IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 31 May 2023, accepted 21 June 2023, date of publication 30 June 2023, date of current version 7 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3291023

RESEARCH ARTICLE

Subgradient Descent Learning Over Fading Multiple Access Channels With Over-the-Air Computation

TAMIR L. S. GEZ AND KOBI COHEN, (Senior Member, IEEE)
School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Be'er Sheva 8410501, Israel
Corresponding author: Kobi Cohen (yakovsec@bgu.ac.il)

This research was supported by the Israel Science Foundation under Grant 2640/20.

ABSTRACT We focus on a distributed learning problem in a communication network, consisting of N distributed nodes and a central parameter server (PS). The PS is responsible for performing the computation based on data received from the nodes, which are transmitted over a multiple access channel (MAC). The objective function for this problem is the sum of the local loss functions of the nodes. This problem has gained attention in the field of distributed sensing systems, as well as in the area of federated learning (FL) recently. However, current approaches to solving this problem rely on the assumption that the loss functions are continuously differentiable. In this paper, we first address the case where this assumption does not hold. We develop a novel algorithm called Sub-Gradient descent Multiple Access (SGMA) to solve the learning problem over MAC. SGMA involves each node transmitting an analog shaped waveform of its local subgradient over MAC, and the PS receiving a superposition of the noisy analog signals, resulting in a bandwidth-efficient over-the-air (OTA) computation used to update the learned model. We analyze the performance of SGMA and prove that it has a convergence rate that approaches that of the centralized subgradient algorithm in large networks. Simulation results using real datasets show the effectiveness of SGMA.

INDEX TERMS Distributed learning, gradient descent (GD)-type learning, subgradient methods, federated learning (FL), multiple access channel (MAC), over-the-air (OTA) computation.

I. INTRODUCTION

We consider a distributed learning problem in a communication network, which consists of many distributed edge nodes and a central parameter server (PS). The objective of the PS is to solve the following optimization problem:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\Theta} \frac{1}{N}\sum_{n=1}^{N} f_n(\boldsymbol{\theta}) \qquad (1)$$

based on data received from the nodes. The term $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^d$ is the $d \times 1$ parameter vector which needs to be optimized. The solution $\boldsymbol{\theta}^*$ is known as the empirical risk minimizer. In machine learning (ML) tasks, we often write $f_n(\boldsymbol{\theta}) = \ell(\boldsymbol{x}_n, y_n; \boldsymbol{\theta})$, which is the loss of the prediction on the input-output data pair sample $(\boldsymbol{x}_n, y_n)$, where $\boldsymbol{x}_n$ refers to the input vector and y_n refers to the label, made with model parameter $\boldsymbol{\theta}$. Traditional ML algorithms solve (1) in a centralized manner. The traditional approach to machine learning involves storing all of the data in a central location and using a centralized optimization algorithm, such as gradient descent, to process the data. However, this method can be inefficient for data-intensive applications due to the high storage and latency requirements. FL is a collaborative machine learning framework that addresses these issues by allowing distributed nodes to process and share a function of their locally-held data with a central PS without the need to upload the entire dataset. This approach is particularly well-suited for mobile applications, such as those found in 5G, IoT, and cognitive radio systems, where communication resources are limited [2], [3], [4], [5], and due to privacy concerns

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Moinul Hossain.

[6], [7]. Therefore, FL has emerged as a promising solution to the challenges faced by traditional centralized ML algorithms, and has received increasing attention in recent years. In FL, the training process is distributed across a number of nodes, each of which is associated with a local loss function or gradient. These nodes communicate with a central PS to solve the optimization problem (1), sending their local output to the PS, which aggregates the data and updates the global model. The updated model is then transmitted back to the nodes, and the process repeats. This approach has potential applications in distributed sensing and control systems as well, and has been the subject of extensive related research (see related work in Section I-C).

A. LEARNING WITH OTA COMPUTATION

In traditional communication schemes, data signals are transmitted over separate, orthogonal channels (such as FDM or TDM communications), with each channel dedicated to a single node's transmission. Despite efforts to optimize the scheduling of node transmissions in order to reduce bandwidth and energy requirements [8], [9], [10], [11], [12], [13], [14], [15], this approach still requires a linear increase in bandwidth as the number of nodes increases, and also consumes more energy due to the additive noise in each dimension. Alternatively, nodes can transmit their data using MAC for tasks that only require the aggregation of transmitted signals (such as the sum of local gradients used to update the model at the central processing node). By exploiting the nature of the wireless channel, this approach allows for OTA aggregation of the data signals transmitted by the nodes, reducing the bandwidth requirement and eliminating the dependence on the number of nodes. The specifics of this technique have been thoroughly examined in [16].

OTA federated learning schemes present significant importance in the context of the edge/fog computing continuum. These schemes leverage the capabilities of edge and fog computing infrastructure to enable distributed ML algorithms while addressing the challenges of bandwidth, latency, and energy requirements in FL systems. Specifically, edge and fog computing aim to bring computation and processing closer to the data source, reducing latency and enabling real-time decision-making. OTA FL schemes aligns with this objective by facilitating distributed learning directly on edge devices. OTA schemes involve each node transmitting an analog shaped waveform of its data (e.g., local subgradient in this paper) over MAC, and the server receiving a superposition of the noisy analog signals, resulting in a bandwidth-efficient OTA computation used to update the learned model. This enables rapid updates and minimizes the latency associated with transmitting data to the server. By contrast to traditional orthogonal multiple access schemes (e.g., TDMA, FDMA), where the bandwidth requirement increases linearly with the number of nodes N, in OTA schemes, the bandwidth requirement is independent of N. Furthermore, traditional orthogonal multiple access schemes consume more energy due to the additive noise in each dimension. These advantages are significant for time-sensitive applications, such as autonomous vehicles, industrial IoT, and real-time surveillance.

Overall, OTA FL schemes play a significant role in extending the capabilities of edge and fog computing systems. They provide resource efficiency, low latency, and FL at the network edge. The potential applications are vast, ranging from healthcare monitoring and smart manufacturing to smart grid management and autonomous systems, where edge devices can learn and adapt with efficient resource consumption. For example, in the context of intelligent systems, smart cities, and environmental monitoring, OTA FL can enhance the efficiency and intelligence of various infrastructure and autonomous systems. Edge devices, such as traffic sensors, surveillance cameras, and environmental sensors, can learn from local data to optimize traffic management, improve public safety, monitor environmental conditions, and anomalous processes. OTA FL enables collaborative learning among these devices, with significant improvements in bandwidth and energy consumption, as well as low latency. Moreover, it has been explored in the context of fading MAC. In a recent study [17], the authors proposed a solution that tackles channel fading through the use of dynamic learning rate. Other relevant studies in this context explored the use of power control and beamforming techniques to mitigate the effects of channel fading in OTA systems [18], [19], [20], [21], [22], [23].

In this paper, we adopt a gradient-based transmission scheme over MAC as studied recently in [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], and [34] and subsequent studies. In [24], [25], [33], and [34], the authors developed the compressed analog distributed stochastic gradient descent method, in which a sparse parameter gradient vector is transmitted by the nodes over a MAC. In [33], power control was used to eliminate the fading distortion, where nodes in deep fading do not transmit to satisfy the power constraint. In [27], the fading distortion is mitigated at the receiver by using multiple antennas, where the fading diminishes as the number of antennas approaches infinity. Channel communication characteristics have been studied further in [31]. In our previous work [30], [35], we have developed and analyzed gradient-based learning, and accelerated learning methods without using power control or beamforming to cancel the fading effect. In [28], the authors developed the federated edge learning algorithm that schedules entries of the gradient vector based on the channel condition. Energy-efficiency aspects have been studied in [29]. Quantization methods of gradient transmissions were developed in [32].

B. MAIN RESULTS

Gradient-based algorithms for OTA learning, as discussed in Subsection I-A, typically rely on the assumption that the loss functions are continuously differentiable. In this paper, we first tackle the learning problem in cases where this assumption does not necessarily hold. Our main contributions are as follows. First, we propose a novel Sub-Gradient-descent Multiple Access (SGMA) algorithm that can solve the learning problem when the loss function is not necessarily differentiable. SGMA involves the transmission of analog shaped waveforms of local subgradients by each node, which can be used even when the loss function is not differentiable. Unlike some previous approaches [24], [25], [27], [28], [31], [33], [34], SGMA does not use power control or beamforming to cancel the effect of the channel gain. Instead, the central PS updates the model based on the noisy (due to additive noise) and distorted (due to channel fading) subgradients received from the nodes. Second, we provide theoretical analysis of SGMA, establishing a finite-sample error bound for both convex and strongly convex loss functions. We also develop specific design principles for the learning step and the scaling of transmission energy that allow SGMA to achieve the convergence rate of the centralized subgradient method in large networks. Specifically, for the strongly convex case, we show that the error scales as $O\left(\frac{1}{k}\right)$, where k is the number of iterations. For the convex case, we show that the error scales as $O\left(\frac{1}{\sqrt{k}}\right)$. Third, in order to evaluate the effectiveness of the proposed SGMA algorithm, we conducted simulation experiments using real datasets and compared its performance to existing methods. The simulation results demonstrate that SGMA significantly outperforms existing methods.

It should be noted that there is no requirement for nodes to have equal-sized local datasets. While heterogeneity in data sizes among nodes can impact convergence analysis in alternative schemes where nodes upload the updated mode itself, e.g., trained local neural networks (see for example our recent work [16], [36]), our focus here is on gradient-based transmissions. Specifically, in these schemes gradient-based functions are uploaded (or subgradient in our case) rather than the weights themselves. The local gradient-based functions at the nodes are computed and summed over the entire local data, allowing the aggregation at the PS to generate the desired global gradient. However, it is worth noting that in practice, it is advantageous to have relatively equal-sized local datasets among nodes for other implementation considerations. This helps ensure that computational resources and processing time are relatively equal among the nodes.

C. OTHER RELATED WORK

Earlier and more recent studies on inference tasks using MAC have typically assumed that the observation distributions are known, and have focused on model-dependent settings (see e.g., [37], [38], [39], [40], [41], [42], [43] and our previous work [44], [45], [46]). However, in the context of ML and FL tasks, which are the focus of this paper, this assumption does not hold. As a result, different algorithms and methods are required to address these tasks, and this research direction has received increasing attention in recent years, as discussed in Subsection I-A. Other aspects that have been explored in recent years in the context of OTA learning involve the use of heterogeneous data [3], [16], [36], [47] redundant data [48], the use of sub-Gaussian fading and noise distributions in OTA computation [49], digital gradient transmissions [32], relay transmissions [50], and privacy over MAC [51].

II. SYSTEM MODEL AND PROBLEM STATEMENT

Consider a network consisting of N nodes indexed by the set $\mathcal{N} = \{1, 2, \ldots, N\}$ and a PS. Each node communicates directly with the PS. The transmission time is slotted, indexed by $\{t_i\}$, $i = 1, 2, \ldots$. Each node $n \in 1, \ldots, N$ experiences at time t_k a block-fading channel $\tilde{h}_{n,k}$ with gain $h_{n,k} \triangleq |\tilde{h}_{n,k}| \in \mathbb{R}_+$ and phase $\phi_{n,k} \triangleq \angle\tilde{h}_{n,k} \in \{x \in \mathbb{R}| -\pi \leq x \leq \pi\}$. The channel fading is assumed to be i.i.d. across time and nodes, with mean μ_h and variance σ_h^2 as in [25], [28], [30], and [33]. Each node is associated with a local loss function f_n, and the objective function at the PS is to minimize the average loss (i.e., empirical risk):

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\Theta} F(\boldsymbol{\theta}), \tag{2}$$

where

$$F(\boldsymbol{\theta}) \triangleq \frac{1}{N}\sum_{n=1}^{N} f_n(\boldsymbol{\theta}). \tag{3}$$

As commonly assumed in the literature [30], [36], [52], [53], we assume for purposes of analysis convexity and strong-convexity of f_n, and a bounded expectation of the subgradient energy: $\mathbb{E}[\|\boldsymbol{\partial} f_n(\cdot)\|^2] \leq M$. Each node n is only aware of its local loss function f_n. By contrast to previous studies of OTA FL that assumed differentiable loss functions, here the loss function is not necessarily differentiable. Therefore, we develop OTA algorithm based on the subgradient $\boldsymbol{\partial} f_n(\boldsymbol{\theta})$ which always exists for all node $n \in \mathcal{N}$ and $\boldsymbol{\theta} \in \Theta$ [54].

III. THE SUB-GRADIENT-DESCENT MULTIPLE ACCESS (SGMA) ALGORITHM

In SGMA, the nodes transmit their local subgradient as an analog signal to the PS using a noisy fading MAC. The PS receives an aggregated OTA noisy distorted subgradients, updates the model based on the received signal, and broadcasts the updated model back to the nodes. This process repeats until convergence, as illustrated in Fig. 1. The subsequent sections provide a detailed exposition of the algorithmic steps and a convergence analysis of the proposed method. Since the gradient may not always exist in the present setting, we formulate and analyze SGMA using subgradient updates. Also, SGMA does not require power control or beamforming techniques to mitigate the impact of channel fading on the performance of the algorithm. Here, the PS updates the model based on the noisy distorted gradients directly. Avoiding power control or beamforming simplifies the implementation as discussed in [30].

We adopt the OTA FL transmission scheme, where $\boldsymbol{s}(t) = (s_1(t), \ldots, s_d(t))$, $0 \leq t < T$, denotes a vector of d orthogonal baseband equivalent normalized waveforms, satisfying $\int_0^T s_m^2(t)dt = 1$, $\int_0^T s_m(t)s_r(t)dt = 0$, for $m \neq r$. Prior to transmitting data signals to the PS, each node possesses

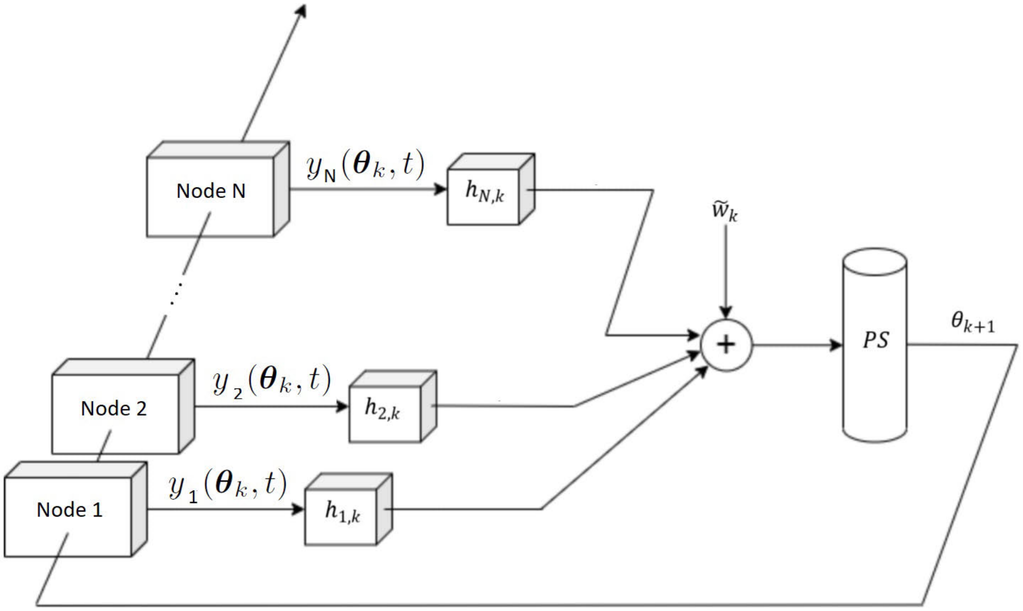

FIGURE 1. An illustration of the wireless network considered in this paper. Each node communicates directly with the PS at the network edge, which updates the global model and broadcasts the updated model back to the nodes.

knowledge of the channel state information (CSI), which is typically obtained by estimating the CSI based on a pilot signal transmitted by the PS. This assumption is commonly made in the relevant literature (as can be seen in the references cited in Subsection I-A). Note that the bandwidth requirement only depends on d but independent of N. Let $\boldsymbol{\theta}_{k+1}$ be the updated parameter model at iteration k. Let $\boldsymbol{g}_n(\boldsymbol{\theta}_k) \triangleq \boldsymbol{\partial f}_n(\boldsymbol{\theta}_k)$ denote the subgradient for node n at value $\boldsymbol{\theta}_k$. All nodes compute their local subgradient, and send an analog function of it to the PS:

$$y_n(\boldsymbol{\theta}_k, t) \triangleq \sqrt{E_N} e^{-j\phi_{n,k}} \boldsymbol{g}_n(\boldsymbol{\theta}_k)^T \boldsymbol{s}(t), \ 0 \leq t < T. \quad (4)$$

Here, E_N is the transmission energy coefficient set to satisfy the energy requirement, and $e^{-j\phi_{n,k}}$ is used to correct the phase reflection to yield coherent aggregated signals at the receiver, need to be estimated only with error of less than $\pi/2$ to have positive channel gains at the receiver as discussed in [30]. After matched filtering the received signal at the PS by the waveform $s_j(t)$ for each dimension j, we have:

$$\tilde{\boldsymbol{v}}_k \triangleq \sum_{n=1}^{N} \sqrt{E_N} h_{n,k} \boldsymbol{g}_n(\boldsymbol{\theta}_k) + \tilde{\boldsymbol{w}}_k, \quad (5)$$

where $\tilde{\boldsymbol{w}}_k$ is a zero-mean additive Gaussian noise vector, distributed as $\tilde{\boldsymbol{w}}_k \sim \mathcal{N}(\boldsymbol{0}, \sigma_w^2 \boldsymbol{I}_d)$, where $\boldsymbol{I}_d$ is the $d \times d$ identity matrix. Let us define:

$$\boldsymbol{v}_k \triangleq \frac{\tilde{\boldsymbol{v}}_k}{N\sqrt{E_N}} = \frac{1}{N} \sum_{n=1}^{N} h_{n,k} \boldsymbol{g}_n(\boldsymbol{\theta}_k) + \boldsymbol{w}_k, \quad (6)$$

where $\boldsymbol{w}_k \triangleq \frac{\tilde{\boldsymbol{w}}_k}{N\sqrt{E_N}} \sim \mathcal{N}(\boldsymbol{0}, \frac{\sigma_w^2}{N^2 E_N} \boldsymbol{I}_d)$. The PS updates the model $\boldsymbol{\theta}_{k+1}$ as follows:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha_k \boldsymbol{v}_k, \quad (7)$$

where α_k is the step size. Then, the PS broadcasts the updated model back to the nodes via an error-free channel, as commonly assumed in the OTA learning literature [24], [25], [27], [28], [29], [30], [31], [32], [33], [34], as the bandwidth requirement for the downlink transmission does not scale with N, and digital communications can be implemented [16]. The nodes set their updated model to $\boldsymbol{\theta}_{k+1}$ and start the next iteration until convergence. Note that $\boldsymbol{v}_k$ represents a noisy distorted version of the global subgradient. The effect on the convergence rate will be analyzed in Section IV. The pseudocode of the SGMA algorithm is given in Algorithm 1.

It should be noted that the uplink phase (from nodes to PS) and the downlink phase (from PS to nodes) are utilized in separate time slots due to the serial implementation of the learning task. During the uplink phase, nodes upload their data, which is then aggregated at the PS. Subsequently, the PS updates the model and broadcasts the updated model back to the nodes. This iterative process continues until convergence. Consequently, time-division duplexing (TDD) is employed to facilitate communication between these two phases over the same channel.

A. COMMUNICATION TIME AND COMPUTATIONAL COMPLEXITY

Regarding the communication time, each iteration involves both uplink transmissions (from nodes to PS) and downlink

Algorithm 1 SGMA Algorithm

```
initializing: PS broadcasts $\boldsymbol{\theta}_0$ for all nodes
for iteration $k = 0, 1, \ldots$ do
    Each node calculates local subgradient $\boldsymbol{g}_n(\boldsymbol{\theta}_k)$
    Each node transmits simultaneously a linear combination of $d$
        amplified orthogonal analog signals $y_n(z_k, t)$ according to (4)
    PS receives the aggregated signal and computes $\boldsymbol{v}_k$ according to (6)
    PS updates its estimate $\boldsymbol{\theta}_{k+1}$ according to (7)
    PS broadcasts $\boldsymbol{\theta}_{k+1}$ to the nodes
end for (until convergence)
```

transmissions (from PS to nodes). Prior to the uplink transmissions, the PS broadcasts a control signal to all nodes, allowing them to estimate the channel phase. Similar mechanisms are also required in other OTA schemes and communication schemes that involve channel phase or channel gain estimation. Also, the inclusion of uplink and downlink phases is not unique to our proposed method but is a common requirement in FL tasks using other communication schemes as well.

When it comes to broadcasting the control signal to the nodes, it requires $O(1)$ transmission time in both OTA transmission schemes and traditional digital communication schemes. This time complexity is independent of the network size N and the problem dimension d. For the downlink transmission, both OTA transmission schemes and traditional digital communication schemes require $O(d)$ transmission time.

The significant advantage of OTA schemes lies in the uplink transmissions. In OTA transmission schemes, the uplink transmission requires $O(d)$ transmission time, which is independent of the network size N. In contrast, in traditional digital communication schemes, the uplink transmission requires $O(d \cdot N)$ transmission time due to the linear scaling of the bandwidth requirement with N.

In terms of computational complexity, at the node (say node i), computing the subgradient requires $O(d \cdot n_i)$ computations, where n_i is the data size at node i. Updating the model at the PS requires $O(d)$ computations. The computational complexity order is similar among the competitor learning algorithms.

IV. PERFORMANCE ANALYSIS

The error, or excess risk, of gradient descent-type algorithms is typically defined as the difference between the objective value of the loss function at iteration k and the optimal value [54], [55]:

$$\mathbb{E}[F(\boldsymbol{\theta}_k)] - F(\boldsymbol{\theta}^*), \tag{8}$$

where the expectation is taken with respect to the randomness of the generated estimate $\boldsymbol{\theta}_k$, i.e., with respect to the random channel fading and the additive noise.

We start by analyzing the performance under the strongly convex case.

Theorem 1: Let the objective function be μ-strongly convex, M be the bound of the expected subgradient energy, $\alpha_k = \frac{1}{\mu(k+1)}$, and $\boldsymbol{\theta}^*$ denote the solution of the optimization problem in (2). Then, $\forall k$, the error under SGMA is bounded by:

$$\mathbb{E}\left[F(\boldsymbol{\theta}_k)\right] - F(\boldsymbol{\theta}^*) \leq \frac{2\tilde{M}^2}{\mu(k+1)}, \tag{9}$$

where

$$\tilde{M}^2 = \mu_h^2 M^2 + \frac{\sigma_h^2}{N} M^2 + \frac{d\sigma_w^2}{E_N N^2}. \tag{10}$$

The proof is given in the Appendix. Theorem 1 requires the following convergence conditions for the strongly convex case: (i) The objective function is μ-strongly convex; (ii) the step size is set to $\alpha_k = \frac{1}{\mu(k+1)}$; and (iii) bounded expected subgradient energy. The theorem implies that the error scales as $O\left(\frac{1}{k}\right)$, which is the convergence rate of the centralized subgradient method, and the bound approaches the best centralized leading constant as $N \rightarrow \infty$, when the transmission energy is set to $E_N = \Omega\left(N^{\epsilon-2}\right)$ for some $\epsilon > 0$. This implies that the system can improve performance by increasing the number of nodes which participate in the learning task while making the total transmission energy in the network arbitrarily close to zero.

Next, we analyze the performance under the convex case.

Theorem 2: Let the objective function be a convex function, M be the bound of the expected subgradient energy, $\boldsymbol{\theta}^*$ denote the solution of the optimization problem in (2), and $\Omega \geq \frac{1}{2}\|\boldsymbol{\theta}_0 - \boldsymbol{\theta}^*\|^2$. Then, $\forall k$, the error under SGMA is bounded by:

$$\mathbb{E}\left[F(\boldsymbol{\theta}_k)\right] - F(\boldsymbol{\theta}^*) \leq \frac{\tilde{\Omega} + \frac{1}{2}\sum_{i=1}^{k} \alpha_k^2 \tilde{M}^2}{\sum_{i=1}^{k} \alpha_k}, \tag{11}$$

where

$$\tilde{\Omega} = \frac{\Omega}{\mu_h}, \quad \text{and} \quad \tilde{M}^2 = \mu_h^2 M^2 + \frac{\sigma_h^2}{N} M^2 + \frac{d\sigma_w^2}{E_N N^2}. \tag{12}$$

The proof is given in the Appendix. Theorem 2 requires the following convergence conditions for the convex case: (i) The objective function is convex; (ii) the step size is set to $\alpha_k = \frac{1}{k^{1+q}}$ for $q > 0$ (as shown in the Appendix); and (iii) bounded expected subgradient energy. The theorem implies that the error scales as $O\left(\frac{1}{\sqrt{k}}\right)$, which is the convergence rate of the centralized subgradient method. We show in the proof that for other different selections of α_k with the knowledge of the time horizon, we can achieve the convergence rate of $O\left(\frac{1}{\sqrt{k}}\right)$ as well. Also, the bound approaches the best centralized leading constant as $N \rightarrow \infty$, when the transmission energy is set to $E_N = \Omega\left(N^{\epsilon-2}\right)$ for some $\epsilon > 0$.

As for $\boldsymbol{\theta}_0$, it represents the initial parameter values or the starting point of the optimization process. It is common in learning algorithms and optimization to initialize the starting point randomly based on prior knowledge about the optimum vicinity to speed up the convergence time. However, it does not improve the convergence rate order (see our previous work on projected SGD [56] and references therein). With no prior knowledge about the parameter set, $\boldsymbol{\theta}_0$ is selected randomly.

V. SIMULATION RESULTS

In this section, we present numerical examples to showcase the performance of the SGMA algorithm in two distinct problem settings. For the first set of simulations, we consider a federated learning task where the goal is to predict the release year of a song based on its audio features. We utilize the real-data Million Song Dataset [57] and partition the data among multiple edge devices. In the second set of simulations, we examine a federated learning problem involving real data on global stock prices. Specifically, we aim to predict the closing stock price of the following day based on historical price data [58]. The code for the simulations can be found at [59].

In each simulation, we compare the performance of SGMA with other smooth algorithms, as well as algorithms that have been shown to perform well in federated learning tasks in the literature: (i) the Error Compensated Entry-wise Scheduled Analog Distributed Stochastic Gradient Descent (ECESA-DSGD) algorithm [33], which transmits the gradient at each iteration only if the channel state exceeds a certain threshold; (ii) the FDM-GD algorithm, which assigns each node a dedicated orthogonal channel for transmission and calculates the mean signal at the PS [16]; (iii) ECESA-DSG, which employs the logic of ECESA-DSGD but utilizes a subgradient method; and (iv) the FDM-SGD algorithm, which is similar to FDM-GD but transmits the subgradient instead of the smooth function gradient. In all cases, the transmission parameters are set such that the average transmitted energy per node is the same for all algorithms. It is known that for non-smooth problems, the use of smoothing can hinder convergence to the minimum by the level of smoothing. For example, when addressing the absolute value function, $f(x) = |x|$, one may employ a smoothing technique such as the Huber function, defined as:

$$f_\mu(x) = \begin{cases} \frac{x^2}{2\mu}, & \text{if } |x| \le \mu, \\ |x| - \frac{\mu}{2}, & \text{otherwise,} \end{cases} \tag{13}$$

and the function is μ-Lipschitz continuous.

It has been established in [60] that the error rate will be bounded by the following:

$$f(x_t) - f_* \le \mathcal{O}\left(\frac{\|A\|_2^2 D_X^2}{\mu t} + \mu D_Y^2\right), \tag{14}$$

where we define for matrix $\boldsymbol{A}$ the norm: $\|\boldsymbol{A}\|_2 = \max_{\boldsymbol{x}}\{\|\boldsymbol{A}\boldsymbol{x}\|_2 : \|\boldsymbol{x}\|_2 \le 1\}$, where $\boldsymbol{A}$ is the Nesterov smoothing scale, and $D_Y^2 = \max_{\boldsymbol{y}\in Y}\{d(\boldsymbol{y})\}$, where D_Y is the bounding diameter over the image space. Similarly, D_X is defined as the bounding diameter over the source space.

In each set of simulations, we modeled the problem using the elastic loss function, which is defined as follows:

$$\begin{aligned} &L_{elastic}(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{y}) \\ &\quad = \sum_{n=0}^{M} (\boldsymbol{\theta}\boldsymbol{x}_n^T - y_n)^2 + \alpha \sum_{i=0}^{N} |\theta_i| + \frac{\beta}{2}\sum_{i=0}^{N}\theta_i^2 \\ &\quad = L_{MSE}(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{y}) + \alpha\|\boldsymbol{\theta}\|_1 + \frac{\beta}{2}\|\boldsymbol{\theta}\|_2^2, \end{aligned} \tag{15}$$

where the problem can be defined to be a Lasso problem by setting $\beta = 0$, and a Ridge problem by setting $\alpha = 0$. This problem is μ-strongly convex with $\mu = \frac{\beta}{2}$, and convex when $\beta = 0$. Note that the absolute value function does not possess a derivative at zero. Therefore, we define the subgradient of the absolute value function as the sign function, with a value of zero at zero, which is a valid subgradient [54], enabling us to define the derivative as:

$$\frac{\partial L_{elastic}(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{y})}{\partial \boldsymbol{\theta}} = \sum_{n=0}^{M} (\boldsymbol{\theta}\boldsymbol{x}_n^T - y_n)\boldsymbol{x}_n^T + \alpha sign(\boldsymbol{\theta}) + \beta\boldsymbol{\theta}. \tag{16}$$

For the smoothed function we obtain:

$$\begin{aligned} f(\boldsymbol{\theta}) &= \|\boldsymbol{\theta}\|_1 = \sum_{i=0}^{N} |\theta_i| \\ \Rightarrow f_\mu(\boldsymbol{\theta}) &= \sum_{|\theta_i| \le \mu} \frac{\theta_i^2}{2\mu} + \sum_{|\theta_i| \ge \mu} \left(|x| - \frac{\mu}{2}\right), \end{aligned} \tag{17}$$

and the derivative is defined by:

$$\frac{\partial L_{elastic}(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{y})}{\partial \boldsymbol{\theta}} = \sum_{n=0}^{M} (\boldsymbol{\theta}\boldsymbol{x}_n^T - y_n)\boldsymbol{x}_n^T + \alpha\frac{\partial f_\mu(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} + \beta\boldsymbol{\theta}, \tag{18}$$

where:

$$\left[\frac{\partial f_\mu(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]_i = \begin{cases} \frac{x}{\mu}, & \text{if } |x| \le \mu, \\ sign(x), & \text{otherwise.} \end{cases} \tag{19}$$

The selection of α and β plays an important role in striking a balance between model complexity and fitting error, with regularization loss to constrain the weights of the local models and prevent overfitting. The parameter α controls the weight of L_1 regularization, promoting sparse solutions that have many zero weights, while β controls the weight of L_2 regularization, pushing the weights towards zero and penalizing large weights. It is essential to choose appropriate values for α and β to ensure that the regularization is neither too weak nor too strong. In practice, these hyperparameters are often tuned using a validation set to find an efficient trade-off between model complexity and fitting error.

In the experiments, we chose the values of α and β based on their respective penalty in Lasso and Ridge/Elastic net regularization. In Lasso, α controls the weight of the L_1 penalty, promoting sparsity by encouraging many coefficients to be zero. In Ridge/Elastic net, β controls the weight of the L_2 penalty, discouraging large coefficients. In Experiment 1, we modeled the problem as a Lasso problem to prioritize sparsity in the learned model coefficients. Given the high-dimensional feature space, it was essential to reduce the number of features utilized in the model. By applying Lasso regularization, we achieved this sparsity by setting small coefficients to zero, effectively eliminating them from the model. We chose α to be 0.1, which offered a moderate level of sparsity while still retaining sufficient features to attain satisfactory accuracy. In Experiment 2, we modeled the problem as a Ridge/Elastic net problem to prevent overfitting in the model. Although the feature space was smaller, we aimed to ensure the learned model's robustness when exposed to new data. Ridge/Elastic net regularization allowed us to control overfitting by reducing the magnitude of the coefficients while preserving all the features in the model. We chose β to be 0.1, striking a suitable balance between diminishing coefficient magnitude and including all the features in the model. Our choice of regularization parameters was driven by the specific requirements of each experiment, aiming to achieve a favorable compromise between accuracy, sparsity, and generalization. The values we selected allowed us to strike an efficient trade-off considering these factors.

A. EXPERIMENT 1: PREDICTING A RELEASE YEAR OF A SONG

To begin, we consider the application of our algorithm to the task of predicting the release year of a song based on its audio attributes using federated learning. The Million Song Dataset [1], which spans the years 1922 to 2011, is utilized for this purpose. Each song is represented by a feature vector of size 90, referred to as the audio attributes, which serves as the input to the model, with the release year serving as the output. We chose to model the problem as a Lasso problem, resulting in a convex model (as shown in Equation (15)). For this purpose, we set $\beta = 0$. Additionally, we normalized the input and output values to the range [0,1], as is standard for Lasso problems. We also set $\mu = 10^{-16}$ for the smoothing technique and chose the number of edge nodes to be 100.

The results can be seen in Figs. 2, 3 for low ($\sigma_w = 1$) and high ($\sigma_w = 10$) noise levels, respectively. For each run, the energy of the signal was normalized to 1. As for the low noise level, all algorithms converge, and SGMA converges significantly more quickly. Additionally, it can be observed that SGMA is more stable compared to the other algorithms, and it achieves the smallest error. For the high noise level, only SGMA converges. It can be seen again that SGMA achieves the best convergence speed. These results demonstrated the significance of using SGMA for OTA computation of non-differentiable loss functions.

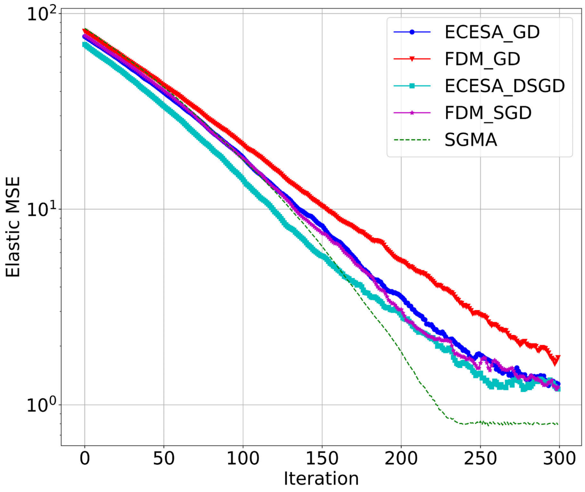

FIGURE 2. Algorithm comparison for the federated learning setting of predicting a release year of a song, for a low noise level.

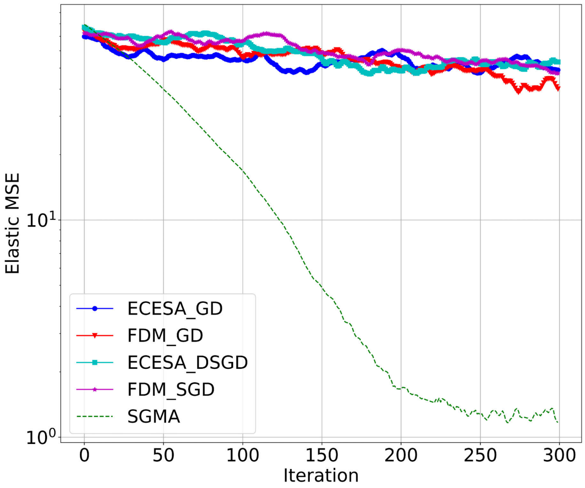

FIGURE 3. Algorithm comparison for the federated learning setting of predicting a release year of a song, for a high noise level.

B. EXPERIMENT 2: PREDICTING STOCK VALUE AT THE CLOSING CALL

In this part of the study, we attempted to predict the value of a stock at the end of a day based on its price for the previous N days. The Stock price data set [58], which contains daily stock prices for the New York stock exchange from 2012 to 2016, was utilized for this purpose. The data includes information on 10,000 stock prices. We selected a 20-day window of historical data for each stock and sought to predict the price at the following day based on this information. We split the data so that the input is the $n+21$ sample, for input samples from n to $n+20$ each time. We chose to model the problem as an Elastic problem, which is a μ-strongly convex problem, in order to facilitate comparison with other models. We set the smooth parameter to $\mu = 10^{-20}$. Furthermore, by setting

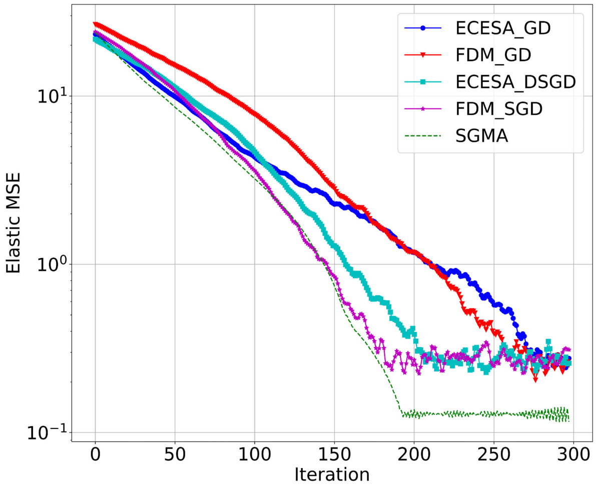

FIGURE 4. Algorithm comparison for the federated learning setting of predicting a stock value, for a low noise level.

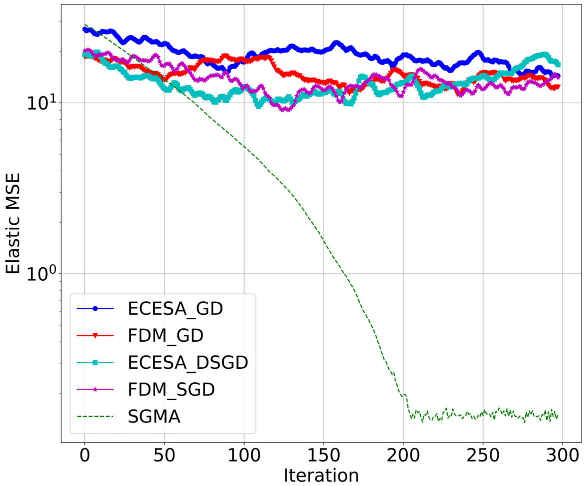

FIGURE 5. Algorithm comparison for the federated learning setting of predicting a stock value, for a high noise level.

β to a non-zero value in the loss function (15), our model becomes $\frac{\beta}{2}$-strongly convex.

The results can be seen in Figs. 4, 5 for low ($\sigma_w = 1$) and high ($\sigma_w = 10$) noise levels, respectively. As in the previous simulations, for low noise levels, all algorithms converge, with SGMA exhibiting significantly faster convergence. Additionally, it can be observed again that SGMA is more stable compared to the other algorithms, and it achieves the smallest error. For the high noise level, only SGMA converges again. The simulation results show that in all tested experiments, SGMA consistently demonstrated superior performance.

VI. DISCUSSION

As can be seen, the proposed SGMA algorithm significantly outperforms other methods. In what follows, we provide an interpretation of the results, which can be attributed to two key factors. Firstly, the utilization of OTA computation in SGMA leads to higher SNR at the receiver. This advantage arises from the fact that additive noise is introduced in only one dimension, in contrast to digital communication schemes where noise is added in N dimensions. By reducing the impact of noise, SGMA achieves improved performance in terms of convergence and accuracy. Secondly, SGMA stands out as the first OTA scheme specifically designed to handle non-differentiable loss functions over fading channels. This capability is crucial in scenarios where the loss functions involved are non-differentiable, which is often encountered in practical machine learning applications. SGMA allows all nodes to simultaneously transmit their sub-gradients over MAC, ensuring convergence to the best-known centralized optimizer under carefully selected parameter settings. This analytical demonstration further reinforces the effectiveness and practical applicability of SGMA.

It is worth emphasizing that the observed results are significant in a wider context, holding broader implications beyond the specific problems simulated in this paper. Similar outcomes can be expected in other types of problems where non-differentiable objective functions are encountered. For instance, certain formulations of SVMs incorporate a hinge loss function, which is known to be non-differentiable. The hinge loss imposes a penalty when the predicted value falls outside a predefined margin constraint. In such cases, SGMA's ability to handle non-differentiable loss functions becomes particularly advantageous. Another relevant example is robust optimization, which aims to identify solutions that are resilient to uncertainties or outliers present in the data. In robust optimization, the objective function often incorporates non-differentiable components such as absolute deviation or the Huber loss, which exhibit non-differentiability at specific points. In these problems, SGMA has significant advantage, as it collaboratively learns and optimizes the model without relying on the differentiability assumption of the objective function.

The suggested method has the potential to significantly impact various federated learning applications, demonstrating its relevance and potential across various domains. For instance, the utilization of Lasso optimization in Experiment 1 and Elastic Net optimization in Experiment 2 reflects their wide application in machine learning. These optimization techniques find utility in tasks related to image processing and computer vision, such as image denoising, image segmentation, and feature extraction. They are also employed in regression and classification problems that aim to construct accurate models for predicting outcomes or classifying instances based on an extensive set of input features. Such applications span domains like environmental monitoring, finance, healthcare, marketing, and more. By harnessing SGMA's capability to efficiently solve these optimization problems within resource-constrained communication networks, its potential for contributing to federated learning systems in the Internet of Things (IoT) and autonomous networks is significant. In these contexts, edge devices such as traffic sensors, surveillance cameras, drones, body sensors, and environmental sensors can collaboratively learn and optimize the desired model.

Despite its advantages, the suggested method has limitations as well. The first limitation is that OTA schemes, particularly the suggested SGMA, require coherent transmissions over MAC. This requirement may restrict the scalability and adaptability of SGMA in larger or more diverse networks. To overcome this limitation, a common approach is to adopt a hybrid strategy, where OTA is employed in clustered geographical areas, while OFDMA is used between clusters. The second limitation is that OTA schemes may not perform well in scenarios with very low SNR and small network sizes, as analog transmissions cannot be reliably decoded. In such cases, digital transmissions are preferred to ensure reliable signal decoding.

There are several promising avenues for future research in this area. One potential direction is to investigate hierarchical SGMA, where nodes communicate in a hierarchical topology instead of the star topology considered in this paper. This would provide greater flexibility for data transmission, especially in scenarios where link connectivity between nodes and the server is unstable. Developing SGMA with hierarchical communications poses challenges in terms of practical design and theoretical convergence analysis. Another research direction is to design SGMA with a hybrid approach, where OTA computation is implemented in clustered geographical areas while using OFDMA between clusters. This approach would address the limitation of coherent transmissions required for all nodes over MAC. However, implementing this scheme introduces challenges in terms of efficient channel allocation among nodes. It would be desirable to develop dynamic spectrum access algorithms that can achieve the best performance by balancing learning performance and resource consumption.

VII. CONCLUSION

We have developed a novel subgradient-based learning algorithm, SGMA, for distributed optimization problems over noisy fading MAC. We have theoretically developed finite bounds on the error for both convex and strongly convex cases, and showed that SGMA achieves the centralized model error bound as the number of nodes increases and transmission energy is set to $\Omega\left(N^{\epsilon-2}\right)$. Extensive simulation results using real data sets have been presented to demonstrate the superior performance of SGMA compared to existing methods. Furthermore, we have demonstrated that under moderate to high levels of noise, SGMA should be employed instead of smoothing techniques in order to achieve faster and more stable convergence of the model.

APPENDIX

In this appendix, we provide the proofs for Theorems 1, 2. The proof will be divided into two parts. In the first part, we will derive error bounds for each case, and then in the second part, we will examine the convergence rate by considering a range of different learning rates. First, let us define:

$$\nabla f = \{g(\boldsymbol{x}) \| f(\boldsymbol{y}) \geq g(\boldsymbol{x})^T(\boldsymbol{y}-\boldsymbol{x}), \quad \forall \boldsymbol{y} \in \boldsymbol{X}\}. \tag{20}$$

Under the assumption that f is a convex function, the partial derivative holds:

$$f(\boldsymbol{y}) \geq f(\boldsymbol{\theta}_k) + g^T(\boldsymbol{\theta}_k)(\boldsymbol{y}-\boldsymbol{\theta}_k), \quad \forall \boldsymbol{y} \in \boldsymbol{X} \tag{21}$$

We will first prove Theorem 2 and then proceed to prove Theorem 1.

A. PROOF OF THEOREM 2

We utilize the projection function $\Pi_{\boldsymbol{X}}(\cdot)$ onto some space $\boldsymbol{X}$, so that the update iteration step becomes:

$$\boldsymbol{\theta}_{k+1} = \Pi_{\boldsymbol{X}}(\boldsymbol{\theta}_k - \alpha_k \boldsymbol{v}_k). \tag{22}$$

Note that

for $\boldsymbol{X} = \mathbb{R}$, we recover the expression within the projection function. Now, let us denote $\boldsymbol{\theta}^*$ as the solution, then:

$$\boldsymbol{r}_k^2 = \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2, \qquad \boldsymbol{r}_{k+1}^2 = \|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2. \tag{23}$$

Consequently, we can write:

$$\begin{aligned} \boldsymbol{r}_k^2 &= \|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2 \overset{(22)}{=} \|\Pi_{\boldsymbol{X}}(\boldsymbol{\theta}_k - \alpha_k \boldsymbol{v}_k) - \boldsymbol{\theta}^*\|^2 \\ &\leq \|\boldsymbol{\theta}_k - \alpha_k \boldsymbol{v}_k - \boldsymbol{\theta}^*\|^2 \\ &= \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2 - 2\alpha_k \boldsymbol{v}_k^T(\boldsymbol{\theta}_k - \boldsymbol{\theta}^*) + \alpha_k^2 \|\boldsymbol{v}_k\|^2 \\ &= \boldsymbol{r}_k^2 - 2\alpha_k \boldsymbol{v}_k^T(\boldsymbol{\theta}_k - \boldsymbol{\theta}^*) + \alpha_k^2 \|\boldsymbol{v}_k\|^2. \end{aligned} \tag{24}$$

Additionally, we have:

$$\nabla \boldsymbol{G}(\boldsymbol{\theta}_k) = \frac{1}{N}\sum_{n=1}^{N} g_n(\boldsymbol{\theta_k}), \tag{25}$$

$$\mathbb{E}[\boldsymbol{v}_k] = \mu_h \mathbb{E}[\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)], \tag{26}$$

and

$$\begin{aligned} &\mathbb{E}[\|\boldsymbol{v}_k\|^2] \\ &= \frac{1}{N^2}\sum_{n=1}^{N}\sum_{m=1}^{N} \mathbb{E}[(h_{n,k} g_n(\boldsymbol{\theta}_k)^T (h_{m,k} g_m(\boldsymbol{\theta}_k)] + \frac{d\sigma_w^2}{E_N N^2} \\ &= \mu_h^2 \mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \frac{d\sigma_w^2}{E_N N^2}. \end{aligned} \tag{27}$$

By taking the expectation of (24), we obtain:

$$\begin{aligned} \mathbb{E}[\boldsymbol{r}_{k+1}^2] \leq \mathbb{E}[\boldsymbol{r}_k^2] &- 2\alpha_k \mu_h \mathbb{E}[\nabla \boldsymbol{G}(\boldsymbol{\theta}_k - \boldsymbol{\theta}^*)] \\ &+ \alpha_k^2 \mu_h^2 \mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] \\ &+ \alpha_k^2 \frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2 \frac{d\sigma_w^2}{E_N N^2}, \end{aligned} \tag{28}$$

and we can write (28) as:

$$\begin{aligned} &2\alpha_k \mu_h \mathbb{E}[\nabla \boldsymbol{G}(\boldsymbol{\theta}_k - \boldsymbol{\theta}^*)] \\ &\quad \leq \mathbb{E}[\boldsymbol{r}_k^2] - \mathbb{E}[\boldsymbol{r}_{k+1}^2] + \alpha_k^2 \mu_h^2 \mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] \\ &\qquad + \alpha_k^2 \frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2 \frac{d\sigma_w^2}{E_N N^2}. \end{aligned} \tag{29}$$

Since f is convex, from (21), we can write:

$$\mathbb{E}[g_n^T(\boldsymbol{\theta}_k)(\boldsymbol{\theta}_k - \boldsymbol{\theta}^*)] \geq \mathbb{E}[f(\boldsymbol{\theta}_k) - f(\boldsymbol{\theta}^*)]. \tag{30}$$

By summing from $n = 1$ to $n = N$, and dividing each side by N, we obtain:

$$\mathbb{E}[\nabla \boldsymbol{G}^T(\boldsymbol{\theta}_k)(\boldsymbol{\theta}_k - \boldsymbol{\theta}^*)] \geq \mathbb{E}[F(\boldsymbol{\theta}_k) - F(\boldsymbol{\theta}^*)]. \tag{31}$$

Substituting (31) into (28), we obtain:

$$\begin{aligned} 2\alpha_k\mu_h\mathbb{E}[F(\boldsymbol{\theta}_k) - F(\boldsymbol{\theta}^*)] \\ \leq \mathbb{E}[\boldsymbol{r}_k^2] - \mathbb{E}[\boldsymbol{r}_{k+1}^2] + \alpha_k^2\mu_h^2\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] \\ + \alpha_k^2\frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2\frac{d\sigma_w^2}{E_N N^2}. \end{aligned} \tag{32}$$

By summing both sides of (32) from $k = 1$ to $k = T$, we get:

$$\begin{aligned} \sum_{k=1}^{T}\alpha_k\mathbb{E}[f(\boldsymbol{\theta}_k) - f(\boldsymbol{\theta}^*)] \leq \frac{1}{2\mu_h}\sum_{k=1}^{T}\Big(\mathbb{E}[\boldsymbol{r}_k^2] - \mathbb{E}[\boldsymbol{r}_{k+1}^2] \\ + \alpha_k^2\mu_h^2\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] \\ + \alpha_k^2\frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2\frac{d\sigma_w^2}{E_N N^2}\Big) \\ \overset{\text{telescoping series}}{=} \frac{1}{2\mu_h}\Big[\mathbb{E}[r_1^2] - \mathbb{E}[r_{T+1}^2] \\ + \sum_{k=1}^{T}(\alpha_k^2\mu_h^2\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2\frac{d\sigma_w}{E_N N^2})\Big] \\ \leq \frac{1}{2\mu_h}\Big(\mathbb{E}[r_1^2] + \sum_{k=1}^{T}(\alpha_k^2\mu_h^2\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2\frac{d\sigma_w}{E_N N^2})\Big). \end{aligned} \tag{33}$$

We can bound the left side by:

$$\begin{aligned} \sum_{k=1}^{T}\alpha_k\mathbb{E}[F(\boldsymbol{\theta}_k) - F(\boldsymbol{\theta}^*)] \\ \geq \sum_{k=1}^{T}\alpha_k \min_{1\leq l\leq T}\big(\mathbb{E}[F(\boldsymbol{\theta}_l) - F(\boldsymbol{\theta}^*)]\big) \\ = \Big(\sum_{k=1}^{T}\alpha_k\Big)\min_{1\leq l\leq T}\big(\mathbb{E}[F(\boldsymbol{\theta}_l) - F(\boldsymbol{\theta}^*)]\big), \end{aligned} \tag{34}$$

and:

$$\sum_{k=1}^{T}\alpha_k\mathbb{E}\big[F(\boldsymbol{\theta}_k) - F(\boldsymbol{\theta}^*)\big] \geq \Big(\sum_{k=1}^{T}\alpha_k\Big)\mathbb{E}\big[F(\hat{\boldsymbol{\theta}}_T) - F(\boldsymbol{\theta}^*)\big], \tag{35}$$

where $\hat{\boldsymbol{\theta}}_T = \frac{\sum_{k=1}^{T}\alpha_k\boldsymbol{\theta}_k}{\sum_{k=1}^{T}\alpha_k} \in \boldsymbol{X}$ is a convex series. Let us denote the error as:

$$\begin{aligned} \varepsilon_T &= \mathbb{E}\big[F(\hat{\boldsymbol{\theta}}_T) - F(\boldsymbol{\theta}^*)\big] \\ \text{or} \quad \varepsilon_T &= \min_{1\leq l\leq T}\big(\mathbb{E}[F(\boldsymbol{\theta}_l) - F(\boldsymbol{\theta}^*)]\big). \end{aligned} \tag{36}$$

For both definitions, we can conclude from (31), (34), (35) and (36) the following error bound:

$$\begin{aligned} \varepsilon_T \leq \frac{1}{2\mu_h\sum_{k=1}^{T}\alpha_k}\Big[\mathbb{E}[\boldsymbol{r}_1^2] \\ + \sum_{k=1}^{T}\Big(\alpha_k^2\mu_h^2\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \alpha_k^2\frac{d\sigma_w}{E_N N^2}\Big)\Big]. \end{aligned} \tag{37}$$

Let us define:

$$\mathbb{E}\big[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|\big] \leq M \Rightarrow \mathbb{E}\big[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\big] \leq M^2, \tag{38}$$

$$\frac{1}{2}\boldsymbol{r}_1^2 = \frac{1}{2}\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}^*\|^2 \leq \Omega. \tag{39}$$

Now, we can write (37) as:

$$\begin{aligned} \varepsilon_T \leq \frac{1}{\mu_h\sum_{k=1}^{T}\alpha_k} \\ \times\Big[\Omega + \frac{1}{2}\sum_{k=1}^{T}\Big(\alpha_k^2\mu_h^2M^2 + \alpha_k^2\frac{\sigma_h^2}{N}M^2 + \alpha_k^2\frac{d\sigma_w}{E_N N^2}\Big)\Big] \\ = \frac{\tilde{\Omega} + \frac{1}{2}\sum_{k=1}^{T}\alpha_k^2\tilde{M}^2}{\sum_{k=1}^{T}\alpha_k}, \end{aligned} \tag{40}$$

where $\tilde{\Omega} = \frac{\Omega}{\mu_h}$ and $\tilde{M}^2 = \mu_h^2M^2 + \frac{\sigma_h^2}{N}M^2 + \frac{d\sigma_w^2}{E_N N^2}$.

Next, we check the convergence for various step sizes.

For a constant step size $\alpha_k = \alpha$, we can write (40) as:

$$\varepsilon_T \leq \frac{\tilde{\Omega} + \frac{1}{2}\sum_{k=1}^{T}\alpha^2\tilde{M}^2}{\sum_{k=1}^{T}\alpha} = \frac{\tilde{\Omega}}{T\alpha} + \frac{\tilde{M}^2}{2}\alpha \xrightarrow{T\to\infty} \frac{\tilde{M}^2}{2}\alpha. \tag{41}$$

It can be seen that the error upper-bound does not diminish to zero as T grows to infinity, which shows one of the drawbacks of using arbitrary constant step sizes. In addition, for optimizing the upper bound, we can select the optimal step size to be $\alpha^* = \frac{\sqrt{2\tilde{\Omega}}}{\tilde{M}\sqrt{T}}$ which implies $\varepsilon_T \leq \frac{\tilde{\Omega}}{\tilde{M}\sqrt{2T}}$. Then, with the optimal choice we get $\varepsilon_T \sim \mathcal{O}(\frac{\tilde{\Omega}}{\tilde{M}\sqrt{T}})$. However, one disadvantage of using that constant step size is that the number of iterations required for convergence, T, is typically unknown beforehand. This makes it difficult to determine the optimal value of α^*, as it depends on the value of T.

Next, conisder a scaled step size $\alpha_k = \frac{\alpha}{\|\nabla G(\boldsymbol{\theta}_k)\|}$. In this case, the error behaves as:

$$\varepsilon_T \leq \frac{\tilde{\Omega} + \frac{\frac{1}{2}\alpha^2\tilde{M}^2}{M^2}T}{\frac{\alpha T}{M}} \xrightarrow{T\to\infty} \frac{\tilde{M}^2}{2M}\alpha. \tag{42}$$

Similarly, we can select the optimal α by minimizing the right-hand side, i.e., $\alpha = \alpha^* = \frac{M\sqrt{2\tilde{\Omega}}}{M\sqrt{T}}$. Then, we get the following step size:

$$\alpha_k = \frac{M\sqrt{2\tilde{\Omega}}}{\tilde{M}\sqrt{T}\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|}, \tag{43}$$

which yields $\varepsilon_T \leq \frac{\tilde{M}\sqrt{\tilde{\Omega}}}{\sqrt{2T}}$.

The same convergence rate is achieved as in the constant step size case. However, the same issue of not knowing the value of T beforehand persists when choosing α_k, as it impacts the optimal value of the step size.

Third, consider a non summable but diminishing step size. Here we assume that $\sum_{k=1}^{\infty}\alpha_k = \infty,\ \lim_{k\to\infty}\alpha_k = 0$. Then, the error behaves as:

$$\begin{aligned}\varepsilon_T &\le \frac{\tilde{\Omega}+\frac{1}{2}\sum_{k=1}^{T}\alpha_k^2\tilde{M}^2}{\sum_{k=1}^{T}\alpha_k}\\ &\le \frac{\tilde{\Omega}+\frac{1}{2}\tilde{M}^2\sum_{k=1}^{T_1}\alpha_k^2}{\sum_{k=1}^{T}\alpha_k}+\frac{\frac{1}{2}\tilde{M}^2\sum_{k=T_1+1}^{T}\alpha_k^2}{\sum_{k=T_1+1}^{T}\alpha_k},\end{aligned} \tag{44}$$

where $1 \le T_1 \le T$, and $T \to \infty$, for selecting large T_1, the first term on the right-hand side goes to 0, since α_k is non-summable, the second term also goes to 0, because α_k^2 always approaches zero faster than α_k. Consequently, we have $\varepsilon_T \to 0$ as $T \to \infty$. An example of a step size choice is $\alpha_k \sim \mathcal{O}(\frac{1}{k^q})$ with $q \in (0, 1]$. As in the above cases, if we choose $\alpha_k = \frac{\sqrt{2\tilde{\Omega}}}{\tilde{M}\sqrt{T}}$, then $\varepsilon_T \le \mathcal{O}(\frac{\sqrt{\tilde{\Omega}}\tilde{M}\ln(T)}{\sqrt{T}}))$, and if we choose the averaging from $\frac{T}{2}$ instead of 1, we have: $\min_{\frac{T}{2}\le l\le T}\varepsilon_l \le \mathcal{O}(\frac{\sqrt{\tilde{\Omega}}\tilde{M}}{\sqrt{T}})$.

Fionally, consider a non-summable but square summable step size. In this case we have: $\sum_{k=1}^{\infty}\alpha_k = \infty$, $\sum_{k=1}^{\infty}\alpha_k^2 \le \infty$, e.g., $\alpha_k \sim \mathcal{O}(\frac{1}{k})$. Then, the error behaves as:

$$\varepsilon_T \le \frac{\tilde{\Omega}+\frac{1}{2}\sum_{k=1}^{T}\alpha_k^2\tilde{M}^2}{\sum_{k=1}^{T}\alpha_k}\xrightarrow{T\to\infty} 0. \tag{45}$$

As a typical choice of $\alpha_k = \frac{1}{t^{1+q}}$ for $q > 0$, this results in the rate of $\mathcal{O}(\frac{1}{\sqrt{T}})$, which completes the proof.

B. PROOF OF THEOREM 1

We now assume that f is μ-strongly convex. Thus,

$$f(y) \ge f(x) + g^T(x)(y-x) + \frac{\mu}{2}\|x-y\|^2. \tag{46}$$

We can rewrite (31) as:

$$\begin{aligned}&\mathbb{E}\left[\nabla \boldsymbol{G}^T(\boldsymbol{\theta}_k)(\boldsymbol{\theta}_k-\boldsymbol{\theta}^*)\right]\\ &\quad\ge \mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta}^*)\right]+\mathbb{E}\left[\frac{\mu}{2}\|\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\|^2\right]\\ &\quad= \mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta}^*)\right]+\frac{\mu}{2}\boldsymbol{r_k}^2.\end{aligned} \tag{47}$$

Substituting (47) into (29), we can rewrite (32) as:

$$\begin{aligned}&2\alpha_k\mu_h\left[\mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta}^*)\right]+\frac{\mu}{2}\boldsymbol{r}_k^2\right]\\ &\quad\le \mathbb{E}\left[\boldsymbol{r}_k^2\right]-\mathbb{E}\left[\boldsymbol{r}_{k+1}^2\right]+\alpha_k^2\mu_h^2\mathbb{E}\left[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\right]\\ &\qquad+\alpha_k^2\frac{\sigma_h^2}{N}\mathbb{E}\left[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\right]+\alpha_k^2\frac{d\sigma_w^2}{E_N N^2}.\end{aligned} \tag{48}$$

Next, consider two choices for the learning step-size. First, consider $\alpha_k = \frac{1}{\mu k}$. For this step-size we can write (48) as:

$$\begin{aligned}&2\frac{1}{\mu k}\mu_h\mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta}^*)\right]+\frac{\mu_h}{k}\boldsymbol{r}_k^2\\ &\quad\le \mathbb{E}[\boldsymbol{r}_k^2]-\mathbb{E}[\boldsymbol{r}_{k+1}^2]+\frac{\mu_h^2}{(\mu k)^2}\mathbb{E}\left[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\right]\\ &\qquad+\frac{\sigma_h^2}{N(\mu k)^2}\mathbb{E}\left[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\right]+\frac{1}{(\mu k)^2}\frac{d\sigma^2}{E_N N^2}.\end{aligned} \tag{49}$$

Then, by: $\tilde{\Omega} = \frac{\Omega}{\mu_h}$ and $\tilde{M}^2 = \mu_h^2M^2 + \frac{\sigma_h^2}{N}M^2 + \frac{d\sigma_w^2}{E_N N^2}$, and by manipulating the inequality in (49), we obtain:

$$\begin{aligned}&\mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta}^*)\right] \le \frac{\mu}{2\mu_h}(k-\mu_h)\mathbb{E}\left[\boldsymbol{r}_k^2\right]\\ &\quad-\frac{\mu k}{2\mu_h}\mathbb{E}[\boldsymbol{r}_{k+1}^2]+\frac{1}{(\mu k)^2}\\ &\quad\times\left(\mu_h^2\mathbb{E}\left[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\right]+\frac{\sigma_h^2}{N}\mathbb{E}\left[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2\right]+\frac{d\sigma_w^2}{E_N N^2}\right)\\ &\le \frac{\mu}{2\mu_h}(k-\mu_h)\mathbb{E}[\boldsymbol{r}_k^2]-\frac{\mu k}{2\mu_h}\mathbb{E}[\boldsymbol{r}_{k+1}^2]+\frac{1}{(\mu k)^2}\tilde{M}^2.\end{aligned} \tag{50}$$

By summing both sides in (50) over $k = 1, \ldots, T$ we obtain:

$$\begin{aligned}\sum_{k=1}^{T}\mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta}^*)\right] &\le \sum_{k=1}^{T}\frac{1}{(\mu k)^2}\tilde{M}^2\\ &= \frac{1}{2\mu}\tilde{M}^2(\ln T+1),\end{aligned} \tag{51}$$

and from (36), we can write:

$$\varepsilon_T \le \frac{1}{2\mu T}\tilde{M}^2(\ln T+1). \tag{52}$$

Therefore, we obtain: $\varepsilon_T \sim \mathcal{O}\left(\frac{\tilde{M}^2}{\mu T}\ln T\right)$.

Next, consider the following step size: $\alpha_k = \frac{1}{\mu(k+1)}$. In this case, we can write (48) as:

$$\begin{aligned}&\mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta})^*\right]\\ &\quad\le \frac{\mu}{4\mu_h}(k+1-2\mu_h)\mathbb{E}[\boldsymbol{r}_k^2]-\frac{\mu(k+1)}{\mu_h}\mathbb{E}[\boldsymbol{r}_{k+1}^2]\\ &\qquad+\frac{2}{\mu(k+1)}\\ &\qquad\left[\mu_h^2\mathbb{E}[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2]+\frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2]+\frac{d\sigma_w^2}{E_N N^2}\right]\end{aligned} \tag{53}$$

Then, by multiplying both sides of (53) by k, we obtain:

$$\begin{aligned}&k\mathbb{E}\left[F(\boldsymbol{\theta}_k)-F(\boldsymbol{\theta})^*\right]\\ &\quad\le \frac{\mu}{4\mu_h}(k+1-2\mu_h)k\mathbb{E}[\boldsymbol{r}_k^2]-\frac{\mu k(k+1)}{\mu_h}\mathbb{E}[\boldsymbol{r}_{k+1}^2]\\ &\qquad+\frac{2k}{\mu(k+1)}\\ &\qquad\left[\mu_h^2\mathbb{E}[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2]+\frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla\boldsymbol{G}(\boldsymbol{\theta}_k)\|^2]+\frac{d\sigma_w^2}{E_N N^2}\right]\end{aligned}$$

$$\begin{aligned}
&\leq \frac{\mu}{4\mu_h}(k+1-2\mu_h)k\mathbb{E}[\boldsymbol{r}_k^2] - \frac{\mu k(k+1)}{\mu_h}\mathbb{E}[\boldsymbol{r}_{k+1}^2] \\
&+ \frac{2}{\mu}\left[\mu_h^2\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \frac{\sigma_h^2}{N}\mathbb{E}[\|\nabla \boldsymbol{G}(\boldsymbol{\theta}_k)\|^2] + \frac{d\sigma_w^2}{E_N N^2}\right] \\
&\leq \frac{\mu}{4\mu_h}(k+1-2\mu_h)k\mathbb{E}[\boldsymbol{r}_k^2] - \frac{\mu k(k+1)}{\mu_h}\mathbb{E}[\boldsymbol{r}_{k+1}^2] + \frac{\tilde{M}^2}{\mu}.
\end{aligned} \tag{54}$$

Now, by summing (54) from $k = 1$ to $k = T$, we get:

$$\sum_{k=1}^{T} k\mathbb{E}[F(\boldsymbol{\theta}_k) - F(\boldsymbol{\theta}^*)] \leq \frac{\tilde{M}^2 T}{\mu}. \tag{55}$$

Since F is μ-strongly convex, we can write:

$$\sum_{k=1}^{T} k\mathbb{E}[F(\boldsymbol{\theta}_k) - F(\boldsymbol{\theta}^*)] \geq \frac{T(T+1)}{2}\varepsilon_T. \tag{56}$$

From (55) and (56), we obtain:

$$\varepsilon_T \leq \frac{2\tilde{M}^2}{\mu(T+1)}. \tag{57}$$

Finally, We can see that $\varepsilon_T \sim \mathcal{O}\left(\frac{\tilde{M}^2}{\mu T}\right)$, which completes the proof.

ACKNOWLEDGMENT

An earlier version of this paper was presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2023 [DOI: 10.1109/ICASSP49357.2023.10095134].

REFERENCES

[1] T. L. S. Gez and K. Cohen, "Subgradient descent learning with over-the-air computation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.

[2] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.

[3] M. S. H. Abad, E. Ozfatura, D. GUndUz, and O. Ercetin, "Hierarchical federated learning ACROSS heterogeneous cellular networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8866–8870.

[4] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 310–323, Jan. 2019.

[5] D. Livne and K. Cohen, "PoPS: Policy pruning and shrinking for deep reinforcement learning," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 4, pp. 789–801, May 2020.

[6] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[7] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140699–140725, 2020.

[8] S. Appadwedula, V. V. Veeravalli, and D. L. Jones, "Decentralized detection with censoring sensors," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1362–1373, Apr. 2008.

[9] R. S. Blum and B. M. Sadler, "Energy efficient signal detection in sensor networks using ordered transmissions," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3229–3235, Jul. 2008.

[10] R. S. Blum, "Ordering for estimation and optimization in energy efficient sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2847–2856, Jun. 2011.

[11] K. Cohen and A. Leshem, "Energy-efficient detection in wireless sensor networks using likelihood ratio and channel state information," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 8, pp. 1671–1683, Sep. 2011.

[12] P. Braca, S. Marano, and V. Matta, "Single-transmission distributed detection via order statistics," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 2042–2048, Apr. 2012.

[13] K. Cohen and Q. Zhao, "Active hypothesis testing for anomaly detection," *IEEE Trans. Inf. Theory*, vol. 61, no. 3, pp. 1432–1450, Mar. 2015.

[14] J. Zhang, Z. Chen, R. S. Blum, X. Lu, and W. Xu, "Ordering for reduced transmission energy detection in sensor networks testing a shift in the mean of a Gaussian graphical model," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2178–2189, Apr. 2017.

[15] B. Huang, K. Cohen, and Q. Zhao, "Active anomaly detection in heterogeneous processes," *IEEE Trans. Inf. Theory*, vol. 65, no. 4, pp. 2284–2301, Apr. 2019.

[16] T. Gafni, N. Shlezinger, K. Cohen, Y. C. Eldar, and H. V. Poor, "Federated learning: A signal processing perspective," *IEEE Signal Process. Mag.*, vol. 39, no. 3, pp. 14–41, May 2022.

[17] C. Xu, S. Liu, Z. Yang, Y. Huang, and K.-K. Wong, "Learning rate optimization for federated learning exploiting over-the-air computation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3742–3756, Dec. 2021.

[18] N. Zhang and M. Tao, "Gradient statistics aware power control for over-the-air federated learning in fading channels," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–6.

[19] H. Hellström, V. Fodor, and C. Fischione, "Over-the-air federated learning with retransmissions," in *Proc. IEEE 22nd Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Sep. 2021, pp. 291–295.

[20] B. Jiang, J. Du, C. Jiang, Y. Shi, and Z. Han, "Communication-efficient device scheduling via over-the-air computation for federated learning," in *Proc. GLOBECOM IEEE Global Commun. Conf.*, Dec. 2022, pp. 173–178.

[21] M. Kim, A. Lee Swindlehurst, and D. Park, "Beamforming vector design and device selection in over-the-air federated learning," *IEEE Trans. Wireless Commun.*, early access, Mar. 8, 2023, doi: 10.1109/TWC.2023.3251339.

[22] M. Kim and D. Park, "Joint beamforming and learning rate optimization for over-the-air federated learning," *IEEE Trans. Veh. Technol.*, early access, May 16, 2023, doi: 10.1109/TVT.2023.3276786.

[23] C. Chen, Y.-H. Chiang, H. Lin, J. C. S. Lui, and Y. Ji, "Joint client selection and receive beamforming for over-the-air federated learning with energy harvesting," *IEEE Open J. Commun. Soc.*, vol. 4, pp. 1127–1140, 2023.

[24] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, 2020.

[25] M. M. Amiri and D. Gündüz, "Over-the-air machine learning at the wireless edge," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2019, pp. 1–5.

[26] T. Sery and K. Cohen, "A sequential gradient-based multiple access for distributed learning over fading channels," in *Proc. 57th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2019, pp. 303–307.

[27] M. M. Amiri, T. M. Duman, and D. Gündüz, "Collaborative machine learning at the wireless edge with blind transmitters," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2019, pp. 1–5.

[28] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.

[29] Q. Zeng, Y. Du, K. Huang, and K. K. Leung, "Energy-efficient radio resource allocation for federated edge learning," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–6.

[30] T. Sery and K. Cohen, "On analog gradient descent learning over multiple access fading channels," *IEEE Trans. Signal Process.*, vol. 68, pp. 2897–2911, 2020.

[31] A. Abdi, Y. M. Saidutta, and F. Fekri, "Analog compression and communication for federated learning over wireless MAC," in *Proc. IEEE 21st Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, May 2020, pp. 1–5.

[32] W.-T. Chang and R. Tandon, "Communication efficient federated learning over multiple access channels," 2020, *arXiv:2001.08737*.

[33] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.

[34] K. Ozfatura, E. Ozfatura, and D. Gunduz, "Distributed sparse SGD with majority voting," 2020, *arXiv:2011.06495*.

[35] R. Paul, Y. Friedman, and K. Cohen, "Accelerated gradient descent learning over multiple access fading channels," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 2, pp. 532–547, Feb. 2022.

[36] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-air federated learning from heterogeneous data," *IEEE Trans. Signal Process.*, vol. 69, pp. 3796–3811, 2021.

[37] G. Mergen and L. Tong, "Type based estimation over multiaccess channels," *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.

[38] G. Mergen, V. Naware, and L. Tong, "Asymptotic detection performance of type-based multiple access over multiaccess fading channels," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 1081–1092, Mar. 2007.

[39] S. Marano, V. Matta, L. Tong, and P. Willett, "A likelihood-based multiple access for estimation in sensor networks," *IEEE Trans. Signal Process.*, vol. 55, no. 11, pp. 5155–5166, Nov. 2007.

[40] P. Zhang, I. Nevat, G. W. Peters, and L. Clavier, "Event detection in sensor networks with non-linear amplifiers via mixture series expansion," *IEEE Sensors J.*, vol. 16, no. 18, pp. 6939–6946, Sep. 2016.

[41] A. Anandkumar and L. Tong, "Type-based random access for distributed detection over multiaccess fading channels," *IEEE Trans. Signal Process.*, vol. 55, no. 10, pp. 5032–5043, Oct. 2007.

[42] F. Li, J. S. Evans, and S. Dey, "Decision fusion over noncoherent fading multiaccess channels," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4367–4380, Sep. 2011.

[43] J. A. Maya, L. Rey Vega, and C. G. Galarza, "Optimal resource allocation for detection of a Gaussian process using a MAC in WSNs," *IEEE Trans. Signal Process.*, vol. 63, no. 8, pp. 2057–2069, Apr. 2015.

[44] K. Cohen and A. Leshem, "Performance analysis of likelihood-based multiple access for detection over fading channels," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2471–2481, Apr. 2013.

[45] K. Cohen and D. Malachi, "A time-varying opportunistic multiple access for delay-sensitive inference in wireless sensor networks," *IEEE Access*, vol. 7, pp. 170475–170487, 2019.

[46] K. Cohen and A. Leshem, "Spectrum and energy efficient multiple access for detection in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 66, no. 22, pp. 5988–6001, Nov. 2018.

[47] J.-H. Ahn, O. Simeone, and J. Kang, "Wireless federated distillation for distributed edge learning with heterogeneous data," in *Proc. IEEE 30th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2019, pp. 1–6.

[48] Y. Sun, S. Zhou, and D. Gündüz, "Energy-aware analog aggregation for federated learning with redundant data," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–7.

[49] M. Frey, I. Bjelaković, and S. Stańczak, "Over-the-air computation for distributed machine learning and consensus in large wireless networks," in *Compressed Sensing in Information Processing*. Cham, Switzerland: Springer, 2022, pp. 401–434.

[50] S. Tang, H. Yin, C. Zhang, and S. Obana, "Reliable over-the-air computation by amplify-and-forward based relay," *IEEE Access*, vol. 9, pp. 53333–53342, 2021.

[51] M. Seif, R. Tandon, and M. Li, "Wireless federated learning with local differential privacy," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2020, pp. 2604–2609.

[52] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM J. Optim.*, vol. 19, no. 4, pp. 1574–1609, Jan. 2009.

[53] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for SVM," *Math. Program.*, vol. 127, no. 1, pp. 3–30, Mar. 2011.

[54] Y. Nesterov, "Introductory lectures on convex programming volume I: Basic course," *Lecture Notes*, vol. 3, no. 4, p. 5, 1998.

[55] Y. Nesterov, *Introductory Lectures on Convex Optimization* (International Series of Monographs on Physics), vol. 87. Cham, Switzerland: Springer, 2004.

[56] K. Cohen, A. Nedic, and R. Srikant, "On projected stochastic gradient descent algorithm with weighted averaging for least squares regression," *IEEE Trans. Autom. Control*, vol. 62, no. 11, pp. 5974–5981, Nov. 2017.

[57] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, "The million song dataset," in *Proc. 12th Int. Conf. Music Inf. Retr. (ISMIR)*, 2011, pp. 1–6.

[58] (2018). *New York Stock Exchange*. Kaggle. [Online]. Available: https://www.kaggle.com/daiearth22/predict-msft-via-linear-regression/data

[59] T. L. Gez and K. Cohen. (2023). *Code for Simulations for Paper: Subgradient Descent Learning Over Fading Multiple Access Channels With Over-the-Air Computation*. [Online]. Available: https://github.com/TamirGez/SGMA

[60] N. Z. Shor, *Minimization Methods for Non-Differential Functions* (Springer Series in Computational Mathematics). Berlin, Germany: Springer-Verlag, 1985.

TAMIR L. S. GEZ received the B.Sc. degree in electrical and computer engineering from the Ben-Gurion University of the Negev, Israel, in 2015. He is currently pursuing the M.Sc. degree with the School of Electrical and Computer Engineering. He has seven years of professional experience, as a Computer Vision Engineer in the high tech industry. His research interests include stochastic optimization and learning, with applications in large-scale systems.

KOBI COHEN (Senior Member, IEEE) received the B.Sc. and Ph.D. degrees in electrical engineering from Bar-Ilan University, Ramat Gan, Israel, in 2007 and 2013, respectively. He was with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, from August 2014 to July 2015, and with the Department of Electrical and Computer Engineering, University of California, Davis, from November 2012 to July 2014, as a Postdoctoral Research Associate. In October 2015, he joined the School of Electrical and Computer Engineering, Ben-Gurion University of the Negev (BGU), Be'er Sheva, Israel, where he is currently an Associate Professor. He is a member of the Cyber Security Research Center and the Data Science Research Center, BGU. His research interests include statistical inference and learning, signal processing, communication networks, decision theory and stochastic optimization with applications to large-scale systems, cyber systems, and wireless and wireline networks. Other selected Awards and Honors, include highlighting in top 50 popular paper list, IEEE Transactions on Wireless Communications, in 2019 and 2020, for paper: "Deep multi-user reinforcement learning for distributed dynamic spectrum access," highlighting in popular paper list, *IEEE Signal Processing Magazine*, in 2022, for paper: "Federated learning: A signal processing perspective," receiving the Best Paper Award from the International Symposium on Modeling and Optimization in Mobile, Ad hoc and Wireless Networks (WiOpt), in 2015, the Feder Family Award (second prize), awarded by the Advanced Communication Center at Tel Aviv University in 2011, the President Fellowship from 2008 to 2012, and top Honor List's prizes from Bar-Ilan University in 2006, 2010, and 2011. Since 2021, he has been an Associate Editor of the IEEE Transactions on Signal Processing and the IEEE Transactions on Wireless Communications.