}_i||_F \cdot ||W_i^c \ast X^c|| + ||e_i^c \ast N_i||_F\right)$. Applying Lemma 14 with error parameter $\frac{\delta}{4}$, we have $\mathbb{E}[S] \geq 0.4t/5$ with $g_j$ being drawn independently. Then we apply a Chernoff bound to argue that $\mathbb{P}(|S| > 0.2t/3) \geq 1 - \exp(-\Omega(t))$.

Fixing a coordinate $s \in [r]$. By the median property we have $||g_j(1, s, :)||_F^2 \geq ||g_j||_F^2 \geq 0.5$. Since $|S| \geq 0.2t/3$, we know that at least $t/3$ horizontal slices with $j \in S$ have $||g_j(1, s, :)||_F^2 \geq ||g_j||_F^2$. Therefore, the average value of $||g_j(1, s, :)||_F^2$ over all $j \in [S]$ much be at least $\frac{1}{3}|S| \geq \frac{0.2t}{3} \geq ||g_j||_F^2$. This means that the average of $||g_j||_F^2$ over all $j \in [S]$ much be at least $||g_j||_F^2/3$. On the other hand, we also know that the average squared Frobenius norm in $S$ is at most $B^2/3$ by the definition of $S$. Then, the lemma is proved.

We then provide a strong concentration bound for the median of multiple independent samples to the tensor least squares minimization step.

Lemma 16. Let $\Omega$ be a sample set in which each element is included independently with probability $p_{\mathcal{X}} \geq \frac{\mu(X^c) \log nk}{\delta^2 n}$. Let $\mathcal{Y} = \text{MedianLS}(\mathcal{P}_\Omega(T), \Omega, X, r)$. Then, we have with probability $1 - 1/n^3$ that $\mathcal{Y} = T \ast X + \mathcal{G}$ with $\mathcal{G}$ satisfying:

$$||e_i^c \ast \mathcal{G}||_F \leq \delta \left(||e_i^c \ast \mathcal{M}_i||_F \cdot ||W_i^c \ast X^c|| + ||e_i^c \ast N_i||_F\right).$$

Proof: Using the Split($\Omega, t$) process, we know that the sample sets $\Omega_1, ..., \Omega_t, ..., \Omega_m$ are independent and each set $\Omega_j$ includes each element with probability at least $p_{\mathcal{X}}/t$. The output satisfies $\mathcal{Y}_j = \text{median} (\mathcal{Y}_1, ..., \mathcal{Y}_t, ..., \mathcal{Y}_m)$, where $\mathcal{Y}_j = T \ast X + \mathcal{G}_j$. Then median($\mathcal{Y}_1, ..., \mathcal{Y}_t, ..., \mathcal{Y}_m$) = $T \ast X + \mathcal{G}_t$. Therefore, apply Lemma 15 combining the fact $t = O(\log n)$, we take a union bound over all $n$ horizontal slices of $\mathcal{G}$ to conclude this lemma.

E. Convergence of Noisy Tensor-Column Subspace Iteration
Alg. 8 describes our noisy tensor-column subspace iteration, where $G_\delta(\mathcal{Y}_t)$ denotes the Gram-Schmidt process which orthonormalizes the lateral slices of the tensor $Z_\delta$. The detailed steps of the Gram-Schmidt process for third-order tubal-rank tensor is given in [25]. Note that Alg. 8 is different from the recently proposed power method [38]; 1) we simultaneously compute multiple top-k eigenslices while the power method considered only the top-1 eigenslice, 2) in each iteration $t$, the computation in Alg. 8 is perturbed by a tensor $G_\delta$ which can be adversarially and adaptively chosen, and 3) the Gram-Schmidt process for third-order tubal-rank tensor is introduced to manipulate $G_\delta$. 


Algorithm 8 Noisy Tensor-Column Subspace Iteration

Input: Tensor $T \in \mathbb{R}^{n \times n \times k}$, number of iterations $L$, target dimension $r$
Let $X_0 \in \mathbb{R}^{n \times r \times k}$ be an orthonormal tensor.
For $\ell = 1$ to $L$
Let $G_\ell \in \mathbb{R}^{n \times r \times k}$ be an arbitrary perturbation.
$Z_\ell \leftarrow T \ast X_{\ell-1} + G_\ell,$
$X_\ell \leftarrow GS(Z_\ell).$
Output: Tensor $X_L \in \mathbb{R}^{n \times r \times k}$.

For the matrix case, an important observation of [33]–[35] is that the least squares minimization can be analyzed as a noisy update step of the well known subspace iteration (or power method). Therefore, the convergence of the alternating minimization iteration is equivalent to the convergence of the noisy subspace iteration. The corresponding convergence analysis exploits the tangent function of the largest principle angle between the subspace $U$ spanned by the first $r$ singular vectors of the input matrix and the $r$-dimensional space spanned by the columns of the iterate $X_\ell$.

To show the convergence results of noisy tensor column subspace iteration, we use the largest principle angle between two tensor-column subspaces as the potential function. Borrowing idea from [38], we show that the noisy tensor-column subspace iteration can be transformed to $k$ parallel noisy subspace iterations in the frequency domain.

Lemma 17. The noisy tensor-columns subspace iteration in Alg. 8 converges at a geometric rate.5

Proof: The key iterative operations in Alg. 8 are

$$Z_\ell \leftarrow T \ast X_{\ell-1} + G_\ell,$$
$$X_\ell \leftarrow GS(Z_\ell).$$

Introducing the cft() operation in (38), we know that (74) be represented as follows:

$$\text{cft}(Y_\ell) \leftarrow \text{cft}(X) \ast \text{cft}(X_{\ell-1}) + \text{cft}(G_\ell).$$

This implies that (74) equals to $k$ parallel standard noisy subspace iteration in the frequency domain. Therefore, combining the convergence results of [35] that noisy subspace iteration converges at a geometric rate, our noisy tensor-columns subspace iteration in Alg. 8 will also converge at a geometric rate.

In the following, we first provide the definitions of principal angles and corresponding inequalities for the matrix case [35]. Then, we need to establish explicit inequalities along the iteration process, so that we will be able to bound the recovery error of Alg. 2.

Definition 17. Largest principal angle. Let $X, Y \in \mathbb{R}^{n \times r}$ be orthonormal bases for subspaces $S_X, S_Y$, respectively. Then,

the sine of the largest principal angle between $S_X$ and $S_Y$ is defined as $\sin \theta(S_X, S_Y) = \| (I - XX^\dagger) Y \|$. 

Lemma 18. (Matrix Local Convergence) Let $0 \leq \epsilon \leq 1/4$, $\Delta = \max_{1 \leq \ell \leq L} ||G_\ell||$, and $\gamma_r = 1 - \sigma_{r+1}/\sigma_r$. Assume that $||W^\dagger X_0|| \leq 1/4$ and $\sigma_r \geq 8/\gamma_r. Then$

$$||W^\dagger X_{\ell}|| \leq \max\{\epsilon, 2||W^\dagger X_0|| \exp(-\gamma_r \ell/2)\}$$ (76)

Similarly we can prove the following lemma for our tensor case.

Lemma 19. (Tensor Local Convergence) Let $0 \leq \epsilon \leq 1/4$, $\Delta = \max_{1 \leq \ell \leq L} ||G_\ell||$, and $\gamma_r = 1 - \sigma_{r+1}/\sigma_r$. Assume that $||W^\dagger X_0|| \leq 1/4$ and $\sigma_r \geq 8/\gamma_r. Then$

$$||W^\dagger X_{\ell}|| \leq \max\{\epsilon, 2||W^\dagger X_0|| \exp(-\gamma_r \ell/2)\}$$ (77)

Proof: According to Lemma 4, we have

$$||W^\dagger X_{\ell}|| = ||W^\dagger X_{\ell}|| = ||W^\dagger X_{\ell}||.$$ (78)

This means that the largest principle angle between $W^\dagger$ and $X_\ell$ equals to that of these two tensor-column subspaces in the frequency domain.

Note that $||W^\dagger X_0|| \leq 1/4$ will be provided in Lemma 4, thus $||W^\dagger X_0|| \leq 1/4$. Let $\Delta = \max_{1 \leq \ell \leq L} ||G_\ell||$, $\gamma_r = 1 - \sigma_{r+1}/\sigma_r$, and $\gamma_r \geq 8/\gamma_r. From Definition 5 and 7, we know that a tensor $T$ with tubal-rank $r$ has a corresponding block diagonal matrix $T$ with rank $r k$. Therefore, applying Lemma 18 we get

$$||W^\dagger X_{L}|| \leq \max\{\epsilon, 2||W^\dagger X_0|| \exp(-\gamma_r L/2)\} = \max\{\epsilon, 2||W^\dagger X_0|| \exp(-\gamma_r L/2)\}.$$ (79)

Combining with (78), the lemma is proved.

To prove the convergence of the noisy tensor-column subspace iteration, we show that the error term $||G_\ell||$ decrease as $\ell$ increases and Alg. 8 starts to converge. We define the following condition as a convergence bound for this type of shrinking error.

Definition 18. (Tensor $\epsilon$-admissible). Let $\gamma_r = 1 - \sigma_{r+1}/\sigma_r$. We say that the pair of tensors $(X_{\ell-1}, G_\ell)$ is $\epsilon$-admissible for the noisy tensor-column subspace iteration in Alg. 8 if

$$||G_\ell|| \leq \frac{1}{32} \gamma_r \sigma_r ||W^\dagger X_{\ell-1}|| + \epsilon \frac{32}{32} \gamma_r \sigma_r.$$ (80)

One can say that a sequence of tensors $\{(X_{\ell-1}, G_\ell)\}$ is $\epsilon$-admissible for the noisy tensor-column subspace iteration in Alg. 8 if each element of this sequence is $\epsilon$-admissible. In the following we will use the notation $\{G_{\ell}\}_{\ell=1}^L$ as a shorthand for $\{(X_{\ell-1}, G_\ell)\}_{\ell=1}^L$.

With Lemma 19 and Definition 18, we are able to get the following convergence guarantee for admissible noise tensors.

Theorem 5. Let $\gamma_r = 1 - \sigma_{r+1}/\sigma_r$, and $\epsilon \leq 1/2$. Assume that the sequence of noisy tensors $\{G_{\ell}\}$ is $(\epsilon/2)$-admissible for the noisy tensor-columns subspace iteration in Alg. 8 and that $||W^\dagger X_0|| \leq 1/4$. Then, we have $||W^\dagger X_{L}|| \leq \epsilon$ for any $L \geq 4\gamma_r^{-1} \log(1/\epsilon)$. 

---

5We do not explicitly state the convergence rate because the one in [35] depends on the condition number, while we encounter a constant convergence rate from our experiments.
Proof: We prove by induction that for every integer \( t \geq 0 \) after \( L_t = 4t\gamma_{rk}^{-1} \) steps, we have \( ||W^t \ast X_{L_t}|| \leq \max\{2^{-t+1}, \epsilon\} \). For the base case \( t = 0 \), the lemma holds because of the assumption that \( ||W^0 \ast X_0|| \leq 1/4 \). For \( t \geq 1 \), we assume that \( ||W^t \ast X_{L_t}|| \leq \max\{2^{-(t+1)}, \epsilon\} \). Apply Lemma 19 with \( X_0 = X_{L_t} \), error parameter \( \max\{2^{-t+2}, \epsilon\} \) and \( L = L_{t+1} - L_t = 4/\gamma_{rk} \). The conditions of the lemma are satisfied due to the assumption that \( \{G_t\} \) is \( \epsilon/2 \)-admissible. Therefore, we get
\[
||W^t \ast X_{L_{t+1}}|| \leq \max\{\epsilon, 2 \max\{2^{-(t+1)}, \epsilon\} \cdot \exp(-\gamma_{rk}(L_{t+1} - L_t)/2)\} \leq \max\{\epsilon, 2^{-t+2}\} \]
(81)

F. Incoherence via the SmoothQR Procedure

As a requirement for our proof in Appendix B, we need to show that each intermediate solution \( Y_t \) (accordingly \( X_t \)) has small coherence. Lemma 21 states that applying the SmoothQR factorization (in Alg. 6) on \( Y_t \) will return a tensor \( Z \) satisfying this coherence requirement. Note that before orthonormalizing \( X_t \), a small Gaussian perturbation \( \mathcal{H} \) is added to \( Y_t \). There exists such noisy term that will cause little effect as long as its norm is bounded by that of \( G_t \).

Lemma 20. Let \( G \in \mathbb{R}^{n \times r \times k} \) be any tensor with \( ||G|| \leq 1 \), \( W \in \mathbb{R}^{n \times (n-r) \times k} \) be an \((n-r)\) dimensional tensor-column subspace with orthogonal projection \( P_W \), and \( H \in \mathbb{R}^{n \times r \times k} \) be a random Gaussian tensor. Assume that \( r = o(n/\log n) \) (where the order \( o(n) \) is defined at the beginning of Section II). Then, with probability \( 1 - \exp(-\Omega(n)) \), we have
\[
\sigma_{rk}(P_W(G + H)) \geq \Omega(\tau).
\]
Proof: Consider a tensor \( X \in \mathbb{R}^{r \times 1 \times k} \) with \( ||X||_F = 1 \), we have
\[
||P_W(G + H) \ast X||^2 > ||P_W \ast H \ast X||^2 - ||P_W \ast G \ast X||^2.
\]
(82)

Note that \( g = H \ast X \in \mathbb{R}^{n \times 1 \times k} \) follows the distribution \( N(0, \tau^2/2) \), where \( P_W \ast G \ast X \) has spectral norm at most 1, and \( W \) is a \( n \times k \) dimensional tensor-column subspace, and \( h = P_W \ast H \ast X \) follows the distribution \( N(0, \tau^2/2) \). Then, we need to lower bound \( ||h||^2 - ||y, h|| \). Since \( E||h||^2 > \tau^2/2 \), by standard concentration bounds for the norm of a Gaussian variable, we get
\[
P\{|||h||^2 \leq \tau^2/4\} \leq \exp(-\Omega(n)).
\]
(83)

On the other hand, \( \langle y, h \rangle \) is distributed like a one-dimensional Gaussian variable of variance at most \( \tau^2/n \). By Gaussian tail bounds, \( P\{|\langle y, h \rangle^2 > \tau^2/8\} \leq \exp(-\Omega(n)) \). Therefore, with probability 1 - \( \exp(-\Omega(n)) \), we have
\[
||P_W(G + H) \ast X|| > \Omega(\tau)
\]
for all unit tensors \( X \in \mathbb{R}^{r \times 1 \times k} \), i.e., \( \sigma_{rk}(P_W(G + H)) > \Omega(\tau) \).

Note that \( \exp(\Omega(r \log r)) = \exp(o(n)) \), hence this event occurs with probability \( 1 - \exp(\Omega(n)) \).

We introduce a variant of \( \mu \)-coherence, i.e., \( \rho \)-coherence, that applies to tensors rather than tensor-column subspaces. The next lemma (Lemma 21) show that adding a Gaussian noise term leads to a bound on the coherence after applying the QR-factorization.

Definition 19. \((\rho \)-coherence\). Given a tensor \( G \in \mathbb{R}^{n \times r \times k} \) we let \( \rho(G) = \max_{i \in [n]} ||\hat{e}_i \ast G||_F^2 \).

Lemma 21. Let \( r = \Omega(n/\log n) \) and \( \tau \in (0, 1) \). Let \( U \in \mathbb{R}^{n \times r \times k} \) be an orthonormal tensor, and \( G \in \mathbb{R}^{n \times r \times k} \) be a tensor such that \( ||G|| \leq 1 \). Let \( \mathcal{H} \sim N(0, \tau^2/2) \) be a random Gaussian tensor. Then, with probability \( 1 - \exp(-\Omega(n)) - n^{-5} \), there exists an orthonormal tensor \( \mathcal{Q} \in \mathbb{R}^{n \times 2r \times k} \) such that
\[
\begin{align*}
\mathcal{R}(\mathcal{Q}) &= \mathcal{R}(\mathcal{U} \vert \mathcal{G} + \mathcal{H}) \quad \text{where } \mathcal{R}(\mathcal{Q}) \text{ denotes the range of } \mathcal{Q}; \\
\mu(\mathcal{Q}) &\leq \Omega(\frac{1}{\tau}(\rho(\mathcal{G}) + \mu(\mathcal{U} + \log n))).
\end{align*}
\]
Proof: First, \( \mathcal{R}(\mathcal{U} \vert \mathcal{G} + \mathcal{H}) = \mathcal{R}(\mathcal{U} \vert (I - \mathcal{U} \ast \mathcal{U}^\dagger) \ast (\mathcal{G} + \mathcal{H})) \). Applying the QR-factorization to \( \mathcal{U} \ast \mathcal{B} \), we can find two orthonormal tensors \( Q_1, Q_2 \in \mathbb{R}^{n \times r \times k} \) such that \( Q_1, Q_2 = \mathcal{U} \ast \mathcal{B} \ast \mathcal{R}^{-1} \) where \( \mathcal{R} \in \mathbb{R}^{r \times r \times k} \). Since \( \mathcal{U} \) is already orthonormal, we can have \( Q_1 = \mathcal{U} \). Furthermore, the lateral slices of \( B \) are orthogonal to \( U \) and thus we apply the QR-factorization to \( U \) and \( B \) independently.

Applying Lemma 20 to the \((n-r)\)-dimensional tensor-column subspace \( U^\perp \) and the tensor \( G + \mathcal{H} \), we get that with probability \( 1 - \exp(-\Omega(n)) \), \( \sigma_{rk}(\mathcal{B}) \geq \Omega(\tau) \). Assume that this hold in the following.

We verify the second condition. We have
\[
\frac{n}{r} ||\hat{e}_i \ast \mathcal{Q}||_F^2 = \frac{n}{r} ||\hat{e}_i \ast \mathcal{U}||_F^2 + \frac{n}{r} ||\hat{e}_i \ast \mathcal{B} \ast \mathcal{R}^{-1}||_F^2
\]
\[
\geq \mu(\mathcal{U}) + \frac{n}{r} ||\hat{e}_i \ast \mathcal{B} \ast \mathcal{R}^{-1}||_F^2.
\]
(84)

On the other hand, we also have
\[
\frac{n}{r} ||\hat{e}_i \ast \mathcal{B} \ast \mathcal{R}^{-1}||_F^2 \leq \frac{n}{r} ||\hat{e}_i \ast \mathcal{B}||_F^2 ||\mathcal{R}^{-1}||_F^2
\]
\[
\leq \rho(\mathcal{B})^2 + 2p(\mathcal{B} \ast \mathcal{R}^{-1}),
\]
(85)

where we used the fact that \( ||\mathcal{R}^{-1}||_F = 1/\sigma_{rk}(\mathcal{R}) = O(1/\tau) \). Moreover, we have
\[
\frac{n}{r} ||\hat{e}_i \ast \mathcal{B} \ast \mathcal{R}^{-1}||_F^2 \leq 2p(\mathcal{B} \ast \mathcal{R}^{-1}) + 2p(I - \mathcal{U} \ast \mathcal{U}^\dagger \ast \mathcal{H})
\]
\[
\leq 2p(G) + 2p(\mathcal{U} \ast \mathcal{U}^\dagger \ast \mathcal{G}) + 2p(I - \mathcal{U} \ast \mathcal{U}^\dagger \ast \mathcal{G}).
\]
(86)

Note that \( \rho(\mathcal{U} \ast \mathcal{U}^\dagger \ast \mathcal{G}) \leq \mu(\mathcal{U}) ||\mathcal{U} \ast \mathcal{U}^\dagger||_F^2 \leq \mu(\mathcal{U}) \).

Combining the Lemma 22 (in the following), we have \( (I - \mathcal{U} \ast \mathcal{U}^\dagger) \ast \mathcal{H} \leq O(\log n) \) with probability \( 1 - n^{-5} \). Summing up the probability concludes the lemma.

Lemma 22. Let \( \mathcal{P} \) be the projection onto an \((n - r)\)-dimensional tensor-column subspace. Let
\[ \mathcal{H} \sim N(0, 1/n)^{n \times r \times k}. \text{ Then, } \rho(P \ast \mathcal{H}) \leq O(\log n) \text{ with probability } 1 - 1/n^2. \]

Proof: Let \( P = (I - U \ast U^\dagger) \) for some \( r \)-dimensional tensor-column basis \( U \). Then,
\[
\rho(P \ast \mathcal{H}) \leq \rho(\mathcal{H}) + \rho(U \ast U^\dagger \ast \mathcal{H}). \tag{87}
\]
Using concentration bounds for the norm of each horizontal slice of \( \mathcal{H} \) and a union bound over all horizontal slices, it follows that
\[
\rho(\mathcal{H}) \leq O(\log n) \text{ with probability } 1 - 1/2n^5,
\rho(U \ast U^\dagger \ast \mathcal{H}) \leq \rho(U)\|U^\dagger \ast \mathcal{H}\|^2. \tag{88}
\]
Note that \( U^\dagger \ast \mathcal{H} \) is a Gaussian tensor following the distribution \( N(0, 1/n)^{r \times r \times k} \) and its largest singular value satisfies \( \|U^\dagger \ast \mathcal{H}\|^2 \leq O(r \log n/n) \) with probability \( 1 - \frac{1}{2n} \). Summing up the probability concludes the lemma.

The next theorem states that when SmoothQR is called on an input of the form \( T * \mathcal{X} + \mathcal{G} \) with suitable parameters, the algorithm outputs a tensor of the form \( \mathcal{X}' = QR(T * \mathcal{X} + \mathcal{G} + \mathcal{H}) \) whose coherence is bounded in terms of \( \mathcal{G} \) and \( \rho(\mathcal{G}) \), and \( \mathcal{H} \) satisfies a bound on its norm.

Theorem 6. Let \( \tau > 0, r = \Omega(n/\log n), \mathcal{G} \in \mathbb{R}^{n \times r \times k}, \) and \( \mathcal{X} \in \mathbb{R}^{n \times r \times k} \) be an orthonormal tensor such that \( v \geq \max\{|\mathcal{G}|, |\mathcal{N} \ast \mathcal{X}|\} \). There exists a constant \( C > 0 \), assume that
\[
\mu \geq \frac{C}{\tau^2} \left( \frac{\rho(\mathcal{G}) + \rho(\mathcal{N} \ast \mathcal{X})}{v^2} \right) + \log n, \tag{89}
\]
then, for every \( \varepsilon \leq \tau v \) satisfying \( \log(n/\varepsilon) \leq n \) and every \( \mu \leq n \) we have with probability \( 1 - O(n^{-4}) \), the algorithm SmoothQR (Alg. 6) terminates in \( (n/\varepsilon) \) steps and outputs \( (\mathcal{X}', \mathcal{H}) \) such that \( \mu(\mathcal{X}') \leq \mu \) and \( \|\mathcal{H}\| \leq \tau v \).

Proof: If the algorithm SmoothQR (Alg. 6) terminates in an iteration where \( \sigma \leq \tau^2 v^2/4 \) (proved in Lemma 23), we claim that in this case, with probability \( 1 - \exp(-\Omega(n)) \) we must have that \( \|\mathcal{H}\| \leq \tau v \). Assume that the algorithm SmoothQR (Alg. 6) terminates in an iteration where \( \sigma \leq \tau^2 v^2/r \), then the algorithm takes at most \( t = O(\log(n/\varepsilon)) \leq O(n) \) steps.

Let \( \mathcal{H}_1, \ldots, \mathcal{H}_t \) denote the random Gaussian tensors generated in each step. We claim that each of them satisfies \( \|\mathcal{H}\| \leq \tau v \). Note that for all \( t \) we have \( \mathbb{E}[\|\mathcal{H}_t\|^2 \leq \tau^2 v^2/4 \). The claim therefore follows directly from tail bounds for the Frobenius norm of Gaussian random tensors and holds with probabilities \( 1 - \exp(-\Omega(n)) \).

Lemma 23. With probability \( 1 - O(n^{-4}) \), Alg. 6 terminates in an iteration where \( \sigma \leq \tau^2 v^2/4 \), where \( \sigma \) is the intermediate parameter used in Alg. 6.

Proof: Consider the first iteration in which \( \sigma \leq \tau^2 v^2/8 \). Let us define \( \mathcal{G}' = (\mathcal{N} \ast \mathcal{X} + \mathcal{G})/2v \). Apply Lemma 21 to the tensor \( \mathcal{G}' \), which satisfies the required assumption that \( \|\mathcal{G}'\|^2 \leq 1 \). Lemma 21 states that with probability \( 1 - O(n^{-4}) \), there is an orthonormal \( n \times 2r \times k \) tensor \( \mathcal{Q} \) such that
\[
\mathcal{R}(\mathcal{Q}) = \mathcal{R}(\{\mathcal{G}' + \mathcal{H} \}) = \mathcal{R}(\{\mathcal{G} + \mathcal{N} \ast \mathcal{X} + \mathcal{H} \}),
\mu(\mathcal{Q}) \leq O(\frac{1}{\tau^2} (\rho(\mathcal{G}) + \mu(U + \log n)))) \tag{90}
\]
On one hand, we have
\[
\mathcal{R}(\mathcal{A}') = \mathcal{R}(T * \mathcal{X} + \mathcal{G} + \mathcal{H}) \subset \mathcal{R}(\{\mathcal{G} + \mathcal{N} \ast \mathcal{X} + \mathcal{H} \}) = \mathcal{R}(\mathcal{W}), \tag{91}
\]
where we use the fact that \( \mathcal{U} \) is an orthonormal basis for the range of \( \mathcal{M} \ast \mathcal{X} = \mathcal{U} \Theta \ast \mathcal{U}^\dagger \ast \mathcal{X} \). On the other hand, \( \mu(\mathcal{A}') \leq 2\mu(\mathcal{Q}) \leq \mu. \) This lemma is proved as long as \( C \) is large enough.

Lemma 24. Let \( \mathcal{X}, \mathcal{Y} \) be \( r \) and \( r' \) dimensional tensor-column subspaces, respectively, such that \( \mathcal{X} \subset \mathcal{Y} \). Then, we have
\[
\mu(\mathcal{X}') \leq \frac{1}{\tau} \mu(\mathcal{Y}). \tag{92}
\]
Proof: We know that \( \mu(\mathcal{Y}) \) is rotationally invariant. Therefore, without loss of generality we assume that \( \mathcal{Y} = [\mathcal{X} | \mathcal{X}'] \) for some orthonormal tensor \( \mathcal{X}' \). Here, we identify \( \mathcal{X} \) and \( \mathcal{Y} \) with orthonormal bases. Therefore,
\[
\mu(\mathcal{X}) = n \max_{r \in [n]} |\mathcal{e}'_r \ast \mathcal{X}'|^2_F \leq n \max_{r \in [n]} \left( |\mathcal{e}'_r \ast \mathcal{X}'|^2_F + |\mathcal{e}'_r \ast \mathcal{X}''|^2_F \right) \leq \frac{r'}{r} \mu(\mathcal{Y}). \tag{92}
\]

References
[1] X. Liu, S. Aeron, V. Aggarwal, and X. Wang, “Low-tubal-rank tensor completion using alternating minimization,” Proc. SPIE, vol. 9848, May 2016, Art. no. 984809.
[2] R. G. Baraniuk, “More is less: Signal processing and the data deluge,” Science, vol. 331, no. 6018, pp. 717–719, Feb. 2011.
[3] A. Cichocki et al., “Tensor decompositions for signal processing applications: From two-way to multiway component analysis,” IEEE Signal Process. Mag., vol. 32, no. 2, pp. 145–163, Mar. 2015.
[4] E. J. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” IEEE Trans. Inf. Theory, vol. 52, no. 2, pp. 489–509, Feb. 2004.
[5] Q. Song, H. Ge, J. Caverlee, and X. Hu, “Tensor completion algorithms in big data analytics,” 2017, arXiv:1711.10105. [Online]. Available: https://arxiv.org/abs/1711.10105
[6] R. Little and D. B. Rubin, Statistical Analysis With Missing Data. Hoboken, NJ, USA: Wiley, 2014.
[7] L. Kong, L. He, X.-Y. Liu, Y. Gu, M.-Y. Wu, and X. Liu, “Privacy-preserving compressive sensing for crowdsensing based trajectory recovery,” in Proc. IEEE ICDCS, Jun./Jul. 2015, pp. 31–40.
[8] X.-Y. Liu, S. Aeron, V. Aggarwal, X. Wang, and M.-Y. Wu, “Adaptive sampling of RF fingerprints for fine-grained indoor localization,” IEEE Trans. Mobile Comput., vol. 15, no. 10, pp. 2411–2423, Oct. 2016.
[9] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, “Spatio-temporal compressive sensing and Internet traffic matrices,” ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 4, pp. 267–278, 2009.
[10] S. Rallapalli, L. Qiu, Y. Zhang, and Y.-C. Chen, “Exploiting temporal stability and low-rank structure for localization in mobile networks,” in Proc. Mobihoc, 2010, pp. 161–172.
Y.-C. Chen, L. Qiu, Y. Zhang, Z. Hu, and G. Xue, “Robust network compressive sensing,” in Proc. MobiCom, 2014, pp. 545–556.

L. Kong, M. Xia, X.-Y. Liu, X. Liu, and M.-Y. Wu, “Data loss and reconstruction in sensor networks,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 1654–1662.

X.-Y. Liu, Y. Zhu, L. Kong, C. Liu, Y. Gu, and M.-Y. Wu, “CDC: Compact data collection for wireless sensor networks,” IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 8, pp. 2188–2197, Aug. 2015.

M. Hardt and M. Wootters, “Fast matrix completion without the condition number,” in Proc. Advances in Neural Inf. Process. Syst. (NIPS), 2014, pp. 2861–2869.

K. Branan, “Third-order tensors as linear operators on a space of matrices,” Linear Algebra Appl., vol. 433, no. 7, pp. 1241–1253, 2010.

R. Sun and Z. Q. Luo, “Guaranteed matrix completion via non-convex factorization,” in Proc. IEEE 56th Annu. Symp. Found. Comput. Sci., Oct. 2015, pp. 270–281.

V. Aggarwal, A. A. Mahinmkar, H. Ma, Z. Zhang, S. Aeron, and W. Willinger, “Inferring smartphone service quality using tensor methods,” in Proc. 12th Int. Conf. Netw. Service Manage. (CNSM), Oct./Nov. 2016, pp. 263–267.

P. Jain and S. Oh, “Provable tensor factorization with missing data,” in Proc. ACM NIPS, 2014, pp. 1431–1439.

H. Kasai and B. Mishra, “Low-rank tensor completion: A Riemannian manifold preconditioning approach,” in Proc. ACM ICML, 2016, pp. 1012–1021.

C. Davis and W. M. Kahan, “The rotation of eigenvectors by a perturbation,” SIAM J. Numer. Anal., vol. 7, pp. 1–46, 1970.

J. Matousek, Lectures on Discrete Geometry. New York, NY, USA: Springer-Verlag, 2001.

Video Results, Accessed: Oct. 2016. [Online]. Available: http://web.ics.purdue.edu/~vaneet/software/tsvd_altmin/video.zip

Shuchin Aeron (M’08–SM’15) received the Ph.D. degree in computer science from the Huazhong University of Science and Technology in 2010, the Ph.D. degree in computer science from Shanghai Jiao Tong University in 2016, and the M.S. degree in electrical engineering from Columbia University in 2017, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering. His research interests include tensor theory, non-convex optimization, deep learning, big data analysis, and data privacy.

Xiao-Yang Liu (S’15) received the B.Eng. degree in computer science from the Huazhong University of Science and Technology in 2010, the Ph.D. degree in computer science from Shanghai Jiao Tong University in 2016, and the M.S. degree in electrical engineering from Columbia University in 2017, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering. His research interests lie in the area of signal processing, information theory, and optimal sampling and recovery. He was a recipient of the Best Thesis Award from the College of Engineering and the Department of ECE in 2009, the Center of Information and Systems Engineering (CISE) Award from Boston University in 2006, and the Schumacher-Doll Research Grant in 2007.
Vaneet Aggarwal (S’08–M’11–SM’15) received the B.Tech. degree from IIT Kanpur, Kanpur, India, in 2005, and the M.A. and Ph.D. degrees from Princeton University, Princeton, NJ, USA, in 2007 and 2010, respectively, all in electrical engineering.

He has been an Associate Professor with Purdue University, West Lafayette, IN, USA, since January 2015. He was a Senior Member of Technical Staff Research with AT&T Labs-Research, NJ, from 2010 to 2014, an Adjunct Assistant Professor with Columbia University, NY, from 2013 to 2014, and the VAJRA Adjunct Professor with IISc Bengaluru from 2018 to 2019. His current research interests are in communications and networking, cloud computing, and machine learning.

Dr. Aggarwal received the Princeton University’s Porter Ogden Jacobus Honorific Fellowship in 2009, the AT&T Vice President Excellence Award in 2012, the AT&T Key Contributor Award in 2013, the AT&T Senior Vice President Excellence Award in 2014, the 2017 Jack Neubauer Memorial Award recognizing the Best Systems Paper published in the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and the 2018 Infocom Workshop HotPOST Best Paper Award. He is currently on the Editorial Board of the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, and the IEEE/ACM TRANSACTIONS ON NETWORKING.

Xiaodong Wang (S’98–M’98–SM’04–F’08) received the Ph.D. degree in electrical engineering from Princeton University. He is currently a Professor of electrical engineering with Columbia University, New York, NY, USA. His research interests fall in the general areas of computing, signal processing, and communications. He has authored extensively in these areas. He has authored the book entitled *Wireless Communication Systems: Advanced Techniques for Signal Reception* (Prentice Hall, 2003). His current research interests include wireless communications, statistical signal processing, and genomic signal processing. He received the 1999 NSF CAREER Award, the 2001 IEEE Communications Society and Information Theory Society Joint Paper Award, and the 2011 IEEE Communication Society Award for Outstanding Paper on New Communication Topics. He has served as an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the IEEE TRANSACTIONS ON INFORMATION THEORY. He is also an ISI Highly Cited Author.