BIT Numerical Mathematics (2022) 62:171–194
https://doi.org/10.1007/s10543-021-00877-w

BIT

Randomized Kaczmarz for tensor linear systems

Anna Ma[1] · Denali Molitor[2]

Received: 10 June 2020 / Accepted: 22 April 2021 / Published online: 17 May 2021

Abstract
Solving linear systems of equations is a fundamental problem in mathematics. When the linear system is so large that it cannot be loaded into memory at once, iterative methods such as the randomized Kaczmarz method excel. Here, we extend the randomized Kaczmarz method to solve multi-linear (tensor) systems under the tensor–tensor t-product. We present convergence guarantees for tensor randomized Kaczmarz in two ways: using the classical matrix randomized Kaczmarz analysis and taking advantage of the tensor–tensor t-product structure. We demonstrate experimentally that the tensor randomized Kaczmarz method converges faster than traditional randomized Kaczmarz applied to a naively matricized version of the linear system. In addition, we draw connections between the proposed algorithm and a previously known extension of the randomized Kaczmarz algorithm for matrix linear systems.

Keywords Multilinear systems · Randomized Kaczmarz · Tensor product

Mathematics Subject Classification 65F10 · 65F20 · 65F25 · 15A69

Communicated by Daniel Kressner.

This work began at the 2019 workshop for Women in Science of Data and Math (WISDM) held at the Institute for Computational and Experimental Research in Mathematics (ICERM). This workshop is partially supported by an NSF ADVANCE grant (award #1500481) to the Association for Women in Mathematics (AWM). Ma was partially supported by U.S. Air Force Award FA9550-18-1-0031 led by Roman Vershynin. Molitor is grateful to and was partially supported by NSF CAREER DMS #1348721 and NSF BIGDATA DMS #1740325 led by Deanna Needell. The authors would also like to thank Misha Kilmer for her advising during the WISDM workshop and valuable feedback that improved earlier versions of this manuscript.

✉ Anna Ma
anna.ma@uci.edu

Denali Molitor
dmolitor@math.ucla.edu

[1] University of California, Irvine, Irvine, USA

[2] University of California, Los Angeles, Los Angeles, USA

1 Introduction

Methods for processing and analyzing large datasets have seen rapid development and use in signal processing and machine learning. Data are often organized in a two dimensional (user-item, pixel-frame, etc.) fashion because a vast majority of the existing methods operate on data that are stored as matrices and vectors [17,18,21]. However, in reality, data can be higher-order multidimensional arrays and this restriction to the one or two dimensional representations often destroys inherent structure (for example, spatial or temporal structure) Tensor methods aim to retain and take advantage of natural multidimensional structure.

Here, we consider the fundamental problem of solving large linear systems of equations for third-order tensors under the t-product. In the matrix linear system setting, randomized iterative methods are a popular choice for solving or finding approximate solutions to systems that are too large to load into memory at once [7,20,34]. One such randomized iterative method is known as the randomized Kaczmarz method. The randomized Kaczmarz method (MRK)[1] is closely related to other popular randomized iterative methods such as stochastic gradient descent and coordinate descent and is commonly used in computed tomography (CT imaging) and other signal processing applications [10,26]. In this work, we extend the Kaczmarz method to tensor linear systems under the t-product. Before summarizing our contributions, we first present the Randomized Kaczmarz method, it's related works, and briefly motivate the use of the tensor–tensor t-product.

1.1 Randomized Kaczmarz

Randomized Kaczmarz is an iterative method for approximating solutions to linear systems of equations [13]. For a linear system $\mathbf{A}x = b$, a row $\mathbf{A}_{i:}$ is chosen at each iteration of MRK and the current iterate x (approximate solution) is projected onto the solution space

$$\mathbf{A}_{i:}x = b_i.$$

The RK method is advantageous for very large linear systems that cannot be loaded into memory at once due to its low memory footprint. Extensions to MRK include greedy [1,4,6,11,23,28,29] and block [25] variants to speed convergence and versions for inconsistent linear systems [24,41].

For a fixed sampling distribution over row indices i, MRK converges exponentially in expectation [10,34]. The standard MRK update for a linear system $\mathbf{A}x = b$ is given by

$$x^{t+1} = x^t - \mathbf{A}_{i_t:}^* \frac{\langle \mathbf{A}_{i_t:}, x^t \rangle - b_{i_t}}{\left\| \mathbf{A}_{i_t:} \right\|^2}, \tag{1.1}$$

[1] While the randomized Kaczmarz literature typically abbreviates randomized Kaczmarz as RK, throughout this work, MRK is used to distinguish the matrix and tensor versions of randomized Kaczmarz.

where i_t is the row index selected at iteration t and $\mathbf{A}^*_{i_t:}$ is the transpose of the i_tth row of $\mathbf{A}$.

Randomized Kaczmarz is closely related to the popular optimization technique, stochastic gradient descent (SGD) [26] which has been considered under tensor frameworks. For example, a tensor stochastic gradient descent was recently implemented to train tensor neural networks under the t-product [27]. The emphasis of the aforementioned work is a tensor neural network framework for multidimensional data and not the algorithmic analysis of SGD under the t-product, whereas this work focuses on analyzing tensor RK for tensor linear systems and connecting the extended algorithm to existing methods.

This and other tensor-based approaches are motivated by the fact that tensors arise in many applications and working with tensors directly, as opposed to naively flattening tensors into matrices can preserve significant structures and have computational advantages. A popular approach for working with tensors is to use the tensor–tensor *t-product*.

1.2 Tensor linear systems

The tensor–tensor t-product [16] is a bilinear operation between tensors, that allows for a linear algebraic-like framework. A tensor linear system under the t-product is formulated as follows. Let $\mathscr{X} \in \mathbb{C}^{\ell\times p\times n}$ be an unknown third-order tensor representing a three-dimensional data array. For example, this three-dimensional data could represent a video, color image, temporal data, or three-dimensional density values. A tensor linear system under the t-product is written as:

$$\mathscr{A}\mathscr{X} = \mathscr{B}, \tag{1.2}$$

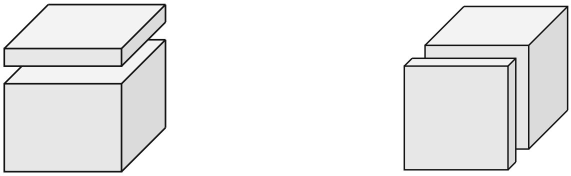
with $\mathscr{A} \in \mathbb{C}^{m\times\ell\times n}$, $\mathscr{X} \in \mathbb{C}^{\ell\times p\times n}$ and $\mathscr{B} \in \mathbb{C}^{m\times p\times n}$.

Initially motivated for tensor factorization, use of the t-product has become prominent in the tensor and signal processing community. The t-product has proved useful in applications such as dictionary learning [32,37], low-rank tensor completion [31,38–40], facial recognition [12], and neural networks [27,36]. T-product tensor linear systems also arise when using the Boundary Element Method[2] for analyzing electromagnetism and acoustic properties of spherically symmetric objects [2,5,35]. We provide further details about the t-product in Sect. 2.2.

1.3 Contributions

We extend the randomized Kaczmarz method to solve linear systems of third-order tensors under the t-product and denote the method TRK. To the best of our knowledge, no other works have considered solving large-scale t-product linear tensor systems with stochastic iterative methods. We provide theoretical guarantees for the TRK method and demonstrate its performance empirically. Furthermore, we remark on

[2] These linear systems are typically written as linear systems with block circulant matrices which are equivalent to the t-product as discussed in Definition 2.2.



(a) Horizontal slice $\mathscr{M}_{i::}$ of tensor $\mathscr{M}$. **(b)** Frontal slice $\mathscr{M}_{::k}$ of tensor $\mathscr{M}$.

Fig. 1 Horizontal slice $\mathscr{M}_{i::}$ and frontal slice $\mathscr{M}_{::k}$ of tensor $\mathscr{M}$

connections between TRK and the existing block-variant of MRK (BRK) [9], when working in the Fourier domain. This work serves as an example for extending methods for tensors under the t-product and how the properties of the t-product in the Fourier domain can be used to analyze convergence in this setting. Lastly, we demonstrate the computational advantage of using TRK over MRK for real and synthetic data by considering performance comparisons of the TRK method with a naively matricized MRK applied to a flattened tensor system as well as systems with equivalent memory constraints.

2 Background and notation

In this section, we present notation and several linear algebraic definitions for tensors under the t-product.

2.1 Notation

Throughout, calligraphic capital letters represent tensors, bold capital letters represent matrices, and lower case letters represent vectors and scalars. The index i is reserved for indexing horizontal slices of tensors (see Fig. 1a), rows of matrices, and entries of vectors. The index j is similarly reserved for indexing column slices of tensors and columns of matrices. The index k is reserved for indexing frontal slices of tensors as illustrated in Fig. 1b.

For matrices $\mathbf{M}$, we use the notation $\mathbf{M}_{i:}$ and $\mathbf{M}_{:j}$ to represent the ith row and jth column respectively. We use $\mathscr{M}_{i::}$ to represent horizontal slices and $\mathscr{M}_{::k}$ to represent frontal slices of a third-order tensor $\mathscr{M}$ as shown in Fig. 1. Because frontal slices of tensors are heavily used throughout this work, to condense notation, bold sub-scripted capital letters, $\mathbf{M}_k$, represents the kth frontal slice of $\mathscr{M}$ equivalently given by $\mathscr{M}_{::k}$, unless otherwise stated (for example, the $n \times n$ discrete Fourier transform (DFT) matrix $\mathbf{F}_n$ and $n \times n$ identity matrix $\mathbf{I}_n$).

The squared Frobenius norm $|| \cdot ||_F^2$ for matrices and tensors denotes the sum of squares of all scalar elements. For a matrix $\mathbf{M}$, $||\mathbf{M}||_F^2 = \sum_{ij} \mathbf{M}_{ij}^2$ and for a third-order tensor $\mathscr{M}$, $||\mathscr{M}||_F^2 = \sum_{ijk} \mathscr{M}_{ijk}^2$. We use $\sigma_{\min}^{+}(\mathbf{M})$ to denote the smallest nonzero singular value and $\mathbf{M}^{\dagger}$ to denote the pseudoinverse of the matrix $\mathbf{M}$.

Equation (2.1) shows how a third-order tensor $\mathscr{M} \in \mathbb{C}^{m \times \ell \times n}$ is unfolded into a $\mathbb{C}^{m\ell \times n}$ matrix:

$$\text{unfold}\,(\mathscr{M}) = \begin{pmatrix} \mathscr{M}_{::0} \\ \vdots \\ \mathscr{M}_{::n-1} \end{pmatrix} = \begin{pmatrix} \mathbf{M}_0 \\ \vdots \\ \mathbf{M}_{n-1} \end{pmatrix}. \tag{2.1}$$

To revert the unfolding of a tensor $\mathscr{M}$ we can fold the matrix in Eq. (2.1) such that $\text{fold}\,(\text{unfold}\,(\mathscr{M})) = \mathscr{M}$. To condense notation, when using both indices and transposes, the transpose are applied to the tensor or matrix slice, that is $\mathbf{M}_{i:}^{*} = (\mathbf{M}_{i:})^{*}$ and $\mathscr{M}_{i::}^{*} = (\mathscr{M}_{i::})^{*}$.

The tensor product of tensors $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$ and $\mathscr{B} \in \mathbb{C}^{\ell \times p \times n}$ is written as $\mathscr{A}\mathscr{B} \in \mathbb{C}^{m \times p \times n}$. Similarly, for matrices $\mathbf{A}, \mathbf{B}$, their matrix product is written as $\mathbf{AB}$. We do not consider the products between tensors and matrices. Throughout, we use $\mathscr{A}$ and $\mathbf{A}$ to represent the measurement tensor and matrix, $\mathscr{X}$, $\mathbf{X}$, and x to represent signal tensor, matrix and vector and $\mathscr{B}$, $\mathbf{B}$, and b to represent the observed measurements for the linear systems

$$\mathscr{A}\mathscr{X} = \mathscr{B}, \quad \mathbf{AX} = \mathbf{B}, \text{ and } \mathbf{A}x = b.$$

Lastly, the index t is reserved only to indicate iteration number and the shorthand $i \in [m-1]$ denotes $i \in \{0, 1, 2, ..., m-1\}$.

2.2 Tensor linear algebra

We now provide background on the tensor–tensor t-product. Under the t-product one, can recover many standard linear algebraic properties such as transposes, orthogonality, inverses and projections [16].

The t-product is defined in terms of block-circulant matrices.

Definition 2.1 For $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$, let $\text{bcirc}\,(\mathscr{A})$ denote the block-circulant matrix

$$\text{bcirc}\,(\mathscr{A}) = \begin{pmatrix} \mathbf{A}_0 & \mathbf{A}_{n-1} & \mathbf{A}_{n-2} & \dots & \mathbf{A}_1 \\ \mathbf{A}_1 & \mathbf{A}_0 & \mathbf{A}_{n-1} & \dots & \mathbf{A}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{n-1} & \mathbf{A}_{n-2} & \mathbf{A}_{n-3} & \dots & \mathbf{A}_0 \end{pmatrix} \in \mathbb{C}^{mn \times \ell n}.$$

While the definitions and results here are specific to the t-product which uses the Fourier transform, this product has been generalized to a class of tensor products that use arbitrary invertible linear operators [14,33].

Definition 2.2 The tensor–tensor t-product is defined as

$$\mathscr{A}\mathscr{B} = \text{fold}\,(\text{bcirc}\,(\mathscr{A})\,\text{unfold}\,(\mathscr{B})) \in \mathbb{C}^{m \times p \times n},$$

where $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$ and $\mathscr{B} \in \mathbb{C}^{\ell \times p \times n}$.

Definition 2.3 A tensor $\mathscr{X} \in \mathbb{C}^{\ell \times p \times n}$ is said to be in the row space of tensor $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$ if ALL the columns of unfold $(\mathscr{X})$ are in the row space of bcirc $(\mathscr{A})$. We denote this as $\mathscr{X} \in$ rowsp $(\mathscr{A})$.

Definition 2.4 The $m \times m \times n$ identity tensor, denoted $\mathscr{I}$, is the tensor whose first frontal slice is the $m \times m$ identity matrix and whose remaining entries are all zeros.

The identity tensor satisfies $\mathscr{M}\mathscr{I} = \mathscr{I}\mathscr{M} = \mathscr{M}$ for all tensors $\mathscr{M}$ with compatible sizes.

Definition 2.5 The conjugate transpose of a tensor $\mathscr{M} \in \mathbb{C}^{m \times \ell \times n}$ is denoted $\mathscr{M}^*$ and is produced by taking the conjugate transpose of all frontal slices and reversing the order of the second to last frontal slices $1, \dots, n-1$.

Note that this definition ensures $(\mathscr{M}^*)^* = \mathscr{M}$ and $(\mathscr{A}\mathscr{B})^* = \mathscr{B}^*\mathscr{A}^*$. A tensor is symmetric if $\mathscr{M}^* = \mathscr{M}$.

Definition 2.6 A tensor $\mathscr{M}$ is invertible if there exists an inverse tensor $\mathscr{M}^{-1}$ such that

$$\mathscr{M}\mathscr{M}^{-1} = \mathscr{M}^{-1}\mathscr{M} = \mathscr{I}.$$

Note that for an invertible tensor $\mathscr{M}$,

$$\mathscr{M}^* \left(\mathscr{M}^{-1}\right)^* = \mathscr{M}^{-1}\mathscr{M} = \mathscr{I} \quad \text{and} \quad \left(\mathscr{M}^{-1}\right)^* \mathscr{M}^* = \mathscr{M}\mathscr{M}^{-1} = \mathscr{I}.$$

Thus, we have $(\mathscr{M}^*)^{-1} = \left(\mathscr{M}^{-1}\right)^*$.

Definition 2.7 A tensor $\mathscr{Q} \in \mathbb{C}^{m \times p \times n}$ is orthogonal if

$$\mathscr{Q}^*\mathscr{Q} = \mathscr{I} = \mathscr{Q}\mathscr{Q}^*.$$

3 Tensor randomized Kaczmarz

Tensor randomized Kaczmarz is a Kaczmarz-type iterative method designed for t-product tensor linear systems. One notable difference between the t-product tensor and matrix linear systems is the interaction of the measurements $\mathscr{A}_{i::}$ and $\mathbf{A}_{i:}$ with the signals $\mathscr{X}$ and x. For the products $\mathbf{A}_{i:}x = b_i$ and $\mathbf{A}_{i:}\mathbf{X} = \mathbf{B}_{i:}$, each value in the signal $\mathbf{X}$ or x is multiplied by a single element of the measurement $\mathbf{A}_{i:}$. In the tensor measurement product,

$$\mathscr{A}_{i::}\mathscr{X} = \text{fold}\left(\text{bcirc}\left(\mathscr{A}_{i::}\right) \text{unfold}\left(\mathscr{X}\right)\right) \in \mathbb{C}^{1 \times n \times p}.$$

Since bcirc $(\mathscr{A}_{i::}) \in \mathbb{C}^{n \times \ell n}$, each element of $\mathscr{X}$ is multiplied by n elements in $\mathscr{A}_{i::}$ and affects n entries of the resulting product $\mathscr{B}_{i::}$. Equivalently, each frontal face of $\mathscr{X}$ is multiplied by each frontal face of $\mathscr{A}_{i::}$. See Kilmer and Martin [16] for more details and intuition for the t-product.

Algorithm 1 Tensor RK

```
Input: $\mathscr{X}^0 \in \mathbb{C}^{\ell\times p\times n}$, $\mathscr{A} \in \mathbb{C}^{m\times\ell\times n}$, $\mathscr{B} \in \mathbb{C}^{m\times p\times n}$, and probabilities $p_0, \dots, p_{m-1}$ corresponding to each horizontal slice of $\mathscr{A}$
for $t = 0, 1, 2, \dots$ do
    Sample $i_t \sim p_i$
    $\mathscr{X}^{t+1} = \mathscr{X}^t - \mathscr{A}^*_{i_t::}\left(\mathscr{A}_{i_t::}\mathscr{A}^*_{i_t::}\right)^{-1}\left(\mathscr{A}_{i_t::}\mathscr{X}^t - \mathscr{B}_{i_t::}\right).$
Output: last iterate $\mathscr{X}^{t+1}$
```

Making use of the algebraic properties of the t-product, we can write the TRK update for tensor linear systems as

$$\mathscr{X}^{t+1} = \mathscr{X}^t - \mathscr{A}^*_{i_t::}\left(\mathscr{A}_{i_t::}\mathscr{A}^*_{i_t::}\right)^{-1}\left(\mathscr{A}_{i_t::}\mathscr{X}^t - \mathscr{B}_{i_t::}\right). \tag{3.1}$$

The $\mathscr{A}_{i::}$ are horizontal slices of the tensor $\mathscr{A}$ as depicted in Fig. 1a. The index i_t used at each iteration is selected according to a probability distribution over the row indices $i \in [m-1]$. The TRK algorithm is detailed in Algorithm 1.

At each iteration, the current iterate $\mathscr{X}^t$ is projected onto the solution space of the sub-sampled system $\mathscr{A}_{i_t::}\mathscr{X} = \mathscr{B}_{i_t::}$. Note that this is the natural analogue of the MRK update, which projects the current iterate x^t onto the solution space of $\mathbf{A}_{i_t}x = b_{i_t}$. Tensor orthogonal projections under the t-product are discussed briefly by Kilmer et al. [15] and in more detail by Miao et al. [22].

The multiplication by $\left(\mathscr{A}_{i_t::}\mathscr{A}^*_{i_t::}\right)^{-1}$ in the TRK update serves an analogous role to normalization by the squared row norms, $\left\|\mathbf{A}_{i_t:}\right\|^2$in the MRK update. We assume throughout that $\mathscr{A}_{i::}\mathscr{A}^*_{i::}$ is invertible for all $i \in [m-1]$. The matrix bcirc $\left(\mathscr{A}_{i::}\mathscr{A}^*_{i::}\right)$ is invertible if and only if the tensor $\mathscr{A}_{i::}\mathscr{A}^*_{i::}$ is. Its inverse can be calculated explicitly as follows.

Lemma 3.1 *The inverse of $\mathscr{A}_{i::}\mathscr{A}^*_{i::}$ under the t-product is*

$$\left(\mathscr{A}_{i::}\mathscr{A}^*_{i::}\right)^{-1} = fold\left(\frac{1}{\sqrt{n}}\mathbf{F}^*_n\, diag\left(\mathbf{D}^{-1}\right)\right),$$

where $\mathbf{F}_n$ is the $n \times n$ Discrete Fourier Transform (DFT) matrix and $\mathbf{D}$ is a diagonal matrix such that bcirc $\left(\mathscr{A}_{i::}\mathscr{A}^*_{i::}\right) = \mathbf{F}^*_n\mathbf{D}\mathbf{F}_n$.

The inverse, $\left(\mathscr{A}_{i::}\mathscr{A}^*_{i::}\right)^{-1}$, can be derived using the definition t-product generalization of the Moore-Penrose pseudoinverse bcirc $\left(\mathscr{M}^\dagger\right) = \text{bcirc}\left(\mathscr{M}\right)^\dagger$ [22].

With the convergence analysis modeling that of MRK, one can derive analogous convergence guarantees for TRK. This result, stated in Theorem 3.1, shows that in expectation, the TRK algorithm will converge linearly to the solution of least Frobenius norm under mild conditions.

Theorem 3.1 *Let $\mathscr{X}^*$ be the tensor of minimal Frobenius norm such that $\mathscr{A}\mathscr{X}^* = \mathscr{B}$ and $\mathscr{X}^t$ be the tth approximation of $\mathscr{X}^*$ given by the updates of Eq. (3.1) with initial* *iterate $\mathscr{X}^0$ where $\mathscr{X}^0 \in rowsp(\mathscr{A})$. Let indices $i \in [m-1]$ be sampled independently from a probability distribution $\mathscr{D}$ at each iteration. Denote the orthogonal projection $\mathscr{P}_i = \mathscr{A}_{i::}^* \left(\mathscr{A}_{i::}\mathscr{A}_{i::}^*\right)^{-1} \mathscr{A}_{i::}$. The expected error at the tth iteration satisfies*

$$\mathbb{E}\left[\left\|\mathscr{X}^t - \mathscr{X}^*\right\|_F^2 \middle| \mathscr{X}^0\right] \le \left(1 - \sigma_{\min}^+\left(\mathbb{E}\left[bcirc\left(\mathscr{P}_i\right)\right]\right)\right)^t \left\|\mathscr{X}^0 - \mathscr{X}^*\right\|_F^2,$$

where the expectation is taken over the probability distribution $\mathscr{D}$, $\sigma_{\min}^+(\mathbf{M})$ denotes the smallest nonzero singular value of $\mathbf{M}$, and $\|\mathscr{M}\|_F^2$ is the sum of squared entries of the tensor $\mathscr{M}$.

The proof of Theorem 3.1 mirrors the standard analysis of MRK making use of the linear algebra mimetic properties of the t-product. The original proof of linear convergence of MRK requires the assumption that the matrix $\mathbf{A}$ have full column rank to ensure convergence [34]. This requirement can be relaxed by adding the assumption that the initial iterate reside in the row space of $\mathbf{A}$ [30,41]. For TRK, this requirement becomes $\mathscr{X}^0 \in \text{rowsp}(\mathscr{A})$. Note that this condition can be satisfied by choosing an initial iterate $\mathscr{X}^0$ of all zeros.

In addition to using this standard analysis, one can also take advantage of the block-diagonal structure of tensor linear systems in the Fourier domain and prove convergence in this setting. We proceed with this approach in the following section.

4 Analysis of TRK in the Fourier domain

The t-product can be computed efficiently using the Fast Fourier Transform (FFT), since circulant matrices are diagonalized by the DFT. In this section, we describe how the TRK update can be performed efficiently in the Fourier domain. We derive a convergence guarantee for TRK and additionally demonstrate that TRK is equivalent to performing block MRK with specific block structure on the linear system in the Fourier domain.

4.1 Notation and preliminary facts

We first introduce some additional notation and basic facts that will be used throughout this section. The notation and definitions are adopted from [16]. Let $\mathscr{M} \in \mathbb{C}^{m\times\ell\times n}$ and $\widehat{\mathscr{M}}$ denote the tensor resulting from applying the DFT matrix to each of the $1 \times 1 \times n$ tube fibers of $\mathscr{M}$. In previous literature, this operation is referred to as a *mode-3 FFT*. Fact 2 of [16], guarantees that

$$\text{bdiag}\left(\widehat{\mathscr{M}}\right) := \left(\mathbf{F}_n \otimes \mathbf{I}_m\right) \text{bcirc}\left(\mathscr{M}\right) \left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right) = \begin{pmatrix} \widehat{\mathbf{M}}_0 & & & \\ & \widehat{\mathbf{M}}_1 & & \\ & & \ddots & \\ & & & \widehat{\mathbf{M}}_{n-1} \end{pmatrix}, \quad (4.1)$$

where $\widehat{\mathbf{M}}_k \in \mathbb{C}^{m\times\ell}$ is the kth frontal face of $\widehat{\mathscr{M}}$, $\otimes$ denotes the Kronecker product, $\mathbf{F}_n$ is the $n \times n$ DFT matrix, and bdiag $\left(\widehat{\mathscr{M}}\right)$ is the block diagonal matrix formed by the frontal faces of $\widehat{\mathscr{M}}$.

This fact allows us to reformulate the tensor linear system Equation 1.2 as:

$$\begin{pmatrix} \widehat{\mathbf{A}}_0 & & & \\ & \widehat{\mathbf{A}}_1 & & \\ & & \ddots & \\ & & & \widehat{\mathbf{A}}_{n-1} \end{pmatrix} \begin{pmatrix} \widehat{\mathbf{X}}_0 \\ \widehat{\mathbf{X}}_1 \\ \vdots \\ \widehat{\mathbf{X}}_{n-1} \end{pmatrix} = \begin{pmatrix} \widehat{\mathbf{B}}_0 \\ \widehat{\mathbf{B}}_1 \\ \vdots \\ \widehat{\mathbf{B}}_{n-1} \end{pmatrix}, \tag{4.2}$$

where $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k = \widehat{\mathbf{B}}_k$ can be solved as individual sub-systems that can be solved independently.

The following fact is useful for working with the t-product in the Fourier domain. A proof of this fact can be found in the appendix.

Fact 1 *For appropriately sized tensors $\mathscr{A}$ and $\mathscr{B}$,*

1. *bdiag $(\widehat{\cdot})$ is a multiplicative function*

$$bdiag\left(\widehat{\mathscr{A}\mathscr{B}}\right) = bdiag\left(\widehat{\mathscr{A}}\right) bdiag\left(\widehat{\mathscr{B}}\right).$$

2. *Addition and $\widehat{\cdot}$ are commutative $\widehat{\mathscr{A}+\mathscr{B}} = \widehat{\mathscr{A}} + \widehat{\mathscr{B}}$.*
3. *The conjugate transpose commutes with bdiag $(\widehat{\cdot})$, bdiag $\left(\widehat{\mathscr{A}^*}\right) = bdiag\left(\widehat{\mathscr{A}}\right)^*$. Additionally, if bcirc $(\mathscr{A})$ is symmetric, bdiag $\left(\widehat{\mathscr{A}}\right)$ is also symmetric.*
4. *The inverse commutes with bdiag $(\widehat{\cdot})$, bdiag $\left(\widehat{\mathscr{A}^{-1}}\right) = bdiag\left(\widehat{\mathscr{A}}\right)^{-1}$.*

4.2 TRK convergence guarantee

Now, it is clear that Eq. (4.2) can be solved by applying MRK to every sub-system $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k = \widehat{\mathbf{B}}_k$ independently (asynchronously). However, when only the i^{th} horizontal slice of $\mathscr{A}$ is available, each sub-system is solved *synchronously*, i.e., in such a way that MRK uses the i^{th} row in *every* sub-system to iterate on the approximate solution. This case arises naturally when the tensor $\mathscr{A}$ is so large that only single horizontal slices of $\mathscr{A}$ can be accessed at a time or in an online setting in which horizontal slices of the tensor $\mathscr{A}$ are streaming in one at a time. This synchronous setting is equivalent to TRK (Algorithm 1) with calculations performed in the Fourier domain and has the convergence guarantees provided by Theorem 4.1. The proof of Theorem 4.1 is deferred to Appendix B.

Theorem 4.1 *Let $\mathscr{X}^*$ be the tensor of minimal Frobenius norm such that $\mathscr{A}\mathscr{X}^* = \mathscr{B}$ and $\mathscr{X}^t$ be the tth approximation of $\mathscr{X}^*$ given by applying RK to each of the subsystems $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k = \widehat{\mathbf{B}}_k$ synchronously. Furthermore, suppose the initial iterate $\mathscr{X}^0 \in rowsp(\mathscr{A})$ and indices $i_t \in [m-1]$ are sampled uniformly at random at each iteration and each* *subsystem. The expected error at the t^{th} iteration satisfies*

$$\mathbb{E}\left[\left\|\mathscr{X}^t-\mathscr{X}^*\right\|_F^2\Big|\mathscr{X}^0\right]\leq\left(1-\min_{k\in[n-1]}\frac{\left[\sigma_{\min}^+\left(\widehat{\mathbf{A}}_k\right)\right]^2}{m\left\|\widehat{\mathbf{A}}_k\right\|_{\infty,2}^2}\right)^t\left\|\mathscr{X}^0-\mathscr{X}^*\right\|_F^2,$$

where $\|\widehat{\mathbf{A}}_k\|_{\infty,2}^2 := \max_i\left[\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k\right]$, $\widehat{\mathbf{A}}_k$ *is the k^{th} frontal slice of $\widehat{\mathscr{A}}$, and $\sigma_{min}^+(\cdot)$ denotes the smallest nonzero singular value.*

When we are not restricted to using the same row $i_t\in[m-1]$ for all subsystems $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k=\widehat{\mathbf{B}}_k$, Eq. (4.2) can be solved by applying RK individually to each subsystem. Such an approach results in the following Corollary.

Corollary 4.1 *Let $\mathscr{X}^*$ be the tensor of minimal Frobenius norm such that $\mathscr{A}\mathscr{X}^*=\mathscr{B}$ and $\mathscr{X}^t$ be the tth approximation of $\mathscr{X}^*$ given by applying an iteration of RK to each of the subsystems $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k=\widehat{\mathbf{B}}_k$ synchronously with initial iterate $\mathscr{X}^0$ where $\mathscr{X}^0\in rowsp\,(\mathscr{A})$ and indices $i_t\in[m-1]$ sampled uniformly at random at each iteration and each subsystem. The expected error at the $(t)^{th}$ iteration satisfies*

$$\mathbb{E}\left[\left\|\mathscr{X}^t-\mathscr{X}^*\right\|_F^2\Big|\mathscr{X}^0\right]\leq\left(1-\min_{k\in[n-1]}\frac{\left[\sigma_{\min}^+\left(\widehat{\mathbf{A}}_k\right)\right]^2}{\left\|\widehat{\mathbf{A}}_k\right\|_F^2}\right)^t\left\|\mathscr{X}^0-\mathscr{X}^*\right\|_F^2,$$

where $\sigma_{min}^+(\cdot)$ denotes the smallest nonzero singular value.

The result stated in Theorem 4.1 is a direct result of the the convergence guarantees from the randomized Kaczmarz literature [30,34,41]. In particular, for over-determined consistent linear systems of the form $\mathbf{A}x=b$ with initial iterate x^0 in the row space of $\mathbf{A}$,

$$\mathbb{E}\left[\|x^t-x\|^2\right]\leq\left(1-\frac{\left[\sigma_{\min}^+(\mathbf{A})\right]^2}{\|\mathbf{A}\|_F^2}\right)^t\|x^0-x\|^2. \tag{4.3}$$

Applying RK to each of the sub-systems $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k=\widehat{\mathbf{B}}_k$, and taking the minimum expected decrease in error over k, obtains Corollary 4.1.

4.3 Equivalence of TRK and block MRK applied in the Fourier domain

TRK can be interpreted as the previously studied block MRK algorithm [25] with additional structural restrictions. In block MRK, one projects the current iterate onto the solution space of a set of constraints (set of rows of the linear system) as opposed to the solution space with respect to a single row. In practice, block MRK can lead to a significant speed up over MRK [25].

Here we show the equivalence of TRK and block MRK performed in the Fourier domain with specific block partitions and remark on the convergence rate implications in the block MRK setting. Making use of the equivalence of TRK and block MRK in the Fourier domain, TRK can be implemented efficiently using methods for matrices as detailed in Algorithm 2. Using Eq (4.1), the tensor linear system can be rewritten as a block diagonal system in the Fourier domain as shown in Eq. (4.2).

The system shown in Eq. (4.2) can be solved using block MRK such that the resulting iterate is equivalent to the TRK iterate in the following way. Let

$$\tau_i = \{km + i \mid k \in [n-1]\}, \tag{4.4}$$

denote in set of indices corresponding to a randomly selected block of the measurement matrix in Eq. (4.2). This choice of τ_i corresponds to selecting the ith row of each $\widehat{\mathbf{A}}_k$ in bdiag $\left(\widehat{\mathscr{A}}\right)$, i.e., each row of $\widehat{\mathscr{A}_{i::}}$ appears along the diagonal of bdiag $\left(\widehat{\mathscr{A}}\right)_{\tau_i}$ and therefore, bdiag $\left(\widehat{\mathscr{A}}\right)_{\tau_i}$ = bdiag $\left(\widehat{\mathscr{A}_{i::}}\right)$.

For a randomly selected row index $i_t \in [m-1]$, the block MRK update for Eq. (4.2) is aptly written as:

$$\begin{aligned}
&\text{unfold}\left(\widehat{\mathscr{X}^{t+1}}\right)\\
&\quad = \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{bdiag}\left(\widehat{\mathscr{A}}\right)^{\dagger}_{\tau_{i_t}} \left(\text{bdiag}\left(\widehat{\mathscr{A}}\right)_{\tau_{i_t}} \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{unfold}\left(\widehat{\mathscr{B}}\right)_{\tau_{i_t}}\right)\\
&\quad = \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right)^{\dagger} \left(\text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right) \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{unfold}\left(\widehat{\mathscr{B}}\right)_{\tau_{i_t}}\right).
\end{aligned} \tag{4.5}$$

Using Eq. (4.1) and Fact 1, we can show

$$\begin{aligned}
\text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right)^{\dagger} &= \text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right)^{*} \left(\text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right) \text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right)^{*}\right)^{-1}\\
&= \text{bdiag}\left(\widehat{\mathscr{A}^{*}_{i_t::}} \left(\widehat{\mathscr{A}_{i_t::}}\widehat{\mathscr{A}^{*}_{i_t::}}\right)^{-1}\right)
\end{aligned}$$

Therefore, noting the following equalities and folding the right and left sides of the equation into tensors, we derive the iterate update for $\widehat{\mathscr{X}^{t+1}}$ from the block MRK update:

$$\begin{aligned}
&\text{unfold}\left(\widehat{\mathscr{X}^{t+1}}\right)\\
&\quad = \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{bdiag}\left(\widehat{\mathscr{A}^{*}}_{i::} \left(\widehat{\mathscr{A}_{i_t::}}\widehat{\mathscr{A}^{*}}_{i_t::}\right)^{-1}\right) \left(\text{bdiag}\left(\widehat{\mathscr{A}_{i_t::}}\right) \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{unfold}\left(\widehat{\mathscr{B}}\right)_{\tau_{i_t}}\right)\\
&\quad = \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - (\mathbf{F}_n \otimes \mathbf{I}_\ell)\text{unfold}\left(\mathscr{A}_{i_t::}{}^{*} \left(\mathscr{A}_{i_t::}\mathscr{A}_{i_t::}{}^{*}\right)^{-1} \left(\mathscr{A}_{i_t::}\mathscr{X}^{t} - \mathscr{B}_{i_t::}\right)\right)\\
&\quad = \text{unfold}\left(\widehat{\mathscr{X}^{t}}\right) - \text{unfold}\left(\widehat{\mathscr{A}^{*}}_{i_t::} \left(\widehat{\mathscr{A}_{i_t::}}\widehat{\mathscr{A}^{*}}_{i_t::}\right)^{-1} \left(\widehat{\mathscr{A}_{i_t::}}\widehat{\mathscr{X}^{t}} - \widehat{\mathscr{B}_{i_t::}}\right)\right)\\
&\quad \Rightarrow \widehat{\mathscr{X}^{t+1}} = \widehat{\mathscr{X}^{t}} - \widehat{\mathscr{A}^{*}}_{i_t::} \left(\widehat{\mathscr{A}_{i_t::}}\widehat{\mathscr{A}^{*}}_{i_t::}\right)^{-1} \left(\widehat{\mathscr{A}_{\tau_{i_t::}}}\widehat{\mathscr{X}^{t}} - \widehat{\mathscr{B}_{i_t::}}\right).
\end{aligned} \tag{4.6}$$

Since the FFT is applied to each tube fiber of $\mathscr{A}$ independently, $\widehat{\mathscr{A}_{i::}} = \widehat{\mathscr{A}}_{i::}$. To see that Eq. (4.6) is equivalent to Eq. (3.1) one can use Fact 1 to show that $\widehat{\mathscr{X}^{t+1}} = \widehat{\mathscr{X}^{t+1}}$, that is taking the inverse FFT on the tubes of $\widehat{\mathscr{X}^{t+1}}$ will return the TRK update Eq. (3.1).

Remark 4.1 The contraction rate for block MRK applied to the linear system Eq. (4.2) with iterates as shown in Eq. (4.5) is

$$\rho_{\mathrm{BRK}} = 1 - \frac{\left[\sigma^{+}_{\min}\left(\mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)\right)\right]^2}{mn \max_i \lambda_{\max}\left(\mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)_{\tau_i} \mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)^{*}_{\tau_i}\right)}. \tag{4.7}$$

The contraction coefficient ρ_{BRK} is a direct result of the theoretical guarantees for block MRK shown in [25,30]. Note that due to the block-diagonal structure, the numerator of the second term of Eq. (4.7) can be simplified to

$$\sigma^{+}_{\min}\left(\mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)\right) = \min_{k\in[n-1]} \sigma^{+}_{\min}(\widehat{\mathbf{A}}_k).$$

Using the fact that $\mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)_{\tau_i} = \mathrm{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)$ along with Fact 1, it can be easily shown that $\mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)_{\tau_i} \mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)^{*}_{\tau_i} = \mathrm{bdiag}\left(\widehat{\mathscr{A}_{i::}}\widehat{\mathscr{A}^{*}}_{i::}\right)$. Thus, the denominator of Eq. (4.7) can be simplified to:

$$\begin{aligned}\max_i \lambda_{\max}\left(\mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)_{\tau_i} \mathrm{bdiag}\left(\widehat{\mathscr{A}}\right)^{*}_{\tau_i}\right) &= \max_i \lambda_{\max}\left(\mathrm{bdiag}\left(\widehat{\mathscr{A}_{i::}}\widehat{\mathscr{A}^{*}_{i::}}\right)\right)\\ &= \max_i \max_k \left[\widehat{\mathscr{A}_{i::}}\widehat{\mathscr{A}^{*}_{i::}}\right]_k\\ &= \max_k \left\|\widehat{\mathbf{A}}_k\right\|^2_{\infty,2},\end{aligned}$$

where the norm in the last equality is as defined in Theorem 4.1. Putting this all together, the contraction rate for block MRK applied to Eq. (4.2) is

$$\rho_{\mathrm{BRK}} = 1 - \frac{\min_k \left[\sigma^{+}_{\min}(\widehat{\mathbf{A}}_k)\right]^2}{mn \max_k \left\|\widehat{\mathbf{A}}_k\right\|^2_{\infty,2}}. \tag{4.8}$$

Compared to the convergence rate derived for TRK in Theorem 4.1, the standard block MRK convergence guarantee is weaker (slower). However, it should be noted that the standard analysis for the convergence of block MRK is not restricted to block diagonal systems. Thus, although block MRK applied to Eq. (4.2) with predetermined blocks τ_i is equivalent to the proposed TRK update, the standard block MRK guarantee is weaker since the TRK analysis takes advantage of the block diagonal structure of the system in the Fourier domain.

Remark 4.2 The block-diagonal system in Eq. (4.2) is highly parallelizable. Specifically, each component block of the system $\widehat{\mathbf{A}}_k\widehat{\mathbf{X}}_k = \widehat{\mathbf{B}}_k$ for $k \in [n-1]$ can be solved independently. For m extremely large, however, loading a single $\widehat{\mathbf{A}}_k$ into memory maybe be impossible. In such settings, a randomized iterative method such as TRK is advantageous. The block-diagonal structure of the subsampled system in the Fourier domain also allows the update for each component block to be computed in parallel.

Algorithm 2 Tensor RK computed in the Fourier domain

```
**Input:** $\mathscr{A} \in \mathbb{C}^{m\times \ell \times n}$, $\mathscr{B} \in \mathbb{C}^{m\times p \times n}$, $\mathscr{X}^0 \in \mathbb{C}^{\ell\times p \times n}$ with $\mathscr{X}^0 \in \operatorname{rowsp}(\mathscr{A})$, and probabilities $p_0, \dots, p_{m-1}$ corresponding to each horizontal slice of $\mathscr{A}$
Compute $\widehat{\mathscr{X}^0}$, $\widehat{\mathscr{A}}$, $\widehat{\mathscr{B}}$ as in Eq. (4.1)
**for** $t = 0, 1, 2, \dots$ **do**
    Sample $i_t \sim p_i$
    **for** $k = 0, 1, \dots, n-1$ **do**
        $\widehat{\mathbf{X}_k^{t+1}} = \widehat{\mathbf{X}_k^t} - \left(\widehat{\mathbf{A}}_{k_{i_t:}}\right)^\dagger \left(\widehat{\mathbf{A}}_{k_{i_t:}} \widehat{\mathbf{X}_k^t} - \widehat{\mathbf{B}}_{i_t:}\right)$
Recover $\mathscr{X}^{t+1}$ from $\widehat{\mathscr{X}^{t+1}}$
**Output:** last iterate $\mathscr{X}^{t+1}$
```

Remark 4.3 The equivalence between TRK and block MRK with blocks indexed by Eq. (4.4) also reveal a straightforward analysis for the comparison of the computational complexity between TRK and MRK. The per iteration complexity of MRK using rows $\mathbf{A}_{i:} \in \mathbb{R}^{1\times \ell n}$ is $\mathscr{O}(\ell n)$ and the per iteration complexity of TRK using rows $\mathscr{A}_{i::} \in \mathbb{R}^{1\times \ell \times n}$ is $\mathscr{O}(\ell n^2)$.

5 Experiments

In this section, we present numerical experiments comparing MRK and TRK. The implementation of the TRK algorithm used is as outlined in Algorithm 1, unless otherwise noted. First, we show empirically that with an increasing number of measurements m, the contraction coefficient for TRK is smaller than that of MRK indicating a stronger convergence guarantee. Next, we compare the performance of TRK with that of MRK applied to a matrix linear system where the memory complexity of the measurement matrix is preserved. Then, we move on to the setting in which one is given tensor measurements $\mathscr{B}$ and compare the performance of TRK with that of MRK applied to the unfolded tensor system

$$\operatorname{bcirc}(\mathscr{A}) \operatorname{unfold}(\mathscr{X}) = \operatorname{unfold}(\mathscr{B}).$$

These experiments demonstrate the computational benefits of using TRK given by Eq. (3.1) over applying standard MRK to an unfolded system.

5.1 Contraction coefficients of TRK and MRK

In this experiment, the contraction coefficient of the proposed TRK is compared to that of MRK. In order to apply the standard MRK method to recover the three-dimensional signal $\mathscr{X}$, we unfold the tensor $\mathscr{X}$ into the matrix $\operatorname{unfold}(\mathscr{X}) \in \mathbb{C}^{\ell n \times p}$ and collect measurements $\mathbf{B} \in \mathbb{C}^{\mu \times p}$ of the signal $\mathscr{X}$ via the measurement matrix $\mathbf{A} \in \mathbb{C}^{\mu \times n\ell}$, resulting in the matrix linear system

$$\mathbf{A}\operatorname{unfold}(\mathscr{X}) = \mathbf{B}. \tag{5.1}$$

After each iteration of MRK applied to Eq. (5.1), the iterate $\text{unfold}\left(\mathscr{X}^{t+1}\right)$ satisfies

$$\mathbf{A}_{i_t:}\text{unfold}\left(\mathscr{X}^{t+1}\right) = \text{unfold}\,(\mathscr{B})_{i_t:}\,.$$

Thus, the constraint is applied to each column of $\text{unfold}\,(\mathscr{X})$ or equivalently each column slice of $\mathscr{X}$ independently. Note that the measurement matrix $\mathbf{A}$ will have the same number of elements as the measurement tensor $\mathscr{A}$ if $\mu = m$.

Assuming that the rows of $\mathbf{A}$ are normalized, MRK applied to matrix linear systems has a contraction coefficient of

$$1 - \left[\sigma^+_{min}(\mathbf{A})\right]^2 / m. \tag{5.2}$$

For TRK, the contraction coefficient from Theorem 4.1 is

$$1 - \min_{k \in [n-1]} \frac{\left[\sigma^+_{\min}\left(\widehat{\mathbf{A}}_k\right)\right]^2}{m \left\|\widehat{\mathbf{A}}_k\right\|^2_{\infty,2}}. \tag{5.3}$$

In this experiment, horizontal slices $\mathscr{A}_{i::}$ have unit Frobenius norm and indices $i \in [m-1]$ are selected uniformly at random at each iteration.

The measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times \ell n}$ and measurement tensor $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$ are generated as follows. The entries of $\mathbf{A} \in \mathbb{C}^{m \times \ell n}$ are drawn i.i.d. from a standard Gaussian distribution and then each row is normalized to have unit norm. The entries of $\mathscr{A} \in \mathbb{R}^{m \times \ell \times n}$ are also drawn i.i.d. from a standard Gaussian distribution but horizontal slices $\mathscr{A}_{i::}$ (as opposed to matrix rows) of $\mathscr{A}$ are normalized to have unit Frobenius norm. Note that both the tensor $\mathscr{A}$ and matrix $\mathbf{A}$ in this experiment have the same memory complexity of $\mathscr{O}(m\ell n)$. The contraction coefficients, computed via Eq. (5.2) for matrices $\mathbf{A}$ and Eq. (5.3) for tensors $\mathscr{A}$, with a varying number of measurements m are presented in Fig. 2. Here, the dimensions $\ell = 20$ and $n = 10$ are fixed. For each number of measurements m, the contraction coefficients are averaged over 50 random realizations of the measurement tensor or matrix.

In this experiment, the contraction coefficients for MRK and TRK differ, with TRK being smaller (i.e., faster convergence) for larger m. Thus, in the large-scale setting where $m \gg \ell n$, TRK is expected to converge faster than MRK, as we will see in the experiments of Sect. 5.2. When a small number of measurements m are used, MRK has a smaller contraction coefficient than TRK, however, we are primarily concerned with the setting in which $m \gg \ell n$ as this is the typical use case for Kaczmarz methods.

5.2 Empirical performance of TRK and MRK

We now compare the empirical performance of MRK and TRK on linear systems $\mathscr{A}\mathscr{X} = \mathscr{B}$ and $\mathbf{AX} = \mathbf{Y}$. Similar to the previous experiment, the dimensions of $\mathscr{A}$ and $\mathbf{A}$ are selected to require a similar measurement complexity while solving for unknown signals of comparable dimensions. More specifically, for the tensor system we have $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$ and $\mathscr{X} \in \mathbb{C}^{\ell \times p \times n}$, while for the matrix system, we have $\mathbf{A} \in \mathbb{C}^{m \times \ell n}$

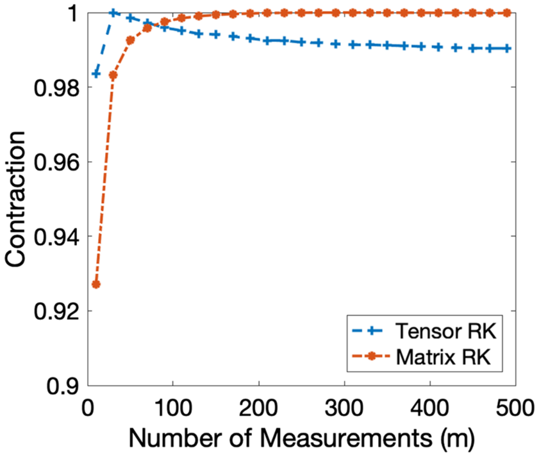


Fig. 2 Comparison between contraction coefficients of MRK (Eq. 5.2) applied to a matrix linear system and TRK (Eq. 5.3) applied to a tensor system

and $\mathbf{X} \in \mathbb{C}^{\ell n \times p}$. The entries of $\mathscr{A}$ and $\mathbf{A}$ are initialized with i.i.d. standard Gaussian entries then normalized to have unit horizontal slice and matrix row Frobenius norm respectively. The entries of the signals $\mathscr{X}$ and $\mathbf{X}$ are drawn i.i.d. from a standard Gaussian distribution and the empirical results presented here are averaged over 20 random runs of TRK and MRK. For TRK, we use the implementation outlined in Algorithm 2.

Figure 3 compares the empirical performances of the two algorithms for an over-determined system with $m = 500$, $\ell = 20$, $n = 10$, and $p = 10$. We refer to a tensor linear system as over-determined if the Fourier transformed systems of Eq. (4.2) is over-determined, i.e., if $m \geq \ell$. In the over-determined setting, we plot the convergence of the algorithms with respect to iterations (left plot) as well as CPU time (right plot). We observe that in both settings, TRK outperforms MRK in terms of iterations and CPU times. While, visually, MRK does not seem to be making progress towards the solution in either setting, it is in fact converging slowly. This should not be surprising given the equivalence between TRK and block MRK. In particular, one can think of TRK as block MRK acting on n rows at a time (whereas MRK only works on on row at a time).

We additionally consider the setting in which one is immediately provided the measurement tensor $\mathscr{A} \in \mathbb{C}^{m \times \ell \times n}$ and corresponding measurements $\mathscr{B} \in \mathbb{C}^{m \times p \times n}$ and can choose between performing signal recovery using TRK or by unfolding the tensor system and solving $\text{bcirc}\,(\mathscr{A})\,\text{unfold}\,(\mathscr{X}) = \text{unfold}\,(\mathscr{B})$ using MRK.

The tensor $\mathscr{A}$ is initialized with i.i.d. standard Gaussian entries and the measurement matrix $\mathbf{A}$ is taken to be $\mathbf{A} = \text{bcirc}\,(\mathscr{A})$. Here, $m = 100$, $\ell = 15$, $n = 10$, and $p = 30$. Fig. 4 plots the resulting empirical performance averaged over 20 random runs of TRK and MRK when choosing between signal recovery using TRK or MRK for a given tensor measurement system. We again see that TRK converges at a much faster rate than MRK in this setting.

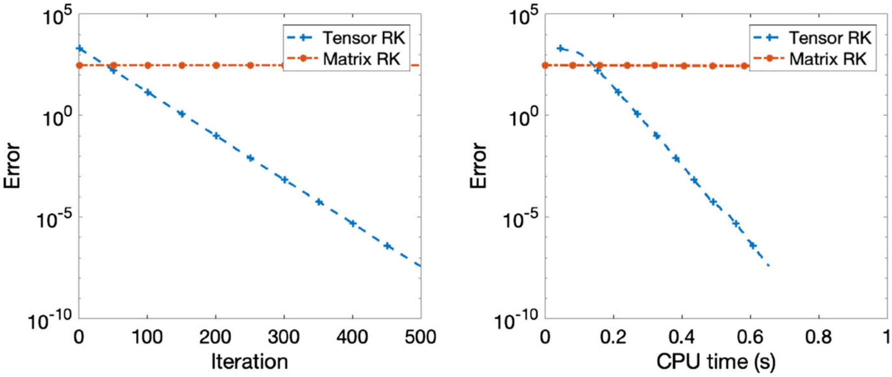


Fig. 3 Comparison MRK and TRK when the measurement matrix (or tensor) has a fixed memory budget of $\mathscr{O}(m\ell n)$ bits when $m = 500$, $\ell = 20$, and $n = 10$

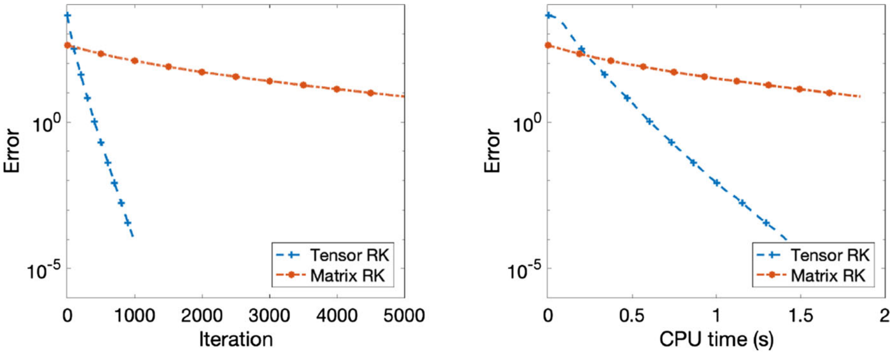


Fig. 4 Performance comparison of MRK on matricized linear system and TRK on tensor linear system

5.3 Empirical performance of TRK and block MRK

To support the theoretical guarantees and remarks regarding the equivalence of TRK and block MRK, experimental results comparing the empirical performance of the two algorithms are presented in this section. In Fig. 5, TRK and block MRK are used to solve a tensor linear system. TRK solves the tensor system via the update in Eq. (3.1) while block MRK is performed on the transformed system in the Fourier domain given in Eq. (4.2) with predetermined blocks $\tau_i = \{km + i \mid k \in [n-1]\}$. The measurement tensor $\mathscr{A} \in \mathbb{R}^{100\times30\times5}$ and signal tensor $\mathscr{X} \in \mathbb{R}^{30\times15\times5}$ contains i.i.d. standard Gaussian entries. All approximation errors are averaged over 20 runs of the respective algorithm. The theoretical upper bounds, titled in the legend with 'UB', are computed using Theorem 4.1 for TRK and Eq. (4.8) for block MRK. Figure 5 clearly shows that TRK and block MRK perform similarly across iterations as expected since the two methods are shown to be equivalent in Sect. 4.3. As remarked, the TRK upper bound shown in Theorem 4.1 has a slight advantage over the general block MRK convergence guarantees as these do not make use of the block diagonal structure of Eq. (4.2). Experiments comparing CPU times for TRK and block MRK are omitted, as the two

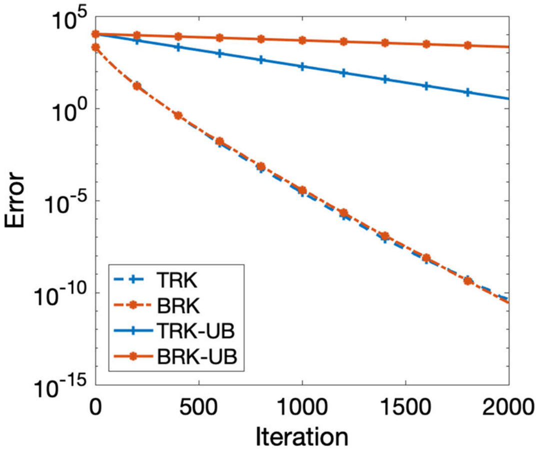


Fig. 5 Performance of TRK and block MRK on a tensor linear system. 'TRK-UB' and 'BRK-UB' indicate the theoretical upper bounds of TRK and block MRK respectively

methods are equivalent as shown in Sect. 4.3 and highly optimized algorithms exist for the matrix implementation.

5.4 Performance on CT and video Data

In the following experiments, we evaluate the performance of TRK on real world CT and video data sets.

In the first experiment, the underlying signal $\mathscr{X}$ is a $512 \times 512 \times 11$ tensor where each 512×512 frontal slice is a 2-dimensional slice of the C1-vertebrae. The images for this experiment were obtained from the Laboratory of Human Anatomy and Embryology, University of Brussels (ULB), Belgium [3]. To set up the tensor linear system, we randomly generate a Gaussian matrix $\mathscr{A} \in \mathbb{R}^{10000 \times 512 \times 11}$ and take the t-product between $\mathscr{A}$ and $\mathscr{X}$ to get the the measurement tensor $\mathscr{B}$. We compare the performance of TRK on this system with a similar matrix linear system where the memory complexity of the measurement matrix is fixed (i.e., $\mathbf{A} \in \mathbb{R}^{10000 \times (512 \cdot 11)}$). The results of this experiment are provided in Fig. 6. Here we can see that a tensor system with measurement tensor $\mathscr{A} \in \mathbb{R}^{10000 \times 512 \times 11}$ equipped with TRK can more efficiently recover the underlying signal $\mathscr{X}$ compared to a matrix linear system with matrix $\mathbf{A} \in \mathbb{R}^{10000 \times (512 \cdot 11)}$ in terms of the number of FLOPS required. Furthermore, even before reaching an error of 10^0, the recovered scan looks visually identical to the ground truth.

Next, we demonstrate the performance of TRK on video data where the frontal slices of the tensor $\mathscr{X}$ are frames from the 1929 film "Finding His Voice" [8]. Each video frame is 181×251 and there are a total of 20 frames. For the measurement tensor, similar to the previous experiment, we randomly generate the tensor by populating its entries with i.i.d. samples from a standard Gaussian distribution and the number of measurements $m = 5000$ for TRK. Figure 7 presents the results from this experiment. Again we see here that that a tensor system, with measurement tensor $\mathscr{A} \in \mathbb{R}^{5000 \times 181 \times 20}$, can more efficiently recover the underlying signal $\mathscr{X}$ compared

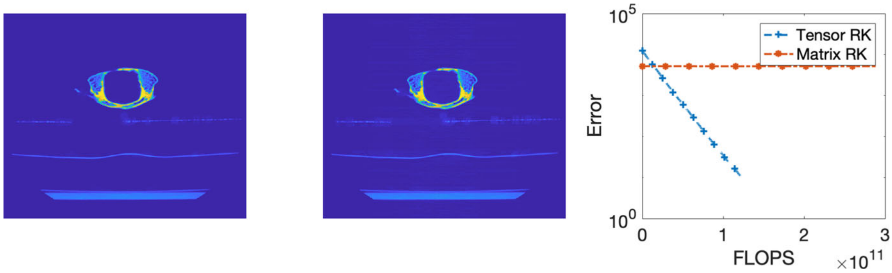


Fig. 6 (Left) ground truth slice from CT data set. (Center) recovered CT slice from TRK. (Right) evolution of approximation error with respect to FLOPS

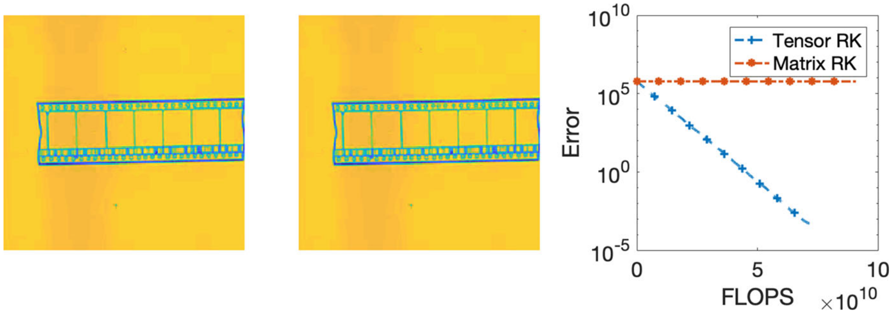


Fig. 7 (left) Ground truth slice from Video data set. (center) Recovered Video slice from TRK. (right) Evolution of approximation error with respect to FLOPS

to a matrix linear system, with measurement matrix $\mathbf{A} \in \mathbb{R}^{5000\times(181\cdot 20)}$, when the the storage requirement of the measurement tensor or matrix is fixed.

6 Conclusion

This work extends the randomized Kaczmarz literature to solve large-scale tensor linear systems under the t-product. The proposed tensor randomized Kaczmarz (TRK) algorithm solves large-scale tensor linear systems and is guaranteed to convergence exponentially in expectation. Connections to the block randomized Kaczmarz are made and empirical results are provided to support derived theoretical guarantees. This work further provides a framework to extend other stochastic iterative methods that arise in literature such as the randomized extended Kaczmarz algorithm, randomized Gauss-Seidel algorithm, coordinate descent, sketch-and-project [10], and many more.

A Proof of Fact 1

The following properties of block circulant matrices will be useful in proving Fact 1.

Fact 2 (Lemma 1.iii [19]) *For tensors* $\mathscr{A}$ *and* $\mathscr{B}$, *bcirc* $(\mathscr{A}\mathscr{B}) =$ *bcirc* $(\mathscr{A})$ *bcirc* $(\mathscr{B})$.

Fact 3 (Theorem 6.ii [19]) *The block circulant operator bcirc* $(\cdot)$ *commutes with the conjugate transpose,*

$$bcirc\left(\mathscr{M}^*\right) = bcirc\left(\mathscr{M}\right)^*.$$

1. Part 1 of Fact 1 states

$$\text{bdiag}\left(\widehat{\mathscr{A}\mathscr{B}}\right) = \text{bdiag}\left(\widehat{\mathscr{A}}\right)\text{bdiag}\left(\widehat{\mathscr{B}}\right).$$

Proof Let $\mathscr{A} \in \mathbb{C}^{m\times\ell\times n}$ and $\mathscr{B} \in \mathbb{C}^{\ell\times p\times n}$, then

$$\begin{aligned}\text{bdiag}\left(\widehat{\mathscr{A}\mathscr{B}}\right) &\overset{(4.1)}{=} (\mathbf{F}_n \otimes \mathbf{I}_m)\,\text{bcirc}\,(\mathscr{A}\mathscr{B})\left(\mathbf{F}_n^* \otimes \mathbf{I}_p\right)\\ &\overset{Fact\ 2}{=} (\mathbf{F}_n \otimes \mathbf{I}_m)\,\text{bcirc}\,(\mathscr{A})\,\text{bcirc}\,(\mathscr{B})\left(\mathbf{F}_n^* \otimes \mathbf{I}_p\right)\\ &= (\mathbf{F}_n \otimes \mathbf{I}_m)\,\text{bcirc}\,(\mathscr{A})\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right)(\mathbf{F}_n \otimes \mathbf{I}_\ell)\,\text{bcirc}\,(\mathscr{B})\left(\mathbf{F}_n^* \otimes \mathbf{I}_p\right)\\ &\overset{(4.1)}{=}\text{bdiag}\left(\widehat{\mathscr{A}}\right)\text{bdiag}\left(\widehat{\mathscr{B}}\right).\end{aligned}$$

□

2. Part 2 of Fact 1 states

$$\widehat{\mathscr{A}+\mathscr{B}} = \widehat{\mathscr{A}} + \widehat{\mathscr{B}}.$$

Proof Let $\mathscr{A} \in \mathbb{C}^{m\times\ell\times n}$ and $\mathscr{B} \in \mathbb{C}^{m\times\ell\times n}$, then

$$\begin{aligned}\text{bdiag}\left(\widehat{\mathscr{A}+\mathscr{B}}\right) &\overset{(4.1)}{=} (\mathbf{F}_n \otimes \mathbf{I}_m)\,\text{bcirc}\,(\mathscr{A}+\mathscr{B})\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right)\\ &= (\mathbf{F}_n \otimes \mathbf{I}_m)\,(\text{bcirc}\,(\mathscr{A}) + \text{bcirc}\,(\mathscr{B}))\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right)\\ &= (\mathbf{F}_n \otimes \mathbf{I}_m)\,\text{bcirc}\,(\mathscr{A})\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right) + (\mathbf{F}_n \otimes \mathbf{I}_m)\,\text{bcirc}\,(\mathscr{B})\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right)\\ &\overset{(4.1)}{=}\text{bdiag}\left(\widehat{\mathscr{A}}\right) + \text{bdiag}\left(\widehat{\mathscr{B}}\right).\end{aligned}$$

□

3. Part 3 of Fact 1 states that

$$\text{bdiag}\left(\widehat{\mathscr{M}^*}\right) = \text{bdiag}\left(\widehat{\mathscr{M}}\right)^*.$$

Additionally, it states that if bcirc $(\mathscr{M})$ is symmetric, bdiag $\left(\widehat{\mathscr{M}}\right)$ is also symmetric.

Proof Let $\mathscr{M} \in \mathbb{C}^{m \times \ell \times n}$. Then

$$\begin{aligned}
\operatorname{bdiag}\left(\widehat{\mathscr{M}^*}\right) &\overset{(4.1)}{=} (\mathbf{F}_n \otimes \mathbf{I}_\ell)\operatorname{bcirc}\left(\mathscr{M}^*\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_m\right) \\
&\overset{Fact\ 3}{=} (\mathbf{F}_n \otimes \mathbf{I}_\ell)\operatorname{bcirc}\left(\mathscr{M}\right)^*\left(\mathbf{F}_n^* \otimes \mathbf{I}_m\right) \\
&= \left[(\mathbf{F}_n \otimes \mathbf{I}_m)\operatorname{bcirc}\left(\mathscr{M}\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right)\right]^* \\
&\overset{(4.1)}{=} \operatorname{bdiag}\left(\widehat{\mathscr{M}}\right)^*.
\end{aligned}$$

To see that $\operatorname{bdiag}\left(\widehat{\mathscr{M}}\right)$ is also symmetric when $\operatorname{bcirc}\left(\mathscr{M}\right)$ is symmetric, note that

$$\operatorname{bdiag}\left(\widehat{\mathscr{M}}\right)^* \overset{(4.1)}{=} \left[(\mathbf{F}_n \otimes \mathbf{I}_m)\operatorname{bcirc}\left(\mathscr{M}\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_n\right)\right]^*.$$

□

4. Finally, part 4 of Fact 1 states

$$\operatorname{bdiag}\left(\widehat{\mathscr{M}^{-1}}\right) = \operatorname{bdiag}\left(\widehat{\mathscr{M}}\right)^{-1}.$$

Proof Let $\mathscr{M} \in \mathbb{C}^{m \times m \times n}$. Note that $\operatorname{bcirc}\left(\mathscr{I}_m\right) = \mathbf{I}_{mn}$. Using Fact 2 and Eq. (4.1),

$$\begin{aligned}
&\operatorname{bdiag}\left(\widehat{\mathscr{M}^{-1}}\right)\operatorname{bdiag}\left(\widehat{\mathscr{M}}\right) \\
&\quad\overset{(4.1)}{=} (\mathbf{F}_n \otimes \mathbf{I}_m)\operatorname{bcirc}\left(\mathscr{M}^{-1}\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_m\right)(\mathbf{F}_n \otimes \mathbf{I}_m)\operatorname{bcirc}\left(\mathscr{M}\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_m\right) \\
&\quad= (\mathbf{F}_n \otimes \mathbf{I}_m)\operatorname{bcirc}\left(\mathscr{M}^{-1}\right)\operatorname{bcirc}\left(\mathscr{M}\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_m\right) \\
&\quad\overset{Fact\ 2}{=} (\mathbf{F}_n \otimes \mathbf{I}_m)\operatorname{bcirc}\left(\mathscr{I}_m\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_m\right) \\
&\quad= \mathbf{I}_{mn}.
\end{aligned}$$

Analogously, one can show $\operatorname{bdiag}\left(\widehat{\mathscr{M}}\right)\operatorname{bdiag}\left(\widehat{\mathscr{M}^{-1}}\right) = \mathbf{I}_{mn}$. □

B Proof of Theorem 4.1

We now prove Theorem 4.1.

Proof Let $\widehat{\mathscr{P}_i}$ be the tensor formed by applying FFTs to each tube fiber of $\mathscr{P}_i = \mathscr{A}_{i::}^*\left(\mathscr{A}_{i::}\mathscr{A}_{i::}^*\right)^{-1}\mathscr{A}_{i::}$ and $\mathscr{E}^t = \mathscr{X}^t - \mathscr{X}^*$. By Eq. (4.1), we have that

$$\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right) = (\mathbf{F}_n \otimes \mathbf{I}_\ell)\operatorname{bcirc}\left(\mathscr{P}_i\right)\left(\mathbf{F}_n^* \otimes \mathbf{I}_\ell\right),$$

is a block diagonal matrix with blocks $\left(\widehat{\mathbf{P}}_i\right)_k$, where $\left(\widehat{\mathbf{P}}_i\right)_k$ is the kth frontal slice of the tensor $\widehat{\mathscr{P}_i}$. We note that the projected error can be rewritten as

$$\begin{aligned}\mathbb{E}\left[\left\|\mathscr{P}_i\mathscr{E}^t\right\|_F^2\right] &= \sum_{j=1}^{p}\langle\mathbb{E}\left[\operatorname{bcirc}\left(\mathscr{P}_i\right)\right]\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}, \operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}\rangle\\ &= \sum_{j=1}^{p}\mathbb{E}\left[\langle\left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{bcirc}\left(\mathscr{P}_i\right)\left(\mathbf{F}_n^*\otimes\mathbf{I}_\ell\right)\left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}, \left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}\rangle\right]\\ &= \sum_{j=1}^{p}\mathbb{E}\left[\langle\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right)\left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}, \left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}\rangle\right].\end{aligned}$$

Note that the rows of $\operatorname{bcirc}\left(\mathscr{A}_{i::}\right)$ are also rows of $\operatorname{bcirc}\left(\mathscr{A}\right)$. Thus, $\mathscr{X}^{t+1}\in\operatorname{rowsp}\left(\mathscr{A}\right)$. Since $\mathscr{X}^*$ is the tensor of least Frobenius norm, $\mathscr{X}^*\in\operatorname{rowsp}\left(\mathscr{A}\right)$. Therefore $\mathscr{E}=\mathscr{X}^t-\mathscr{X}^*\in\operatorname{rowsp}\left(\mathscr{A}\right)$ as long as $\mathscr{X}^0\in\operatorname{rowsp}\left(\mathscr{A}\right)$.

Now, since $\mathbb{E}\left[\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right)\right]$ is symmetric and $\mathscr{E}^t\in\operatorname{rowsp}\left(\mathscr{A}\right)$, by Fact 1,

$$\mathbb{E}\left[\left\|\mathscr{P}_i\mathscr{E}^t\right\|_F^2\right]\geq\sigma_{\min}^{+}\left(\mathbb{E}\left[\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right)\right]\right)\left\|\left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)\right\|_F^2. \tag{B.1}$$

Note that,

$$\begin{aligned}\left\|\left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)\right\|_F^2 &= \sum_{j=1}^{p}\langle\left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}, \left(\mathbf{F}_n\otimes\mathbf{I}_\ell\right)\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}\rangle\\ &= \sum_{j=1}^{p}\langle\operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}, \operatorname{unfold}\left(\mathscr{E}^t\right)_{:j}\rangle\\ &= \left\|\operatorname{unfold}\left(\mathscr{E}^t\right)\right\|_F^2\\ &= \left\|\mathscr{E}^t\right\|_F^2.\end{aligned}$$

Since $\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right)$ is block diagonal,

$$\sigma_{\min}^{+}\left(\mathbb{E}\left[\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right)\right]\right)=\min_{k\in[n-1]}\sigma_{\min}^{+}\left(\mathbb{E}\left[\left(\widehat{\mathbf{P}}_i\right)_k\right]\right).$$

Factoring $\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right)$ and using Fact 1,

$$\begin{aligned}\operatorname{bdiag}\left(\widehat{\mathscr{P}_i}\right) &= \operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}^*}\right)\operatorname{bdiag}\left(\widehat{\left(\mathscr{A}_{i::}\mathscr{A}_{i::}^*\right)^{-1}}\right)\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)\\ &= \operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)^*\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)^{-1}\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)\\ &= \operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)^*\left[\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}^*}\right)\right]^{-1}\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right).\end{aligned}$$

Noting that $\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)\operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}^*}\right)$ is a diagonal matrix, one can see that $\left(\widehat{\mathbf{P}}_i\right)_k$ is the projection onto $\left(\widehat{\mathscr{A}_{i::}}\right)_k$ by rewriting the kth frontal face of $\widehat{\mathscr{P}_i}$ as

$$\left(\widehat{\mathbf{P}}_i\right)_k = \frac{\left(\widehat{\mathscr{A}_{i::}}\right)_k^* \left(\widehat{\mathscr{A}_{i::}}\right)_k}{\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k}.$$

We can thus rewrite Eq. (B.1) as

$$\mathbb{E}\left[\left\|\mathscr{P}_i\mathscr{E}^t\right\|_F^2\right] \geq \min_{k\in[n-1]} \sigma_{\min}^+\left(\mathbb{E}\left[\frac{\left(\widehat{\mathscr{A}_{i::}}\right)_k^* \left(\widehat{\mathscr{A}_{i::}}\right)_k}{\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k}\right]\right)\left\|\mathscr{E}^t\right\|_F^2. \tag{B.2}$$

The expectation of Eq. (B.1) can now be calculated explicitly. For simplicity, we assume that the row indices i are sampled uniformly. As in MRK extensions and literature, many other sampling distributions could be used.

To derive a lower bound for the smallest singular value in Eq. (B.2), define

$$\left\|\widehat{\mathbf{A}}_k\right\|_{\infty,2}^2 := \max_i\left[\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k\right]. \tag{B.3}$$

The values $\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k$ are necessarily positive for all $k \in [n-1]$ when $\mathscr{A}_{i::}\mathscr{A}_{i::}^*$ is invertible for all $i \in [m-1]$. as

$$\begin{aligned}
\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k &= \operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_{kk} \\
&= \operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}}\right)_k \operatorname{bdiag}\left(\widehat{\mathscr{A}_{i::}^*}\right)_k \\
&= (\mathbf{F}_n)_{k:}\operatorname{bcirc}\left(\mathscr{A}_{i::}\right)\operatorname{bcirc}\left(\mathscr{A}_{i::}^*\right)(\mathbf{F}_n)_{k:}^* \\
&= (\mathbf{F}_n)_{k:}\operatorname{bcirc}\left(\mathscr{A}_{i::}\right)\operatorname{bcirc}\left(\mathscr{A}_{i::}\right)^*(\mathbf{F}_n)_{k:}^* \\
&= \left\|\operatorname{bcirc}\left(\mathscr{A}_{i::}\right)^*(\mathbf{F}_n)_{k:}^*\right\|_2^2.
\end{aligned}$$

Now, it can be easily verified that

$$\begin{aligned}
\sigma_{\min}^+\left(\mathbb{E}\left[\frac{\left(\widehat{\mathscr{A}_{i::}}\right)_k^* \left(\widehat{\mathscr{A}_{i::}}\right)_k}{\left(\widehat{\mathscr{A}_{i::}\mathscr{A}_{i::}^*}\right)_k}\right]\right) &\geq \sigma_{\min}^+\left(\frac{1}{m}\sum_{i=0}^{m-1}\frac{\left(\widehat{\mathscr{A}_{i::}}\right)_k^* \left(\widehat{\mathscr{A}_{i::}}\right)_k}{\left\|\widehat{\mathbf{A}}_k\right\|_{\infty,2}^2}\right) \\
&= \frac{\left[\sigma_{\min}^+\left(\widehat{\mathbf{A}}_k\right)\right]^2}{m\left\|\widehat{\mathbf{A}}_k\right\|_{\infty,2}^2}.
\end{aligned}$$

The projected error of Eq. (B.2) then becomes

$$\mathbb{E}\left[\left\|\mathscr{P}_i\mathscr{E}^t\right\|_F^2\right] \geq \min_{k\in[n-1]} \frac{\left[\sigma_{\min}^{+}\left(\widehat{\mathbf{A}}_k\right)\right]^2}{m\left\|\widehat{\mathbf{A}}_k\right\|_{\infty,2}^2}\left\|\mathscr{E}^t\right\|_F^2.,$$

We can thus rewrite the guarantee in Theorem 4.1 for uniform random sampling of the row indices i as

$$\mathbb{E}\left[\left\|\mathscr{X}^{t+1}-\mathscr{X}^*\right\|_F^2\middle|\mathscr{X}^0\right] \leq \left(1-\min_{k\in[n-1]} \frac{\left[\sigma_{\min}^{+}\left(\widehat{\mathbf{A}}_k\right)\right]^2}{m\left\|\widehat{\mathbf{A}}_k\right\|_{\infty,2}^2}\right)^{t+1}\left\|\mathscr{X}^0-\mathscr{X}^*\right\|_F^2.$$

□

References

1. Agmon, S.: The relaxation method for linear inequalities. Can. J. Math. **6**, 382–392 (1954)
2. Ahn, C.H., Jeong, B.S., Lee, S.Y.: Efficient hybrid finite element-boundary element method for 3-dimensional open-boundary field problems. IEEE Trans. Magn. **27**, 4069–4072 (1991)
3. Bone and joint ct-scan data. https://isbweb.org/data/vsj/
4. Censor, Y.: Row-action methods for huge and sparse systems and their applications. SIAM Rev. **23**(4), 444–466 (1981)
5. Czuprynski, K.D., Fahnline, J.B., Shontz, S.M.: Parallel boundary element solutions of block circulant linear systems for acoustic radiation problems with rotationally symmetric boundary surfaces. In: INTER-NOISE and NOISE-CON Congress and Conference Proceedings, vol. 2012, pp. 2812–2823. Institute of Noise Control Engineering (2012)
6. De Loera, J.A., Haddock, J., Needell, D.: A sampling Kaczmarz–Motzkin algorithm for linear feasibility. SIAM J. Sci. Comput. **39**(5), S66–S87 (2017)
7. Drineas, P., Mahoney, M.W.: Randnla: randomized numerical linear algebra. Commun. ACM **59**(6), 80–90 (2016)
8. Finding his Voice. Western Electric Company (1929). https://archive.org/details/FindingH1929
9. Elfving, T.: Block-iterative methods for consistent and inconsistent linear equations. Numer. Math. **35**(1), 1–12 (1980)
10. Gower, R.M., Richtárik, P.: Randomized iterative methods for linear systems. SIAM J. Matrix Anal. Appl. **36**(4), 1660–1690 (2015)
11. Haddock, J., Needell, D.: On Motzkin's method for inconsistent linear systems. BIT **59**(2), 387–401 (2019)
12. Hao, N., Kilmer, M.E., Braman, K., Hoover, R.C.: Facial recognition using tensor–tensor decompositions. SIAM J. Imaging Sci. **6**(1), 437–463 (2013)
13. Kaczmarz, M.S.: Angenäherte auflösung von systemen linearer gleichungen. Bull. Acad. Polonaise Sci. Lett. **35**, 355–357 (1937)
14. Kernfeld, E., Kilmer, M., Aeron, S.: Tensor–tensor products with invertible linear transforms. Linear Algebra Appl. **485**, 545–570 (2015)
15. Kilmer, M.E., Braman, K., Hao, N., Hoover, R.C.: Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging. SIAM J. Matrix Anal. Appl. **34**(1), 148–172 (2013)
16. Kilmer, M.E., Martin, C.D.: Factorization strategies for third-order tensors. Linear Algebra Appl. **435**(3), 641–658 (2011)
17. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009)
18. Liu, Z., Zhao, H.V., Elezzabi, A.Y.: Block-based adaptive compressed sensing for video. In: IEEE Image Proc., pp. 1649–1652. IEEE (2010)

19. Lund, K.: The tensor t-function: a definition for functions of third-order tensors. Numer. Linear Algebra Appl. **27**(3), e2288 (2020)
20. Ma, A., Needell, D., Ramdas, A.: Convergence properties of the randomized extended Gauss-Seidel and Kaczmarz methods. SIAM J. Matrix Anal. A **36**(4), 1590–1604 (2015)
21. Majumdar, A., Ward, R.K.: Face recognition from video: an MMV recovery approach. In: Int. Conf. Acoust. Spee, pp. 2221–2224. IEEE (2012)
22. Miao, Y., Qi, L., Wei, Y.: Generalized tensor function via the tensor singular value decomposition based on the t-product. Linear Algebra Its Appl. **590**, 258–303 (2020)
23. Motzkin, T.S., Schoenberg, I.J.: The relaxation method for linear inequalities. Can. J. Math **6**, 393–404 (1954)
24. Needell, D.: Randomized Kaczmarz solver for noisy linear systems. BIT **50**(2), 395–403 (2010)
25. Needell, D., Tropp, J.A.: Paved with good intentions: analysis of a randomized block Kaczmarz method. Linear Algebra Appl. **441**, 199–221 (2014)
26. Needell, D., Ward, R., Srebro, N.: Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. Adv. Neural Inf. Process. Syst. **27**, 1017–1025 (2014)
27. Newman, E., Horesh, L., Avron, H., Kilmer, M.: Stable tensor neural networks for rapid deep learning. arXiv preprint arXiv:1811.06569 (2018)
28. Nutini, J., Sepehry, B., Virani, A., Laradji, I., Schmidt, M., Koepke, H.: Convergence rates for greedy Kaczmarz algorithms. In: UAI (2016)
29. Petra, S., Popa, C.: Single projection Kaczmarz extended algorithms. Numer. Algorithms **73**(3), 791–806 (2016)
30. Richtárik, P., Takác, M.: Stochastic reformulations of linear systems: algorithms and convergence theory. SIAM J. Matrix Anal. Appl. **41**(2), 487–524 (2020)
31. Semerci, O., Hao, N., Kilmer, M.E., Miller, E.L.: Tensor-based formulation and nuclear norm regularization for multienergy computed tomography. IEEE Trans. Image Process. **23**(4), 1678–1693 (2014)
32. Soltani, S., Kilmer, M.E., Hansen, P.C.: A tensor-based dictionary learning approach to tomographic image reconstruction. BIT **56**(4), 1425–1454 (2016)
33. Song, G., Ng, M.K., Zhang, X.: Robust tensor completion using transformed tensor singular value decomposition. Numer. Linear Algebra Appl. **27**(3), e2299 (2020). https://doi.org/10.1002/nla.2299
34. Strohmer, T., Vershynin, R.: A randomized Kaczmarz algorithm with exponential convergence. J. Fourier Anal. Appl. **15**(2), 262 (2009)
35. Vescovo, R.: Electromagnetic scattering from cylindrical arrays of infinitely long thin wires. Electron. Lett. **31**(19), 1646–1647 (1995)
36. Wang, X., Che, M., Wei, Y.: Tensor neural network models for tensor singular value decompositions. Comput. Optim. Appl. **75**(3), 753–777 (2020)
37. Zhang, Z., Aeron, S.: Denoising and completion of 3d data via multidimensional dictionary learning. In: Int. Join. Conf. Artif., pp. 2371–2377 (2016)
38. Zhang, Z., Aeron, S.: Exact tensor completion using t-SVD. IEEE Trans. Signal Process. **65**(6), 1511–1526 (2017)
39. Zhang, Z., Ely, G., Aeron, S., Hao, N., Kilmer, M.: Novel methods for multilinear data completion and de-noising based on tensor-SVD. In: CVPR, pp. 3842–3849. IEEE (2014)
40. Zhou, P., Lu, C., Lin, Z., Zhang, C.: Tensor factorization for low-rank tensor completion. IEEE Trans. Image Process. **27**(3), 1152–1163 (2018)
41. Zouzias, A., Freris, N.M.: Randomized extended Kaczmarz for solving least squares. SIAM J. Matrix Anal. Appl. **34**(2), 773–793 (2013)