Federated Learning via Over-the-Air Computation

Kai Yang, *Student Member, IEEE*, Tao Jiang, *Student Member, IEEE*, Yuanming Shi, *Member, IEEE*, and Zhi Ding, *Fellow, IEEE*

***Abstract*— The stringent requirements for low-latency and privacy of the emerging high-stake applications with intelligent devices such as drones and smart vehicles make the cloud computing inapplicable in these scenarios. Instead, *edge machine learning* becomes increasingly attractive for performing training and inference directly at network edges without sending data to a centralized data center. This stimulates a nascent field termed as *federated learning* for training a machine learning model on computation, storage, energy and bandwidth limited mobile devices in a distributed manner. To preserve data privacy and address the issues of unbalanced and non-IID data points across different devices, the federated averaging algorithm has been proposed for global model aggregation by computing the weighted average of locally updated model at each selected device. However, the limited communication bandwidth becomes the main bottleneck for aggregating the locally computed updates. We thus propose a novel *over-the-air computation* based approach for fast global model aggregation via exploring the superposition property of a wireless multiple-access channel. This is achieved by joint device selection and beamforming design, which is modeled as a sparse and low-rank optimization problem to support efficient algorithms design. To achieve this goal, we provide a difference-of-convex-functions (DC) representation for the sparse and low-rank function to enhance sparsity and accurately detect the fixed-rank constraint in the procedure of device selection. A DC algorithm is further developed to solve the resulting DC program with global convergence guarantees. The algorithmic advantages and admirable performance of the proposed methodologies are demonstrated through extensive numerical results.**

***Index Terms*— Federated learning, over-the-air computation, edge machine learning, sparse optimization, low-rank optimization, difference-of-convex-functions, DC programming.**

I. Introduction

THE astounding growth in data volume promotes widespread artificial intelligent applications such as image recognition and natural language processing [1], thanks to the recent breakthroughs in machine learning (ML) techniques particularly deep learning, as well as the unprecedented levels of computing power [2]. Nowadays the typical machine learning procedure including the training process and the inference process, is supported by the cloud computing, i.e., a centralized cloud data center with the broad accessibility of computation, storage and the whole dataset. However, the emerging intelligent mobile devices and high-stake applications such as drones, smart vehicles and augmented reality, call for the critical requirements of low-latency and privacy. This makes the cloud computing based ML methodologies inapplicable [3]. Therefore, it becomes increasingly attractive to possess data locally at the edge devices and then performing training/inference directly at the edge, instead of sending data to the cloud or networks. This emerging technique is termed as *edge ML* [4], which is supported by mobile edge computing [5], [6] via pushing the cloud computing services to the network edges. The main bottleneck is the limited computation, storage, energy and bandwidth resources to enable mobile edge intelligent services. To address this issue, there is a growing body of recent works to reduce the storage overhead, time and power consumption in the inference process using the model compression methods via hardware and software co-design [7], [8]. Furthermore, various advanced distributed optimization algorithms [9]–[13] have been proposed to speed up the training process by taking advantages of the computing power and distributed data over multiple devices.

Recently, a nascent field called *federated learning* [12]–[16] investigates the possibility of distributed learning directly on the mobile devices to enjoy the benefits of better privacy and less network bandwidth. It is particular useful in situations where data are generated at mobile devices but it is undesirable/infeasible to transmit the data to servers. It has promising applications [14], [15] such as smart retail, smart healthcare, financial services, mobile content predictions, etc. However, a number of challenges arise to deploy the federated learning technique. 1) The collected non-IID (not independent and identically distributed) data across the network (i.e., the data is generated by distinct distributions across different devices), imposes significant statistical challenges to fit a mode from the non-IID data [13], [17]. 2) Large communication loads across mobile devices limit the scalability for federated learning to efficiently exchange locally computed updates at each device [12], [18]. 3) The heterogeneity of computation, storage and communication capabilities across different devices brings unique system challenges to tame latency for on-device distributed training, e.g., the stragglers (i.e., devices that run slow) may cause significant delays [10], [19]. 4) The arbitrarily adversarial behaviors of the

Manuscript received February 18, 2019; revised June 7, 2019, September 19, 2019, and October 26, 2019; accepted December 11, 2019. Date of publication January 8, 2020; date of current version March 10, 2020. This work was supported in part by the National Nature Science Foundation of China under Grant 61601290 and in part by the National Science Foundation under Grant CNS-1702752 and Grant ECCS-1711823. The associate editor coordinating the review of this article and approving it for publication was K. Choi. *(Corresponding author: Yuanming Shi.)*

Kai Yang is with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China, also with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: yangkai@shanghaitech.edu.cn).

Tao Jiang and Yuanming Shi are with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China (e-mail: jiangtao1@shanghaitech.edu.cn; shiym@shanghaitech.edu.cn).

Zhi Ding is with the Department of Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616 USA (e-mail: zding@ucdavis.edu).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2019.2961673

devices (e.g,. Byzantine failures [20]) bring critical security issues for large-scale distributed learning, which will incur a major degradation of the learning performance [21]. 5) System implementation issues such as the unreliable device connectivity, interrupted execution and slow convergence compared with learning on centralized data [14]. In particular, the federated averaging (FedAvg) algorithm [12] turns out to be a promising way to efficiently average the locally updated model at each device with unbalanced and non-IID data, thereby reducing the number of communication rounds between the center node and the end devices.

In this paper, we focus on designing the fast model aggregation approach for the FedAvg algorithm to improve the communication efficiency and speed up the federated learning system. We observe that the global model aggregation procedure consists of the transmission of locally computed updates from each device, followed by the computation of their weighted average at a central node. In consideration of both computation and communication, we shall propose a co-design approach for fast model aggregation by leveraging the principles of *over-the-air computation* (AirComp) [22]. Aircomp can improve the communication efficiency and reduce the required bandwidth [22], [23] over the traditional communication-and-computation separation method. This is achieved by exploring the superposition property of a wireless multiple-access channel to compute the desired function (i.e., the weighted average function) of distributed locally computed updates via concurrent transmission. Recent research works on AirComp have achieved significant progresses from the point of view of information theory [22], signal processing [23], transceiver design [24], [25], channel state information acquisition [26], synchronization issues [27], [28], the AirComp based model aggregation problem poses unique challenges as we need to simultaneously minimize the function distortion and maximize the number of involved devices. This is based on the key observations that the aggregation errors may lead to a notable drop of the prediction accuracy, while the convergence of training can be accelerated with more involved devices [12], [29]. To improve the communication efficiency and statistical performance of federated learning, we shall propose a joint device selection and receiver beamforming design approach to find the maximum selected devices with the mean-squared-error (MSE) requirement for fast model aggregation via AirComp. Note that selecting more devices can improve the convergence rate of federated learning, but may be infeasible under the target MSE requirement of model aggregation. Larger aggregation error will lead to poorer model accuracy. This tradeoff between learning and aggregation is also considered in the recent parallel work [30], which quantifies the device population by excluding the devices with weak channel coefficients under deep channel fading and assuming IID located devices. In contrast, we propose to select maximum number of devices given arbitrary values of channel coefficients from the point of view of mathematical optimization.

However, the joint device selection and beamforming design problem is essentially a computationally difficult mixed combinatorial optimization problem with nonconvex quadratic constraints. Specifically, device selection needs to maximize a combinatorial objective function, while the MSE requirement yields nonconvex quadratic constraints due to the multicasting duality for receiver beamforming design in AirComp [25]. To address the computational issue, we propose a sparse and low-rank modeling approach to assist efficient algorithms design. This is achieved by finding a sparse representation for the combinatorial objective function, followed by reformulating the nonconvex quadratic constraints as affine constraints with an additional rank-one matrix constraint by adopting the matrix lifting technique [31]. For the sparse optimization problem, ℓ_1-norm is a celebrated convex surrogate for the nonconvex ℓ_0-norm. The nonconvex smoothed ℓ_p-norm supported by the iteratively reweighted algorithm is a promising way to enhance the sparsity level [32], [33]. However, its convergence results rely on the carefully chosen smoothing parameter. Although the semidefinite relaxation (SDR) technique convexifies the nonconvex quadratic constraints as a linear constraint via dropping the rank-one constraint in the lifting problem, the performance degenerates with large number of antennas as its weak capability of inducing low-rank structures [34].

To address the limitations of existing algorithms for solving the presented sparse and low-rank optimization problem, we propose a unified difference-of-convex-functions (DC) approach to induce both the sparsity and low-rank structures. Specifically, to enhance sparsity, we adopt a novel DC representation for the ℓ_0-norm [35], which is given by the difference of the ℓ_1-norm and the Ky Fan k-norm [36], i.e., sum of the largest k absolute values. We also provide a DC representation for the rank-one constraint of the positive semidefinite matrix by setting the difference between its trace norm and spectral norm as zero. Based on the novel DC representations for the sparse function and low-rank constraint, we propose to induce the sparse structure in the first step as a guideline for the priority of selecting devices. In the second step, we solve a number of feasibility detection problems to find the maximum selected devices via accurately satisfying the rank-one constraint. Our proposed DC approach for enhancing sparsity is parameter free. The exact detection of the rank-one constraint is critical for accurately detecting the feasibility of nonconvex quadratic constraints in the procedure of device selection. Furthermore, the computationally efficient DC Algorithm (DC) with global convergence guarantee is developed by successively solving the convex relaxation of primal problem and dual problem of the DC program. These algorithmic advantages make the proposed DC approach for sparse and low-rank optimization outperform state-of-the-art approaches considerably.

A. Contributions

In this paper, we propose a novel fast global model aggregation approach for on-device federated learning via over-the-air computation. To improve the performance and the convergence rate for federated learning, we propose a joint device selection and beamforming approach by selecting maximum number of devices under target MSE requirement. It is formulated as a sparse and low-rank optimization problem, followed by proposing to enhance sparsity and accurately detect rank-one constraint with a novel DC approach. We then develop a DC

algorithm via successively convex relaxation with established convergence rate.

The main contributions of the paper are summarized as follows:

1) We design a novel fast model aggregation approach for federated learning via exploiting signal superposition property of a wireless multiple-access channel using the principles of over-the-air computation. This idea is achieved by joint device selection and beamforming design to improve the statistical learning performance.
2) A sparse and low-rank modeling approach is provided to support efficient algorithms design for the joint device selection and beamforming problem, which is essentially a highly intractable combinatorial optimization problem with nonconvex quadratic constraints.
3) To address the limitations of existing algorithms for sparse and low-rank optimization, we propose a unified DC representation approach to induce both the sparse and low-rank structures. The proposed DC approach has the capability of accurately detecting the feasibility of nonconvex quadratic constraints, which is critical in the procedure of device selection.
4) We further develop a DC algorithm for the presented nonconvex DC program via successive convex relaxation. The global convergence rate of the DC algorithm is further established by rewriting the DC function as the difference of strongly convex functions.

The superiority of the proposed DC approach for accurately feasibility detection and device selection will be demonstrated through extensive numerical results. It turns out that our proposed approaches can achieve better prediction accuracy and faster convergence rate in the experiments of training support vector machine (SVM) classifier on CIFAR-10 dataset.

B. Organization

The remaining part of this work is organized as follows. Section II introduces the system model of on-device distributed federated learning and problem formulation for fast model aggregation. Section III presents a sparse and low-rank modeling approach for model aggregation. Section IV provides the DC representation framework for solving the sparse and low-rank optimization problem, while in Section V the DC Algorithm is developed and its convergence rate is also established. The performances of the proposed approaches and other state-of-the-art approaches are illustrated in Section VI. We conclude this work in Section VII.

II. System Model and Problem Formulation

In this section, the on-device distributed federated learning system is presented. Based on the principles of over-the-air computation, we propose a computation and communication co-design approach based on the principles of over-the-air computation for fast model aggregation of locally computed updates at each device to improve the global model.

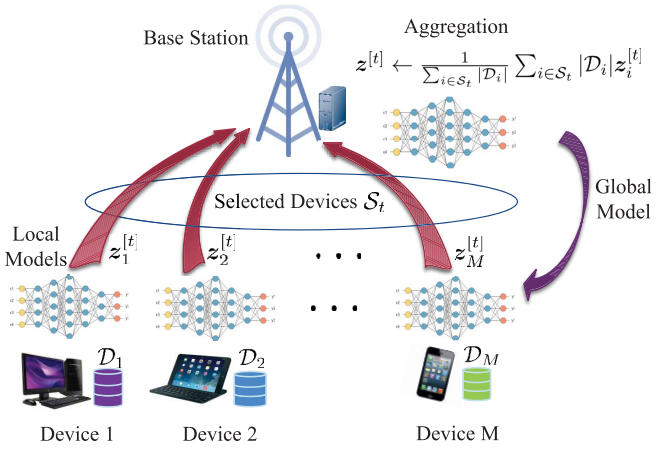


Fig. 1. On-device distributed federated learning system.

A. On-Device Distributed Federated Learning

On-device federated learning system keeps the training data at each device and learns a shared global model, which enjoys lots of benefits such as low-latency, low power consumption as well as preserving users' privacy [12]. Fig. 1 illustrates the federated learning system with M single-antenna mobile devices and one computing enabled base station (BS) equipped with N antennas to support the following distributed machine learning task:

$$\underset{\boldsymbol{z}\in\mathbb{R}^d}{\text{minimize}} \quad f(\boldsymbol{z}) = \frac{1}{T}\sum_{j=1}^{T} f_j(\boldsymbol{z}), \tag{1}$$

where $\boldsymbol{z}$ is the model parameter vector to be optimized with dimension d and T is the total number of data points. This model is widely used in linear regression, logistic regression, support vector machine, as well as deep neural networks. Typically, each function f_j is parameterized by $\ell(\boldsymbol{z};\boldsymbol{x}_j,y_j)$, where ℓ is a loss function with the input-output data pair as $(\boldsymbol{x}_j,y_j)$. Here, $\mathcal{D}=\{(\boldsymbol{x}_j,y_j):j=1,\cdots,T\}$ denotes the dataset involved in the training process. The local dataset at device i is denoted as $\mathcal{D}_i\subseteq\mathcal{D}$.

Limited network bandwidth is the main bottleneck for global model aggregation of federated learning. To reduce the number of communication rounds for global model aggregation, the federated averaging (FedAvg) algorithm [12] has recently been proposed. Specifically, at the t-th round:

1) The BS selects a subset of mobile devices $\mathcal{S}_t\subseteq\{1,\cdots,M\}$;
2) The BS sends the updated global model $\boldsymbol{z}^{[t-1]}$ to the selected devices $\mathcal{S}_t$;
3) Each selected device $i\in\mathcal{S}_t$ runs a local update algorithm (e.g., stochastic gradient algorithm) based on its local dataset $\mathcal{D}_i$ and the global model $\boldsymbol{z}^{[t-1]}$, whose output is the updated local model $\boldsymbol{z}_i^{[t]}$;
4) The BS aggregates all the local updates $\boldsymbol{z}_i^{[t]}$ with $i\in\mathcal{S}_t$, i.e., computing their weighted average as the updated global model $\boldsymbol{z}^{[t]}$.

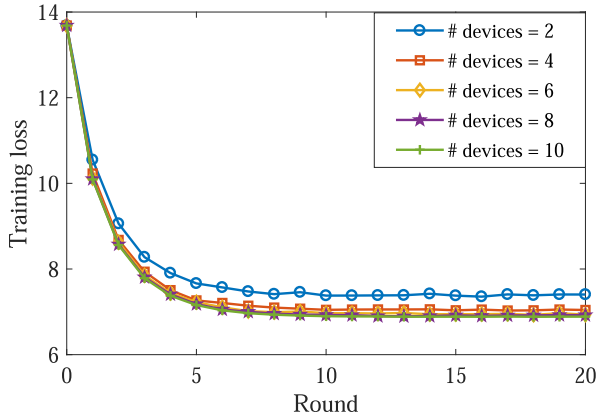
The federated averaging framework is thus presented in Algorithm 1.

Algorithm 1: Federated Averaging (FedAvg) Algorithm

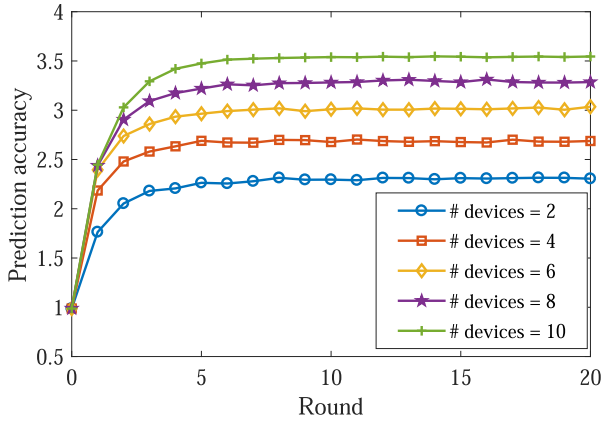
BS executes:
initialize $\boldsymbol{w}_0$.
for *each round* $t = 1, 2, \cdots$ **do**
  $\mathcal{S}_t \leftarrow$ select a subset of M devices;
  broadcast global model $\boldsymbol{z}^{[t-1]}$ to devices in $\mathcal{S}_t$.
  for *each mobile device* $i \in \mathcal{S}_t$ ***in parallel*** **do**
    $\boldsymbol{z}_i^{[t]} \leftarrow \text{LocalUpdate}(\mathcal{D}_i, \boldsymbol{z}^{[t-1]})$
  end
  $\boldsymbol{z}^{[t]} \leftarrow \frac{1}{\sum_{i\in\mathcal{S}_t}|\mathcal{D}_i|}\sum_{i\in\mathcal{S}_t}|\mathcal{D}_i|\boldsymbol{z}_i^{[t]}$ **(aggregation)**
end

In this paper, we aim at improving the communication efficiency for on-device distributed federated learning by developing a fast model aggregation approach for locally computed updates in the FedAvg algorithm. A key observation for the FedAvg algorithm is that the statistical learning performance can be improved by selecting more workers in each round [12], [29]. As an illustrative example in Fig. 2, we train an support vector machine (SVM) classifier on the CIFAR-10 dataset [37] with FedAvg algorithm and show the training loss and the relative prediction accuracy over the number of selected devices. The relative prediction accuracy is defined as the prediction accuracy over the accuracy of random classification, where the prediction accuracy is given by $\frac{\text{\# of correct predictions}}{\text{size of the test set}}$ and the accuracy of random classification is given by $\frac{1}{\text{total number of classes}} = 0.1$. The federated learning system consists of 10 mobile devices in total and the selected devices are chosen uniformly at random for each round. However, selecting more devices also brings higher communication overhead for aggregating the local computed updates at each selected device.

Note that the model aggregation procedure requires the computation of the weighted average of locally computed updates and the communication from selected mobile devices to the BS. Therefore, in this paper we develop a novel communication and computation co-design approach for fast model aggregation. Our approach is based on the principles of *over-the-air computation* [22] by leveraging the signal superposition property of a multiple-access channel. The advantages of the over-the-air computation beyond traditional communication-and-computation separation method that computes a linear function of messages across distributed mobile nodes at the center node have been demonstrated in terms of higher communication efficiency and lower bandwidth [22], [23]. They are consistent with the goal of aggregating local model updates in federated learning. Furthermore, we notice that the aggregation error may cause a notable drop of the prediction accuracy [20]. The aggregation error could be measured by mean-squared-error in equation (7). To address this issue, we shall develop efficient transceiver strategies to minimize the distortion error for model aggregation via over-the-air computation. Based on the above key observations, in this paper, we focus on the following two aspects to improve the statistical learning performance in on-device distributed federated learning system:



(a) Training loss



(b) Relative prediction accuracy

Fig. 2. The training loss and prediction accuracy with different number of randomly selected devices for FedAvg. We train an support vector machine (SVM) classifier on the CIFAR-10 dataset and adopt the stochastic gradient descent algorithm [37] as the local update algorithm for each device. Each curve is averaged for 10 times.

- Maximize the number of selected devices at each round to improve the convergence rate in the distributed training process;
- Minimize the model aggregation error to improve the prediction accuracy in the inference process.

B. Over-the-Air Computation for Aggregation

Over-the-air computation has become a promising approach for fast wireless data aggregation via computing a nomographic function (e.g., arithmetic mean) of distributed data from multiple transmitters [23]. By integrating computation and communication through exploiting the signal superposition property of a multiple-access channel, over-the-air computation can accomplish the computation of target function via concurrent transmission, thereby significantly improving the communication efficiency compared with orthogonal transmission. The key observation in the FedAvg algorithm is that the global model is updated through computing the weighted average of locally computed updates at each selected device, which falls in the category of computing nomographic functions of distributed data. In this paper, we shall propose the over-the-air computation approach for communication efficient aggregation in federated learning system.

Specifically, the target vector for aggregating local updates in FedAvg algorithm is given by

$$\boldsymbol{z} = \psi\left(\sum_{i\in\mathcal{S}} \phi_i(\boldsymbol{z}_i)\right), \tag{2}$$

where $\boldsymbol{z}_i$ is the updated local model at the i-th device, $\phi_i = |\mathcal{D}_i|$ is the pre-processing scalar at device i, $\psi = \frac{1}{\sum_{k\in\mathcal{S}}|\mathcal{D}_k|}$ is the post-processing scalar at the BS, and $\mathcal{S}$ is the selected set of mobile devices. The symbol vector for each local model before pre-processing $\boldsymbol{s}_i := \boldsymbol{z}_i \in \mathbb{C}^d$ is assumed to be normalized with unit variance, i.e., $\mathbb{E}(\boldsymbol{s}_i\boldsymbol{s}_i^{\mathsf{H}}) = \boldsymbol{I}$. At each time slot $j \in \{1, \cdots, d\}$, each device sends the signal $s_i^{(j)} \in \mathbb{C}$ to the BS. We denote

$$g^{(j)} = \sum_{i\in\mathcal{S}} \phi_i\left(s_i^{(j)}\right) \tag{3}$$

as the target function to be estimated through over-the-air computation at the j-th time slot.

To simplify the notation, we omit the time index by writing $g^{(j)}$ and $s_i^{(j)}$ as g and s_i, respectively. The received signal at the BS is given by

$$\boldsymbol{y} = \sum_{i\in\mathcal{S}} \boldsymbol{h}_i b_i s_i + \boldsymbol{n}, \tag{4}$$

where $b_i \in \mathbb{C}$ is the transmitter scalar, $\boldsymbol{h}_i \in \mathbb{C}^N$ is the channel vector between device i and the BS, and $\boldsymbol{n} \sim \mathcal{CN}(\boldsymbol{0}, \sigma^2\boldsymbol{I})$ is the noise vector. The transmit power constraint at device i is given by

$$\mathbb{E}(|b_i s_i|^2) = |b_i|^2 \le P_0 \tag{5}$$

with $P_0 > 0$ as the maximum transmit power. The estimated value before post-processing at the BS is given as

$$\hat{g} = \frac{1}{\sqrt{\eta}}\boldsymbol{m}^{\mathsf{H}}\boldsymbol{y} = \frac{1}{\sqrt{\eta}}\boldsymbol{m}^{\mathsf{H}}\sum_{i\in\mathcal{S}} \boldsymbol{h}_i b_i s_i + \frac{\boldsymbol{m}^{\mathsf{H}}\boldsymbol{n}}{\sqrt{\eta}}, \tag{6}$$

where $\boldsymbol{m} \in \mathbb{C}^N$ is the receiver beamforming vector and η is a normalizing factor. Each element of the target vector can thus be obtained as $\hat{z} = \psi(\hat{g})$ at the BS.

The distortion of $\hat{g}$ with respect to the target value g given in equation (3), which quantifies the over-the-air computation performance for global model aggregation in the FedAvg algorithm, is measured by the mean-squared-error (MSE) defined as

$$\mathsf{MSE}(\hat{g}, g) = \mathbb{E}\left(|\hat{g} - g|^2\right) = \sum_{i\in\mathcal{S}} \left|\frac{\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i b_i}{\sqrt{\eta}} - \phi_i\right|^2 + \sigma^2\frac{\|\boldsymbol{m}\|^2}{\eta}. \tag{7}$$

Motivated by [34], we have the following proposition for transmitter beamformers:

Proposition 1: Given arbitrarily chosen receiver beamforming vector $\boldsymbol{m}$, the optimal transmitter scalar that minimizes the MSE is given by the following zero-forcing transmitter:

$$b_i = \sqrt{\eta}\phi_i \frac{(\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i)^{\mathsf{H}}}{\|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2}. \tag{8}$$

Proof: See Appendix A. ∎

Due to the transmit power constraint (5) for transmit scalar b_i given in (8), we have

$$\eta = \min_{i\in\mathcal{S}} \frac{P_0\|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2}{\phi_i^2}. \tag{9}$$

The MSE is thus given as

$$\mathsf{MSE}(\hat{g}, g; \mathcal{S}, \boldsymbol{m}) = \frac{\|\boldsymbol{m}\|^2\sigma^2}{\eta} = \frac{\sigma^2}{P_0}\max_{i\in\mathcal{S}}\phi_i^2\frac{\|\boldsymbol{m}\|^2}{\|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2}. \tag{10}$$

Remark 1: Note that we use a single beamforming vector for the BS instead of M beamforming vectors. Indeed, $\boldsymbol{m}^{\mathsf{H}}\boldsymbol{y}$ is a general linear operation for mapping the received signal $\boldsymbol{y}$ to an estimated target function value $\hat{g}$. If we use multiple beamforming vectors $\boldsymbol{m}_1, \cdots, \boldsymbol{m}_M \in \mathbb{C}^N$ for their respective messages $s_1, \cdots, s_M$ and then estimate the target function g by computing their linear combination, we get the following equations

$$\hat{s}_i = \boldsymbol{m}_i^{\mathsf{H}}\boldsymbol{y}, \quad \forall i = 1, \cdots, M \tag{11}$$

$$\hat{g} = \sum_{i=1}^{M} c_i\hat{s}_i = \sum_{i=1}^{M} c_i\boldsymbol{m}_i^{\mathsf{H}}\boldsymbol{y}, \tag{12}$$

where $c_i \in \mathbb{C}$. Therefore, we can always find a single beamforming vector $\boldsymbol{m} = \sum_{i=1}^{M} c_i^*\boldsymbol{m_i} \in \mathbb{C}^N$ to achieve the same performance of multiple beams where c_i^* is the conjugate of the complex number c_i. Thus, using a single beamforming vector in over-the-air computation achieves the same performance as using multiple beamforming vectors.

C. Problem Formulation

As discussed in Section II-A, we shall maximize the number of selected devices while introducing small aggregation error with over-the-air computation. We thus formulate it as the following mixed combinatorial optimization problem

$$\underset{\mathcal{S},\boldsymbol{m}\in\mathbb{C}^N}{\text{maximize}} \ |\mathcal{S}| \ \text{ subject to } \left(\max_{i\in\mathcal{S}}\phi_i^2\frac{\|\boldsymbol{m}\|^2}{\|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2}\right) \le \gamma, \tag{13}$$

where $\gamma > 0$ is the MSE requirement for global model aggregation and $|\mathcal{S}|$ denotes the cardinality of the set $\mathcal{S}$, i.e., the set of selected devices for uploading locally updated models.

Unfortunately, the mixed combinatorial optimization problem (13) is highly intractable due to the combinatorial objective function $|\mathcal{S}|$ and the nonconvex MSE constraint with coupled combinatorial variable $\mathcal{S}$ and continuous variable $\boldsymbol{m}$. To address the nonconvexity of MSE function, [34] finds the connections between the nonconvex MSE constraint (13) and the nonconvex quadratic constraints for efficient algorithm designing. Enlightened by this observation, we will show that problem (13) can be equivalently solved by maximizing the number of feasible nonconvex quadratic constraints. Specifically, to support efficient algorithms design, we shall propose a sparse representation approach to find the maximum number of involved devices, followed by reformulating the nonconvex quadratic constraints as affine constraints with an additional rank-one constraint by the matrix lifting technique.

Remark 2: Note that the proposed transceiver design with over-the-air computation relies on the perfect channel state

information (CSI). To avoid the high overhead of CSI feedback, we can perform the transceiver design at the base station by solving problem (13) and computing equation (8). Then only channel state information at the base station is required. After computing the values of transmit scalars, the base station shall feed back each transmit scalar b_i to device i. Channel training for estimating CSI at the base station can be accomplished by transmitting pilot sequences from each mobile device [38, Chapter 4.1]. The feedback problem can be addressed using unquantized analog feedback or quantized digital feedback [39].

III. Sparse and Low-Rank Optimization for On-Device Distributed Federated Learning

In this section, we propose a sparse and low-rank optimization modeling approach for on-device distributed federated learning with device selection.

A. Sparse and Low-Rank Optimization

To support efficient algorithms design, we first rewrite problem (13) as the mixed combinatorial optimization problem with nonconvex quadratic constraints as presented in Proposition 2.

Proposition 2: Problem (13) is equivalent to the following mixed combinatorial optimization problem:

$$\begin{aligned} \underset{\mathcal{S},\boldsymbol{m}\in\mathbb{C}^N}{\text{maximize}} \ & |\mathcal{S}| \\ \text{subject to } & \|\boldsymbol{m}\|^2-\gamma_i|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2\le 0, \quad i\in\mathcal{S}, \\ & \|\boldsymbol{m}\|^2\ge 1, \end{aligned} \tag{14}$$

where $\gamma_i=\gamma/\phi_i^2$. That is, our target becomes maximizing the number of feasible MSE constraints $\|\boldsymbol{m}\|^2-\gamma_i|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2\le 0$ under the regularity condition $\|\boldsymbol{m}\|^2\ge 1$.

Proof: Problem (13) can be reformulated as

$$\begin{aligned} \underset{\mathcal{S},\boldsymbol{m}\in\mathbb{C}^N}{\text{maximize}} \ & |\mathcal{S}| \\ \text{subject to } & F_i(\boldsymbol{m})=\|\boldsymbol{m}\|^2-\gamma_i|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2\le 0, \quad i\in\mathcal{S} \\ & \boldsymbol{m}\ne\boldsymbol{0}, \end{aligned} \tag{15}$$

which is further equivalently rewritten as

$$\begin{aligned} \underset{\mathcal{S},\boldsymbol{m}\in\mathbb{C}^N}{\text{maximize}} \ & |\mathcal{S}| \\ \text{subject to } & F_i(\boldsymbol{m})/\tau=\|\boldsymbol{m}\|^2/\tau-\gamma_i|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2/\tau\le 0, \quad i\in\mathcal{S} \\ & \|\boldsymbol{m}\|^2\ge\tau,\tau>0. \end{aligned} \tag{16}$$

Then by introducing variable $\tilde{\boldsymbol{m}}=\boldsymbol{m}/\sqrt{\tau}$, problem (16) can be reformulated as

$$\begin{aligned} \underset{\mathcal{S},\tilde{\boldsymbol{m}}\in\mathbb{C}^N}{\text{maximize}} \ & |\mathcal{S}| \\ \text{subject to } & F_i(\tilde{\boldsymbol{m}})=\|\tilde{\boldsymbol{m}}\|^2-\gamma_i|\tilde{\boldsymbol{m}}^{\mathsf{H}}\boldsymbol{h}_i\|^2\le 0, \quad i\in\mathcal{S}, \\ & \|\tilde{\boldsymbol{m}}\|^2\ge 1. \end{aligned} \tag{17}$$

Therefore, problem (13) is equivalent to problem (14), where the regularity condition $\|\boldsymbol{m}\|^2\ge 1$ serves the purpose of avoiding the singularity (i.e., $\boldsymbol{m}=0$). ■

To maximize the number of feasible MSE constraints in problem (14), we can minimize the number of nonzero x_k's [32], i.e.,

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}_+^M,\boldsymbol{m}\in\mathbb{C}^N}{\text{minimize}} \ & \|\boldsymbol{x}\|_0 \\ \text{subject to } & \|\boldsymbol{m}\|^2-\gamma_i|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i\|^2\le x_i,\forall i, \\ & \|\boldsymbol{m}\|^2\ge 1. \end{aligned} \tag{18}$$

The sparsity structure of $\boldsymbol{x}$ indicates the feasibility of each mobile device. If $x_i=0$, the i-th mobile device can be selected while satisfying the MSE requirement.

However, both the MSE constraints and the regularity condition in problem (18) are nonconvex quadratic. A natural way to address it is adopting the matrix lifting technique [40]. Specifically, by lifting $\boldsymbol{m}$ as a rank-one positive semidefinite (PSD) matrix $\boldsymbol{M}=\boldsymbol{m}\boldsymbol{m}^{\mathsf{H}}$, problem (18) can be reformulated as the following sparse and low-rank optimization problem

$$\begin{aligned} \mathscr{P}: \quad \underset{\boldsymbol{x}\in\mathbb{R}_+^M,\boldsymbol{M}\in\mathbb{C}^{N\times N}}{\text{minimize}} \ & \|\boldsymbol{x}\|_0 \\ \text{subject to } & \mathrm{Tr}(\boldsymbol{M})-\gamma_i\boldsymbol{h}_i^{\mathsf{H}}\boldsymbol{M}\boldsymbol{h}_i\le x_i, \quad \forall i, \\ & \boldsymbol{M}\succeq\boldsymbol{0},\mathrm{Tr}(\boldsymbol{M})\ge 1, \\ & \mathrm{rank}(\boldsymbol{M})=1. \end{aligned} \tag{19}$$

Although problem $\mathscr{P}$ is still nonconvex, we shall demonstrate its algorithmic advantages by developing efficient algorithms.

B. Problem Analysis

Problem $\mathscr{P}$ is nonconvex with sparse objective function and low-rank constraint. Sparse and low-rank optimizations have attracted much attention in machine learning, signal processing, high-dimensional statistics, as well as wireless communication [41]–[45]. Although the sparse function and the rank function are both nonconvex and computationally difficult, efficient and provable algorithms have been developed for taming the nonconvexity by exploiting various problem structures.

1) Sparse Optimization: ℓ_1-norm is a natural convex surrogate for the nonconvex sparse function, i.e., ℓ_0-norm. The resulting problem is known as the sum-of-infeasibilities in the literature of optimization [46]. Another known approach for enhancing sparsity is the smoothed ℓ_p-minimization [32] by finding a tight approximation for the nonconvex ℓ_0-norm, followed by the iteratively reweighted ℓ_2-minimization algorithm. However, the smoothing parameters should be chosen carefully since the convergence behavior of iterative reweighted algorithms may be sensitive to them [33], [47].

2) Low-Rank Optimization: Simply dropping the rank-one constraint in problem $\mathscr{P}$ yields the semidefinite relaxation (SDR) technique [31]. The SDR technique is widely used as an effective approach to find approximate solutions for the nonconvex quadratic constrained quadratic programs. If the solution fails to be rank-one, we can obtain a rank-one approximate solution through the Gaussian randomization method [31]. However, when the number of antennas N increases, its performance deteriorates since the probability of returning rank-one solutions is low [34], [48].

To address the limitations of the existing works, in this paper, we shall propose a unified difference-of-convex-functions (DC) programming approach to solve the sparse and low-rank optimization problem $\mathscr{P}$. This approach is able to enhance the sparsity in the objective as well as accurately detect the infeasibility in the nonconvex quadratic constraints, yielding considerably improvements compared with state-of-the-art algorithms. Specifically,

- We will develop a parameter-free DC approach to enhance sparsity, thereby maximizing the number of selected devices.
- Instead of dropping the rank-one constraint directly, we will propose a novel DC approach to guarantee the exact rank-one constraint.

Note that the proposed DC approach has the capability of guarantee the feasibility of rank-one constraint, which is critical for accurately detecting the feasibility of the nonconvex quadratic constraints in the procedure of device selection.

IV. DC Representation for the Sparse and Low-Rank Functions

In this section, we shall propose a unified DC representation framework to problem $\mathscr{P}$ for federated learning. Specifically, a novel DC representation for ℓ_0-norm is used to induce sparsity for device selection. A novel DC representation for the rank function is used to induce rank-one solutions, which can accurately detect the feasibility of nonconvex quadratic programs during the procedure of device selection.

A. DC Representation for Sparse Function

Before introducing the DC representation for the ℓ_0-norm, we first give the definition of Ky Fan k-norm.

Definition 1: Ky Fan k-norm [36]: The Ky Fan k-norm of vector $\boldsymbol{x} \in \mathbb{C}^M$ is a convex function of $\boldsymbol{x}$ and is given by the sum of largest-k absolute values, i.e.,

$$\|\boldsymbol{x}\|_k = \sum_{i=1}^{k} |x_{\pi(i)}|, \tag{20}$$

where π is a permutation of $\{1, \cdots, M\}$ and $|x_{\pi(1)}| \geq \cdots \geq |x_{\pi(M)}|$.

If the ℓ_0-norm is no greater than k, its ℓ_1-norm is equal to its Ky Fan k-norm. Based on this fact, the ℓ_0-norm can be represented by the difference between ℓ_1-norm and Ky Fan k-norm [35]:

$$\|\boldsymbol{x}\|_0 = \min\{k : \|\boldsymbol{x}\|_1 - \|\boldsymbol{x}\|_k = 0, 0 \leq k \leq M\}. \tag{21}$$

B. DC Representation for Low-Rank Constraint

For the positive semidefinite (PSD) matrix $\boldsymbol{M} \in \mathbb{C}^{N \times N}$, the rank-one constraint can be equivalently rewritten as

$$\sigma_i(\boldsymbol{M}) = 0, \forall i = 2, \cdots, N, \tag{22}$$

where $\sigma_i(\boldsymbol{M})$ is the i-th largest singular value of matrix $\boldsymbol{M}$. Note that the trace norm and spectral norm are given by

$$\mathrm{Tr}(\boldsymbol{M}) = \sum_{i=1}^{N} \sigma_i(\boldsymbol{M}) \text{ and } \|\boldsymbol{M}\|_2 = \sigma_1(\boldsymbol{M}), \tag{23}$$

respectively. Therefore, we have the following proposition:

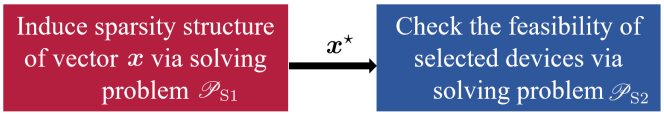


Fig. 3. A two-step framework for device selection.

Proposition 3: For PSD matrix $\boldsymbol{M}$ and $\mathrm{Tr}(\boldsymbol{M}) \geq 1$, we have

$$\mathrm{rank}(\boldsymbol{M}) = 1 \Leftrightarrow \mathrm{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 = 0. \tag{24}$$

Proof: If the rank of PSD matrix $\boldsymbol{M}$ is one, the trace norm is equal to the spectral norm as $\sigma_i(\boldsymbol{M}) = 0$ for all $i \geq 2$. The equation $\mathrm{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 = 0$ implies that $\sigma_i(\boldsymbol{M}) = 0$ for all $i \geq 2$, i.e., $\mathrm{rank}(\boldsymbol{M}) \leq 1$. And we have $\sigma_1(\boldsymbol{M}) > 0$ from $\mathrm{Tr}(\boldsymbol{M}) \geq 1$. Therefore, $\mathrm{rank}(\boldsymbol{M}) = 1$ holds if $\mathrm{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 = 0$. ∎

C. A Unified DC Representation Framework

The main idea of our proposed DC representation framework is to induce the sparsity of $\boldsymbol{x}$ in the first step, which will provide guidelines for determining the priority of selecting devices. Then we shall solve a series of feasibility detection problems to find maximum selected devices such that the MSE requirement is satisfied. This two-step framework is illustrated in Fig. 3. And each step will be accomplished by solving a DC program.

1) Step I: Sparsity Inducing: In the first step, we solve the following DC program for problem $\mathscr{P}$:

$$\begin{aligned} \mathscr{P}_{\mathrm{S1}}: \ \underset{\boldsymbol{x},\boldsymbol{M}}{\text{minimize}} \ & \|\boldsymbol{x}\|_1 - \|\boldsymbol{x}\|_k + \mathrm{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 \\ \text{subject to } & \mathrm{Tr}(\boldsymbol{M}) - \gamma_i \boldsymbol{h}_i^{\mathsf{H}} \boldsymbol{M} \boldsymbol{h}_i \leq x_i, \quad \forall i = 1, \cdots, M \\ & \boldsymbol{M} \succeq \boldsymbol{0}, \quad \mathrm{Tr}(\boldsymbol{M}) \geq 1, \boldsymbol{x} \succeq \boldsymbol{0}. \end{aligned} \tag{25}$$

By sequentially solving problem $\mathscr{P}_{\mathrm{S1}}$, we can obtain the sparse vector $\boldsymbol{x}^\star$ such that the objective value achieves zero through increasing k from 0 to M. Note that the rank one constraint of matrix $\boldsymbol{M}$ shall be satisfied when the objective value equals zero with $\mathrm{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 = 0$.

2) Step II: Feasibility Detection: The solution $\boldsymbol{x}$ obtained in the first step characterizes the gap between the MSE requirement and the achievable MSE for each device. Therefore, in the second step, we propose to select device k with higher priority if x_k is small. The elements of $\boldsymbol{x}$ can be arranged in descending order $x_{\pi(1)} \geq \cdots \geq x_{\pi(M)}$. We will find the minimum k by increasing k from 1 to M such that selecting all devices in $\mathcal{S}^{[k]}$ is feasible, where the set $\mathcal{S}^{[k]}$ is chosen as $\{\pi(k), \pi(k+1), \cdots, \pi(M)\}$.

In detail, if all devices in $\mathcal{S}^{[k]}$ can be selected, the following optimization problem

$$\begin{aligned} \text{find } & \boldsymbol{m} \\ \text{subject to } & \|\boldsymbol{m}\|^2 - \gamma_i |\boldsymbol{m}^{\mathsf{H}} \boldsymbol{h}_i|^2 \leq 0, \quad \forall i \in \mathcal{S}^{[k]} \\ & \|\boldsymbol{m}\|^2 \geq 1 \end{aligned} \tag{26}$$

should be feasible. It can be equivalently reformulated as

$$\begin{aligned} \text{find } & \boldsymbol{M} \\ \text{subject to } & \mathrm{Tr}(\boldsymbol{M}) - \gamma_i \boldsymbol{h}_i^{\mathsf{H}} \boldsymbol{M} \boldsymbol{h}_i \leq 0, \quad \forall i \in \mathcal{S}^{[k]} \\ & \boldsymbol{M} \succeq \boldsymbol{0}, \mathrm{Tr}(\boldsymbol{M}) \geq 1, \mathrm{rank}(\boldsymbol{M}) = 1 \end{aligned} \tag{27}$$

Algorithm 2: DC Representation Framework for Solving Problem $\mathscr{P}$ in Federated Learning With Device Selection

Step 1: sparsity inducing
$k \leftarrow 0$
while *objective value of* $\mathscr{P}_{S1}$ *is not zero* **do**
  Obtain solution $\boldsymbol{x}$ by solving the DC program $\mathscr{P}_{\text{S1}}$
  $k \leftarrow k+1$
end
Step 2: feasibility detection
Order $\boldsymbol{x}$ in descending order as $x_{\pi(1)} \geq \cdots \geq x_{\pi(M)}$
$k \leftarrow 1$
while *objective value of* $\mathscr{P}_{S2}$ *is not zero* **do**
  $\mathcal{S}^{[k]} \leftarrow \{\pi(k), \pi(k+1), \cdots, \pi(M)\}$
  Obtain solution $\boldsymbol{M}$ by solving the DC program $\mathscr{P}_{\text{S2}}$
  $k \leftarrow k+1$
end
Output: $\boldsymbol{m}$ through Cholesky decomposition $\boldsymbol{M} = \boldsymbol{m}\boldsymbol{m}^{\mathsf{H}}$, and the set of selected devices $\mathcal{S}^{[k]} = \{\pi(k), \pi(k+1), \cdots, \pi(M)\}$

using the matrix lifting technique. To guarantee the feasibility of the fixed-rank constraint for accurately detecting the feasibility of MSE constraints, we propose the following DC approach by minimizing the difference between trace norm and spectral norm:

$$\begin{aligned}\mathscr{P}_{\text{S2}}: \ \underset{\boldsymbol{M}}{\text{minimize}} \ & \text{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 \\ \text{subject to} \ & \text{Tr}(\boldsymbol{M}) - \gamma_i \boldsymbol{h}_i^{\mathsf{H}} \boldsymbol{M} \boldsymbol{h}_i \leq 0, \quad \forall i \in \mathcal{S}^{[k]} \\ & \boldsymbol{M} \succeq \boldsymbol{0}, \quad \text{Tr}(\boldsymbol{M}) \geq 1. \end{aligned} \tag{28}$$

That is, when the objective value of problem $\mathscr{P}_{\text{S2}}$ equals zero given set $\mathcal{S}^{[k]}$, we conclude that all devices in $\mathcal{S}^{[k]}$ are selected while satisfying the MSE requirement, i.e., problem (26) is feasible for $\mathcal{S}^{[k]}$. Note that the solution $\boldsymbol{M}^*$ shall be an exact rank-one matrix and a feasible receiver beamforming vector $\boldsymbol{m}$ can be obtained through Cholesky decomposition $\boldsymbol{M}^* = \boldsymbol{m}\boldsymbol{m}^{\mathsf{H}}$.

The proposed DC representation framework for solving the sparse and low-rank optimization problem in federated learning is presented in Algorithm 2. Since the DC program is still nonconvex, in next section, we will develop the DC Algorithm (DC) [49] for the DC optimization problem $\mathscr{P}_{\text{S1}}$ and problem $\mathscr{P}_{\text{S2}}$. We further contribute by establishing the convergence rate of DC algorithm. Due to the superiority of the presented DC representation (24) for rank-one constraint, our proposed DC approach for accurate feasibility detection considerably outperforms the SDR approach [31] by simply dropping the rank-one constraint, which will be demonstrated through numerical experiments in Section V.

V. DC Algorithm for DC Program With Convergence Guarantees

In this section, the DC Algorithm will be developed by successively solving the convex relaxation of primal problem and dual problem of DC program. To further establish the convergence results, we add quadratic terms in convex functions while their difference (i.e., the objective value) remains unchanged. With this technique, we represent the DC objective function as the difference of strongly convex functions, which allows us establish the convergence rate of the DC algorithm.

A. Difference-of-Strongly-Convex-Functions Representation

The DC formulations $\mathscr{P}_{\text{S1}}$ and $\mathscr{P}_{\text{S2}}$ for sparse and low-rank optimization are nonconvex programs with DC objective functions and convex constraints. Although DC functions are nonconvex, they have good problem structures and the DC Algorithm can be developed based on the principles provided in [49]. In order to establish the convergence result of the DC algorithm, we will represent the DC objective function as the difference of strongly convex functions.

Specifically, we can equivalently rewrite problem $\mathscr{P}_{\text{S1}}$ as

$$\underset{\boldsymbol{x},\boldsymbol{M}}{\text{minimize}} \ \ f_1 = \|\boldsymbol{x}\|_1 - \|\!|\boldsymbol{x}|\!\|_k + \text{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 + I_{\mathcal{C}_1}(\boldsymbol{x}, \boldsymbol{M}), \tag{29}$$

and problem $\mathscr{P}_{\text{S2}}$ as

$$\underset{\boldsymbol{M}}{\text{minimize}} \ \ f_2 = \text{Tr}(\boldsymbol{M}) - \|\boldsymbol{M}\|_2 + I_{\mathcal{C}_2}(\boldsymbol{M}), \tag{30}$$

respectively. Here $\mathcal{C}_1, \mathcal{C}_2$ are positive semidefinite cones that integrates the constraints of problem $\mathscr{P}_{\text{S1}}$ and problem $\mathscr{P}_{\text{S2}}$, and the indicator function is defined as

$$I_{\mathcal{C}_1}(\boldsymbol{x}, \boldsymbol{M}) = \begin{cases} 0, & (\boldsymbol{x}, \boldsymbol{M}) \in \mathcal{C}_1 \\ +\infty, & \text{otherwise.} \end{cases} \tag{31}$$

In order to establish the convergence result of the DC algorithm, we rewrite the DC functions f_1, f_2 as the difference of *strongly* convex functions, i.e., $f_1 = g_1 - h_1$ and $f_2 = g_2 - h_2$, where

$$g_1 = \|\boldsymbol{x}\|_1 + \text{Tr}(\boldsymbol{M}) + I_{\mathcal{C}_1}(\boldsymbol{x}, \boldsymbol{M}) + \frac{\alpha}{2}(\|\boldsymbol{x}\|_F^2 + \|\boldsymbol{M}\|_F^2), \tag{32}$$

$$h_1 = \|\!|\boldsymbol{x}|\!\|_k + \|\boldsymbol{M}\|_2 + \frac{\alpha}{2}(\|\boldsymbol{x}\|_F^2 + \|\boldsymbol{M}\|_F^2), \tag{33}$$

$$g_2 = \text{Tr}(\boldsymbol{M}) + I_{\mathcal{C}_2}(\boldsymbol{M}) + \frac{\alpha}{2}\|\boldsymbol{M}\|_F^2, \tag{34}$$

$$h_2 = \|\boldsymbol{M}\|_2 + \frac{\alpha}{2}\|\boldsymbol{M}\|_F^2. \tag{35}$$

By adding quadratic terms, g_1, g_2, h_1, h_2 are all α-strongly convex functions. Then problem (29) and problem (30) admit the uniform structure of minimizing the difference of two strongly convex functions

$$\underset{\boldsymbol{X} \in \mathbb{C}^{m \times n}}{\text{minimize}} \quad f(\boldsymbol{X}) = g(\boldsymbol{X}) - h(\boldsymbol{X}). \tag{36}$$

For complex domain $\boldsymbol{X}$, we shall apply Wirtinger calculus [50] for algorithm design. The DC algorithm is given by constructing sequences of candidates to primal solutions and dual solutions. Since the primal problem (36) and its dual problem are still nonconvex, convex relaxation is further needed.

B. DC Algorithm for Sparse and Low-Rank Optimization

According to the Fenchel's duality [51], the dual problem of problem (36) is given by

$$\underset{\boldsymbol{Y}\in\mathbb{C}^{m\times n}}{\text{minimize}} \quad h^*(\boldsymbol{Y}) - g^*(\boldsymbol{Y}), \tag{37}$$

where g^* and h^* are the conjugate functions of g and h, respectively. The conjugate function is defined as

$$g^*(\boldsymbol{Y}) = \sup_{\boldsymbol{X}\in\mathbb{C}^{m\times n}} \langle \boldsymbol{X}, \boldsymbol{Y}\rangle - g(\boldsymbol{X}), \tag{38}$$

where $\langle \boldsymbol{X}, \boldsymbol{Y}\rangle = \mathrm{Real}\big(\mathrm{Tr}(\mathrm{X}^{\mathrm{H}}\mathrm{Y})\big)$ defines the inner product of two matrices [50]. The t-th iteration of the simplified DC algorithm is to solve the convex approximation of primal problem and dual problem by linearizing the concave part:

$$\boldsymbol{Y}^{[t]} = \arg\inf_{\boldsymbol{Y}\in\mathcal{Y}} \; h^*(\boldsymbol{Y}) - [g^*(\boldsymbol{Y}^{[t-1]}) + \langle \boldsymbol{Y} - \boldsymbol{Y}^{[t-1]}, \boldsymbol{X}^{[t]}\rangle], \tag{39}$$

$$\boldsymbol{X}^{[t+1]} = \arg\inf_{\boldsymbol{X}\in\mathcal{X}} \; g(\boldsymbol{X}) - [h(\boldsymbol{X}^{[t]}) + \langle \boldsymbol{X} - \boldsymbol{X}^{[t]}, \boldsymbol{Y}^{[t]}\rangle]. \tag{40}$$

According to the Fenchel biconjugation theorem [51], equation (39) can be rewritten as

$$\boldsymbol{Y}^{[t]} \in \partial_{\boldsymbol{X}^{[t]}} h, \tag{41}$$

$\partial_{\boldsymbol{X}^{[t]}} h$ is the subgradient of h with respect to $\boldsymbol{X}$ at $\boldsymbol{X}^{[t]}$.

Therefore, iterations $\boldsymbol{x}^{[t]}, \boldsymbol{M}^{[t]}$ of the DC algorithm for problem $\mathscr{P}_{\mathrm{S1}}$ are constructed as the solution to the following convex optimization problem

$$\begin{aligned} \underset{\boldsymbol{x},\boldsymbol{M}}{\text{minimize}} \;& g_1 - \langle \partial_{\boldsymbol{x}^{[t-1]}} h_1, \boldsymbol{x}\rangle - \langle \partial_{\boldsymbol{M}^{[t-1]}} h_1, \boldsymbol{M}\rangle \\ \text{subject to } & \mathrm{Tr}(\boldsymbol{M}) - \gamma_i \boldsymbol{h}_i^{\mathsf{H}} \boldsymbol{M} \boldsymbol{h}_i \le x_i, \quad \forall i = 1, \cdots, M, \\ & \boldsymbol{M} \succeq \boldsymbol{0}, \quad \mathrm{Tr}(\boldsymbol{M}) \ge 1, \boldsymbol{x} \succeq \boldsymbol{0}. \end{aligned} \tag{42}$$

The iteration $\boldsymbol{M}^{[t]}$ for problem $\mathscr{P}_{\mathrm{S2}}$ is given by the solution to the following optimization problem

$$\begin{aligned} \underset{\boldsymbol{M}}{\text{minimize}} \;& g_2 - \langle \partial_{\boldsymbol{M}^{[t-1]}} h_2, \boldsymbol{M}\rangle \\ \text{subject to } & \mathrm{Tr}(\boldsymbol{M}) - \gamma_i \boldsymbol{h}_i^{\mathsf{H}} \boldsymbol{M} \boldsymbol{h}_i \le 0, \quad \forall i \in \mathcal{S}^{[k]}, \\ & \boldsymbol{M} \succeq \boldsymbol{0}, \quad \mathrm{Tr}(\boldsymbol{M}) \ge 1. \end{aligned} \tag{43}$$

The subgradient of h_1 and h_2 are given by

$$\partial_{\boldsymbol{x}} h_1 = \partial \|\boldsymbol{x}\|_k + \alpha \boldsymbol{x}, \quad \partial_{\boldsymbol{M}} h_1 = \partial_{\boldsymbol{M}} h_2 = \partial \|\boldsymbol{M}\|_2 + \alpha \boldsymbol{M}. \tag{44}$$

The subgradient of $\|\boldsymbol{x}\|_k$ can be computed by [35]

$$i\text{-th entry of } \partial \|\boldsymbol{x}\|_k = \begin{cases} \mathrm{sign}(x_i), & |x_i| \ge |x_{(k)}| \\ 0, & |x_i| < |x_{(k)}|. \end{cases} \tag{45}$$

The subgradient of $\|\boldsymbol{M}\|_2$ is given by the following proposition.

Proposition 4: The subgradient of $\|\boldsymbol{M}\|_2$ can be computed as $\boldsymbol{v}_1 \boldsymbol{v}_1^{\mathsf{H}}$, where $\boldsymbol{v}_1 \in \mathbb{C}^N$ is the eigenvector of the largest eigenvalue $\sigma_1(\boldsymbol{M})$.

Proof: The subdifferential of orthogonal invariant norm $\|\boldsymbol{M}\|_2$ for PSD matrix $\boldsymbol{M}$ is given by [52]

$$\partial \|\boldsymbol{M}\|_2 = \mathrm{conv}\{\boldsymbol{V}\mathrm{diag}(\boldsymbol{d})\boldsymbol{V}^{\mathsf{H}} : \boldsymbol{d} \in \partial \|\boldsymbol{\sigma}(\boldsymbol{M})\|_\infty\}, \tag{46}$$

where conv denotes the convex hull of a set and $\boldsymbol{M} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$ is the singular value decomposition of $\boldsymbol{M}$, and $\boldsymbol{\sigma}(\boldsymbol{M}) = [\sigma_i(\boldsymbol{M})] \in \mathbb{C}^N$ is the vector formed by all singular values of $\boldsymbol{M}$. Since $\sigma_1(\boldsymbol{M}) \ge \cdots \ge \sigma_N(\boldsymbol{M}) \ge 0$, we have

$$[1, \underbrace{0, \cdots, 0}_{N-1}]^{\mathsf{H}} \in \partial \|\boldsymbol{\sigma}(\boldsymbol{M})\|_\infty. \tag{47}$$

Therefore, one subgradient of $\|\boldsymbol{M}\|_2$ is given by $\boldsymbol{v}_1 \boldsymbol{v}_1^{\mathsf{H}}$. ■

C. Computational Complexity and Convergence Analysis

The computational cost of the proposed DC algorithm consists of solving a sequence of the DC program $\mathscr{P}_{\mathrm{S1}}$ in step I, plus solving the DC program $\mathscr{P}_{\mathrm{S2}}$ in step II. In step I, we shall solve problem $\mathscr{P}_{\mathrm{S1}}$ by increasing k from 0 to M. To address each DC program $\mathscr{P}_{\mathrm{S1}}$, the SDP problem (42) should be solved at the t-th iteration. The computational cost of solving problem (42) using the second-order interior point method [46] is $\mathcal{O}((N^2 + M)^3)$ at each iteration. In step II, problem $\mathscr{P}_{\mathrm{S2}}$ shall be addressed by iteratively solving the SDP problem (43). The computational cost of solving problem (43) using the interior point method is $\mathcal{O}(N^6)$ at each iteration. Note that the "reweighted+SDR" approach requires iteratively solving an SDP (i.e., ℓ_2-minimization problem) and the "ℓ_1+ SDR" approach only requires solving a single SDP (i.e., ℓ_1-minimization problem) in step I. The computational cost of each SDP in step I for both approaches is $\mathcal{O}((N^2 + M)^3)$ at each iteration using the interior point method. In step II, both of the "reweighted+SDR" approach and the "ℓ_1+SDR" approach requires solving a single SDP problem with complexity $\mathcal{O}(N^6)$ at each iteration using the interior point method. Thus, the proposed DC algorithm has higher computation complexity than other comparison solutions inexchange for a high-quality solution, while the "reweighted+SDR" approach is more complex than the "ℓ_1+SDR" approach.

Based on [49, Proposition 2] and [53, Proposition 1], we have provided the convergence results of the DC algorithm for problem $\mathscr{P}_{\mathrm{S1}}$ and problem $\mathscr{P}_{\mathrm{S2}}$ in the following proposition, where the metric of convergence rate is chosen following [53].

Proposition 5: The sequence $\{(\boldsymbol{M}^{[t]}, \boldsymbol{x}^{[t]})\}$ generated by iteratively solving problem (42) for problem $\mathscr{P}_{\mathrm{S1}}$ has the following properties:

(i) Any limit point of the sequence $\{(\boldsymbol{M}^{[t]}, \boldsymbol{x}^{[t]})\}$ is a critical point of f_1 (29) given arbitrary initial point, and the sequence of $\{f_1^{[t]}\}$ is strictly decreasing and convergent.

(ii) For any $t = 0, 1, \cdots$, we have

$$\mathrm{Avg}\Big(\|\boldsymbol{M}^{[t]} - \boldsymbol{M}^{[t+1]}\|_F^2\Big) \le \frac{f_1^{[0]} - f_1^\star}{\alpha(t+1)}, \tag{48}$$

$$\mathrm{Avg}\Big(\|\boldsymbol{x}^{[t]} - \boldsymbol{x}^{[t+1]}\|_2^2\Big) \le \frac{f_1^{[0]} - f_1^\star}{\alpha(t+1)}, \tag{49}$$

where $f_1^\star$ is the global minimum of f_1 and $\mathrm{Avg}\Big(\|\boldsymbol{M}^{[t]} - \boldsymbol{M}^{[t+1]}\|_F^2\Big)$ denotes the average of the sequence $\{\|\boldsymbol{M}^{[i]} - \boldsymbol{M}^{[i+1]}\|_F^2\}_{i=0}^t$.

Likewise, the sequence $\{(\boldsymbol{M}^{[t]}\}$ generated by iteratively solving problem (43) for problem $\mathscr{P}_{\text{S2}}$ has the following properties:

(iii) Any limit point of the sequence $\{\boldsymbol{M}^{[t]}\}$ is a critical point of f_2 (30) given arbitrary initial point, and the sequence of $\{f_2^{[t]}\}$ is strictly decreasing and convergent.

(iv) For any $t = 0, 1, \cdots$, we have

$$\text{Avg}\Big(\|\boldsymbol{M}^{[t]} - \boldsymbol{M}^{[t+1]}\|_F^2\Big) \leq \frac{f_2^{[0]} - f_2^{\star}}{\alpha(t+1)}. \tag{50}$$

where $f_2^{\star}$ is the global minimum of f_2.

Proof: Please refer to Appendix B for details. ■

VI. Simulation Results

In this section, we conduct numerical experiments to compare the proposed DC method with state-of-the-art approaches for federated learning with device selection. The channel coefficient vectors $\boldsymbol{h}_i$'s between the BS and each mobile device follow the i.i.d. complex normal distribution, i.e., $\boldsymbol{h}_i \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$. The average transmit signal-to-noise-ratio (SNR) P_0/σ^2 is chosen as 20 dB. We assume that all devices have the same number of data points, i.e., $|\mathcal{D}_1| = \cdots = |\mathcal{D}_M|$, for which the pre-processing post-processing pair can be chosen as $\phi_i = 1, \psi = 1/|\mathcal{S}|$.

A. Feasibility Detection

Consider a typical Internet of Things (IoT) network setting with $M = 20$ active mobile devices for federated learning. The BS is equipped with $N = 6$ antennas. Note that there are possibly a large number of devices to be connected to the Internet via one base station while only a small fraction of devices are active simultaneously due to sporadic traffic. This sporadic property of IoT data traffic can be exploited to support massive device connectivity via jointly detecting active devices and estimating channel coefficients [54]. The performance of feasibility detection, i.e., checking the feasibility of selected devices, is a critical step for the device selection. We first evaluate the convergence behavior of the proposed DC algorithm for detecting the feasibility of selecting all mobile devices, i.e., problem $\mathscr{P}_{\text{S2}}$ with $\mathcal{S}^{[k]} = \{1, \cdots, 20\}$. The results with $\gamma = 5$ dB and $\gamma = 3$ dB are shown in Fig. 4. It reveals that the objective value achieves zero for $\gamma = 5$ dB but cannot achieve zero for $\gamma = 3$ dB, which demonstrates that the proposed DC algorithm returns a rank-one solution when $\gamma = 5$ dB but fails to do the same when $\gamma = 3$ dB.

We then compare the performance of feasibility detection with the proposed DC approach by solving $\mathscr{P}_{\text{S2}}$ with the following state-of-the-art approaches:

- **SDR** [31]: Simply dropping the rank-one constraint of problem (26) yields the semidefinite relaxation (SDR) approach for the feasibility detection problem.
- **Global Optimization** [55]: In [55], a global optimization approach is proposed with exponential time complexity in the worst case. We set the relative error tolerance as $\epsilon = 10^{-5}$ and take its performance as our benchmark.

The results averaged over 500 times are shown in Fig. 5, which demonstrates that the proposed DC-based approach

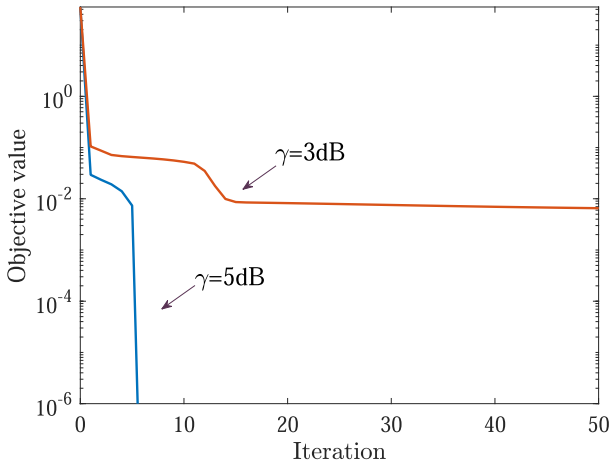


Fig. 4. Convergence of the proposed DC algorithm.

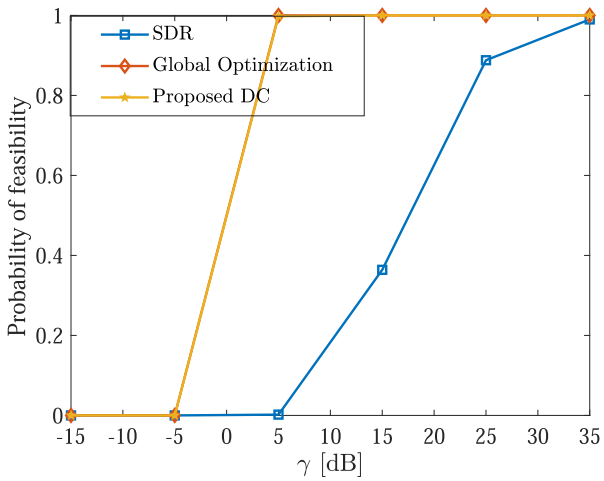


Fig. 5. Probability of feasibility with different algorithms.

outperforms SDR approach significantly and achieves the near-optimal performance compared with the global optimization approach, and thus yields accurate feasibility detection.

We further evaluate the performance of the proposed DC approach over the number of antennas. Under different target MSE requirement, the results averaged over 500 channel realizations are illustrated in Fig. 6. It demonstrates that fast aggregation from mobile devices under a more stringent MSE requirement can be accomplished by increasing the number of antennas at the BS.

B. Number of Selected Devices over Target MSE

Consider a network with 20 mobile devices and a 6-antenna BS. Under the presented two-step framework and ordering rule in Algorithm 2, we compare the proposed DC Algorithm 2 for device selection with the following state-of-the-art approaches:

- ℓ_1+**SDR** [46] [31]: The ℓ_1-norm minimization is adopted to induce the sparsity of $\boldsymbol{x}$ in Step 1, and the nonconvex quadratic constraints are addressed with SDR in Step 1 and Step 2.
- **Reweighted** ℓ_2+ **SDR** [32]: We take the smoothed ℓ_p-norm for sparsity inducing of $\boldsymbol{x}$ in Step 1, which is

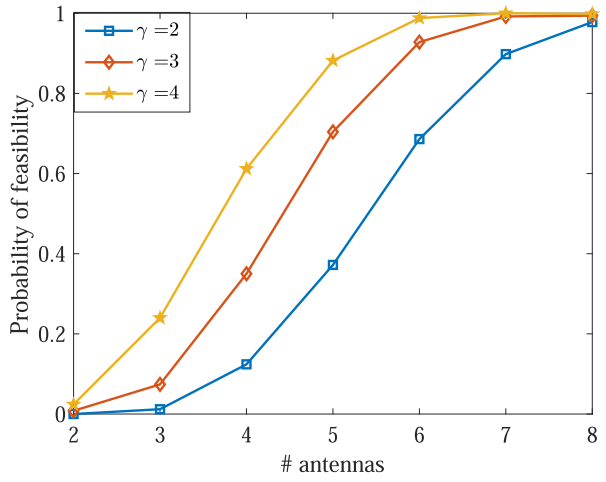


Fig. 6. Probability of feasibility over the number of BS antennas with the proposed DC approach.

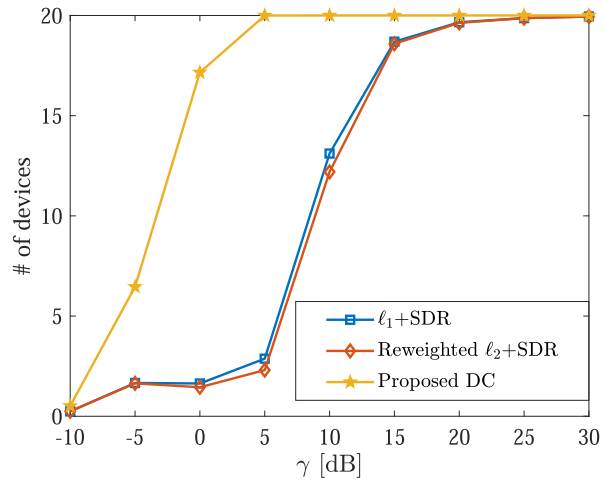
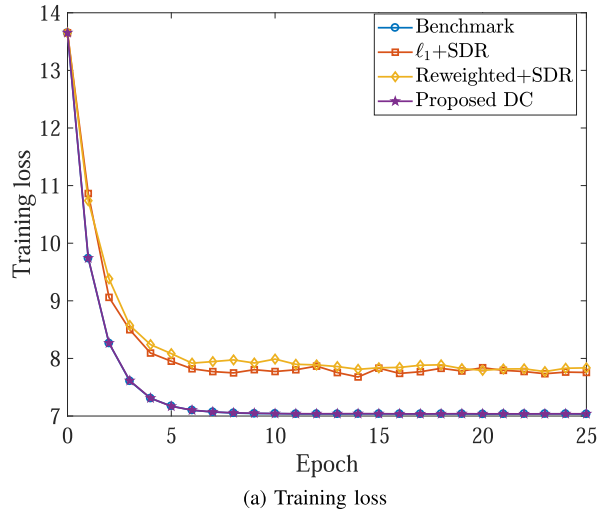
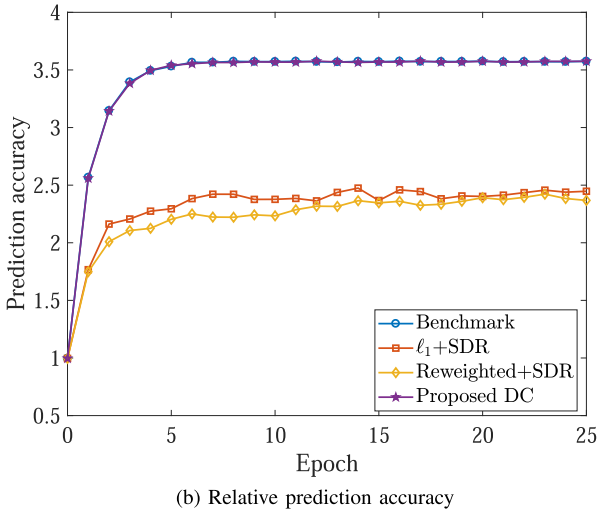


Fig. 7. Average number of selected devices with different algorithms.



(a) Training loss



(b) Relative prediction accuracy

Fig. 8. a) Convergence of different device selection algorithms for FedAvg. b) The relationship between communication rounds and test accuracy over random classification of the trained model. Each client updates its local model with stochastic gradient descent algorithm.

solved by the reweighted ℓ_2-minimization algorithm. The SDR approach is used to address the nonconvex quadratic program in Step 1 and Step 2.

The average results over 500 channel realizations with different approaches for sparsity inducing and feasibility detection are illustrated in Fig. 7. It is demonstrated that the novel sparsity and low-rankness inducing approach via the proposed DC algorithm is able to select more devices than other state-of-the-art approaches.

C. Performance of Proposed DC Approach for Distributed Federated Learning

To show the performance of the proposed DC approach for device selection in distributed federated learning, we further train a support vector machine (SVM) classifier on CIFAR-10 dataset [37] with a 6-antenna BS and 20 mobile devices. CIFAR-10 is a commonly used dataset of images for classification and contains 10 different classes of objects. The benchmark is chosen as the case where all devices are selected and all local updates are aggregated without aggregation error. We average over 10 channel realizations and the performances of all algorithms with $\gamma = 5$dB are illustrated in Fig. 8. Here we choose the size of training set and test set as 50000 and 10000, respectively. The simulation results demonstrate that the proposed DC approach achieves lower training loss and higher prediction accuracy as shown in Fig. 8a and Fig. 8b, respectively.

VII. Conclusion

In this paper, we proposed a novel fast global model aggregation approach for federated learning based on the principles of over-the-air computation. To improve the statistical learning performance for on-device distributed training, we developed a novel sparse and low-rank modeling approach to maximize the selected devices with the MSE requirements for model aggregation. We provided a unified DC representation framework to induce sparsity and low-rankness, which is supported by the convergence guaranteed DC algorithm via successive convex relaxation. Simulation results demonstrated the admirable

performance of the proposed approaches compared with the state-of-the-art algorithms.

There are still some interesting open problems on the fast model aggregation for on-device federated learning including:

- This work assumes the perfect channel state information during receiver beamforming. It would be interesting to investigate the impacts of channel uncertainty in model aggregation.
- The security issues are also critical for model aggregation, though it is beyond the scope of this paper. It is also interesting to propose a robust approach against the malicious attacks during model aggregation.
- The proposed DC approach for feasibility detection has comparable performance with the global optimization approach through numerical experiments. But it remains challenging to characterize its optimality conditions of the DC approach.
- It is interesting to further reduce the computational complexity of the proposed DC algorithm.

APPENDIX A
PROOF OF PROPOSITION 1

The sequence $\{b_i\}$ given by Proposition 1 has the zero-forcing structure which enforces

$$\sum_{i\in\mathcal{S}}\left|\boldsymbol{m}^{\mathsf{H}}\boldsymbol{h}_i b_i-\phi_i\right|^2=0. \tag{51}$$

In addition, the MSE satisfies

$$\mathsf{MSE}(\hat{g},g)\geq\sigma^2\|\boldsymbol{m}\|^2. \tag{52}$$

Therefore, the MSE is minimized by the zero-forcing transmitter beamforming vectors $\{b_i\}$'s given in Proposition 1.

APPENDIX B
PROOF OF PROPOSITION 5

Without loss of generality, we shall only present the proof of properties (i) and (ii), while properties (iii) and (iv) can be proved with the same merit. For the sequence $\{(\boldsymbol{M}^{[t]},\boldsymbol{x}^{[t]})\}$ generated by iteratively solving problem (42), we denote the dual variables as $\boldsymbol{Y}_M^{[t]}\in\partial_{\boldsymbol{M}^{[t]}}h_1,\boldsymbol{Y}_x^{[t]}\in\partial_{\boldsymbol{x}^{[t]}}h_1$. Due to the strong convexity of h_1, we have

$$\begin{aligned}h_1^{[t+1]}-h_1^{[t]}\geq\langle\Delta_t\boldsymbol{M},\boldsymbol{Y}_M^{[t]}+\langle\Delta_t\boldsymbol{x},\boldsymbol{Y}_x^{[t]}\rangle\rangle\\+\frac{\alpha}{2}\big(\|\Delta_t\boldsymbol{M}\|_F^2+\|\Delta_t\boldsymbol{x}\|_2^2\big),\end{aligned} \tag{53}$$

$$\langle\boldsymbol{M}^{[t]},\boldsymbol{Y}_M^{[t]}\rangle+\langle\boldsymbol{x}^{[t]},\boldsymbol{Y}_x^{[t]}\rangle=h_1^{[t]}+h_1^{\star[t]}, \tag{54}$$

where $\Delta_t\boldsymbol{M}=\boldsymbol{M}^{[t+1]}-\boldsymbol{M}^{[t]}$ and $\Delta_t\boldsymbol{x}=\boldsymbol{x}^{[t+1]}-\boldsymbol{x}^{[t]}$. Adding $g_1^{[t+1]}$ at both sides of (53), we obtain that

$$\begin{aligned}f_1^{[t+1]}\leq g_1^{[t+1]}-h_1^{[t]}-\langle\Delta_t\boldsymbol{M},\boldsymbol{Y}_M^{[t]}\rangle+\langle\Delta_t\boldsymbol{x},\boldsymbol{Y}_x^{[t]}\rangle\\-\frac{\alpha}{2}\big(\|\Delta_t\boldsymbol{M}\|_F^2+\|\Delta_t\boldsymbol{x}\|_2^2\big).\end{aligned} \tag{55}$$

For the update of primal variable $\boldsymbol{M}$ and $\boldsymbol{x}$ according to equation (40), we have $\boldsymbol{Y}_M^{[t]}\in\partial_{\boldsymbol{M}^{[t+1]}}g_1,\boldsymbol{Y}_x^{[t]}\in\partial_{\boldsymbol{x}^{[t+1]}}g_1$. This implies that

$$\begin{aligned}g_1^{[t]}-g_1^{[t+1]}\geq\langle-\Delta_t\boldsymbol{M},\boldsymbol{Y}_M^{[t]}\rangle+\langle-\Delta_t\boldsymbol{x},\boldsymbol{Y}_x^{[t]}\rangle\\+\frac{\alpha}{2}\big(\|\Delta_t\boldsymbol{M}\|_F^2+\|\Delta_t\boldsymbol{x}\|_2^2\big),\end{aligned} \tag{56}$$

$$\langle\boldsymbol{M}^{[t+1]},\boldsymbol{Y}_M^{[t]}\rangle+\langle\boldsymbol{x}^{[t+1]},\boldsymbol{Y}_x^{[t]}\rangle=g_1^{[t+1]}+g_1^{\star[t]}. \tag{57}$$

Similarly, by adding $-h_1^{[t]}$ at both sides of equation (56), we have

$$\begin{aligned}f_1^{[t]}\geq g_1^{[t+1]}-h_1^{[t]}+\langle-\Delta_t\boldsymbol{M},\boldsymbol{Y}_M^{[t]}\rangle+\langle-\Delta_t\boldsymbol{x},\boldsymbol{Y}_x^{[t]}\rangle\\+\frac{\alpha}{2}\big(\|\Delta_t\boldsymbol{M}\|_F^2+\|\Delta_t\boldsymbol{x}\|_2^2\big).\end{aligned} \tag{58}$$

From equation (54) and equation (57), we deduce that

$$g_1^{[t+1]}-h_1^{[t]}+\langle-\Delta_t\boldsymbol{M},\boldsymbol{Y}_M^{[t]}\rangle+\langle-\Delta_t\boldsymbol{x},\boldsymbol{Y}_x^{[t]}\rangle=f_1^{\star[t]}, \tag{59}$$

where $f_1^\star=h_1^\star-g_1^\star$. Combining equation (55), (58) and (59), it is derived that

$$\begin{aligned}f_1^{[t]}&\geq f_1^{\star[t]}+\frac{\alpha}{2}\big(\|\Delta_t\boldsymbol{M}\|_F^2+\|\Delta_t\boldsymbol{x}\|_2^2\big)\\&\geq f_1^{[t+1]}+\alpha\big(\|\Delta_t\boldsymbol{M}\|_F^2+\|\Delta_t\boldsymbol{x}\|_2^2\big).\end{aligned} \tag{60}$$

Then the sequence $\{f_1^{[t]}\}$ is non-increasing. Since $f_1\geq 0$ always holds, we conclude that the sequence $\{f_1^{[t]}\}$ is strictly decreasing until convergence, and we have

$$\begin{aligned}0&\leq\lim_{t\to\infty}\big(\|\boldsymbol{M}^{[t]}-\boldsymbol{M}^{[t+1]}\|_F^2+\|\boldsymbol{x}^{[t]}-\boldsymbol{x}^{[t+1]}\|_2^2\big)\\&\leq\lim_{t\to\infty}(f_1^{[t]}-f_1^{\star[t]})=0.\end{aligned} \tag{61}$$

For every limit point, $f_1^{[t+1]}=f_1^{[t]}$, we have

$$\begin{aligned}\|\boldsymbol{M}^{[t]}-\boldsymbol{M}^{[t+1]}\|_F^2=0,\quad\|\boldsymbol{x}^{[t]}-\boldsymbol{x}^{[t+1]}\|_2^2=0,\\f^{[t+1]}=f^{\star[t]}=f^{[t]}.\end{aligned} \tag{62}$$

Then it is followed by

$$\begin{aligned}h^{\star[t]}+h^{[t+1]}&=g^{[t]}+g^{[t+1]}\\&=\langle\boldsymbol{M}^{[t+1]},\boldsymbol{Y}_M^{[t]}\rangle+\langle\boldsymbol{x}^{[t+1]},\boldsymbol{Y}_x^{[t]}\rangle,\end{aligned} \tag{63}$$

i.e.,

$$\boldsymbol{Y}_M^{[t]}\in\partial_{\boldsymbol{M}^{[t+1]}}h_1,\boldsymbol{Y}_x^{[t]}\in\partial_{\boldsymbol{x}^{[t+1]}}h_1. \tag{64}$$

Therefore, $\boldsymbol{Y}_M^{[t]}\in\partial_{\boldsymbol{M}^{[t+1]}}g_1\cap\partial_{\boldsymbol{M}^{[t+1]}}h_1,\boldsymbol{Y}_x^{[t]}\in\partial_{\boldsymbol{x}^{[t+1]}}g_1\cap\partial_{\boldsymbol{x}^{[t+1]}}h_1$. It is concluded that $(\boldsymbol{M}^{[t+1]},\boldsymbol{x}^{[t+1]})$ is a critical point of $f_1=g_1-h_1$.

In addition, since

$$\begin{aligned}&\mathrm{Avg}\Big(\|\boldsymbol{M}^{[t]}-\boldsymbol{M}^{[t+1]}\|_F^2+\|\boldsymbol{x}^{[t]}-\boldsymbol{x}^{[t+1]}\|_2^2\Big)\\&\leq\sum_{i=0}^{t}\frac{1}{\alpha(t+1)}(f_1^{[i]}-f_1^{[i+1]})\qquad(65)\\&\leq\frac{1}{\alpha(t+1)}(f_1^{[0]}-f_1^{[t+1]})\qquad(66)\\&\leq\frac{1}{\alpha(t+1)}(f_1^{[0]}-f_1^\star),\qquad(67)\end{aligned}$$

we conclude that property (ii) holds, i.e.,

$$\mathrm{Avg}\Big(\|\boldsymbol{M}^{[t]}-\boldsymbol{M}^{[t+1]}\|_F^2\Big)\leq\frac{f_1^{[0]}-f_1^\star}{\alpha(t+1)}, \tag{68}$$

$$\mathrm{Avg}\Big(\|\boldsymbol{x}^{[t]}-\boldsymbol{x}^{[t+1]}\|_2^2\Big)\leq\frac{f_1^{[0]}-f_1^\star}{\alpha(t+1)}. \tag{69}$$

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] I. Stoica *et al.*, "A berkeley view of systems challenges for AI," 2017, *arXiv:1712.05855*. [Online]. Available: https://arxiv.org/abs/1712.05855

[3] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Towards an intelligent edge: Wireless communication meets machine learning," 2018, *arXiv:1809.00343*. [Online]. Available: https://arxiv.org/abs/1809.00343

[4] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[5] T. Q. Dinh, Q. D. La, T. Q. Quek, and H. Shin, "Learning for computation offloading in mobile edge computing," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6353–6367, Dec. 2018.

[6] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, Aug. 2017.

[7] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016, pp. 1–14.

[8] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 126–136, Jan. 2018.

[9] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–13.

[10] S. Wang *et al.*, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2018, pp. 63–71.

[11] C. Karakus, Y. Sun, S. Diggavi, and W. Yin, "Straggler mitigation in distributed optimization through data encoding," in *Proc. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5434–5442.

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artif. Intell. Stat. (AISTATS)*, vol. 54, 2017, pp. 1273–1282.

[13] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4424–4434.

[14] K. Bonawitz *et al.*, "Towards federated learning at scale: System design," 2019, *arXiv:1902.01046*. [Online]. Available: https://arxiv.org/abs/1902.01046

[15] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, Jan. 2019.

[16] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2019, pp. 1387–1395.

[17] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," 2018, *arXiv:1806.00582*. [Online]. Available: https://arxiv.org/abs/1806.00582

[18] S. Wang, F. Roosta, P. Xu, and M. W. Mahoney, "GIANT: Globally improved approximate Newton method for distributed optimization," in *Proc. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 1–11.

[19] S. Li, S. M. Mousavi Kalan, A. S. Avestimehr, and M. Soltanolkotabi, "Near-optimal straggler mitigation for distributed gradient methods," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. Workshops (IPDPSW)*, May 2018, pp. 857–866.

[20] P. Blanchard *et al.*, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 119–129.

[21] Y. Chen, L. Su, and J. Xu, "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent," *ACM Meas. Anal. Comput. Syst.*, vol. 1, no. 2, p. 44, 2017.

[22] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3498–3516, Oct. 2007.

[23] M. Goldenbaum, H. Boche, and S. Stanczak, "Harnessing interference for analog function computation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 61, no. 20, pp. 4893–4906, Oct. 2013.

[24] L. Chen, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, "Over-the-air computation for IoT networks: Computing multiple functions with antenna arrays," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 5296–5306, Dec. 2018.

[25] G. Zhu and K. Huang, "MIMO over-the-air computation for high-mobility multi-modal sensing," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6089–6103, Aug. 2019.

[26] F. Ang, L. Chen, N. Zhao, Y. Chen, and F. R. Yu, "Robust design for massive CSI acquisition in analog function computation networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2361–2373, Mar. 2019.

[27] O. Abari, H. Rahul, D. Katabi, and M. Pant, "AirShare: Distributed coherent transmission made seamless," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 1742–1750.

[28] M. Goldenbaum and S. Stanczak, "Robust analog function computation via wireless multiple-access channels," *IEEE Trans. Commun.*, vol. 61, no. 9, pp. 3863–3877, Sep. 2013.

[29] J. Wang and G. Joshi, "Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms," 2018, *arXiv:1808.07576*. [Online]. Available: https://arxiv.org/abs/1808.07576

[30] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, to be published.

[31] Z. Luo, N. D. Sidiropoulos, P. Tseng, and S. Zhang, "Approximation bounds for quadratic optimization with homogeneous quadratic constraints," *SIAM J. Optim.*, vol. 18, no. 1, pp. 1–28, Jan. 2007.

[32] Y. Shi, J. Cheng, J. Zhang, B. Bai, W. Chen, and K. B. Letaief, "Smoothed-minimization for green cloud-RAN with user admission control," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 1022–1036, Apr. 2016.

[33] H. Wang, F. Zhang, Q. Wu, Y. Hu, and Y. Shi, "Nonconvex and non-smooth sparse optimization via adaptively iterative reweighted methods," 2018, *arXiv:1810.10167*. [Online]. Available: https://arxiv.org/abs/1810.10167

[34] L. Chen, X. Qin, and G. Wei, "A uniform-forcing transceiver design for over-the-air function computation," *IEEE Wireless Commun. Lett.*, vol. 7, no. 6, pp. 942–945, Dec. 2018.

[35] J.-Y. Gotoh, A. Takeda, and K. Tono, "DC formulations and algorithms for sparse optimization problems," *Math. Program.*, vol. 169, no. 1, pp. 141–176, May 2018.

[36] K. Fan, "Maximum properties and inequalities for the eigenvalues of completely continuous operators," *Proc. Nat. Acad. Sci. USA*, vol. 37, no. 11, pp. 760–766, Nov. 1951.

[37] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep. 4, 2009.

[38] T. L. Marzetta, E. G. Larsson, H. Yang, and H. Q. Ngo, *Fundamentals of Massive MIMO*. Cambridge, U.K.: Cambridge Univ. Press, 2016.

[39] G. Caire, N. Jindal, M. Kobayashi, and N. Ravindran, "Multiuser MIMO achievable rates with downlink training and channel state feedback," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2845–2866, Jun. 2010.

[40] N. Sidiropoulos, T. Davidson, and Z.-Q. Luo, "Transmit beamforming for physical-layer multicasting," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 2239–2251, Jun. 2006.

[41] Y. Shi, J. Zhang, and K. B. Letaief, "Group sparse beamforming for green cloud-RAN," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2809–2823, May 2014.

[42] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. IEEE*, vol. 98, no. 6, pp. 948–958, Jun. 2010.

[43] 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.

[44] Y. Shi, J. Zhang, and K. B. Letaief, "Low-rank matrix completion for topological interference management by Riemannian pursuit," *IEEE Trans. Wireless Commun.*, vol. 15, vol. 7, pp. 4703–4717, Jul. 2016.

[45] Y. Shi, J. Zhang, W. Chen, and K. B. Letaief, "Generalized sparse and low-rank optimization for ultra-dense networks," *IEEE Commun. Mag.*, vol. 56, no. 6, pp. 42–48, Jun. 2018.

[46] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[47] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2008, pp. 3869–3872.

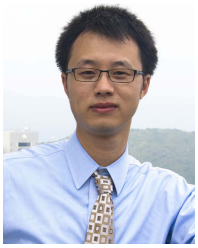
[48] E. Chen and M. Tao, "ADMM-based fast algorithm for multi-group multicast beamforming in large-scale wireless systems," *IEEE Trans. Commun.*, vol. 65, no. 6, pp. 2685–2698, Jun. 2017.

[49] P. D. Tao and L. T. H. An, "Convex analysis approach to DC programming: Theory, algorithms and applications," *Acta Math. Vietnamica*, vol. 22, no. 1, pp. 289–355, 1997.

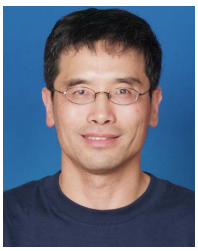
[50] P. Bouboulis, K. Slavakis, and S. Theodoridis, "Adaptive learning in complex reproducing kernel Hilbert spaces employing Wirtinger's subgradients," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 425–438, Mar. 2012.
[51] R. T. Rockafellar, *Convex Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 2015.
[52] G. Watson, "Characterization of the subdifferential of some matrix norms," *Linear Algebra Appl.*, vol. 170, pp. 33–45, Jun. 1992.
[53] K. Khamaru and M. J. Wainwright, "Convergence guarantees for a class of non-convex and non-smooth optimization problems," *J. Mach. Learn. Res.*, vol. 20, no. 154, pp. 1–52, 2019.
[54] T. Jiang, Y. Shi, J. Zhang, and K. B. Letaief, "Joint activity detection and channel estimation for IoT networks: Phase transition and computation-estimation tradeoff," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6212–6225, Aug. 2019.
[55] C. Lu and Y.-F. Liu, "An efficient global algorithm for single-group multicast beamforming," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3761–3774, Jul. 2017.

Kai Yang (Student Member, IEEE) received the B.S. degree in electronic engineering from the Dalian University of Technology, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, also with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, and also with the University of Chinese Academy of Sciences, Beijing, China. His research interests include big data processing, mobile edge/fog computing, mobile edge artificial intelligence, and dense communication networking.

Tao Jiang (Student Member, IEEE) received the B.S. degree in communication engineering from Xidian University, Xi'an, China, in 2017. He is currently pursuing the master's degree with the School of Information Science and Technology, ShanghaiTech University. His research interests include high-dimensional structured estimation and on device distributed learning.



Yuanming Shi (Member, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 2011, and the Ph.D. degree in electronic and computer engineering from The Hong Kong University of Science and Technology (HKUST), in 2015. Since September 2015, he has been with the School of Information Science and Technology, ShanghaiTech University, where he is currently a tenured Associate Professor. He visited the University of California, Berkeley, CA, USA, from October 2016 to February 2017. His research areas include optimization, statistics, machine learning, signal processing, and their applications to 6G, IoT, AI, and FinTech. He was a recipient of the 2016 IEEE Marconi Prize Paper Award in Wireless Communications and the 2016 Young Author Best Paper Award by the IEEE Signal Processing Society. He is an Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.



Zhi Ding (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Cornell University, Ithaca, NY, USA, in 1990. From 1990 to 2000, he was a faculty member of Auburn University, Auburn, AL, USA, and later The University of Iowa, Iowa City, IA, USA. He has held visiting positions in Australian National University, The Hong Kong University of Science and Technology, NASA Lewis Research Center, and USAF Wright Laboratory. He is currently a Professor of electrical and computer engineering with the University of California at Davis, Davis, CA USA. He has active collaboration with researchers from Australia, Canada, China, Finland, Hong Kong, Japan, Korea, Singapore, and Taiwan. He is a coauthor of the text *Modern Digital and Analog Communication Systems* (Oxford University Press, 2019).

He has been an active member of IEEE, serving on technical programs of several workshops and conferences. He was a member of technical committee on Statistical Signal and Array Processing and a member of technical committee on Signal Processing for Communications, from 1994 to 2003. He was the General Chair of the 2016 IEEE International Conference on Acoustics, Speech, and Signal Processing and the Technical Program Chair of the 2006 IEEE Globecom. He was also an IEEE Distinguished Lecturer (Circuits and Systems Society, from 2004 to 2006, and Communications Society, from 2008 to 2009). He served on the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS as a Steering Committee Member, from 2007 to 2009, and its Chair, from 2009 to 2010. He was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, from 1994 to 1997 and from 2001 to 2004, and an Associate Editor of the IEEE SIGNAL PROCESSING LETTERS, from 2002 to 2005.