LOW-RANK OPTIMIZATION FOR DATA SHUFFLING IN WIRELESS DISTRIBUTED COMPUTING

Kai Yang[*]*, Yuanming Shi*[*]*, and Zhi Ding*[†]

[*]School of Information Science and Technology, ShanghaiTech University, Shanghai, China
[†]Dept. of ECE, University of California, Davis, California 95616, USA
E-mail: {yangkai, shiym}@shanghaitech.edu.cn, zding@ucdavis.edu

ABSTRACT

Wireless distributed computing presents new opportunities to execute intelligent tasks on mobile devices for low-latency applications, by wirelessly aggregating the computation and storage resources among mobile devices. However, for low-latency applications, the key bottleneck lies in the exchange of intermediate results among mobile devices for *data shuffling*. To improve communication efficiency therein, we establish a novel interference alignment condition by exploiting the locally computed intermediate values as side information. The low-rank optimization model is further developed to maximize the achieved degrees-of-freedom (DoFs). Unfortunately, existing convex relaxation based approach fails to yield satisfied performance due to the poor structure in the formulated low-rank optimization problem, for which we develop a novel difference-of-convex (DC) programming based algorithm. We show that this new approach can significantly improve communication efficiency and the achievable DoF is independent of the number of mobile devices.

Index Terms— Wireless distributed computing, low rank, data shuffling

1. INTRODUCTION

Machine and deep learning has become a key enabling technology for big data analytics, thereby providing diversified artificial intelligence applications, e.g., computer vision and natural language processing. Moreover, the proliferation of smart mobile devices and Internet-of-Things (IoT) devices, has made it possible to execute real-time and private machine learning applications on the collected input data directly from sensors in end devices. The emerging applications for mobile edge intelligence include augmented reality, smart vehicles, and drones. However, the requirement of ultra-low response latency [1] for executing intensive computation tasks on resource-constrained end devices [2] remains one of the key challenges. Given limited resources of computation, storage and energy at mobile devices, it's generally infeasible to accomplish computation tasks directly on a single device. Wireless distributed computing [3] presents promises to support computation intensive intelligent tasks execution on each end device by pooling the computation and storage resource among the devices.

In wireless distributed computing systems for large-scale intelligent tasks, the dataset (e.g., a feature library of objects) is normally too large for storing in a single mobile device. It thus can be stored across devices, during the *dataset placement phase*, supported by the distributed computing framework such as MapReduce [4]. With the input data (e.g., the feature vector of an image), each mobile device then performs local computation based on the locally stored dataset, which is called the *map phase*. By exchanging the computed intermediate values among devices (i.e., *shuffle phase*), the output (e.g., the inference result of the image) of each mobile device can be constructed with additional local computations (i.e., *reduce phase*). To enable real-time and low-latency applications, the inter-device communication for data shuffling becomes the main bottleneck. Recent works [3] and [5] proposed coding schemes to reduce the communication load (e.g., the number of information bits) for data shuffling in wireline and wireless distributed computing system respectively.

However, in wireless networks with limited spectral resources and interference, it is also critical to improve the communication efficiency (i.e., achieved data rates) for data shuffling. In this paper, we propose a systematic linear coding approach to improve the communication efficiency in the shuffle phase. Specifically, by exploiting the locally computed intermediate values in the map phase as the side information, we establish a novel interference alignment [6] (IA) condition for data shuffling. Note that orthogonal uplink transmission is assumed in [3], while we assume co-frequency transmission in both uplink and downlink to improve spectral efficiency. Based on the proposed IA condition, we further develop a low-rank model to maximize the achievable degrees-of-freedom (DoF), i.e., the first-order characterization for the achievable data rate. Unfortunately, due to the non-convexity of rank function, the resulting low-rank optimization problem is computationally infeasible.

Low-rank approaches have attracted enormous attention in machine learning, high-dimensional statistics, and recommendation system [7]. A growing body of research focuses on developing convex and nonconvex algorithms. In particular, nuclear norm relaxation approach is well-known as the convex envelope of rank function [7]. To further improve the performance, iterative reweighted least square algorithm [8] is proposed by alternating between minimizing weighted Frobenius norm and updating weights. Unfortunately, due to poorly structured affine constraint in the proposed low-rank optimization model, both approaches fail to yield satisfactory performance. We thus present a novel DC (difference-of-convex-functions) [9] programming based approach for the proposed low-rank problem. Specifically, we sequentially determining the minimal k such that the difference between nuclear norm and Ky Fan k-norm [10] (i.e., the sum of largest-k singular values of a matrix) becomes zero. The resulting DC programming subproblem can be solved by the principles of majorization-minimization (MM) algorithm [9]. Numerical experiments show that the DC algorithm significantly outperforms state-of-the-arts, and the achievable DoF remains constant when the number of mobile devices increase.

This work was partly supported by the National Nature Science Foundation of China under Grant No. 61601290, and the Shanghai Sailing Program under Grant No. 16YF1407700.

2. SYSTEM MODEL

In this section, we shall present the wireless distributed computing system, followed by transceiver design with the interference alignment condition to improve the communication efficiency for accomplishing the computation tasks.

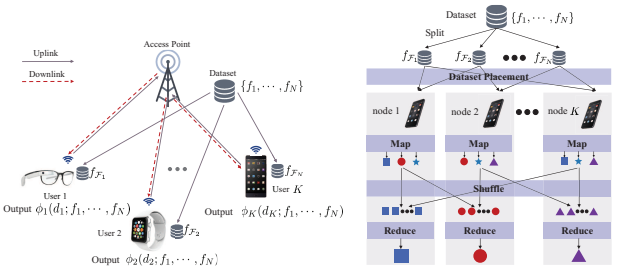
2.1. Computation Model

We consider a wireless distributed computing system with K mobile users, where all users communicate through a common wireless access point (AP) as shown in Fig. 1a. Suppose each mobile user is equipped with L antennas and AP is equipped with M antennas. The dataset (e.g., a feature library of objects in object recognition) of the system is evenly separated to N files $f_1, \cdots, f_N$, each with F bits. Each user k has a computation task $\phi_k(d_k; f_1, \cdots, f_N)$ which maps the input d_k (e.g., the feature vector of an image) to an output result (e.g., the inference result of the image). We consider the scenario that the local memory of each user can only store up to μ files (i.e., μF bits with $\mu < N$), while the entire dataset can be stored collectively by all the users. The index set of files stored at the k-th node is denoted by $\mathcal{F}_k \subseteq [N] = \{1, 2, \cdots, N\}$ where $|\mathcal{F}_k| \leq \mu$ for all $k \in [K]$ and $\cup_{k\in[K]}\mathcal{F}_k = [N]$.

In this paper, we adopt the distributed computing framework such as MapReduce [4] and Spark to accomplish the computation tasks for each mobile users. Specifically, the computation task ϕ_k is assumed to be decomposed as follows [3]

$$\phi_k(d_k; f_1, \cdots, f_N) = h_k(g_{k,1}(d_k; f_1), \cdots, g_{k,N}(d_k; f_N)). \quad (1)$$

Here, $g_{k,t}(d_k; f_t)$ is the *Map* function, which maps input d_k and file f_t into an intermediate value $w_{k,t}$ with E bits, and $h_k(w_{k,1}, \cdots, w_{k,N})$ is the *Reduce* function which maps all required intermediate values $w_{k,1}, \cdots, w_{k,N}$ into the output of the joint computation task. Note that we focus on applications in which the sizes of inputs and intermediate values are small enough so that they can be stored locally and the overhead of collecting inputs is negligible. The overall procedure for distributed computing shall be:



(a) System Model (b) Computation Model

Fig. 1: a) Wireless distributed computing system where $f_{\mathcal{F}_k} = \{f_j : j \in \mathcal{F}_k\}$. b) Distributed computing model.

- **Dataset Placement Phase:** To execute Map Phase, we need to determine file placement $\mathcal{F}_k$ and delivery files in advance.
- **Map Phase:** Compute intermediate values $w_{s,t}$ locally with map functions $g_{s,t}$ for all $s \in [K], t \in \mathcal{F}_k$.
- **Shuffle Phase:** User k shall collect the intermediate values $\{w_{k,t} : t \notin \mathcal{F}_k\}$ that cannot be computed locally and exchange intermediate values wirelessly with others for task ϕ_k.
- **Reduce Phase:** Each mobile user constructs the output value by mapping all required intermediate values into the output value, i.e., $\phi_k(d_k; f_1, \cdots, f_N) = h_k(w_{k,1}, \cdots, w_{k,N})$.

2.2. Communication Model

Given the dataset placement, this paper aims at improving the communication efficiency for the Shuffle Phase. Specifically, we denote the entire set of messages (i.e., all intermediate values $\{w_{k,n} : k \in [K], n \in [N]\}$) by $\{W_1, \cdots, W_T\}$ with $T = KN$. Let $\mathcal{T}_k \subseteq [T]$ be the index set of intermediate values available at mobile user k and $\mathcal{R}_k \subseteq [T]$ be the index set of intermediate values required by mobile user k. Here, we have $\cup_{k\in[K]}\mathcal{T}_k = [T], \mathcal{T}_k \cap \mathcal{R}_k = \emptyset$. As a result, the Shuffle Phase is reformulated as a message delivery problem with side information. As shown in Fig. 1a, the Shuffle Phase consists of *uplink multiple access (MAC) stage* and *downlink broadcasting (BC) stage* to construct the output in the Reduce Phase.

Let $\boldsymbol{x}_k = [\boldsymbol{x}_k[i]] \in \mathbb{C}^{Lr}$ be the transmitted signal at mobile user k, and $\boldsymbol{x}_k[i] \in \mathbb{C}^r$ is the signal transmitted at the i-th antenna over r channel uses. Then the received signal at the s-th antenna of AP in uplink MAC stage over r channel use is given by

$$\boldsymbol{y}[s] = \sum_{k=1}^{K}\sum_{i=1}^{L} H_k^u[s,i]\boldsymbol{x}_k[i] + \boldsymbol{n}^u[s], \quad (2)$$

where $H_k^u[s,i] \in \mathbb{C}$ is the flat-fading channel coefficient between the i-th antenna of the k-th mobile user and the s-th antenna of AP, and $\boldsymbol{n}^u[s] \in \mathbb{C}^r$ is the additive isotropic white Gaussian noise. In this work, we assume the block fading channel where the channel coefficients remain unchanged over r channel uses. The received signal at the AP is given by

$$\boldsymbol{y} = \sum_{k=1}^{K}(\boldsymbol{H}_k^u \otimes \boldsymbol{I}_r)\boldsymbol{x}_k + \boldsymbol{n}^u, \quad (3)$$

where $\otimes$ denotes Kronecker product, $\boldsymbol{y} = [\boldsymbol{y}[s]] \in \mathbb{C}^{Mr}$, $\boldsymbol{H}_k^u = [H_k^u[s,i]] \in \mathbb{C}^{M\times L}$ is the channel coefficient matrix between AP and the k-th mobile user in uplink MAC stage, and $\boldsymbol{n}^u \in \mathbb{C}^{Mr}$ is the additive isotropic white Gaussian noise. After receiving the signal from all mobile users, AP will forward it to each mobile user directly, which means that the signal received by user k is given by

$$\boldsymbol{z}_k = (\boldsymbol{H}_k^d \otimes \boldsymbol{I}_r)\boldsymbol{y} + \boldsymbol{n}_k^d. \quad (4)$$

Here $\boldsymbol{H}_k^d \in \mathbb{C}^{L\times M}$ is the downlink channel coefficient matrix between AP and mobile user k. $\boldsymbol{n}_k^d \in \mathbb{C}^{Lr}$ is the downlink Gaussian noise. Then the overall input-output relationship from mobile user to mobile user can be represented by

$$\boldsymbol{z}_k = \sum_{i=1}^{K}(\boldsymbol{H}_{ki} \otimes \boldsymbol{I}_r)\boldsymbol{x}_i + \boldsymbol{n}_k, \quad (5)$$

in which $\boldsymbol{H}_{ki} = \boldsymbol{H}_k^d\boldsymbol{H}_i^u \in \mathbb{C}^{L\times L}$ is the equivalent channel coefficient matrix from the i-th mobile user to the k-th mobile user, and $\boldsymbol{n}_k = (\boldsymbol{H}_k^d \otimes \boldsymbol{I}_r)\boldsymbol{n}^u + \boldsymbol{n}_k^d$ is the effective additive noise. Here we use the equation $(\boldsymbol{A}\otimes\boldsymbol{B})(\boldsymbol{C}\otimes\boldsymbol{D}) = (\boldsymbol{AC})\otimes(\boldsymbol{BD})$. For message l at user k, the degree-of-freedom (DoF) is defined as

$$\mathrm{DoF}_{k,l} \triangleq \limsup_{\mathrm{SNR}_{k,l}\to\infty} \frac{R_{k,l}}{\log(\mathrm{SNR}_{k,l})}, \quad (6)$$

where $\mathrm{SNR}_{k,l}$ is the signal-to-noise-ratio (SNR) and the rate $R_{k,l}$ is achievable if the error probability can be arbitrarily small with certain coding scheme.

2.3. Interference Alignment Conditions for Linear Coding

Linear schemes for transceiver design are widely used such as in interference alignment [6] and index coding [11] because of the low-complexity and optimality in DoF. Therefore, we focus on linear coding scheme in this work. Let $\boldsymbol{s}_j \in \mathbb{C}^d$ be the representative vector for message W_j with d datastreams where each datastream

carries one degree-of-freedom (DoF). Then the transmitted signal at the i-th antenna of user k is given by

$$\boldsymbol{x}_k[i] = \sum_{j \in \mathcal{T}_k} \boldsymbol{V}_{kj}[i]\boldsymbol{s}_j, \tag{7}$$

where $\boldsymbol{V}_{kj}[i] \in \mathbb{C}^{r\times d}$ is the precoding matrix corresponding to the i-th antenna of mobile user k for message j. Let $\boldsymbol{U}_{kl} \in \mathbb{C}^{d\times Lr}$ be the decoding matrix for each message $W_l, l \in \mathcal{R}_k$. Specifically, we want to decode message W_l from

$$\tilde{\boldsymbol{z}}_{kl} = \boldsymbol{U}_{kl}\boldsymbol{z}_k = \boldsymbol{U}_{kl}\sum_{i=1}^{K}(\boldsymbol{H}_{ki}\otimes \boldsymbol{I}_r)\sum_{j\in\mathcal{T}_i}\boldsymbol{V}_{ij}\boldsymbol{s}_j + \boldsymbol{n}_{kl}, \tag{8}$$

in which $\boldsymbol{V}_{kj} = [\boldsymbol{V}_{kj}[i]] \in \mathbb{C}^{Lr\times d}$ is the precoding matrix, and $\boldsymbol{n}_{kl} = \boldsymbol{U}_{kl}((\boldsymbol{H}_k^d \otimes \boldsymbol{I}_r)\boldsymbol{n}^u + \boldsymbol{n}_k^d)$. We observe that $\tilde{\boldsymbol{z}}_{kl}$ is the linear combination of the entire message set which can be split into three parts: desired message, interferences, and locally available messages, i.e.,

$$\tilde{\boldsymbol{z}}_{kl} = \mathcal{I}_1(\underbrace{\boldsymbol{s}_l}_{\text{desired message}}) + \mathcal{I}_2(\underbrace{\{\boldsymbol{s}_j : j\in\mathcal{T}_k\}}_{\text{locally available messages}}) + \mathcal{I}_3(\underbrace{\{\boldsymbol{s}_j : j\notin\mathcal{T}_k\cup\{l\}\}}_{\text{interferences}}).$$

Specifically, linear operators $\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3$ are given by

$$\mathcal{I}_1(\boldsymbol{s}_l) = \sum_{i:l\in\mathcal{T}_i}\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{il}\boldsymbol{s}_l,$$

$$\mathcal{I}_2(\{\boldsymbol{s}_j : j\in\mathcal{T}_k\}) = \sum_{j\in\mathcal{T}_k}\sum_{i:j\in\mathcal{T}_i}\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{ij}\boldsymbol{s}_j,$$

$$\mathcal{I}_3(\{\boldsymbol{s}_j : j\notin\mathcal{T}_k\cup\{l\}\}) = \sum_{j\notin\mathcal{T}_k\cup\{l\}}\sum_{i:j\in\mathcal{T}_i}\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{ij}\boldsymbol{s}_j.$$

We now propose the following interference alignment conditions

$$\det\left(\sum_{i:l\in\mathcal{T}_i}\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{il}\right) \neq 0, \tag{9}$$

$$\sum_{i:j\in\mathcal{T}_i}\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{ij} = \boldsymbol{0},\ j\notin\mathcal{T}_k\cup\{l\} \tag{10}$$

to estimate W_l from $\tilde{\boldsymbol{s}}_l = \mathcal{I}_1^{-1}(\tilde{\boldsymbol{z}}_{kl} - \mathcal{I}_2(\{\boldsymbol{s}_j : j\in\mathcal{T}_k\}))$ for all $l\in\mathcal{R}_k, k\in[K]$.

If conditions (9) (10) are satisfied, we can obtain interference-free channels for the transmission of d-dimensional messages over r channel uses. $\text{DoF}_{k,l}$ is thus given by d/r. Hence the symmetric DoF (largest achievable DoF for all k, l) is

$$\text{DoF}_{\text{sym}} = d/r. \tag{11}$$

Consequently, achievable symmetric DoF can be maximized by finding the minimum r such that (9) (10) are satisfied.

3. A LOW-RANK FRAMEWORK FOR DATA SHUFFLING

In this section, we propose a low-rank optimization formulated to maximize the achievable symmetric DoF in the Shuffle Phase, followed by a DC programming based algorithm.

3.1. Low-Rank Optimization Approach

Consider the proposed interference alignment conditions (9) (10) for data shuffling in wireless distributed computing. Without loss of generality, to enable efficient algorithms design, we set $\sum_{i:l\in\mathcal{T}_i}\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{il} = \boldsymbol{I}$ in (9). Let $\boldsymbol{U}_{kl} = [\boldsymbol{U}_{kl}[1],\cdots,\boldsymbol{U}_{kl}[L]], \tilde{\boldsymbol{U}}_{kl}^{\mathsf{H}} = [\boldsymbol{U}_{kl}[1]^{\mathsf{H}},\cdots,\boldsymbol{U}_{kl}[L]^{\mathsf{H}}] \in \mathbb{C}^{Ld\times r}, \tilde{\boldsymbol{U}}^{\mathsf{H}} = [\tilde{\boldsymbol{U}}_{11}^{\mathsf{H}},\cdots,\tilde{\boldsymbol{U}}_{1T}^{\mathsf{H}},\cdots,\tilde{\boldsymbol{U}}_{KT}^{\mathsf{H}}] \in \mathbb{C}^{LdKT\times r}, \tilde{\boldsymbol{V}}_{ij} = [\boldsymbol{V}_{ij}[1],\cdots,\boldsymbol{V}_{ij}[L]] \in \mathbb{C}^{r\times Ld}, \tilde{\boldsymbol{V}} = [\tilde{\boldsymbol{V}}_{11},\cdots,\tilde{\boldsymbol{V}}_{1T},\cdots,\tilde{\boldsymbol{V}}_{KT}] \in \mathbb{C}^{r\times LdKT}, \boldsymbol{X} = \tilde{\boldsymbol{U}}\tilde{\boldsymbol{V}} = [\boldsymbol{U}_{kl}[m]\boldsymbol{V}_{ij}[n]] = [\boldsymbol{X}_{k,l,i,j}[m,n]] \in \mathbb{C}^{LdKT\times LdKT}$. Note that

$$\boldsymbol{U}_{kl}(\boldsymbol{H}_{ki}\otimes\boldsymbol{I}_r)\boldsymbol{V}_{ij} = \sum_{m=1}^{L}\sum_{n=1}^{L}H_{ki}[m,n]\boldsymbol{U}_{kl}[m]\boldsymbol{V}_{ij}[n], \tag{12}$$

where $H_{ki}[m,n]$ is the (m,n)-th entry of matrix $\boldsymbol{H}_{ki}$. We rewrite the interference alignment conditions (9) (10) as

$$\sum_{i:l\in\mathcal{T}_i}\sum_{m=1}^{L}\sum_{n=1}^{L}H_{ki}[m,n]\boldsymbol{X}_{k,l,i,l}[m,n] = \boldsymbol{I}, \tag{13}$$

$$\sum_{i:j\in\mathcal{T}_i}\sum_{m=1}^{L}\sum_{n=1}^{L}H_{ki}[m,n]\boldsymbol{X}_{k,l,i,j}[m,n] = \boldsymbol{0},\ j\notin\mathcal{T}_k\cup\{l\}, \tag{14}$$

which can be denoted as $\mathcal{A}(\boldsymbol{X}) = \boldsymbol{b}$ with the linear operator $\mathcal{A}(\cdot)$ as a function of $\{\boldsymbol{H}_{ki}\}$. Note that the rank of matrix $\boldsymbol{X}$ is equal to the number of channel uses r, i.e.,

$$\text{rank}(\boldsymbol{X}) = r. \tag{15}$$

We thus propose the following low-rank optimization approach to find the maximum achievable symmetric DoF

$$\begin{aligned}\mathscr{P}: \underset{\boldsymbol{X}\in\mathbb{C}^{D\times D}}{\text{minimize}}\ & \text{rank}(\boldsymbol{X})\\ \text{subject to}\ & \mathcal{A}(\boldsymbol{X}) = \boldsymbol{b},\end{aligned} \tag{16}$$

where $D = LdKT$. However, problem $\mathscr{P}$ is computationally infeasible due to the non-convexity of the rank function.

3.2. Problem Analysis

Low-rank approach has caught enormous attention in machine learning, high-dimensional statistics, and recommendation systems [7]. Unfortunately, low-rank problems are highly intractable due to the non-convex rank function, for which various convex and non-convex optimization algorithms have been developed. In particular, nuclear norm [7] has demonstrated its effectiveness as the convex relaxation for the rank function. However, it yields poor performance due to the poor structure of the affine constraint in problem $\mathscr{P}$. For example, in the scenario of two users with $K = N = 2, \mu = d = L = M = 1$, problem $\mathscr{P}$ is given as

$$\begin{aligned}\underset{\boldsymbol{X}}{\text{minimize}}\ & \text{rank}(\boldsymbol{X})\\ \text{subject to}\ & \boldsymbol{X} = \begin{bmatrix}\star & \star & 1/H_{12} & 0\\ 0 & 1/H_{21} & \star & \star\end{bmatrix},\end{aligned} \tag{17}$$

where the value of $\star$ is arbitrary (here we have removed the rows and columns that are all unconstrained). In this case, the nuclear norm approach always returns full rank solution while the optimal rank is 1. To further improve the performance of nuclear norm relaxation and enhance low-rankness, the iterative reweighted least square algorithm IRLS-p [8] ($0 \le p \le 1$) is proposed by alternating between minimizing weighted Frobenius norm and updating weights. However, it still yields poor performance when applied to problem $\mathscr{P}$ given the poorly structured affine constraint. Therefore, we shall present a novel difference-of-convex-functions algorithm (DCA) to achieve considerable performance improvement.

3.3. A DC Algorithmic Approach

We first present a novel difference-of-convex-functions (DC) representation [10] for the rank function, before developing a novel DC algorithmic approach to solve problem $\mathscr{P}$.

Definition 1 *Ky Fan k-norm [12]: The Ky Fan k-norm of a matrix $\boldsymbol{X}$ is the sum of its largest-k singular values, i.e.,*

$$\|\!|\boldsymbol{X}|\!\|_k = \sum_{i=1}^{k} \sigma_i(\boldsymbol{X}), \tag{18}$$

where $\sigma_i(\boldsymbol{X})$ is the i-th largest singular value of $\boldsymbol{X}$. We thus obtain a DC representation for the rank function based on Definition 1.

Proposition 1 *For any matrix $\boldsymbol{X} \in \mathbb{C}^{m\times n}$, the following equation holds:*

$$rank(\boldsymbol{X}) = \min\{k : \|\boldsymbol{X}\|_* - \|\!|\boldsymbol{X}|\!\|_k = 0, k \le \min\{m,n\}\}. \tag{19}$$

Proof 1 *Given $rank(\boldsymbol{X}) = r$ we have $\sigma_i(\boldsymbol{X}) = 0\ \forall i > r$ and $\sigma_i(\boldsymbol{X}) > 0\ \forall i \le r$. Since $\|\boldsymbol{X}\|_* - \|\!|\boldsymbol{X}|\!\|_k = \sum_{i=k+1}^{k} \sigma_i(\boldsymbol{X})$, the minimum k such that $\|\boldsymbol{X}\|_* - \|\!|\boldsymbol{X}|\!\|_k = 0$ will be exactly r. Conversely, $r = \min\{k : \|\boldsymbol{X}\|_* - \|\!|\boldsymbol{X}|\!\|_k = 0\}$ we deduce that $\sigma_i(\boldsymbol{X}) = 0\ \forall i > r$ and $\sigma_i(\boldsymbol{X}) > 0\ \forall i \le r$. Then $rank(\boldsymbol{X}) = r$.*

Algorithm 1: DC Algorithm (DCA) for $\mathscr{P}$

Input: $\mathcal{A}$, $\boldsymbol{b}$,, accuracy ε.
for $k = 1, \cdots, D$ **do**
  Initialize: $\boldsymbol{X}_0^{[k]} \in \mathbb{C}^{D\times D}, t = 1$
  while *not converge* **do**
    Compute $\partial\|\!|\boldsymbol{X}_{t-1}^{[k]}|\!\|_k$
    Obtain the optimal solution $\boldsymbol{X}_t^{[k]}$ of 21
  end
  if $\|\boldsymbol{X}^{[k]}\|_* - \|\!|\boldsymbol{X}^{[k]}|\!\|_k < \varepsilon$ **then**
    return $\boldsymbol{X}^{[k]}$
  end
end
Output: $\boldsymbol{X}^{[k]}$ and rank k.

Therefore, by representing the rank function with Ky Fan k-norm, problem $\mathscr{P}$ can be solved by finding the minimum k such that the optimal objective value is zero in problem

$$\begin{aligned} \underset{\boldsymbol{X}\in\mathbb{C}^{D\times D}}{\text{minimize}} \quad & \|\boldsymbol{X}\|_* - \|\!|\boldsymbol{X}|\!\|_k \\ \text{subject to} \quad & \mathcal{A}(\boldsymbol{X}) = \boldsymbol{b}. \end{aligned} \tag{20}$$

Due to the nonconvex DC objective function, we adopt the majorization-minimization (MM) algorithm [9] to iteratively solve a convex subproblem by linearizing $\|\!|\boldsymbol{X}|\!\|_k$ as $\text{Tr}(\partial\|\!|\boldsymbol{X}_t|\!\|_k, \|\boldsymbol{X}\|_k)$, i.e., solving

$$\begin{aligned} \underset{\boldsymbol{X}\in\mathbb{C}^{D\times D}}{\text{minimize}} \quad & \|\boldsymbol{X}\|_* - \text{Tr}(\partial\|\!|\boldsymbol{X}_{t-1}|\!\|_k, \|\boldsymbol{X}\|_k) \\ \text{subject to} \quad & \mathcal{A}(\boldsymbol{X}) = \boldsymbol{b} \end{aligned} \tag{21}$$

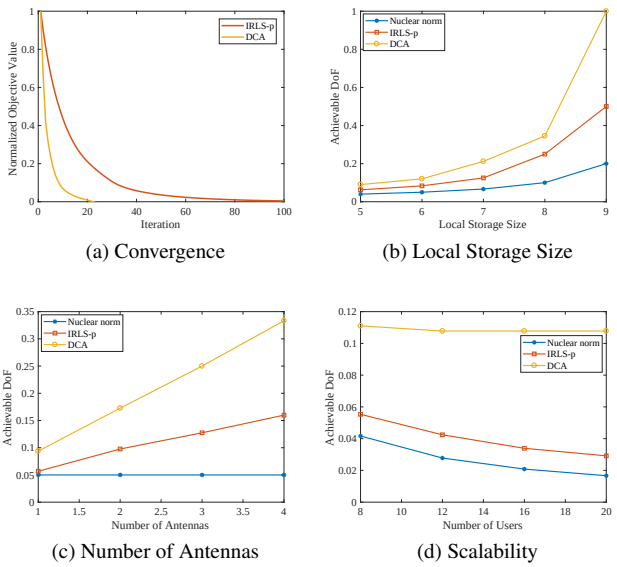
at the t-th iteration. Here $\boldsymbol{X}_{t-1}$ is the solution to (21) in the $t-1$ iteration. $\partial\|\!|\boldsymbol{X}_t|\!\|_k$ [12] is the subdifferential of $\|\!|\boldsymbol{X}|\!\|_k$ at $\boldsymbol{X}_t$ and

$$\partial\|\!|\boldsymbol{X}_t|\!\|_k = \{\boldsymbol{U}\text{diag}(\boldsymbol{q})\boldsymbol{V}^{\mathsf{H}}, \boldsymbol{q} = [\underbrace{1,\cdots,1}_{k},\underbrace{0,\cdots,0}_{D-k}]^T\}, \tag{22}$$

in which $\boldsymbol{X}_t = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$ is the singular value decomposition (SVD) of $\boldsymbol{X}_t$. The whole DC algorithm (DCA) is shown in Algorithm 1.

4. SIMULATION

In this section, we evaluate the convergence rate and achievable DoF in different settings of state-of-art algorithms. For IRLS-p algorithm we set $p = 0.5$ and its parameters are chosen with cross validation. Consider a system with symmetric antennas, i.e., $L = M$, and each mobile user stores μ files uniformly at random. The convergence characteristics of the DCA and IRLS-p algorithm are demonstrated in Fig 2a for a random channel realization, where the objective values are normalized to $(0, 1)$. The maximum achievable symmetric DoFs over local cache size μ and the number of antennas $L = M$ are shown in Figs 2b 2c averaged over 10 channel realizations at each point. For wireless distributed system, whether the coded communication scheme scales to large number of users is another main concern as pointed in [3]. We consider the uniform placement case when each mobile user stores μ files and each file is stored by $\mu K/N$ mobile users in Fig 2d. Numerical results demonstrate that the DCA converges with much fewer iterations in the simulation setting, DCA greatly outperforms the IRLS-p algorithm and nuclear norm approach and achievable symmetric DoF nearly remains unchanged for growing number of users with DCA. Although more requested messages are involved in the system when the number of users grows, opportunities of collaboration for mobile users also increase since each file is stored at more mobile users. However, it still remains an open and future problem to prove the scalability theoretically.



(a) Convergence (b) Local Storage Size

(c) Number of Antennas (d) Scalability

Fig. 2: Numerical experiments: a) The convergence of the DCA algorithm ($r = 12$) and IRLS-p algorithm in which $K = 5, N = 10, \mu = 5$. b) Maximum achievable symmetric DoF over μ under $K = 5, N = 10, L = M = 2$. c) Maximum achievable symmetric DoF over the number of antennas ($K = N = 4, \mu = 1$). d) Maximum achievable symmetric DoF over the number of mobile users with uniform placement strategy where $N = 4, L = M = \mu = 1$.

5. CONCLUSION

In this paper we proposed a novel low-rank optimization approach to improve the communication efficiency for wireless distributed computing among devices. We focus on the data-shuffle phase of the distributed computing and establish a novel interference alignment condition for data shuffling. To address the limitations for existing convex relaxation based approaches, we presented a DC programming based approach to solve the proposed low-rank optimization problem. Numerical results demonstrated that the proposed approach outperforms state-of-art algorithms and the achievable DoF remains constant despite the growing number of users in the network.

6. REFERENCES

[1] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, to appear, 2017.

[2] Song Han, Huizi Mao, and William J Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *Int. Conf. Learn. Representations (ICLR)*, 2016.

[3] S. Li, Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "A scalable framework for wireless distributed computing," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 2643–2654, Oct. 2017.

[4] Jeffrey Dean and Sanjay Ghemawat, "MapReduce: simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[5] S. Li, M. A. Maddah-Ali, Q. Yu, and A. S. Avestimehr, "A fundamental tradeoff between computation and communication in distributed computing," *IEEE Trans. Inf. Theory*, to appear, 2017.

[6] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the k-user interference channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3425–3441, Aug. 2008.

[7] M. A. Davenport and J. Romberg, "An overview of low-rank matrix recovery from incomplete observations," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 608–622, Jun. 2016.

[8] Karthik Mohan and Maryam Fazel, "Iterative reweighted algorithms for matrix rank minimization," *J. Mach. Learn. Res.*, vol. 13, pp. 3441–3473, Nov. 2012.

[9] Pham Dinh Tao and Le Thi Hoai An, "Convex analysis approach to DC programming: Theory, algorithms and applications," *Acta Mathematica Vietnamica*, vol. 22, no. 1, pp. 289–355, 1997.

[10] Jun-ya Gotoh, Akiko Takeda, and Katsuya Tono, "DC formulations and algorithms for sparse optimization problems," *Math. Program.*, to appear, 2017.

[11] H. Maleki, V. R. Cadambe, and S. A. Jafar, "Index coding – an interference alignment perspective," *IEEE Trans. Inf. Theory*, vol. 60, no. 9, pp. 5402–5432, Sept. 2014.

[12] GA Watson, "On matrix approximation problems with Ky Fan k norms," *Numerical Algorithms*, vol. 5, no. 5, pp. 263–272, 1993.