

On-the-Fly Communication-and-Computing for Distributed Tensor Decomposition Over MIMO Channels

Xu Chen, Erik G. Larsson, and Kaibin Huang

Abstract

Distributed tensor decomposition (DTD) is a fundamental data-analytics technique that extracts latent important properties from high-dimensional multi-attribute datasets distributed over edge devices. Conventionally its wireless implementation follows a one-shot approach that first computes local results at devices using local data and then aggregates them to a server with communication-efficient techniques such as *over-the-air computation* (AirComp) for global computation. Such implementation is confronted with the issues of limited storage-and-computation capacities and link interruption, which motivates us to propose a framework of on-the-fly communication-and-computing (FlyCom²) in this work. The proposed framework enables streaming computation with low complexity by leveraging a random sketching technique and achieves progressive global aggregation through the integration of progressive uploading and *multiple-input-multiple-output* (MIMO) AirComp. To develop FlyCom², an on-the-fly sub-space estimator is designed to take real-time sketches accumulated at the server to generate online estimates for the decomposition. Its performance is evaluated by deriving both deterministic and probabilistic error bounds using the perturbation theory and concentration of measure. Both results reveal that the decomposition error is inversely proportional to the population of sketching observations received by the server. To further rein in the noise effect on the error, we propose a threshold-based scheme to select a subset of sufficiently reliable received sketches for DTD at the server. Experimental results validate the performance gain of the proposed selection algorithm and show that compared to its one-shot counterparts, the proposed FlyCom² achieves comparable (even better in the case of large eigen-gaps) decomposition accuracy besides dramatically reducing devices' complexity costs.

X. Chen and K. Huang are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (Email: {chenxu, huangkb}@eee.hku.hk). E. G. Larsson is with the Department of Electrical Engineering (ISY), Linköping University, 58183 Linköping, Sweden (Email: {erik.g.larsson}@liu.se). Corresponding author: K. Huang.

I. INTRODUCTION

In mobile networks, enormous amounts of data are continuously being generated by billions of edge devices. Data analytics can be performed on distributed data to support a broad range of mobile applications, ranging from e-commerce to autonomous driving to IoT sensing [1], [2]. One basic class of techniques is *tensor decomposition*, which extracts a low-dimensional structure from large-scale multi-attribute data with a tensor representation (a high-dimensional counterpart of a matrix) [3], [4]. A popular technique in this class, the Tucker decomposition, is a higher-dimensional extension of the *singular-value decomposition* (SVD) that has supported diverse applications such as Google’s image recognition and Cynefin’s spotting of anomalies. In mobile networks, tensor decomposition can be implemented in a centralized manner, which requires uploading of high-dimensional data from many devices to a central server. However, such implementation is stymied not only by a communication bottleneck but also by data privacy issues [5], [6].

In view of these issues, we focus on *distributed tensor decomposition* (DTD) that avoids direct data uploading and reduces communication overhead by distributing the computation of data tensors to the devices. A direct distributed implementation would call for parallel iterative methods such as alternating least squares [7] and stochastic gradient descent [8], [9] over edge devices, which, however, results in high communication overhead due to slow convergence. On the other hand, DTD can be realized via *one-shot* distributed matrix analysis techniques [10]–[12], since the desired orthogonal factor matrices can be estimated as the principal eigenspaces of unfolding matrices of the tensor along different modes [13]. These one-shot methods improve communication efficiency at a slight cost of decomposition accuracy by following two steps: 1) computing local estimates of the desired factor matrix at each of the devices using local data; 2) uploading and aggregating the local estimates at the server to compute a global estimate. Though alleviated, the communication bottleneck still exists due to the required aggregation of high-dimensional local tensors over potentially many devices. This multi-access problem can be addressed by using a technique called *over-the-air computation* (AirComp), which exploits the waveform superposition property of a multi-access channel to realize over-the-air data aggregation in one shot [5], [14]. AirComp finds applications in communication-efficient distributed computing and learning, and has been especially popular for federated learning (see, e.g., [15]–[17]).

Considering a DTD system with AirComp, this work aims to solve two open problems. The first is the prohibitive cost and latency of computation at resource-constrained devices. A traditional one-shot DTD algorithm requires each device to perform eigenvalue decomposition of a potentially high-dimensional local dataset. The resulting computation complexity increases *super-linearly* with the data dimensions [18], and the consequent latency makes it difficult for DTD to support emerging mission-critical applications [19]. The second problem is that the one-shot transmissions by devices are susceptible to link disruption. Specifically, a loss of connection during the transmission of high-dimensional local principal components can render the already received partial data useless. In other words, the existing designs lack the feature of graceful performance degradation due to fading.

To solve these problems, we propose the novel framework *on-the-fly communication-and-computing* (FlyCom²). Underpinning this framework is the use of a technique from randomized linear algebra, *randomized sketching*, that generates low-dimensional random representations, called *sketches*, of a high-dimensional data sample by projecting it onto randomly generated low-dimensional sub-spaces [20], [21]. This technique has been successfully used in diverse applications ranging from online data tracking [22] to matrix approximation [21]. In FlyCom², in place of the traditional high-dimensional local eigenspaces, each device generates a stream of low-dimensional sketches for uploading to the server. Considering a *multiple-input-multiple-output* (MIMO) channel, the simultaneous transmission of local sketches is enabled by spatially multiplexed AirComp [23]. Upon its arrival at the server, each aggregated sketch is immediately used to improve the global tensor decomposition, giving the name of FlyCom². Since random sketches serve as independent observations of the tensor, the server can produce an estimate for tensor decomposition in every time slot based on the sketches already received. The FlyCom² framework addresses the above-mentioned open problems in several aspects. First, random sketching that involves matrix multiplication has much lower complexity than eigen-decomposition and helps reduce the complexity of on-device computation. Second, the DTD accuracy depends on the number of successfully received sketches and hence is robust to loss of sketches in the transmission. This gives FlyCom² a graceful degradation property in the event of link disruptions or packet losses. Third, as the principal components of a high-dimensional tensor are usually low-dimensional, the progressive DTD at the server is shown to approach its optimal performance quickly as the number of aggregated sketches increases, thereby reining in the communication overhead. Last, the parallel streaming communication and computation in

FlyCom² are more efficient than the sequential operations of the traditional one-shot algorithms due to the communication-computation separation.

In designing the FlyCom² framework, this work makes the following key contributions.

- *On-the-Fly Sub-space Detection*: One key component of the framework is an optimal on-the-fly detector at the server to estimate the tensor's principal eigenspace from the received, noisy (aggregated) sketches. To design a *maximum likelihood* (ML) detector, a whitening technique is used to pre-process the sketches so as to yield an effective global observation, which is shown to have the covariance matrix sharing the same eigenspace as the tensor. Using the result, the ML estimation problem is formulated as a *sub-space alignment problem* and is solved in closed form. It is observed from the solution that the optimal estimate of the desired principal eigenspace approaches its ground truth as the said observation's dimensionality grows (or equivalently more sketches are received).
- *DTD Error Analysis*: The end-to-end performance of the FlyCom²-based DTD system is measured by the squared error of the estimated principal eigenspace *with respect to* (w.r.t.) its ground truth. Using perturbation theory and concentration of measure, bounds are derived on both the error and its expectation. These results reveal that the error consists of one residual component contributed by non-principal components and the other component caused by random sketching. Moreover, the error is observed to be linearly proportional to the number of received sketches, validating the earlier claims on the progressive nature of the designed DTD as well as its feature of graceful degradation. This also suggests a controllable trade-off between the decomposition accuracy and communication overhead, which is useful for practical implementation.
- *Threshold-Based Sketch Selection*: Removing severely channel distorted sketches from use in the sub-space detection can lead to performance improvements. This motivates the design of a sketch-selection scheme that applies a threshold on a scaling factor in MIMO AirComp that reflects the received *signal-to-noise ratio* (SNR) of an aggregated sketch. We show that such a threshold can be efficiently optimized by an enumeration method whose complexity is polynomial in the population of received sketches.

The remainder of the paper is organized as follows. Section II introduces system models and metrics. Section III gives an overview of the proposed FlyCom² framework. Then, Section IV presents the design of the on-the-fly sub-space estimator and its error analysis. The sketch-

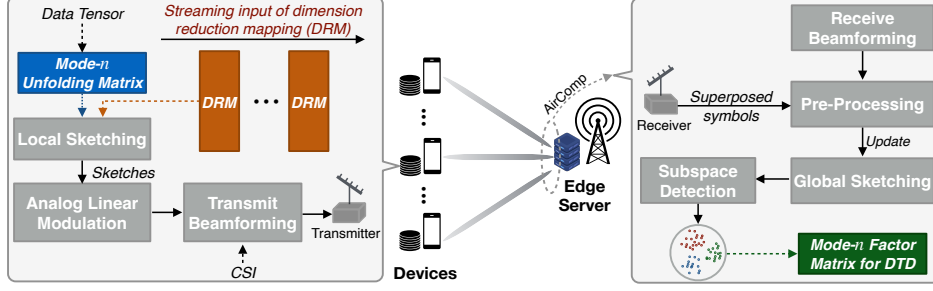


Fig. 1. On-the-fly communication-and-computing for distributed tensor decomposition.

selection scheme is proposed in Section V. Numerical results are provided in Section VI, followed by concluding remarks in Section VII.

II. MODELS, OPERATIONS, AND METRICS

We consider the support of DTD in a MIMO system, as illustrated in Fig. 1. The relevant models, operations and metrics are described in what follows.

A. Distributed Tensor Decomposition

We consider the distributed implementation of the popular Tucker method for tensor decomposition [13]. For ease of notation, the tensor is assumed to have N modes; these modes generalize the concepts of columns and rows in matrices with the first $(N - 1)$ modes corresponding to data features and mode N indexing data samples. For instance, in a surveillance system, images captured by multiple cameras are expressed as local tensors with three modes indicating pixels, colors, and data sample indices, respectively. Let the samples collected by device k be represented by a local tensor $\mathcal{X}_k \in \mathbb{R}^{I_1^{(k)} \times I_2^{(k)} \times \dots \times I_N^{(k)}}$, where $I_n^{(k)}$ denotes the dimensionality of mode n of local tensor k . To simplify notation, we assume that local tensors have the same dimensions for their feature modes: $I_n^{(k)} = I_n, \forall k, 1 \leq n \leq N - 1$. Next, these local tensors are aggregated from K devices to form a global tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ with $I_N = \sum_k I_N^{(k)}$. The Tucker decomposition of \mathcal{X} can be written as [13]

$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_N \mathbf{U}_N \triangleq \tilde{\mathcal{X}}, \quad (1)$$

where $\mathcal{G} \in \mathbb{R}^{r_1 \times r_2 \times \dots \times r_N}$ represents a core tensor [that generalizes the singular values in SVD], $\mathbf{U}_n \in \mathbb{R}^{I_n \times r_n}$ is an orthogonal factor matrix corresponding to the n -th mode, satisfying $\mathbf{U}_n^\top \mathbf{U}_n = \mathbf{I}_{r_n}$ with r_n ($r_n \leq I_n$) representing the number of principal dimensions, and \times_n denotes the

mode- n matrix product [13]. In the sequel, we pursue these factor matrices $\{\mathbf{U}_n\}$ as they reveal the characteristics of the data tensor at different modes. Given $\{\mathbf{U}_n\}$, the computation of \mathcal{G} is straightforward [3]. In centralized computation with full data aggregation, $\{\mathbf{U}_n\}$ can be computed by using the *higher-order SVD* approach [3], [20]. In this approach, the tensor is first flattened along a chosen mode n to yield a matrix $\mathbf{X}^{(n)} \in \mathbb{R}^{I_n \times J_n}$ with $J_n = \prod_{j=1, j \neq n}^N I_j$, termed *mode- n unfolding*; then the desired factor matrix is computed as $\mathbf{U}_n = [\mathbf{u}_1, \dots, \mathbf{u}_{r_n}]$ where \mathbf{u}_i is given by the i -th principal eigenvector of the mode- n unfolding. Let this operation be represented by $\mathcal{S}_{r_n}(\cdot)$ and hence $\mathbf{U}_n = \mathcal{S}_{r_n}(\mathbf{X}^{(n)}(\mathbf{X}^{(n)})^\top)$.

In contrast with its centralized counterpart, DTD computes the eigenspaces of different unfolding matrices distributively. This avoids the aggregation of raw data and preserves the data ownership [10], [11]. Considering the computation of \mathbf{U}_n , DTD goes through the following procedure: 1) local tensors are flattened along chosen mode n to generate local unfoldings, denoted by $\{\mathbf{X}_k^{(n)}\}$; 2) devices compute low-dimensional component $\{\mathbf{S}_k\}$ from local unfoldings $\{\mathbf{X}_k^{(n)}\}$ through dimensionality reduction techniques; 3) the server gathers these local components from devices and aggregates them into a global component, denoted by \mathbf{S} , to yield a global estimate of the ground truth, \mathbf{U}_n . It is worth mentioning that the computation results $\{\mathbf{S}_k\}$ depend on a particular dimensionality reduction technique. For example, when using *principal component analysis* (PCA) [10], [11], $\{\mathbf{S}_k\}$ are computed as the principal eigenspaces of $\{\mathbf{X}_k^{(n)}\}$ at the devices, and then the server averages them to estimate \mathbf{U}_n . In this work, a random sketching approach is adopted as elaborated in Section III.

B. MIMO Over-the-Air Computation

FlyCom² builds on MIMO AirComp to aggregate local results over the air, which is described as follows. First, let N_r and N_t , with $N_r \geq N_t$, denote the numbers of antennas at the edge server and each device, respectively. We assume perfect transmit *channel state information* (CSI) as well as symbol-level and phase synchronization between the devices [23]. Time is slotted and then grouped to form *coherence blocks* with t denoting the block index. In each time slot, an $N_t \times 1$ vector of complex scalar symbols is transmitted over N_t antennas. Then a coherence block spans at least I symbol slots to support the transmission of an $N_t \times I$ matrix. In an arbitrary block, say t , all edge devices transmit simultaneously their $I \times M$ real matrices, denoted as $\{\mathbf{S}_{t,k}\}$, each of which is termed a *matrix symbol*. As a result, the server receives an over-the-air

aggregated matrix symbol, \mathbf{Y}_t , as

$$\mathbf{Y}_t = \mathbf{A}_t \sum_{k=1}^K \mathbf{H}_{t,k} \mathbf{B}_{t,k} \mathbf{S}_{t,k}^\top + \mathbf{A}_t \mathbf{Z}_t,$$

where $\mathbf{H}_{t,k} \in \mathbb{C}^{N_r \times N_t}$ denotes the channel matrix corresponding to device k , \mathbf{Z}_t models additive Gaussian noise with *independent and identically distributed* (i.i.d.) elements of $\mathcal{CN}(0, \sigma^2)$, $\mathbf{A}_t \in \mathbb{C}^{M \times N_r}$ and $\mathbf{B}_{t,k} \in \mathbb{C}^{N_t \times M}$ ($M \leq N_t$) denote receive and transmit beamforming matrices, respectively. To realize AirComp, we consider *zero forcing* (ZF) transmit beamforming that inverts individual MIMO channels [23]. Mathematically, conditioned on a fixed receive beamformer, transmit beamforming matrices are given as

$$\mathbf{B}_{t,k} = (\mathbf{A}_t \mathbf{H}_{t,k})^H (\mathbf{A}_t \mathbf{H}_{t,k} \mathbf{H}_{t,k}^H \mathbf{A}_t^H)^{-1}. \quad (2)$$

The received matrix \mathbf{Y}_t is then rewritten as

$$\mathbf{Y}_t = \sum_{k=1}^K \mathbf{S}_{t,k}^\top + \mathbf{A}_t \mathbf{Z}_t. \quad (3)$$

In the absence of noise, the AirComp in (3) provides the one-shot realization of the desired aggregation operation for DTD. The average transmission power of each device is constrained to not exceed a power budget of P per slot, i.e. $\forall t, k$

$$\begin{aligned} \mathbb{E} [\|\mathbf{B}_{t,k} \mathbf{S}_{t,k}^\top\|_F^2] &= \text{Tr} \left((\mathbf{A}_t \mathbf{H}_{t,k} \mathbf{H}_{t,k}^H \mathbf{A}_t^H)^{-1} \mathbb{E}[\mathbf{S}_{t,k}^\top \mathbf{S}_{t,k}] \right) \\ &\leq IP. \end{aligned} \quad (4)$$

The transmit SNR is then given by $\gamma = \frac{P}{\sigma^2}$.

C. Error Metric

With the above-described MIMO AirComp, a noisy version of \mathbf{U}_n , denoted by $\tilde{\mathbf{U}}_n$, will be computed progressively from a set of received matrices $\{\mathbf{Y}_t\}$ (see Section III). The $\tilde{\mathbf{U}}_n$ deviates from the ground truth due to both distributed computation and channel noise. The resulting error can be a performance metric of DTD in the wireless system. Mathematically, given $\tilde{\mathcal{X}}$ as the tensor derived from $\{\tilde{\mathbf{U}}_n\}$, the error is measured as $\|\mathcal{X} - \tilde{\mathcal{X}}\|_F^2$ that can be bounded as $\|\mathcal{X} - \tilde{\mathcal{X}}\|_F^2 \leq \sum_{n=1}^N \|(\mathbf{I}_{I_n} - \tilde{\mathbf{U}}_n \tilde{\mathbf{U}}_n^\top) \mathbf{X}^{(n)}\|_F^2$ [20]. This suggests that the overall error is determined

by the error of independently decomposing each unfolding matrix. Hence, we define the DTD error as

$$d(\tilde{\mathbf{U}}_n, \mathbf{X}^{(n)}) = \|(\mathbf{I}_{I_n} - \tilde{\mathbf{U}}_n \tilde{\mathbf{U}}_n^\top) \mathbf{X}^{(n)}\|_F^2. \quad (5)$$

III. OVERVIEW OF ON-THE-FLY COMMUNICATION-AND-COMPUTING

To support DTD over edge devices with limited computation power, we propose a FlyCom² framework as shown in Fig. 1. Next, we first briefly introduce the random approach exploited in FlyCom² and then explain how to use FlyCom² to support DTD.

A. Data Dimensionality Reduction via Random Sketching

Recall that the DTD requires data dimensionality reduction on devices prior to transmission. For high-dimensional tensors, the traditional PCA technique becomes too complex for resource-constrained devices. To address this issue, we adopt a technique for random dimensionality reduction, known as *random sketching*, which is simpler than PCA as it only relies on matrix multiplication and also requires a smaller number of passes over the datasets [21]. Specifically, given an $I \times J$ data matrix \mathbf{X} , random sketching uses a $J \times M$ random matrix, termed *dimension reduction mapping* (DRM) and denoted by Ω , to map \mathbf{X} to an $I \times M$ sketch matrix \mathbf{S} with $J \gg M$: $\mathbf{S} = \mathbf{X}\Omega$. The mapping Ω can be composed of i.i.d. Gaussian elements and projects the high-dimensional \mathbf{X} to random directions in a space of low dimensionality. Despite the random projection, the mutual vector distances between the rows of \mathbf{X} can be approximately preserved such that the principal (column) eigenspace of the sketch, \mathbf{S} , constitutes a good approximation of \mathbf{X} . The approximation accuracy grows as M increases and becomes perfect when M is equal to J [21]. Importantly, to estimate an r -dimensional principal eigenspace, random sketching has the complexity of $\mathcal{O}(IJM)$ and requires a single data pass of memory, as opposed to the complexity of $\mathcal{O}(\min\{I, J\}^2 \times \max\{I, J\})$ and $\mathcal{O}(r)$ memory passes in PCA [18].

B. FlyCom²-Based DTD

Based on the preceding random-sketching technique, we propose the FlyCom² framework that decomposes the high-dimensional DTD into on-the-fly processing and transmission of streams of low-dimensional random sketches. Thereby, we not only overcome devices' resource constraints but also achieve a graceful reduction of DTD error as the communication time increases. Without loss of generality, we focus on the computation of the principal eigenspace \mathbf{U}_n for an arbitrary

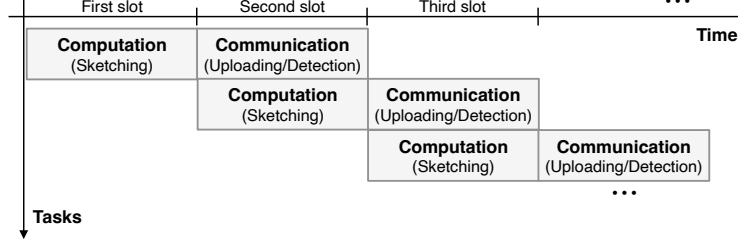


Fig. 2. Parallelization between communication and computation.

data-feature mode n with $n \in [1, \dots, N - 1]$. To simplify notation, the superscript (n) and subscript n are omitted. The detailed operations of FlyCom² are described as follows.

1) *On-the-Fly Computation at Devices*: Each device streams a sequence of low-dimensional local sketches to the server by generating and transmitting them one by one in data packets. First, the progressive computation of local sketches at devices is introduced as follows. Let each local tensor, say \mathcal{X}_k at device k , be flattened along the desired mode to generate the unfolding matrix \mathbf{X}_k . Then, in the (matrix-symbol) slot t , each device k draws i.i.d. $\mathcal{N}(0, 1)$ entries to form a $J \times M$ DRM, denoted by $\Omega_{t,k}$, or retrieves it efficiently from memory [24]. Then an M -dimensional local sketch for \mathbf{X}_k can be computed as $\mathbf{S}_{t,k} = \mathbf{X}_k \Omega_{t,k}$, which is then uploaded to the server immediately before computing the next sketch $\mathbf{S}_{t+1,k}$. This allows efficient communication-and-computation parallelization as shown in Fig. 2.

2) *On-the-Fly Global Random Sketching*: MIMO AirComp is used for low-latency aggregation of the local sketches simultaneously streamed by devices. Local temporal sketches are progressively aggregated at the server by linearly modulating them as MIMO AirComp symbols. Consider the uploading of the t -th local sketches. It follows from (3) that the matrix symbol received at the server can be written as

$$\mathbf{Y}_t^\top = \sum_k \mathbf{X}_k \Omega_{t,k} + \mathbf{Z}_t^\top \mathbf{A}_t^\top. \quad (6)$$

To explain how to use \mathbf{Y}_t in estimating the principal eigenspace of the global unfolding matrix \mathbf{X} , we first consider the case without channel noise, in which $\mathbf{Y}_t^\top = \sum_k \mathbf{X}_k \Omega_{t,k}$. Since the global tensor \mathcal{X} is given by assembling local tensors along mode N , the corresponding global unfolding matrix, denoted by \mathbf{X} , is related to the local unfoldings $\{\mathbf{X}_k\}$ as

$$\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K]. \quad (7)$$

It follows that

$$\mathbf{Y}_t^\top = [\mathbf{X}_1, \dots, \mathbf{X}_K][\boldsymbol{\Omega}_{t,1}^\top, \dots, \boldsymbol{\Omega}_{t,K}^\top]^\top \triangleq \mathbf{X}\mathbf{F}_t,$$

where we define

$$\mathbf{F}_t = [\boldsymbol{\Omega}_{t,1}^\top, \dots, \boldsymbol{\Omega}_{t,K}^\top]^\top.$$

As $\{\boldsymbol{\Omega}_{t,k}\}$ are mutually independent, \mathbf{F}_t has i.i.d. $\mathcal{N}(0, 1)$ elements and can be used as an M -dimensional DRM for randomly sketching \mathbf{X} . Therefore, in the absence of channel noise, \mathbf{Y}_t gives an M -dimensional global sketch for \mathbf{X} . The dimension of the global sketch grows, thereby improving the DTD accuracy, as more aggregated local sketches are received (or equivalently as t progresses), giving the name of on-the-fly global sketching.

3) *On-the-Fly Sub-space Detection at the Server:* In the case with channel noise, the server can produce an estimate of the desired principal eigenspace, \mathbf{U} , based on the noisy observations accumulated up to the current symbol slot. Specifically, in slot t , given the current and past received matrix symbols, $\{\mathbf{Y}_\ell\}_{\ell \leq t}$, and the receive beamformers $\{\mathbf{A}_\ell\}_{\ell \leq t}$ (discussed in the sequel), the server estimates \mathbf{U} as

$$\tilde{\mathbf{U}} = f(\{\mathbf{Y}_\ell\}_{\ell \leq t}, \{\mathbf{A}_\ell\}_{\ell \leq t}), \quad (8)$$

where the estimator $f(\cdot)$ is optimized in the sequel to minimize the DTD error in (5).

Following the above discussion, the procedure for FlyCom²-based DTD is summarized as follows.

To compute the principal eigenspace of the global unfolding matrix \mathbf{X} , initialize $t = 1$, and FlyCom²-based DTD repeats:

- Step 1: Each device, say device k , computes a local sketch using $\mathbf{S}_{t,k} = \mathbf{X}_k \boldsymbol{\Omega}_{t,k}$;
- Step 2: The server receives $\mathbf{Y}_t^\top = \sum_k \mathbf{X}_k \boldsymbol{\Omega}_{t,k} + \mathbf{Z}_t^\top \mathbf{A}_t^\top$ via MIMO AirComp;
- Step 3: The server computes an estimate of the eigenspace of \mathbf{X} : $\tilde{\mathbf{U}} = f(\{\mathbf{Y}_\ell\}_{\ell \leq t}, \{\mathbf{A}_\ell\}_{\ell \leq t})$;
- Step 4: Set $t = t + 1$;

Until $t = T$.

The key component of the FlyCom² framework, the on-the-fly sub-space estimator $f(\cdot)$, is designed in Section IV. The performance of FlyCom²-based DTD is enhanced using a sketch selection algorithm designed in Section V.

IV. OPTIMAL SUB-SPACE DETECTION FOR FLYCOM²

In this section, we design the sub-space detection function of the FlyCom² framework, namely $f(\cdot)$ mentioned in the preceding section. It consists of two stages – pre-processing of received symbols and the subsequent sub-space estimation, which are summarized in Algorithm 1 and designed in the following sub-sections. Furthermore, the resultant DTD error is analyzed.

A. Pre-Processing of Received Matrix Symbols

The pre-processing function is to accumulate received matrix symbols from slot 1 to the current slot, t , and generate from them an effective matrix for the ensuing sub-space detection. The operation is instrumental for on-the-fly detection to obtain a progressive performance improvement. The design of the pre-processing takes several steps. First, since the transmitted symbol $\mathbf{X}\mathbf{F}_t$ is real but the channel noise is complex, the real part of the received symbols, namely \mathbf{Y}_t in (6), gives an effective observation of the transmitted symbol¹. Let $\tilde{\mathbf{Y}}_t$ denote the effective observation in slot t and $\tilde{\mathbf{Z}}_t$ the real part of $\mathbf{A}_t\mathbf{Z}_t$. It follows that

$$\tilde{\mathbf{Y}}_t = \Re\{\mathbf{Y}_t^\top\} = \mathbf{X}\mathbf{F}_t + \tilde{\mathbf{Z}}_t^\top. \quad (9)$$

Second, the relation between the eigenspace of \mathbf{X} and the accumulated observations up to the current slot is derived as follows. To this end, let the SVD of \mathbf{X} be expressed as

$$\mathbf{X} = \mathbf{U}_\mathbf{X}\Sigma_\mathbf{X}\mathbf{V}_\mathbf{X}^\top, \quad (10)$$

where $\Sigma_\mathbf{X}$ comprises descending singular values along its diagonal. Then, the accumulation of the current and past observations, denoted by $\hat{\mathbf{Y}}_t = [\tilde{\mathbf{Y}}_1, \tilde{\mathbf{Y}}_2, \dots, \tilde{\mathbf{Y}}_t]$, is a random Gaussian matrix as shown below.

Lemma 1. The accumulated aggregations, $\hat{\mathbf{Y}}_t$, can be decomposed as

$$\hat{\mathbf{Y}}_t = \mathbf{C}^{\frac{1}{2}}\mathbf{W}\mathbf{D}^{\frac{1}{2}},$$

¹It is possible to transmit the coefficients of $\mathbf{X}\mathbf{F}_t$ over both the in-phase and quadrature channels, which halves air latency. The extension is straightforward (see, e.g. [10, Section II]) but complicates the notation without providing new insights. Hence, only the in-phase channel is used in this work.

where the left covariance matrix $\mathbf{C} = \mathbf{X}\mathbf{X}^\top + \frac{1}{2tM}\sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell) \mathbf{I}_I$, the right covariance matrix $\mathbf{D} = \frac{\text{Tr}(\mathbf{X}^\top \mathbf{X}) \mathbf{I}_{tM} + \frac{1}{2} I \sigma^2 \text{diag}(\mathbf{A}_1 \mathbf{A}_1^H, \dots, \mathbf{A}_t \mathbf{A}_t^H)}{\text{Tr}(\mathbf{X}^\top \mathbf{X}) + \frac{1}{2tM} I \sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell)}$, and \mathbf{W} is a random Gaussian matrix with i.i.d. $\mathcal{N}(0, 1)$ entries.

Proof. See Appendix A □

Third, based on (10), the covariance matrix, \mathbf{C} , in Lemma 1 can be rewritten as

$$\mathbf{C} \triangleq \mathbf{U}_\mathbf{X} \mathbf{\Lambda} \mathbf{U}_\mathbf{X}^\top, \quad (11)$$

where we define $\mathbf{\Lambda} = \Sigma_\mathbf{X}^2 + \frac{1}{2tM}\sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell) \mathbf{I}_I$. Hence, the square root, $\mathbf{C}^{\frac{1}{2}}$, is given as

$$\mathbf{C}^{\frac{1}{2}} = \mathbf{U}_\mathbf{X} \mathbf{\Lambda}^{\frac{1}{2}}. \quad (12)$$

Remark 1 (Effective Sketching with Channel Noise). According to Lemma 1 and (12), the accumulated observations, $\hat{\mathbf{Y}}_t$, gives a sketch of the matrix $\mathbf{U}_\mathbf{X} \mathbf{\Lambda}^{\frac{1}{2}}$ using a Gaussian DRM with the covariance of \mathbf{D} . The matrix $\mathbf{U}_\mathbf{X} \mathbf{\Lambda}^{\frac{1}{2}}$ and the unfolding matrix \mathbf{X} share the eigenspace, $\mathbf{U}_\mathbf{X}$. Furthermore, as $\mathbf{\Lambda}$ retains the singular values in descending order, the top- r principal eigenspace of $\mathbf{U}_\mathbf{X} \mathbf{\Lambda}^{\frac{1}{2}}$ is identical to that of \mathbf{X} for any $1 \leq r \leq M$.

Finally, according to the preceding discussion, the desired principal eigenspace of \mathbf{X} can be estimated from the sketch $\hat{\mathbf{Y}}_t$. It is known that randomized sketching prefers DRMs with i.i.d. entries [21]. To improve the performance, $\hat{\mathbf{Y}}_t$ can be further “whitened” to equalize the right covariance \mathbf{D} . Specifically, let $\hat{\mathbf{Y}}_t$ be right-multiplied by $\mathbf{D}^{-\frac{1}{2}}$ to yield the final *effective observation* in time slot t as

$$\boxed{\Phi_t = \hat{\mathbf{Y}}_t \mathbf{D}^{-\frac{1}{2}} = \mathbf{U}_\mathbf{X} \mathbf{\Lambda}^{\frac{1}{2}} \mathbf{W}.} \quad (13)$$

To compute the covariance matrix \mathbf{D} , the server needs to acquire the value of $\text{Tr}(\mathbf{X}\mathbf{X}^\top) = \sum_k \text{Tr}(\mathbf{X}_k \mathbf{X}_k^\top)$. Note that each term in the summation, say $\text{Tr}(\mathbf{X}_k \mathbf{X}_k^\top)$, relates to the covariance of transmitted symbols, $\mathbb{E}[\mathbf{S}_{t,k}^\top \mathbf{S}_{t,k}]$, as

$$\mathbb{E}[\mathbf{S}_{t,k}^\top \mathbf{S}_{t,k}] = \mathbb{E}[\mathbf{\Omega}_{t,k}^\top \mathbf{X}_k^\top \mathbf{X}_k \mathbf{\Omega}_{t,k}] = \text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k) \mathbf{I}_M.$$

Then, $\text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k)$ can be acquired at the server by one-time feedback.

B. Optimal Sub-space Estimation

In this sub-section, the principal eigenspace of the unfolding matrix \mathbf{X} with dimensions fixed as r , is estimated from the effective observation given in (13) under the ML criterion. First, using (13), the distribution of the observation Φ_t conditioned on \mathbf{U} and Λ is given as

$$\Pr(\Phi_t | \mathbf{U}_\mathbf{X}, \Lambda) = \frac{\exp\left(-\frac{tM}{2} \text{Tr}(\Phi_t^\top \mathbf{U}_\mathbf{X} \Lambda^{-1} \mathbf{U}_\mathbf{X}^\top \Phi_t)\right)}{(2\pi)^{ItM/2} \det(\Lambda)^{tM/2}}.$$

This yields the logarithm of the likelihood function, required for ML estimation, as

$$\begin{aligned} \mathcal{L}(\mathbf{U}_\mathbf{X}; \Phi_t, \Lambda) &= \ln(\Pr(\Phi_t | \mathbf{U}_\mathbf{X}, \Lambda)), \\ &= -\frac{tM}{2} \text{Tr}(\Phi_t^\top \mathbf{U}_\mathbf{X} \Lambda^{-1} \mathbf{U}_\mathbf{X}^\top \Phi_t) \\ &\quad - \frac{ItM}{2} \ln(2\pi) - \frac{tM}{2} \ln(\det(\Lambda)). \end{aligned} \quad (14)$$

Let \mathbf{U} denote the desired r -dimensional principal components of \mathbf{X} , as obtained from splitting $\mathbf{U}_\mathbf{X} = [\mathbf{U}, \mathbf{U}^\perp]$. It is observed from (14) that only the first term depends on the variable \mathbf{U} . Then letting $(\tilde{\mathbf{U}}, \tilde{\mathbf{U}}_\mathbf{X})$ denote an estimate of $(\mathbf{U}, \mathbf{U}_\mathbf{X})$, the ML-estimation problem can be formulated as

$$\begin{aligned} \min_{\tilde{\mathbf{U}}} \quad & \text{Tr}(\Phi_t^\top \tilde{\mathbf{U}}_\mathbf{X} \Lambda^{-1} \tilde{\mathbf{U}}_\mathbf{X}^\top \Phi_t) \\ \text{s.t.} \quad & \tilde{\mathbf{U}}_\mathbf{X}^\top \tilde{\mathbf{U}}_\mathbf{X} = \tilde{\mathbf{U}}_\mathbf{X} \tilde{\mathbf{U}}_\mathbf{X}^\top = \mathbf{I}, \\ & \tilde{\mathbf{U}}_\mathbf{X} = [\tilde{\mathbf{U}}, \tilde{\mathbf{U}}^\perp]. \end{aligned} \quad (15)$$

Despite the non-convex orthogonality constraints, the problem in (15) can be solved optimally in closed form, as follows. First, define the eigenvalue decomposition $\Phi_t \Phi_t^\top = \mathbf{Q} \mathbf{\Gamma} \mathbf{Q}^\top$ with $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_I]$ and $\mathbf{\Gamma} = \text{diag}(\gamma_1, \dots, \gamma_I)$ with eigenvalues arranged in a descending order. Then, given $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_I)$ and $\mathbf{U}_\mathbf{X} = [\mathbf{u}_1, \dots, \mathbf{u}_I]$, the objective function of (15) can be rewritten as

$$\text{Tr}(\Lambda^{-1} \mathbf{U}_\mathbf{X}^\top \Phi_t \Phi_t^\top \mathbf{U}_\mathbf{X}) = \sum_{i=1}^I \sum_{j=1}^I \lambda_j^{-1} \gamma_i (\mathbf{q}_i^\top \mathbf{u}_j)^2.$$

Next, define $x_{ij} = \mathbf{q}_i^\top \mathbf{u}_j$ and rewrite the constraints in (15) as $\sum_{i=1}^I x_{ij}^2 = \mathbf{u}_j^\top \mathbf{Q} \mathbf{Q}^\top \mathbf{u}_j = 1$ and $\sum_{j=1}^I x_{ij}^2 = \mathbf{q}_i^\top \mathbf{U} \mathbf{U}^\top \mathbf{q}_i = 1$. Without loss of optimality, such constraints can be further relaxed as $\sum_{i=1}^I x_{ij}^2 \geq 1$ and $\sum_{j=1}^I x_{ij}^2 \geq 1$. This allows the problem in (15) to be reformulated as a

Algorithm 1: On-the-Fly Sub-space Detection for FlyCom² Based DTD

Initialize: Received in-phase matrix symbols $\{\tilde{\mathbf{Y}}_\ell\}_{\ell \leq t}$ in slot t ;

Perform:

- 1: *Aggregation:* Aggregate all received matrix symbol $\{\tilde{\mathbf{Y}}_\ell\}_{\ell \leq t}$ into $\hat{\mathbf{Y}}_t = [\tilde{\mathbf{Y}}_1, \dots, \tilde{\mathbf{Y}}_t]$;
- 2: *Whitening:* Compute the whitened version, Φ_t , of the aggregated matrix $\hat{\mathbf{Y}}_t$ by (13);
- 3: *Sub-space extraction:* Compute the first r eigenvectors of $\Phi_t \Phi_t^\top$ and aggregate them into $\tilde{\mathbf{U}}$.

Output: $\tilde{\mathbf{U}}$ used as the principal eigenspace of the unfolding matrix \mathbf{X} .

convex problem:

$$\begin{aligned}
 \min_{\{x_{ij}\}} \quad & \sum_{i=1}^I \sum_{j=1}^I \lambda_j^{-1} \gamma_i x_{ij}^2 \\
 \text{s.t.} \quad & \sum_{l=1}^I x_{il}^2 \geq 1, \sum_{l=1}^I x_{lj}^2 \geq 1, \forall i, j.
 \end{aligned} \tag{16}$$

Since $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_I$, the objective of (16) subject to the constraints is lower bounded as

$$\sum_{i=1}^I \sum_{j=1}^I \lambda_j^{-1} \gamma_i x_{ij}^2 \geq \sum_{i=1}^I \lambda_i^{-1} \gamma_i. \tag{17}$$

The lower bound can be achieved by letting $x_{ii} = 1, \forall i$ and $x_{ij} = 0, \forall i \neq j$. The optimal solution for (16) follows as shown below.

Proposition 1. Based on the ML criterion, in slot t , the optimal on-the-fly estimate of the r -dimensional principal components of the unfolding matrix, \mathbf{X} , is denoted as $\tilde{\mathbf{U}}^*$ and given as

$$\tilde{\mathbf{U}}^* = [\mathbf{q}_1, \dots, \mathbf{q}_r] = \mathcal{S}_r(\Phi_t \Phi_t^\top), \tag{18}$$

where Φ_t is the effective observation in slot t as given in (13) and we recall $\mathcal{S}_r(\cdot)$ to yield the r -dimensional principal eigenspace of its argument.

Remark 2 (Minimum Number of FlyCom² Operations). For the result in (18) to hold, the dimensions of the current effective observations Φ_t should be larger than those of \mathbf{U} , i.e. $tM \geq r$. This implies that the FlyCom² should run at least $t \geq r/M$ rounds to enable the estimation of an r -dimensional principal eigenspace of the tensor.

C. DTD Error Analysis

Based on the optimal sub-space detection designed in the preceding sub-section, we mathematically quantify the key feature of FlyCom² that the DTD error gracefully decreases with communication rounds. The existing error analysis for random sketching does not target distributed implementation and hence requires no communication links [20], [21]. The new challenge for the current analysis arises from the need to account for the distortion increased by the MIMO AirComp transmission. In what follows, we derive deterministic and probabilistic bounds on the DTD error defined in (5).

1) *Deterministic Error Bound:* As the unfolding matrix comprises r principal components, its singular values can be represented as $\Sigma_{\mathbf{X}} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_I)$ with $\sigma_1 = \dots = \sigma_r \gg \sigma_{r+1} \geq \dots \geq \sigma_I$, where we assume the same principal singular values following the literature (see, e.g. [20]).

Lemma 2. Consider the DTD of the unfolding matrix \mathbf{X} in tensor decomposition that has an r -dimensional principal eigenspace $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_r]$ and the singular values $\Sigma_{\mathbf{X}}$. The estimation of \mathbf{U} as in (18) yields the DTD error given as

$$d(\tilde{\mathbf{U}}, \mathbf{X}) = \sum_{i=1}^r \sum_{j \geq r+1} (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 + \sum_{i \geq r+1} \sigma_i^2. \quad (19)$$

Proof. See Appendix B. □

On the right hand side, the first term, $\sum_{i=1}^r \sum_{j \geq r+1} (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2$, represents the error due to random sketching; the second term $\sum_{i \geq r+1} \sigma_i^2$ represents the residual error due to non-zero non-principal components of \mathbf{X} .

Next, we make an attempt to characterize the behavior of each error term, $(\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2$. Let $\tilde{\mathbf{u}}_i$ and \mathbf{u}_j denote the i -th and j -th ($i \leq r < j$) eigenvectors of the sample covariance matrix $\frac{1}{tM} \Phi_t \Phi_t^\top$ and the covariance matrix $\mathbf{U}_{\mathbf{X}} \Lambda \mathbf{U}_{\mathbf{X}}^\top$, respectively. The error in (19) is caused by the perturbation $\Delta = \frac{1}{tM} \Phi_t \Phi_t^\top - \mathbf{U}_{\mathbf{X}} \Lambda \mathbf{U}_{\mathbf{X}}^\top$. Using this fact allows us to obtain the following desired result.

Lemma 3. Consider a fixed realization \mathbf{W} in the DRM, Φ_t , in (13) and the error term, $(\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2$, in Lemma 2 is upper bounded as

$$(\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 \leq \max \{4, \delta_{ij}^2\} \frac{\|\Delta \mathbf{u}_j\|_2^2}{\sigma_i^2 - \sigma_j^2}, \quad i \leq r < j,$$

where $\delta_{ij} \triangleq \frac{\min\{2|\tilde{\lambda}_i - \lambda_i|, (\sigma_i^2 - \sigma_j^2)\}}{|\tilde{\lambda}_i - \lambda_j|}$ with λ_i and $\tilde{\lambda}_i$ being the i -th eigenvalues of $\mathbf{U}_\mathbf{X} \mathbf{\Lambda} \mathbf{U}_\mathbf{X}^\top$ and $\frac{1}{tM} \mathbf{\Phi}_t \mathbf{\Phi}_t^\top$, respectively.

Proof. See Appendix C. \square

The upper bound in Lemma 3 suggests two scaling regions of the DTD error, namely $\delta_{ij} \geq 2$ and $\delta_{ij} < 2$. Invoking the well-known Weyl's theorem (see, e.g. [25]), the norm of the perturbation $\mathbf{\Delta}$ and hence the value of δ_{ij} reduce as FlyCom² progresses in time. This is aligned with the result in Fig. 3(a) where the average value of $\{\delta_{ij}\}$ is observed to decrease with increasing communication time t . To simplify the analysis, we focus on the case of $\delta_{ij} \leq 2$, $\forall i \leq r < j$, by assuming sufficiently large t . In this case, the upper bound in Lemma 3 simplifies to

$$(\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 \leq \frac{4 \|\mathbf{\Delta} \mathbf{u}_j\|_2^2}{\sigma_i^2 - \sigma_j^2}, \quad i \leq r < j. \quad (20)$$

Next, based on Lemma 2 and (20), the desired deterministic error bound is derived as follows.

Theorem 1 (Expected Error Bound). Given the receive beamformers $\{\mathbf{A}_\ell\}_{\ell \leq t}$ of FlyCom²-based DTD and $\delta_{ij} \leq 2$, $\forall i \leq r < j$, the expected error can be bounded as

$$\mathbb{E}[d(\tilde{\mathbf{U}}, \mathbf{X})] \leq \frac{4}{tM} \sum_{i=1}^r \sum_{j \geq r+1} \frac{\lambda_j^2 + \lambda_j \text{Tr}(\mathbf{\Lambda})}{\sigma_i^2 - \sigma_j^2} + \sum_{i \geq r+1} \sigma_i^2,$$

where δ_{ij} and λ_j follow those in Lemma 3.

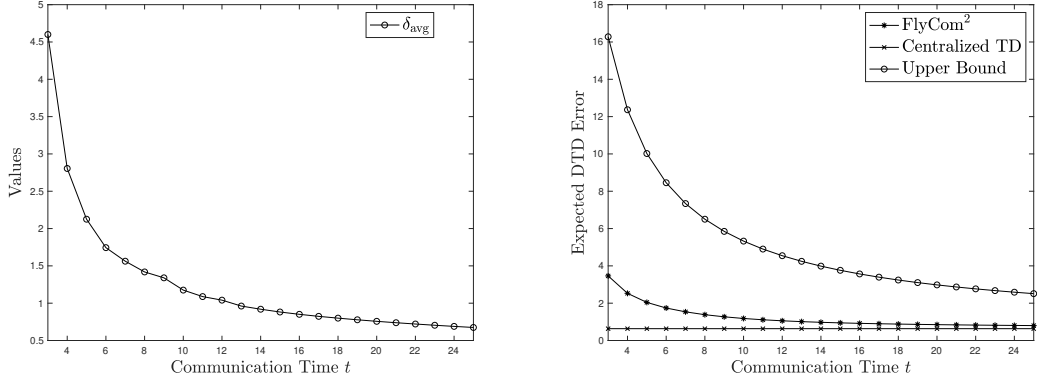
Proof. See Appendix D. \square

The error bound in the Theorem 1 is compared numerically with the exact error and that of centralized tensor decomposition in Fig. 3(b). One can observe the bound to capture the trend of decreasing DTD error as t progresses. In particular, it shows that under a small perturbation,

$$\mathbb{E}[d(\tilde{\mathbf{U}}, \mathbf{X})] \propto \frac{1}{tM}.$$

2) *Probabilistic Error Bound:* We derive in the sequel a probabilistic bound on the DTD error using the method of *concentration of measure*. A relevant useful result is given below.

Lemma 4 (McDiarmid's Inequality [26]). Let g be a positive function on independent variables



(a) Average value of $\{\delta_{ij}\}$ versus communication time (b) Expected DTD error versus communication time

Fig. 3. Validation of theoretical results under the settings of $r = 12$, $I = 100$, $\Sigma_{\mathbf{X}} = \text{diag}(1, \dots, 1, \frac{1}{2}, \frac{1}{3}, \dots, \frac{1}{88})$, and $\mathbf{A}_t \mathbf{A}_t^\top = \frac{1}{10\sigma} \mathbf{I}$

$\{W_m\}$ satisfying the bounded difference property:

$$\sup_{\{W_M\}_{M \neq m}, W_m, W_M} |g(\{W_M\}_{M \neq m}, W_m) - g(\{W_M\}_{M \neq m}, W_M)| \leq c_m, \quad \forall m,$$

with constants $\{c_m\}$ and W_M being i.i.d. as W_m . Then, for any $\epsilon > 0$,

$$\Pr[g(\{W_m\}) - \mathbb{E}[g(\{W_m\})] \geq \epsilon] \leq \exp\left(-\frac{2\epsilon^2}{\sum_m c_m^2}\right).$$

Using Lemma 4, the desired result is obtained as shown below.

Theorem 2 (Probabilistic Error Bound). Given receive beamformers $\{\mathbf{A}_\ell\}_{\ell \leq t}$ and $\delta_{ij} \leq 2$, $\forall i \leq r < j$, for any $\epsilon \geq 0$, the error of the FlyCom²-based DTD can be upper bounded as

$$d(\tilde{\mathbf{U}}, \mathbf{X}) \leq \frac{4(1+\epsilon)}{tM} \sum_{i=1}^r \sum_{j \geq r+1} \frac{\lambda_j^2 + \lambda_j \text{Tr}(\mathbf{\Lambda})}{\sigma_i^2 - \sigma_j^2} + \sum_{i \geq r+1} \sigma_i^2,$$

with the probability of at least $\left[1 - e^{-\frac{\epsilon^2}{2\kappa^8}}\right] \text{erf}\left(\frac{\kappa}{\sqrt{2}}\right)^{tM(I-r)}$, where $\text{erf}(\cdot)$ denotes the error function defined as $\text{erf}(y) = \frac{2}{\sqrt{\pi}} \int_0^y e^{-x^2} dx$ and $\kappa \geq 1$.

Proof. See Appendix E. □

The upper bound on the DTD error in Theorem 2 holds *almost surely* if the constant ϵ is

sufficiently large. Comparing Theorem 1 and Theorem 2, one can make the important observation that as the communication time (t) progresses, both the DTD error and its expectation vanish at the same rate of

$$\text{Error} \propto \frac{1}{tM} \sum_{i=1}^r \sum_{j \geq r+1} \frac{\lambda_j^2 + \lambda_j \text{Tr}(\Lambda)}{\sigma_i^2 - \sigma_j^2}. \quad (21)$$

Another observation is that non-principal components of the tensor contribute to the DTD error but the effect is negligible when the eigen-gap is large.

V. OPTIMAL SKETCH SELECTION FOR FLYCOM²

Discarding aggregated sketches that have been transmitted under unfavourable channel conditions can improve the FlyCom² performance. This motivates us to design a sketch-selection scheme in this section.

A. Threshold Based Sketch Selection

First, we follow the approach in [23] to design the receive beamforming, $\{\mathbf{A}_t\}$, for MIMO AirComp. To this end, we decompose \mathbf{A}_t as $\mathbf{A}_t = \eta_t \mathbf{U}_{\mathbf{A}_t}$, where the positive scalar η_t is called a denoising factor and $\mathbf{U}_{\mathbf{A}_t}$ is an $M \times N_r$ unitary matrix. Following similar steps as in [23], we can show that to minimize the DTD error bounds in Theorem 1 and 2, the beamformer component should be aligned with the channels of devices as

$$\mathbf{U}_{\mathbf{A}_t}^\top = \mathcal{S}_M \left(\frac{1}{K} \sum_k \lambda_{\mathbf{H}_{t,k}} \mathbf{U}_{\mathbf{H}_{t,k}} \mathbf{U}_{\mathbf{H}_{t,k}}^\top \right), \quad (22)$$

where $\lambda_{\mathbf{H}_{t,k}}$ and $\mathbf{U}_{\mathbf{H}_{t,k}}$ denote the N_t -th eigenvalue and the first N_t eigenvectors of $\mathbf{H}_{t,k} \mathbf{H}_{t,k}^\top$, respectively. Furthermore, the denoising factor η_t should cope with the weakest channel by being

$$\eta_t = \max_k \frac{\text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k)}{IP} \text{Tr} \left((\mathbf{U}_{\mathbf{A}_t} \mathbf{H}_{t,k} \mathbf{H}_{t,k}^H \mathbf{U}_{\mathbf{A}_t}^H)^{-1} \right). \quad (23)$$

It follows from (22) and (23) that $\text{Tr}(\mathbf{A}_t^H \mathbf{A}_t) = \eta_t M$ and λ_j in DTD error bounds in Theorem 1 and 2 can be expressed as

$$\lambda_j = \sigma_j^2 + \frac{\sigma^2}{2t} \sum_{\ell \leq t} \eta_\ell, \quad (24)$$

which shows that the error relies on only the denoising factor up to the current time slot. The result also suggests that it is preferable to select from the received sketches $\{\tilde{\mathbf{Y}}_n\}_{\ell \leq t}$ those

associated with small η_ℓ that reflects a favorable channel condition. Naturally, we can derive a threshold based selection scheme as follows:

$$\boxed{\tilde{\mathbf{Y}}_\ell \text{ is selected if } \eta_\ell \leq \eta_{\text{th}}, \forall \ell \leq t,} \quad (25)$$

where the threshold η_{th} is optimized in the sequel.

B. Threshold Optimization

The threshold, η_{th} , in (25), needs to be optimized to minimize the error in (21). Solving the problem is hindered by that the singular values $\{\sigma_j\}$ are not available at the server in advance. We tackle this problem by designing a practical optimization scheme. To this end, we resort to using an upper bound on the DTD error as shown below.

Lemma 5. Let \tilde{M} denote the number of aggregated sketches selected from $\{\tilde{\mathbf{Y}}_\ell\}_{\ell \leq t}$ based on (25) with the threshold η_{th} , the DTD error in (21) satisfies

$$\frac{1}{\tilde{M}} \sum_{i=1}^r \sum_{j \geq r+1} \frac{\lambda_j^2 + \lambda_j \text{Tr}(\mathbf{\Lambda})}{\sigma_i^2 - \sigma_j^2} \leq \frac{c}{\tilde{M}} \left[1 + \frac{r\sigma^2\eta_{\text{th}}}{2 \sum_k \text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k)} \right]^2,$$

where c is a constant.

Proof. See Appendix F. □

Lemma 5 suggests that a sub-optimal threshold can be obtained by minimizing the error upper bound. Let S denote a set and $|S|$ its cardinality. Then, the threshold-optimization problem can be formulated as

$$\begin{aligned} \min_{\eta_{\text{th}}} \quad & \frac{1}{|S|} \left[1 + \frac{r\sigma^2\eta_{\text{th}}}{2 \sum_k \text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k)} \right]^2 \\ \text{s.t.} \quad & S = \{\eta_\ell | \eta_\ell \leq \eta_{\text{th}}, \ell \leq t\}, \end{aligned} \quad (26)$$

where η_ℓ follows the definition in (23). One can observe that the objective function of (26) is a monotonically increasing function w.r.t. the variable η_{th} if \tilde{M} is fixed. Such piecewise monotonicity of the objective function (26) renders a linear-search solution method (e.g. bisection search) infeasible, but allows the optimal solution, η_{th}^* , to be restricted into a finite set, say $\eta_{\text{th}}^* \in \{\eta_1, \eta_2, \dots, \eta_t\}$. Then, finding η_{th}^* is simple by exhausted enumeration as follows. Let \tilde{M}_ℓ

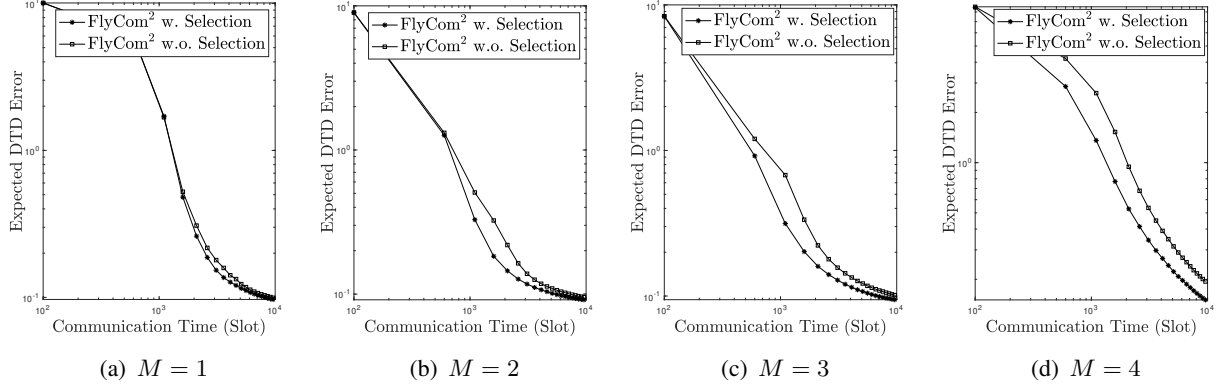


Fig. 4. Error-performance comparison between FlyCom² with and without sketch selection, SNR $\gamma = 10\text{dB}$.

denote the number of selected sketches corresponding to the threshold fixed as $\eta_{\text{th}} = \eta_\ell$. Define

$$\ell^* = \arg \min_{\ell \leq t} \frac{1}{\tilde{M}_\ell} \left[1 + \frac{r\sigma^2\eta_\ell}{2 \sum_k \text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k)} \right]^2.$$

The optimal threshold solving the problem in (26) is $\eta_{\text{th}} = \eta_{\ell^*}$. Two remarks are offered as follows. First, the above research for the optimal threshold has complexity linearly proportional to t , the population of accumulated sketches at the server. Second, the implementation of the optimization at the server requires feedback of a scalar from each device, namely $\text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k)$ from device k .

VI. EXPERIMENTAL RESULTS

A. Experimental Settings

First, the MIMO AirComp system is configured to have the following settings. There are $K = 20$ edge devices connected to the server. The array sizes at each device and the server are set as $N_t = 4$ and $N_r = 16$, respectively. The Rayleigh channel with shadow fading is adopted, in which each MIMO channel is given as $\mathbf{H}_{t,k} = \sqrt{\beta_{t,k}} \hat{\mathbf{H}}_{t,k}$ with $\beta_{t,k}$ following a Gamma distribution $\Gamma(1.2, 0.83)$ (see, e.g. [27]) and $\hat{\mathbf{H}}_{t,k}$ comprising i.i.d. $\mathcal{CN}(0, 1)$ entries. Different channels are independent. Second, we use a synthetic data model following the DTD literature (see, e.g. [20]). Considering the computation of the n -th factor matrix, the unfolding matrix of the data tensor has the size of $I_n \times (\prod_{j=1, j \neq n}^N I_j) = 100 \times 1500$, and its columns are uniformly distributed over devices. Under such settings, each local sketch has the length of $I_n = 100$ that is smaller than a single channel coherence block (see, e.g. [28], for justification). To demonstrate the performance

of the proposed FlyCom² for data with a range of parameterized spectral distribution, the singular values of the unfolding matrix are set to decay with a polynomial rate:

$$\Sigma_{\mathbf{X}} = \text{diag} \left(1, \dots, 1, \frac{1}{2^\xi}, \frac{1}{3^\xi}, \dots, \frac{1}{(I_n - r)^\xi} \right),$$

where the first $r = 12$ principal singular values are fixed as 1 and $\xi > 0$ controls the decay rate of residual values. Furthermore, the left and right eigenspaces of the unfolding matrix are generated as those of random matrices with i.i.d. $\mathcal{N}(0, 1)$ entries [20].

Third, we consider two benchmarking schemes that are variants of SVD-based DTD.

- **Centroid SVD-DTD:** Devices compute local principal eigenspaces $\{\hat{\mathbf{U}}_k\}$ of their on-device data samples by using SVD and the server then aggregates these local results as $\mathbf{P} = \frac{1}{K} \sum_k \hat{\mathbf{U}}_k \hat{\mathbf{U}}_k^\top$. The principal eigenspace of \mathbf{P} represents the centroid of all local estimates $\{\hat{\mathbf{U}}_k\}$ on the Grassmannian manifold and is extracted to form a global estimate of the ground truth [10], [11].
- **Alignment SVD-DTD:** The scheme follows a similar procedure as above except for aggregating local results, $\{\hat{\mathbf{U}}_k\}$, as $\mathbf{P} = \frac{1}{K} \sum_k \hat{\mathbf{U}}_k \mathbf{J}_k$, where the orthogonal matrices $\{\mathbf{J}_k\}$ are alignment matrices that are to be optimized by using past global estimates to improve the system performance [12].

The aggregation operations in both benchmark schemes are implemented using MIMO AirComp [14], [23] as FlyCom² for fair comparison.

B. Performance Gain of Sketch Selection

In Fig. 4, we compare the error performance of FlyCom² between the cases with and without sketch selection. The communication time is measured by the total number of symbol slots used in uploading local sketches, namely tI_n , where I_n is the number of rows of local sketches. We vary the dimension of the receive beamformer, M , from 1 to 4 to achieve different tradeoffs between channel diversity and multiplexing. It is observed from Fig. 4 that the proposed selection scheme helps reduce the expected DTD error for different M . The gain emerges when the communication time exceeds M -dependent threshold (e.g. 6000 time slots for $M = 1$) as the total number of available sketches becomes sufficiently large. Another observation from Fig. 4 is that the DTD error without selection decreases at an approximately linear rate w.r.t. the communication time, which is aligned with our conclusion in (21). In the sequel, FlyCom² is assumed to have sketch selection with M fixed as 2.

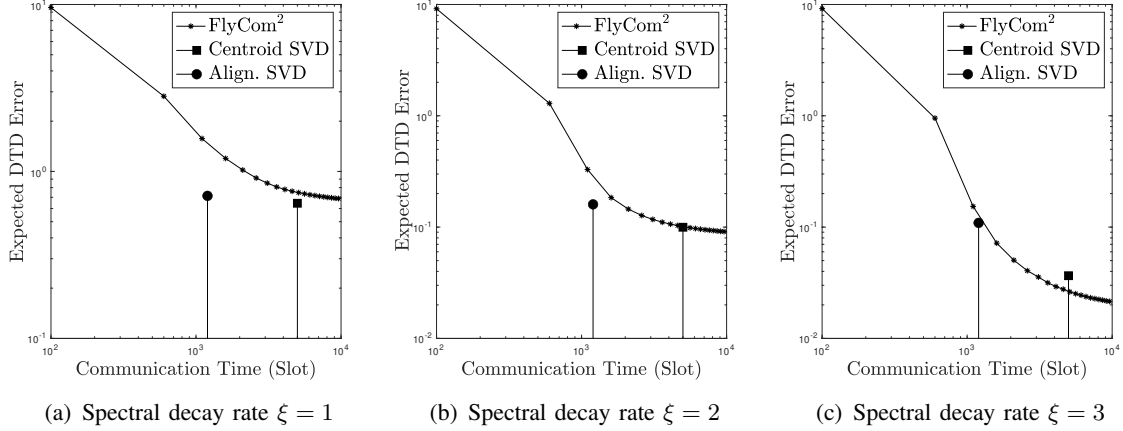


Fig. 5. FlyCom² versus Benchmark schemes, SNR $\gamma = 10$ dB.

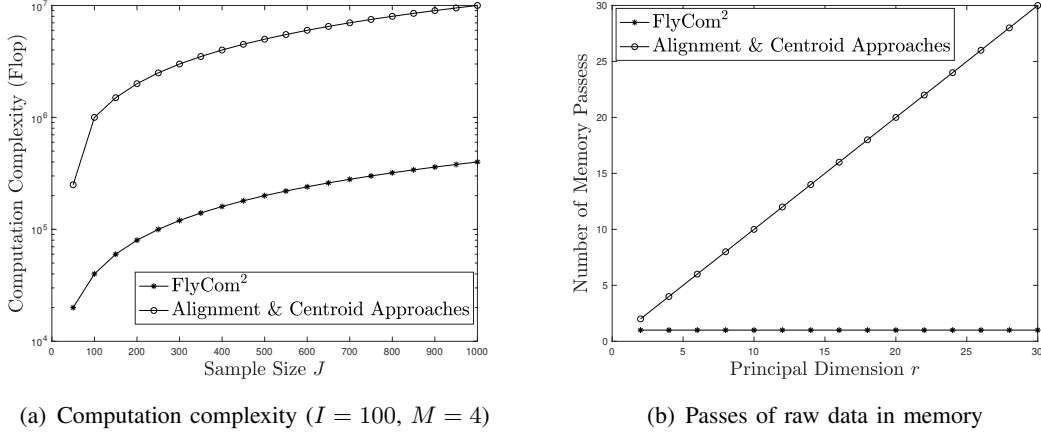


Fig. 6. Computation-cost comparison between FlyCom² based DTD and benchmarking schemes.

C. Error Performance of FlyCom²

While FlyCom² requires much simpler on-device computation than benchmarking schemes (see Section VI-D), we demonstrate in Fig. 5 that it can achieve comparable or even better error performance than the latter. Fig. 5 shows the expected DTD error versus the communication time. The performance of the benchmark schemes with one-shot computation and communication appears as single points in the figure. The results in Fig. 5 show that FlyCom²-based DTD achieves comparable decomposition accuracy as the benchmark schemes with progressing time. Furthermore, its performance is improved by increasing the decay rate (ξ) of the singular values, which validates our conclusion that large eigen-gaps help distinguish principal from non-principal

eigenvectors during random sketching. For instance, for $\xi = 2$, the proposed scheme approaches the centroid and alignment based SVD-DTD in performance for communication time larger than 2600 and 4000 symbol slots, respectively. As ξ increases to 3, the former achieves the same error performance as the alignment-based method while outperforming the centroid based method. Furthermore, one can observe from Fig. 5 that the proposed on-the-fly framework realizes a flexible trade-off between the decomposition accuracy and communication time, which is the distinctive feature of the design.

D. Device Computation Costs of FlyCom²

In Fig. 6, we compare two kinds of computational costs at devices, namely complexity and memory passes, between FlyCom² and benchmark schemes. The complexity refers to the flop count of computation, and the memory passes are equal to the number of memory visits for reading data entries. The computational advantage of FlyCom² is demonstrated by comparing the cost of matrix-vector multiplication in random sketching with that of deterministic SVD used in the one-shot benchmarking schemes. Specifically, given $I \times J$ local unfolding matrices, deterministic SVD has the complexity proportional to $\min\{I, J\}^2 \times \max\{I, J\}$ [18]; based on matrix multiplication, the complexity of FlyCom² to yield an M -dimensional sketch at each time slot is IJM . For the schemes in comparison, their curves of computation complexity versus sample size are plotted in Fig. 6(a). One can observe that the proposed FlyCom² dramatically reduces devices' complexity by more than an order of magnitude. On the other hand, Fig. 6(b) displays the curves of the number of memory passes versus the principal dimensionality, r . The proposed design keeps a constant memory pass for matrix multiplication, as opposed to that of SVD which increases linearly with the principal dimensionality. For example, the number of memory passes is reduced using FlyCom² by 30 times for $r = 30$.

VII. CONCLUSION

We have proposed the FlyCom² framework, that supports the progressive computation of DTD in mobile networks. Through the use of random sketching techniques at devices, the traditional one-shot high-dimensional mobile communication and computation is reduced to low-dimensional operations spread over multiple time slots. Thereby, the resource constraints of devices are overcome. Furthermore, FlyCom² obtains its distinctive feature of progressive improvement of DTD accuracy with increasing communication time, providing robustness against

link disruptions. To develop the FlyCom² based DTD framework, we have designed an on-the-fly sub-space estimator and a sketch-selection scheme to ensure close-to-optimal system performance.

Beyond DTD, high-dimensional communication and computation pose a general challenge for machine learning and data analytics in wireless networks. We expect that FlyCom² can be further developed into a broad approach for efficient deployment of relevant algorithms such as federated learning and distributed optimization. For the current FlyCom² targeting DTD, its extension to accommodate other wireless techniques such as broadband transmission and radio resource management is also a direction worth pursuing.

APPENDIX

A. Proof of Lemma 1

In (9), both the \mathbf{F}_t and $\tilde{\mathbf{Z}}_t$ have i.i.d. zero-mean Gaussian entries, thereby enforcing the observation $\tilde{\mathbf{Y}}_t$ to be a Gaussian matrix. This conclusion holds for all observations and thus the aggregation $\hat{\mathbf{Y}}_t$ is a Gaussian matrix and can be decomposed into $\mathbf{C}^{\frac{1}{2}}\mathbf{W}\mathbf{D}^{\frac{1}{2}}$. Therein, \mathbf{W} has i.i.d. $\mathcal{N}(0,1)$ entries. The covariance matrices \mathbf{C} and \mathbf{D} are computed as follows. First, there is $\mathbb{E}[\hat{\mathbf{Y}}_t\hat{\mathbf{Y}}_t^\top] = \mathbb{E}[\mathbf{C}^{\frac{1}{2}}\mathbf{W}\mathbf{D}\mathbf{W}^\top\mathbf{C}^{\frac{1}{2}}]$, where the right hand side of the equation equals to $\mathbf{C}\text{Tr}(\mathbf{D})$ while the left hand side is given as

$$\begin{aligned}\mathbb{E}[\hat{\mathbf{Y}}_t\hat{\mathbf{Y}}_t^\top] &= \sum_{\ell \leq t} \mathbb{E}[\tilde{\mathbf{Y}}_t\tilde{\mathbf{Y}}_t^\top] \\ &= tM\mathbf{X}\mathbf{X}^\top + \frac{1}{2}\sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell)\mathbf{I}_I.\end{aligned}$$

On the other hand, using $\mathbb{E}[\hat{\mathbf{Y}}_t^\top\hat{\mathbf{Y}}_t] = \mathbb{E}[\mathbf{D}^{\frac{1}{2}}\mathbf{W}^\top\mathbf{C}\mathbf{W}\mathbf{D}^{\frac{1}{2}}] = \mathbf{D}\text{Tr}(\mathbf{C})$, we have

$$\begin{aligned}\mathbf{D}\text{Tr}(\mathbf{C}) &= \mathbb{E}\left[\left[\tilde{\mathbf{Y}}_1, \tilde{\mathbf{Y}}_2, \dots, \tilde{\mathbf{Y}}_t\right]^\top \left[\tilde{\mathbf{Y}}_1, \tilde{\mathbf{Y}}_2, \dots, \tilde{\mathbf{Y}}_t\right]\right] \\ &= \text{diag}\left(\mathbb{E}[\tilde{\mathbf{Y}}_1^\top\tilde{\mathbf{Y}}_1], \dots, \mathbb{E}[\tilde{\mathbf{Y}}_t^\top\tilde{\mathbf{Y}}_t]\right),\end{aligned}$$

where an arbitrary diagonal block, say $\mathbb{E}[\tilde{\mathbf{Y}}_\ell^\top\tilde{\mathbf{Y}}_\ell]$, $\forall \ell \leq t$, can be expressed as

$$\begin{aligned}\mathbb{E}[\tilde{\mathbf{Y}}_\ell^\top\tilde{\mathbf{Y}}_\ell] &= \mathbb{E}[\mathbf{F}^\top\mathbf{X}^\top\mathbf{X}\mathbf{F}] + \mathbb{E}[\tilde{\mathbf{Z}}_t\tilde{\mathbf{Z}}_t^\top] \\ &= \text{Tr}(\mathbf{X}^\top\mathbf{X})\mathbf{I}_M + \frac{1}{2}I\sigma^2\mathbf{A}_\ell\mathbf{A}_\ell^H.\end{aligned}$$

Concluding the above results yields the covariance matrices as $\mathbf{C} = \mathbf{X}\mathbf{X}^\top + \frac{1}{2tM}\sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell) \mathbf{I}_I$ and $\mathbf{D} = \frac{tM \text{Tr}(\mathbf{X}^\top \mathbf{X}) \mathbf{I}_{tM} + \frac{1}{2} I t M \sigma^2 \text{diag}(\mathbf{A}_1 \mathbf{A}_1^H, \dots, \mathbf{A}_t \mathbf{A}_t^H)}{tM \text{Tr}(\mathbf{X}^\top \mathbf{X}) + \frac{1}{2} I \sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell)}$, respectively. This completes the proof.

B. Proof of Lemma 2

First, rewrite the error as

$$\begin{aligned} & \|(\mathbf{I}_I - \tilde{\mathbf{U}}\tilde{\mathbf{U}}^\top)\mathbf{X}\|_F^2 \\ &= \sum_{j=1}^I \sigma_j^2 - \sum_{i=1}^r \sum_{j \neq i} \sigma_j^2 \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 - \sum_{i=1}^r \sigma_i^2 \langle \tilde{\mathbf{u}}_i, \mathbf{u}_i \rangle^2 \\ &= \sum_{j \geq r+1} \sigma_j^2 - \sum_{i=1}^r \sum_{j \neq i} (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2, \end{aligned}$$

where the last step is due to $\langle \tilde{\mathbf{u}}_i, \mathbf{u}_i \rangle^2 = 1 - \sum_{j \neq i} \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2$.

Then, under the assumption of $\sigma_i = \sigma_j$, $\forall i, j \leq r$, the second term on the right side of the above equation can be rewritten as

$$\begin{aligned} & \sum_{i=1}^r \sum_{j \neq i} (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 \\ &= \sum_{i=1}^r \sum_{j \geq r+1}^I (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 \\ & \quad + \underbrace{\sum_{i=1}^r \sum_{j=1, j \neq i}^r (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2}_{=0} \\ &= \sum_{i=1}^r \sum_{j \geq r+1}^I (\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2. \end{aligned}$$

Putting the results above together, the conclusion in Lemma 2 follows.

C. Proof of Lemma 3

According to Remark 1, we have $\lambda_i > \lambda_j$, $\forall i \leq r < j$. Then, using the perturbation theory [29, Theorem 3.1 & Theorem 3.2], the $\langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle$ can be upper bounded as

$$\langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle \leq \min \left\{ \max \left\{ 2, 2 \frac{|\tilde{\lambda}_i - \lambda_i|}{|\tilde{\lambda}_i - \lambda_j|} \right\}, \frac{\lambda_i - \lambda_j}{|\tilde{\lambda}_i - \lambda_j|} \right\} \frac{\|\Delta \mathbf{u}_j\|_2}{\lambda_i - \lambda_j},$$

where λ_i and $\tilde{\lambda}_i$ denote the i -th eigenvalues of $\mathbf{\Lambda}$ and $\frac{1}{tM}\mathbf{\Phi}_t\mathbf{\Phi}_t^\top$, respectively. Based on $\min\{\max\{a, b\}, c\} = \max\{\min\{a, c\}, \min\{b, c\}\} \leq \max\{a, \min\{b, c\}\}$, the upper bound can be further written as

$$\begin{aligned}\langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle &\leq \max \left\{ 2, \min \left\{ \frac{2|\tilde{\lambda}_i - \lambda_i|}{|\tilde{\lambda}_i - \lambda_j|}, \frac{\lambda_i - \lambda_j}{|\tilde{\lambda}_i - \lambda_j|} \right\} \right\} \frac{\|\mathbf{\Delta u}_j\|_2}{\lambda_i - \lambda_j} \\ &= \max \left\{ 2, \frac{2 \min\{|\tilde{\lambda}_i - \lambda_i|, \lambda_i - \lambda_j\}}{|\tilde{\lambda}_i - \lambda_j|} \right\} \frac{\|\mathbf{\Delta u}_j\|_2}{\lambda_i - \lambda_j}.\end{aligned}$$

Recall that $\lambda_i = \sigma_i^2 + \sigma^2 \sum_{\ell \leq t} \text{Tr}(\mathbf{A}_\ell^H \mathbf{A}_\ell) / 2tM$, we have $\lambda_i - \lambda_j = \sigma_i^2 - \sigma_j^2$, which yields

$$\begin{aligned}(\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2 &= \max \left\{ 4, \frac{\min\{4|\tilde{\lambda}_i - \lambda_i|^2, (\sigma_i^2 - \sigma_j^2)^2\}}{|\tilde{\lambda}_i - \lambda_j|^2} \right\} \frac{\|\mathbf{\Delta u}_j\|_2^2}{\sigma_i^2 - \sigma_j^2},\end{aligned}$$

which completes the proof.

D. Proof of Theorem 1

First, the square vector norm, $\|\mathbf{\Delta u}_j\|_2^2$, can be rewritten as

$$\begin{aligned}\|\mathbf{\Delta u}_j\|_2^2 &= \mathbf{u}_j^\top \left(\frac{1}{tM} \mathbf{\Phi}_t \mathbf{\Phi}_t^\top - \mathbf{U}_\mathbf{X} \mathbf{\Lambda} \mathbf{U}_\mathbf{X}^\top \right) \mathbf{u}_j \\ &= \lambda_j^2 - \frac{2\lambda_j^2}{tM} \mathbf{e}_j \mathbf{W} \mathbf{W}^\top \mathbf{e}_j^\top \\ &\quad + \frac{\lambda_j}{(tM)^2} \mathbf{e}_j \mathbf{W} \mathbf{W}^\top \mathbf{\Lambda} \mathbf{W} \mathbf{W}^\top \mathbf{e}_j^\top,\end{aligned}$$

where we define $\mathbf{e}_j = [0, \dots, 0, 1, 0, \dots, 0]$ with the j -th element being 1 and other elements being 0. Since \mathbf{W} has i.i.d. $\mathcal{N}(0, 1)$ entries, the expectation of the upper bound of $(\sigma_i^2 - \sigma_j^2) \langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2$ can be expressed as

$$\begin{aligned}(\sigma_i^2 - \sigma_j^2) \mathbb{E}[\langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2] &\leq \frac{4}{\sigma_i^2 - \sigma_j^2} \mathbb{E}[\|\mathbf{\Delta u}_j\|_2^2] \\ &= \frac{4}{\sigma_i^2 - \sigma_j^2} \left[\frac{\lambda_j}{(tM)^2} \mathbf{e}_j \mathbb{E}[\mathbf{W} \mathbf{W}^\top \mathbf{\Lambda} \mathbf{W} \mathbf{W}^\top] \mathbf{e}_j^\top - \lambda_j^2 \right].\end{aligned}$$

The result of $\mathbf{e}_j \mathbb{E}[\mathbf{W} \mathbf{W}^\top \boldsymbol{\Lambda} \mathbf{W} \mathbf{W}^\top] \mathbf{e}_j^\top$ can be derived by representing $\mathbf{W} = [\mathbf{w}_1^\top, \mathbf{w}_2^\top, \dots, \mathbf{w}_I^\top]^\top$, where \mathbf{w}_i has i.i.d. $\mathcal{N}(0, 1)$ entries and is independent with \mathbf{w}_j with $i \neq j$. In specific, there is

$$\begin{aligned} & \mathbf{e}_j \mathbb{E}[\mathbf{W} \mathbf{W}^\top \boldsymbol{\Lambda} \mathbf{W} \mathbf{W}^\top] \mathbf{e}_j^\top \\ &= \lambda_j \mathbb{E}[\mathbf{w}_j \mathbf{w}_j^\top \mathbf{w}_j \mathbf{w}_j^\top] + \sum_{i \neq j} \lambda_i \mathbb{E}[\mathbf{w}_j \mathbb{E}[\mathbf{w}_i^\top \mathbf{w}_i] \mathbf{w}_j^\top] \\ &= tM [\text{Tr}(\boldsymbol{\Lambda}) + (tM + 1)\lambda_j]. \end{aligned}$$

Putting the above results together yields

$$(\sigma_i^2 - \sigma_j^2) \mathbb{E}[\langle \tilde{\mathbf{u}}_i, \mathbf{u}_j \rangle^2] \leq \frac{4[\lambda_j^2 + \lambda_j \text{Tr}(\boldsymbol{\Lambda})]}{tM(\sigma_i^2 - \sigma_j^2)},$$

which completes the proof.

E. Proof of Theorem 2

Using Lemma 3, we rewrite the error bound as

$$\begin{aligned} & \|(\mathbf{I}_I - \tilde{\mathbf{U}} \tilde{\mathbf{U}}^\top) \mathbf{X}\|_F^2 - \sum_{i \geq r+1} \sigma_i^2 \\ & \leq g(\mathbf{W}) + \sum_{i=1}^r \sum_{j \geq r+1} \frac{4\lambda_j^2}{\sigma_i^2 - \sigma_j^2}, \end{aligned}$$

where the involved random part is defined as $g(\mathbf{W}) \triangleq \text{Tr}(\mathbf{W} \mathbf{W}^\top \boldsymbol{\Lambda} \mathbf{W} \mathbf{W}^\top \boldsymbol{\Lambda}_1) - \text{Tr}(\mathbf{W} \mathbf{W}^\top \boldsymbol{\Lambda}_2)$ with $\boldsymbol{\Lambda}_1 = \sum_{i=1}^r \sum_{j \geq r+1} \frac{4}{\sigma_i^2 - \sigma_j^2} \frac{\lambda_j}{(tM)^2} \mathbf{e}_j^\top \mathbf{e}_j$ and $\boldsymbol{\Lambda}_2 = \sum_{i=1}^r \sum_{j \geq r+1} \frac{8\lambda_j^2}{(\sigma_i^2 - \sigma_j^2)tM} \mathbf{e}_j^\top \mathbf{e}_j$. It then follows that the probabilistic error bound is upper bounded as

$$\begin{aligned} & \Pr \left[\|(\mathbf{I}_I - \tilde{\mathbf{U}} \tilde{\mathbf{U}}^\top) \mathbf{X}\|_F^2 - \sum_{i \geq r+1} \sigma_i^2 \geq \mathbb{E}[g(\mathbf{W})] + \epsilon \right] \\ & \leq \Pr[g(\mathbf{W}) - \mathbb{E}[g(\mathbf{W})] \geq \epsilon]. \end{aligned}$$

Next, rewrite $\mathbf{W} = [\tilde{\mathbf{w}}_1, \tilde{\mathbf{w}}_2, \dots, \tilde{\mathbf{w}}_{tM}]$ and the random variable $g(\mathbf{W})$ can be rewritten as

$$g(\mathbf{W}) = \sum_{m_1, m_2} \tilde{\mathbf{w}}_{m_1}^\top \boldsymbol{\Lambda} \tilde{\mathbf{w}}_{m_2} \tilde{\mathbf{w}}_{m_2}^\top \boldsymbol{\Lambda}_1 \tilde{\mathbf{w}}_{m_1} - \sum_{m_3} \tilde{\mathbf{w}}_{m_3}^\top \boldsymbol{\Lambda}_2 \tilde{\mathbf{w}}_{m_3}.$$

Let $\hat{\mathbf{w}}_m$ be an independent copy of $\tilde{\mathbf{w}}_m$ and then there is

$$\begin{aligned}
& |g'(\{\tilde{\mathbf{w}}_M\}_{M \neq m}, \tilde{\mathbf{w}}_m) - g'(\{\tilde{\mathbf{w}}_M\}_{M \neq m}, \hat{\mathbf{w}}_m)| \\
&= |\hat{\mathbf{w}}_m^\top \Lambda_2 \hat{\mathbf{w}}_m - \tilde{\mathbf{w}}_m^\top \Lambda_2 \tilde{\mathbf{w}}_m \\
&\quad + \tilde{\mathbf{w}}_m^\top \Lambda \tilde{\mathbf{w}}_m \tilde{\mathbf{w}}_m^\top \Lambda_1 \tilde{\mathbf{w}}_m - \hat{\mathbf{w}}_m^\top \Lambda \hat{\mathbf{w}}_m \hat{\mathbf{w}}_m^\top \Lambda_1 \hat{\mathbf{w}}_m \\
&\quad + \sum_{m_1 \neq m} \tilde{\mathbf{w}}_{m_1}^\top \Lambda (\tilde{\mathbf{w}}_m \tilde{\mathbf{w}}_m^\top - \hat{\mathbf{w}}_m \hat{\mathbf{w}}_m^\top) \Lambda_1 \tilde{\mathbf{w}}_{m_1} \\
&\quad + \sum_{m_2 \neq m} \tilde{\mathbf{w}}_{m_2}^\top \Lambda_1 (\tilde{\mathbf{w}}_m \tilde{\mathbf{w}}_m^\top - \hat{\mathbf{w}}_m \hat{\mathbf{w}}_m^\top) \Lambda \tilde{\mathbf{w}}_{m_2}|.
\end{aligned}$$

Note that the above equation does not have an upper bound since the Gaussian random variables go from minus infinity to infinity. To endow on $g(\mathbf{W})$ the bounded difference property, the Gaussian concentration is exploited as follows. Specifically, a $\mathcal{N}(0, 1)$ variable w can be smaller than a threshold, say $\kappa > 1$, with the probability of $p(|w| \leq \kappa) = \text{erf}\left(\frac{\kappa}{\sqrt{2}}\right) \triangleq p_\kappa$, where $\text{erf}(\cdot)$ denotes the error function. Hence, let the event that the abstract value of the last $I - r$ elements in the vectors $\tilde{\mathbf{w}}_m$ and $\hat{\mathbf{w}}_m$ are bounded by κ be denoted by BD and its complement by UBD. Then, there are $\Pr(\text{BD}) = p_\kappa^{tM(I-r)}$ and $\Pr(\text{UBD}) = 1 - p_\kappa^{tM(I-r)}$. Hence, with the probability of $\Pr(\text{BD})$, $|g'(\{\tilde{\mathbf{w}}_M\}_{M \neq m}, \tilde{\mathbf{w}}_m) - g'(\{\tilde{\mathbf{w}}_M\}_{M \neq m}, \hat{\mathbf{w}}_m)|$ can be upper bounded as

$$\begin{aligned}
& |g'(\{\tilde{\mathbf{w}}_M\}_{M \neq m}, \tilde{\mathbf{w}}_m) - g'(\{\tilde{\mathbf{w}}_M\}_{M \neq m}, \hat{\mathbf{w}}_m)| \\
&\leq \kappa^4 \text{Tr}(\Lambda) \text{Tr}(\Lambda_1) - \kappa^2 \text{Tr}(\Lambda_2) + 2(tM - 1)\kappa^4 \text{Tr}(\Lambda) \text{Tr}(\Lambda_1) \\
&= \sum_{i=1}^r \sum_{j \geq r+1} \frac{8\lambda_j}{(\sigma_i^2 - \sigma_j^2)tM} \left[\frac{tM - 1/2}{tM} \kappa^4 \text{Tr}(\Lambda) - \lambda_j \kappa^2 \right] \\
&\leq 2\kappa^4 \sum_{i=1}^r \sum_{j \geq r+1} \frac{4\lambda_j}{(\sigma_i^2 - \sigma_j^2)tM} [\text{Tr}(\Lambda) + \lambda_j] \\
&= 2\kappa^4 \mathbb{E}[g(\mathbf{W})],
\end{aligned}$$

which allows us to leverage the concentration theorem shown in Lemma 4 to give

$$\Pr[g(\mathbf{W}) - \mathbb{E}[g(\mathbf{W})] \geq \epsilon | \text{BD}] \leq \exp\left(-\frac{\epsilon^2}{2\kappa^8 \mathbb{E}^2[g(\mathbf{W})]}\right).$$

Concluding both cases of BD and UBD, the probabilistic error bound can be expressed as

$$\begin{aligned}
& \Pr [g(\mathbf{W}) - \mathbb{E}[g(\mathbf{W})] \geq \epsilon] \\
&= \Pr [g(\mathbf{W}) - \mathbb{E}[g(\mathbf{W})] \geq \epsilon | \text{BD}] \Pr[\text{BD}] \\
&\quad + \Pr [g(\mathbf{W}) - \mathbb{E}[g(\mathbf{W})] \geq \epsilon | \text{UBD}] \Pr[\text{UBD}] \\
&\leq \exp \left(-\frac{\epsilon^2}{2\kappa^8 \mathbb{E}^2[g(\mathbf{W})]} \right) p_\kappa^{tM(I-r)} \\
&\quad + \Pr [g(\mathbf{W}) - \mathbb{E}[g(\mathbf{W})] \geq \epsilon | \text{UBD}] (1 - p_\kappa^{tM(I-r)}) \\
&\leq \exp \left(-\frac{\epsilon^2}{2\kappa^8 \mathbb{E}^2[g(\mathbf{W})]} \right) p_\kappa^{tM(I-r)} + 1 - p_\kappa^{tM(I-r)},
\end{aligned}$$

where the last inequality is due to the fact that a conditional probability is always smaller than

1. Finally, with proper algebraic substitution, we have

$$\begin{aligned}
& \Pr \left[\|(\mathbf{I}_I - \tilde{\mathbf{U}}\tilde{\mathbf{U}}^\top)\mathbf{X}\|_F^2 \leq \mathbb{E}[g(\mathbf{W})](1 + \epsilon) + \sum_{i \geq r+1} \sigma_i^2 \right] \\
&\geq \left[1 - \exp \left(-\frac{\epsilon^2}{2\kappa^8} \right) \right] p_\kappa^{tM(I-r)},
\end{aligned}$$

which completes the proof.

F. Proof of Lemma 5

First of all, for any $i \leq r$ and $j \geq r+1$, we have $\sigma_i^2 - \sigma_j^2 \geq \sigma_r^2 - \sigma_{r+1}^2$, which gives

$$\begin{aligned}
& \frac{1}{\tilde{M}} \sum_{i=1}^r \sum_{j \geq r+1} \frac{\lambda_j^2 + \lambda_j \text{Tr}(\mathbf{\Lambda})}{\sigma_i^2 - \sigma_j^2} \\
&\leq \frac{r}{\tilde{M}(\sigma_r^2 - \sigma_{r+1}^2)} \sum_{j \geq r+1} \lambda_j [\lambda_j + \text{Tr}(\mathbf{\Lambda})].
\end{aligned}$$

Then, define $C = \sum_k \text{Tr}(\mathbf{X}_k^\top \mathbf{X}_k) = \sum_{i=1}^I \sigma_i^2$ and $\eta_{\text{th}} = \zeta_{\text{th}} C$. It follows that $\sigma_j^2 \leq \sigma_i^2 \leq C/r$, $\forall j \geq r+1 > i$, which further gives $r\lambda_j \leq r\sigma_j^2 + \frac{r\sigma^2}{2}\zeta_{\text{th}}C \leq (1 + \frac{r\sigma^2}{2}\zeta_{\text{th}})C$. As a result, one can obtain

$$\sum_{j \geq r+1} r\lambda_j [\lambda_j + \text{Tr}(\mathbf{\Lambda})] \leq (1 + \frac{r\sigma^2}{2}\zeta_{\text{th}})C \sum_{j \geq r+1} [\lambda_j + \text{Tr}(\mathbf{\Lambda})],$$

where the summation term can be upper bounded as

$$\begin{aligned}
& (I - r)\text{Tr}(\mathbf{\Lambda}) + \sum_{j \geq r+1} \lambda_j \\
& \leq (I - r)C + (I - r)\frac{I\sigma^2}{2}\zeta_{\text{th}}C + \frac{I - r}{r}C + (I - r)\frac{\sigma^2}{2}\zeta_{\text{th}}C \\
& \leq \frac{I - r}{(I + 1)r}\left(1 + \frac{r\sigma^2}{2}\zeta_{\text{th}}\right)C.
\end{aligned}$$

Putting the above results together yields

$$\begin{aligned}
& \frac{1}{\tilde{M}} \sum_{i=1}^r \sum_{j \geq r+1} \frac{\lambda_j^2 + \lambda_j \text{Tr}(\mathbf{\Lambda})}{\sigma_i^2 - \sigma_j^2} \\
& \leq \frac{(I - r)C^2}{(\sigma_r^2 - \sigma_{r+1}^2)(I + 1)r} \frac{(1 + \frac{r\sigma^2}{2}\zeta_{\text{th}})^2}{\tilde{M}},
\end{aligned}$$

where $\frac{(I-r)C^2}{(\sigma_r^2 - \sigma_{r+1}^2)(I+1)r}$ can be treated as a constant independent from the variable ζ_{th} . This completes the proof.

REFERENCES

- [1] W. Saad, M. Bennis, and M. Chen, “A vision of 6G wireless systems: Applications, trends, technologies, and open research problems,” *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, 2020.
- [2] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, “Edge artificial intelligence for 6G: Vision, enabling technologies, and applications,” *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 5–36, 2022.
- [3] G. Bergqvist and E. G. Larsson, “The higher-order singular value decomposition: Theory and an application,” *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 151–154, 2010.
- [4] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, “Tensor decomposition for signal processing and machine learning,” *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, 2017.
- [5] M. Chen, D. Gündüz, K. Huang, W. Saad, M. Bennis, A. V. Feljan, and H. V. Poor, “Distributed learning in wireless networks: Recent progress and future challenges,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3579–3605, 2021.
- [6] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, “Federated learning in mobile edge networks: A comprehensive survey,” *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 2020.
- [7] G. Tomasi and R. Bro, “Parafac and missing values,” *Chemometrics Intell. Laboratory Syst.*, vol. 75, no. 2, p. 163–180, 2005.
- [8] K. Shin, L. Sael, and U. Kang, “Fully scalable methods for distributed tensor factorization,” *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 1, pp. 100–113, 2017.
- [9] Z. Zhang, G. Zhu, R. Wang, V. K. N. Lau, and K. Huang, “Turning channel noise into an accelerator for over-the-air principal component analysis,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 7926–7941, 2022.
- [10] X. Chen, E. G. Larsson, and K. Huang, “Analog MIMO communication for one-shot distributed principal component analysis,” *IEEE Trans. Signal Process.*, vol. 70, pp. 3328–3342, 2022.

- [11] J. Fan, D. Wang, K. Wang, and Z. Zhu, “Distributed estimation of principal eigenspaces,” *Ann. Stat.*, vol. 47, no. 6, pp. 3009–3031, Oct. 2019.
- [12] V. Charisopoulos, A. R. Benson, and A. Damle, “Communication-efficient distributed eigenspace estimation,” *SIAM J. Math. Data Sci.*, vol. 3, no. 4, pp. 1067–1092, 2021.
- [13] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.
- [14] G. Zhu, J. Xu, K. Huang, and S. Cui, “Over-the-air computing for wireless data aggregation in massive IoT,” *IEEE Wireless Commun.*, vol. 28, no. 4, pp. 57–65, 2021.
- [15] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, “Over-the-air federated learning from heterogeneous data,” *IEEE Trans. Signal Process.*, vol. 69, pp. 3796–3811, 2021.
- [16] K. Yang, T. Jiang, Y. Shi, and Z. Ding, “Federated learning via over-the-air computation,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, 2020.
- [17] M. Mohammadi Amiri and D. Gündüz, “Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air,” *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, 2020.
- [18] X. Li, S. Wang, and Y. Cai, “Tutorial: Complexity analysis of singular value decomposition and its variants,” [Online] <https://arxiv.org/abs/1906.12085>, 2019.
- [19] P. Popovski, F. Chiarotti, K. Huang, A. E. Kalør, M. Kountouris, N. Pappas, and B. Soret, “A perspective on time toward wireless 6G,” *Proc. IEEE*, vol. 110, no. 8, pp. 1116–1146, 2022.
- [20] Y. Sun, Y. Guo, C. Luo, J. Tropp, and M. Udell, “Low-rank Tucker approximation of a tensor from streaming data,” *SIAM J. Math. Data Sci.*, vol. 2, no. 4, pp. 1123–1150, 2020.
- [21] N. Halko, P. G. Martinsson, and J. A. Tropp, “Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions,” *SIAM Rev.*, vol. 53, no. 2, pp. 217–288, 2011.
- [22] J. A. Tropp, A. Yurtsever, M. Udell, and V. Cevher, “Streaming low-rank matrix approximation with an application to scientific simulation,” *SIAM J. Sci. Comput.*, vol. 41, no. 4, pp. 2430–2463, 2019.
- [23] G. Zhu and K. Huang, “MIMO over-the-air computation for high-mobility multimodal sensing,” *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6089–6103, 2019.
- [24] Y. Sun, Y. Guo, J. A. Tropp, and M. Udell, “Tensor random projection for low memory dimension reduction,” in *Proc. Adv. in Neural Inf. Process. Syst. (NeurIPS)*, Montréal, CA, Dec. 2018.
- [25] B. N. Parlett, *The Symmetric Eigenvalue Problem*. SIAM, Philadelphia, 1998.
- [26] J. L. Doob, “Regularity properties of certain families of chance variables,” *Trans. Am. Math. Soc.*, vol. 47, no. 3, pp. 455–486, 1940.
- [27] A. Yang, Z. He, C. Xing, Z. Fei, and J. Kuang, “The role of large-scale fading in uplink massive MIMO systems,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 477–483, 2016.
- [28] E. Björnson, E. G. Larsson, and T. L. Marzetta, “Massive MIMO: ten myths and one critical question,” *IEEE Commun. Mag.*, vol. 54, no. 2, pp. 114–123, 2016.
- [29] A. Loukas, “How close are the eigenvectors of the sample and actual covariance matrices?” in *Proc. Int. Conf. Mach. Learn. (ICML)*, Sydney, Australia, Aug. 2017.