Robust Federated Learning With Noisy Communication

Fan Ang, Li Chen, Nan Zhao, *Senior Member, IEEE,* Yunfei Chen, *Senior Member, IEEE,* Weidong Wang, F. Richard Yu, *Fellow, IEEE*

Abstract—**Federated learning is a communication-efficient training process that alternate between local training at the edge devices and averaging of the updated local model at the center server. Nevertheless, it is impractical to achieve perfect acquisition of the local models in wireless communication due to the noise, which also brings serious effect on federated learning. To tackle this challenge in this paper, we propose a robust design for federated learning to decline the effect of noise. Considering the noise in two aforementioned steps, we first formulate the training problem as a parallel optimization for each node under the expectation-based model and worst-case model. Due to the non-convexity of the problem, regularizer approximation method is proposed to make it tractable. Regarding the worst-case model, we utilize the sampling-based successive convex approximation algorithm to develop a feasible training scheme to tackle the un-available maxima or minima noise condition and the non-convex issue of the objective function. Furthermore, the convergence rates of both new designs are analyzed from a theoretical point of view. Finally, the improvement of prediction accuracy and the reduction of loss function value are demonstrated via simulation for the proposed designs.**

Index Terms—**Expectation-based model, federated learning, robust design, worst-case model.**

I. INTRODUCTION

FUTURE wireless computing applications demand higher bandwidth, lower latency and more reliable connections with numerous devices [1], [2]. With the burgeoning development of artificial intelligence technologies, the edge devices need to generate a sheer volume of raw data to be transmitted to the center, which results in excessive latency and privacy concerns [3], [4]. To solve this problem, federated learning has been proposed to encounter a paradigm shift from computing at the center to computing at the edge devices [5].

F. Ang, L. Chen, and W. Wang are with Department of Electronic Engineering and Information Science, University of Science and Technology of China.(e-mail: angfan@mail.ustc.edu.cn, {chenli87, wdwang}@ustc.edu.cn).

N. Zhao is with the School of Info. and Commun. Eng., Dalian University of Technology, Dalian 116024 China, and also with National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (email: zhaonan@dlut.edu.cn).

Y. Chen is with the School of Engineering, University of Warwick, Coventry CV4 7AL, U.K. (e-mail: Yunfei.Chen@warwick.ac.uk).

F.R. Yu is with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, K1S 5B6, Canada (email: richard.yu@carleton.ca).

Federated learning can be traced back as federated optimization to decouple the data acquisition and computation at the central server [6]. Federated optimization has recently been extended to deep learning platforms, which was known as federated learning [5], [7]. Federated learning was designed as an iterative process between distributed learning at the edge devices and averaging the updated local models at the central server. In contrast to the conventional centralized training, federated learning is more efficient in communication by uploading no raw data but only local models. To further improve the availability of enormous data from edge devices, federated learning was adopted in several scenes of future wireless networks [8]–[11]. Using federated learning and distributed MEC systems, the authors studied the trade-off between local computing and global aggregation under the given resource-constrained model in [8]. Moreover, the attractive property of lower latency drew attention to exploiting federated learning in latency-sensitive networks, such as vehicular networks [9], [11].

Due to the high-dimensional local model, as well as the long-term training process, the updating step of federated learning still consumes a lot of communication resources. The key issues are to reduce the overhead in the updating steps and to accelerate the training process. A series of research concentrating on reducing the overhead in the updating step was to transmit the compressed gradient vector via exploiting the quantization scheme [12], [13]. Another research focused on scheduling the edge devices to save the transmission bandwidth [14]–[18]. Specifically, some novel updating rules were worked out, which only allowed the edge devices with significant training improvement [15], or the fast responding devices [16], to transmit their gradient vectors in each uploading round. Arranging the adaptive maximum number of transmission-permitted edge devices was also an intelligent way when time was limited [18]. Furthermore, the authors developed a momentum method and cp-stochastic gradient descent algorithm to accelerate the training process for each edge device in local training in [13], [19]. Utilizing the different computation capability of each node, an asynchronous federated learning scheme was proposed to reduce the training delay in [20]–[22].

The aforementioned pioneering works are all based on the assumption that the received signals at both the central server and the edge nodes are perfectly detected. In practice, this is difficult in wireless communications due to imperfect channel estimation, feedback quantization, or delay in signal acquisition on fading channels. In other words, the noise is

indispensable during the training process. Furthermore, neural networks were proved to be not very robust to noise, which leads to the delay in the training process [23].

In conventional centralized learning, a branch of research has been dedicated to eliminate the effects of noise, among which several works used the denoising autoencoder to filter noise, such as contractive auto-encoders and denoising auto-encoders [24], [25], while others considered representing the effect of noise as imposing a penalty during the training process, known as the regularization scheme [26]–[30]. In particular, the addition of noise with infinitesimal variance as the input of training dataset was proved to be equivalent to the punishment on the norm of the weights for some training models [26], [27], whereas the added noise in the model was derived as appending a regularizer in the loss function which pushes the model to find the minima in the flat regions [28], [29]. Besides, the key idea of the Dropout method is to randomly drop units from the neural network during training to simulate the regularization [30].

The centralized learning only occurs on the center and the communication between the center and the edge node is the dataset updating and the optimal broadcasting. Consider the wireless communication, the noise will be brought into the dataset, which can be known as the input noise. However, the communication in federated learning is the local model updating and the global model broadcasting. Therefore, the noise will effect the received model, which can be known as weight noise. In all, the noise problem in centralized learning is different from the one in federated learning. A branch of research focus on improving the robustness of federated learning [31]–[33]. To solve the Byzantine failures, biased local datasets, and poisoning attacks, a robust aggregation rule was proposed which detects and discards bad or malicious local model updates at each training iteration [31]. Consider the characteristic of the training data, the authors proposed a new compression framework that is specifically designed to meet the requirements of the Federated Learning environment as the non-i.i.d. data [32]. In a heterogeneous environment, the authors develop an algorithm which matches the lower bound on the estimation error in dimension and the number of data points to solve the heterogeneous data distribution for federated learning [33]. The aforementioned works concentrate on the construct of the federated learning network rather than the communication noise. However, to the best of our knowledge, no communication noise reduction has been studied for federated learning and it is still an open problem.

Motivated by these observations, we propose a robust federated learning method to alleviate the effects of noise in the training process. Robust designs are first introduced using the expectation-based model and the worst-case model. More specifically, the former model is based on the statistical properties of the noise uncertainty and the latter model represents the fixed uncertainty sets of noise. Furthermore, the corresponding convergence analysis is provided to illustrate the performance of the proposed designs. The main contributions of this work are summarized as follows.

- **Robust design under the expectation-based model.** With the consideration of noise at the central server and the edge nodes, we formulate the training problem using the expectation-based model as a parallel optimization problem for each edge node. To handle the statistical property of noise, as well as the non-convexity of the objective function, we propose a regularization for loss function approximation (RLA) algorithm to approach the objective function and develop the corresponding training process. The proposed solution is superior to the conventional scheme that ignores noise in terms of both prediction accuracy and performance of loss function.

- **Robust design under the worst-case model.** The training problem under the worst-case model meets the challenges that are the unavailable maxima or minima noise condition and the non-convex issue of the objective function. We solve the former problem via the sampling method and tackle the latter one by utilizing the successive convex approximation (SCA) algorithm to generate a feasible descent direction for the training process. The simulation results show that the proposed design outperforms the conventional one for prediction accuracy and values of loss function.

- **Convergence analysis for the proposed designs.** The convergent property of all proposed designs are derived. Specifically, it is found that the proposed training process under the expectation-based model converges at the equivalent rate to the centralized training scheme that ignores noise, and the convergent property of proposed robust design under the worst-case model outperforms the conventional centralized one.

The remainder of the paper is organized as follows. Section II introduces the system model of the federated learning considering noise. Section III presents the formulated problem under the expectation-based model and the worst-case model. The robust design under the expectation-based model and its convergence analysis are developed in Section IV. Section V shows the robust design under the worst-case model and the corresponding convergence analysis. Simulation results are provided in Section VI.

Throughout the paper, we use boldface lowercase to refer to vectors, and lowercase to refer to scalar. Let $(\cdot)^T$ denote the transpose of a vector. Let $|\cdot|$ denote size of the set, $\mathbf{0}$ denotes zero matrix, and $\mathbf{I}$ denotes unit matrix. $\mathbb{E}\{\cdot\}$ is the expectation function.

II. SYSTEM MODEL

We consider a distributed learning system consisting of a single central server and N edge nodes, as shown in Fig. 1. A shared learning process with the global model $\mathbf{w}$ is trained collaboratively by the edge nodes. Each node collects a fraction of labelled training datasets $\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_N$.

The loss function is to facilitate the learning and we define it as $f_j(\mathbf{w}, \mathbf{x}_j, y_j)$ for each data sample j, which consists of the input vector $\mathbf{x}_j$ and the output scalar y_j. For convenience, we rewrite $f_j(\mathbf{w}, \mathbf{x}_j, y_j)$ as $f_j(\mathbf{w})$. The typical loss function in federated learning is shown in Table I. Then the global loss function on all distributed datasets can be defined as
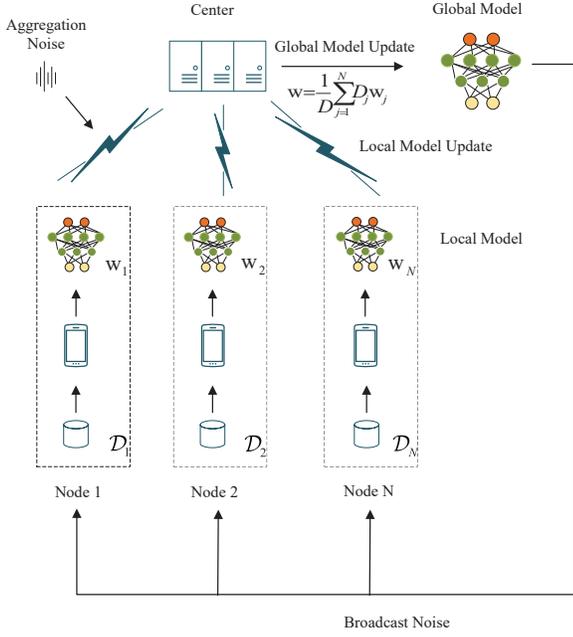
Fig. 1. Federated learning with wireless communication.

TABLE I
SOME EXAMPLES OF LOSS FUNCTION

Model	Loss function
Squared-SVM	$\frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{2}\max{0; 1 - y_j\mathbf{w}^T x_j}$
Linear regression	$\frac{1}{2}\|1 - y_j\mathbf{w}^T x_j\|^2$
K-means	$\frac{1}{2}\min_l \|x_j - \mathbf{w}_l\|^2$
Cross-Entropy	$-\sum y_c p(y = c\|x, \mathbf{w})$

$$F(\mathbf{w}) = \frac{\sum_{j \in \mathcal{D}} f_j(\mathbf{w})}{|\cup_i \mathcal{D}_i|}, \tag{1}$$

where $|\cdot|$ denotes the size of the datasets and each dataset $\mathcal{D}_i$ satisfies $\mathcal{D}_i \cap \mathcal{D}_j = \emptyset$ when $i \neq j$, $i, j = 1, 2, ..., N$. The training target is to minimize the global loss function $F(\mathbf{w})$ according to the distributed learning, i.e., to find

$$\mathbf{w}^* = \arg\min F(\mathbf{w}). \tag{2}$$

One way to search for the optimal $\mathbf{w}^*$ is to update the datasets of the distributed nodes, which only contains the input vector $\mathbf{x}_j$ and the output scalar y_j, called centralized learning. The center completes the training process using the whole datasets, and broadcasts the optimal model from (1) and (2) to all nodes. However, the datasets are generally large in machine learning. Therefore, centralized learning requires numerous communication resources to collect the whole datasets. In other words, the training process will be limited by the communication rates.

Another way to solve (2) is a distributed manner as demonstrated in Fig. 1, which focuses on the model-averaging for the global model $\mathbf{w}$, called federated learning. The global loss function $F(\mathbf{w})$ cannot be directly computed without sharing

datasets among all edge nodes in federated learning. The federated learning algorithm alternates between two stages. In the first stage, the local models at each node are sent to the center for model-averaging via wireless links, and the center updates the global model $\mathbf{w}$. In the second stage, the center broadcasts the current model to all edge nodes at each iteration. Based on the received global model $\mathbf{w}$, each node updates its own model to minimize the local loss function using its own dataset. The updating rules follow:

$$\text{Center}: \mathbf{w} = \frac{\sum_{j=1}^{N} D_j \mathbf{w}_j}{D}, \tag{3a}$$

$$\text{Local}: \mathbf{w}_j = \arg\min F_j(\mathbf{w}), j = 1, 2, ..., N, \tag{3b}$$

where $\mathbf{w}_j$ denotes the local model of node j, D denotes the size of the whole datasets $\cup_j \mathcal{D}_j$, D_j denotes the size of the dataset $\mathcal{D}_j$, $j = 1, 2, ..., N$, $F_j(\mathbf{w})$ is the local loss function of node j with dataset $\mathcal{D}_j$, and can be written as

$$F_j(\mathbf{w}) = \frac{1}{|\mathcal{D}_j|}\sum_{i \in D_j} f_i(\mathbf{w}) = \frac{1}{D_j}\sum_{i \in \mathcal{D}_j} f_i(\mathbf{w}). \tag{4}$$

The training process requires the iterations between (3b) and (3a) until convergence, and each node can obtain the optimal model $\mathbf{w}^*$.

Since the center and each node are connected using wireless links, it inevitably introduces noise. Therefore, the received signal has the aggregation noise at the center via local updating and the broadcasted global model with noise in each iteration for the node j can be modeled as

$$\text{Aggregation}: \tilde{\mathbf{w}} = \mathbf{w} + \Delta\tilde{\mathbf{w}},$$
$$\text{Broadcast}: \tilde{\mathbf{w}}_j = \tilde{\mathbf{w}} + \Delta\tilde{\mathbf{w}}_j, j = 1, 2, ..., N, \tag{5}$$

where $\Delta\tilde{\mathbf{w}}$ refers to the aggregation noise at the center, and $\Delta\tilde{\mathbf{w}}_j$ refers to the broadcast noise for node j.

The imperfect estimation is a major problem in wireless communication. In federated learning, it leads to the changing of optimization in the local update process. The noise in estimation error of the model will make the output data point blurred and make the training difficult to fit the input data point precisely for neural networks. Furthermore, the neural networks were proved to be not robust against noise. In other words, the performance of the learning scheme may be significantly reduced by noise. To solve this problem, robust design is proposed to ensure a certain level of the performance under the uncertainty model.

III. PROBLEM FORMULATION

In this section, we formulate the robust problem using two robust models. According to the different characteristics of the two robust models, the corresponding problem is totally different. We write the corresponding problems in the following.

The aggregation noise and broadcasted noise in (5) can be modelled as the stochastic and the deterministic. The former

is the expectation-based model and the latter is the worst-case model. According to that, each node updates its own model with a different initial point, $\tilde{\mathbf{w}}_j$, the corresponding local loss function $F_j(\mathbf{w})$ is rewritten as $F_j(\tilde{\mathbf{w}}_j)$, $j = 1, 2, ..., N$, and the global loss function $F(\mathbf{w})$ is rewritten as $F(\tilde{\mathbf{w}})$. The iteration process still follows (3a) and (3b).

A. Training Under Expectation-based Model

Expectation-based model is a stochastic method to represent the random condition, which can only be used when statistical properties of noise are available [34]. The stochastic model assumes that the estimation value is a random quantity and its instantaneous value is unknown, but its statistics property, such as the mean and the covariance, is available. In this case, the robust design usually aims at optimizing either the long-term average performance or the outage performance. The corresponding robust model is called the expectation-based model and defined as follows.

Definition 1 (Expectation-based Robust Model [35], [36]): The expectation-based robust model refers to the stochastic property of noise as shown in Fig. 2 (a). For node j, the entries of the uncertainty vector are assumed to be Gaussian distributed with $\mathbb{E}\{\Delta\tilde{\mathbf{w}}_j\} = \mathbf{0}$, and $\mathbb{E}\{\Delta\tilde{\mathbf{w}}_j \cdot \Delta\tilde{\mathbf{w}}_j^T\} = \sigma_j^2\mathbf{I}$, $j = 1, 2, ..., N$, $\mathbb{E}\{\Delta\tilde{\mathbf{w}}_i \cdot \Delta\tilde{\mathbf{w}}_j^T\} = 0$, $i \neq j$ and the aggregation noise at the center is assumed to satisfy $\mathbb{E}\{\Delta\tilde{\mathbf{w}}\} = \mathbf{0}$, and $\mathbb{E}\{\Delta\tilde{\mathbf{w}} \cdot \Delta\tilde{\mathbf{w}}^T\} = \sigma^2\mathbf{I}$.

With the assumption that the aggregation noise $\Delta\tilde{\mathbf{w}}$ and the broadcast noise $\Delta\tilde{\mathbf{w}}_j$ are Gaussian, we can obtain another summed Gaussian noise as $\Delta\mathbf{w}_j$ so that the received value for node j can be expressed as [37]

$$\tilde{\mathbf{w}}_j = \mathbf{w} + \Delta\mathbf{w}_j, j = 1, 2, ..., N, \tag{6}$$

and $\Delta\mathbf{w}_j$ is Gaussian with $\mathbb{E}\{\Delta\mathbf{w}_j\} = \mathbf{0}$, and $\mathbb{E}\{\Delta\mathbf{w}_j \cdot \Delta\mathbf{w}_j^T\} = \sigma_{e_j}^2\mathbf{I}$, $j = 1, 2, ..., N$, where $\sigma_{e_j}^2 = \sigma^2 + \sigma_j^2$.
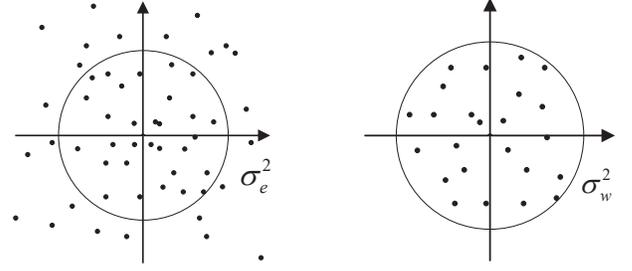
Therefore, using the stochastic property of noise, we should focus on improving the stochastic performance for the network. Furthermore, the optimization object in federated learning is to find the local optimal model $\mathbf{w}_j$ in (3b) and to utilize the combination method to find the global optimal model $\mathbf{w}$ in (3a).

Since the combination method is determinate, we only need to optimize the local model $\mathbf{w}_j$ for each node. Based on the aforementioned analysis, we formulate the robust training problem using the expectation-based model for each node as

$$
\begin{aligned}
\mathrm{P}_1 : \min_{\mathbf{w}} \ & \mathbb{E}\|F_j(\mathbf{w} + \Delta\mathbf{w}_j)\|^2 \\
s.t. \ & \mathbb{E}\{\Delta\mathbf{w}_j\} = \mathbf{0}, j = 1, 2, ..., N, \\
& \mathbb{E}\{\Delta\mathbf{w}_j \cdot \Delta\mathbf{w}_j^T\} = \sigma_{e_j}^2\mathbf{I}, j = 1, 2, ..., N,
\end{aligned}
\tag{7}
$$

where the constraints in P_1 represent the stochastic characteristic of noise from imperfect estimation in wireless communication.

We aim at improving the stochastic performance for the training process. Due to the expectation calculation, the

(a) Noise under expectation-based model in two-dimensional space.

(b) Noise under worst-case model in two-dimensional space.

Fig. 2. Noise under expectation-based model and worst-case model in two-dimensional space

objective function is non-convex. To tackle this challenge, we consider adding the regularizer into the loss function to approximate the objective function and to represent the effect of noise. We provide the corresponding federated learning process in Section IV.

B. Training Under Worst-Case Robust Model

In contrast to the expectation-based model, the worst-case model is a deterministic method to represent the instantaneous condition, which has fixed uncertainty sets, and to maximize the performance under the worst uncertainty [38], [39]. Using the worst-case robust design, we can guarantee a performance level for any value of estimation realization in the uncertainty region. It is applied to design which requires strict constraints, and is more suitable for characterizing instantaneous estimation value with errors. The worst-case approach assumes that the actual estimation value lies in the neighborhood of the uncertainty region with a known nominal estimation value. The size of this region represents the amount of estimation value uncertainty, i.e., the bigger the region is, the more uncertainty there is. We show the brief definition of the worst-case model as follows.

Definition 2 (Worst-Case Robust Model [35], [36]): The worst-case robust model assumes that the estimation lies in a known set of possible values shown as Fig. 2 (b), which can not be exactly known. The norm of the uncertainties vector $\Delta\tilde{\mathbf{w}}$ and $\Delta\tilde{\mathbf{w}}_j$ are bounded by the spherical region, which can be expressed as

$$
\begin{aligned}
\|\Delta\tilde{\mathbf{w}}_j\|^2 &\leq \sigma_j^2, j = 1, 2, ..., N, \\
\|\Delta\tilde{\mathbf{w}}\|^2 &\leq \sigma^2,
\end{aligned}
\tag{8}
$$

where $\sigma_j^2 \geq 0$ denotes the radius of the spherical uncertainty region of the broadcast noise, while $\sigma^2 \geq 0$ denotes the aggregation noise.

Consider the superposition of noise, the uncertainty is expanded to the larger region with the size $\sigma_j^2 + \sigma^2$. Therefore, we reformulate the received value at node j as [37]

$$\tilde{\mathbf{w}}_j = \mathbf{w} + \Delta\mathbf{w}_j, j = 1, 2, ..., N, \tag{9}$$

where $\Delta\mathbf{w}_j$ denotes the whole noise and satisfies $\|\Delta\mathbf{w}_j\|^2 \leq \sigma_{w_j}^2, j = 1, 2, ..., N$.

Similarly, the optimization is to find the local optimal model $\mathbf{w}_j$ in (3b), and follows the aggregation rules in (3a). Therefore, we formulate the robust training problem under the worst-case model as a min-max problem for each node

$$\mathrm{P}_2 : \min_{\mathbf{w}} \max_{\Delta\mathbf{w}_j} F_j(\mathbf{w} + \Delta\mathbf{w}_j)$$
$$s.t.\ \|\Delta\mathbf{w}_j\|^2 \leq \sigma_{w_j}^2, j = 1, 2, ..., N, \tag{10}$$

where the constraints in P_2 represent the noise lies in a spherical region with radius $\sigma_{w_j}^2$.

One challenge to solve the problem is that the worst condition may not be available. The other is the non-convex objective function. We settle the challenges using the sampling method and the SCA algorithm to generate a feasible descent direction for the learning process in Section V.

IV. ROBUST DESIGN USING EXPECTATION-BASED MODEL

In this section, we consider the robust design in federated learning using the expectation-based model. We propose the corresponding RLA algorithm to represent the effects of noise for the expectation-based model so that the local optimal model can be found via optimization.

A. Proposed Training Algorithm

We first model the noise under the expectation-based noise model, which is a stochastic method to represent the random condition, as shown in P_1. We aim at optimizing the average performance based on the expectation-based model. However, the random noise results in the non-convexity property and uncertainty value of the local loss function.

To solve this problem, we propose the RLA to approximate the non-convexity local loss function and utilize the distributed gradient descent to find the optimal global model. The approximation method is inspired by previous works where training with noise was approximated via regularization to enhance the robust of neural networks [28]. We give a brief introduction in the following.

Lemma 1: Training with noise is equal to adding a regularizer $\Omega(\mathbf{w})$, which can be expressed as

$$F(\tilde{\mathbf{w}}) \approx F(\mathbf{w}) + \lambda\Omega(\mathbf{w}), \tag{11}$$

where $F(\cdot)$ denotes the loss function, $\Omega(\cdot)$ is the designed function, $\mathbf{w}$ is the learning model, $\tilde{\mathbf{w}}$ represents the learning model including noise, and λ is a constant.

Proof: Refer to [40]. ∎

There are many regularization strategies in the aforementioned works [26]–[29]. However, there is no specific regularizer that is universally better than any others for the learning algorithm. In other words, there is no best form of regularization. We need to develop a specific form of $\Omega(\mathbf{w})$ using the expectation-based model.

Motivated by this observation, we propose a new regularization term to approximate the original loss function for federated learning in the training process. Using the expectation-based model, we intend to reduce the impact of noise for the training process. Due to the stochastic property of noise, we aim at optimizing the average performance in P_1. We propose the corresponding training problem in the following.

Proposition 1 (Robust Training Under Expectation-based Model): The robust training problem under the expectation-based model in P_1 for each node can be reformulated as

$$\mathrm{P}_3 : \min_{\mathbf{w}} F_j^e(\mathbf{w}), \tag{12}$$

where $F_j^e(\mathbf{w})$ denotes the new loss function for node j and can be written as

$$F_j^e(\mathbf{w}) = F_j(\mathbf{w}) + \sigma_e^2\|\nabla F_j(\mathbf{w})\|^2. \tag{13}$$

Proof: Under the expectation-based model, we can obtain the objective function of P_1 utilizing Taylor expansion according to the work in [27] so that the objective loss function of the optimization problem is written as

$$\mathbb{E}\|F_j(\mathbf{w} + \mathbf{w}_j)\|^2 = \mathbb{E}\|F_j(\mathbf{w}) + \mathbf{w}_j\nabla F_j(\mathbf{w}) + o(\mathbf{w})\|^2$$
$$\approx \mathbb{E}\|F_j(\mathbf{w})\|^2 + \sigma_e^2\|\nabla F_j(\mathbf{w})\|^2. \tag{14}$$

The first term $\mathbb{E}\|F_j(\mathbf{w})\|^2$ refers to the training process with perfect estimation in (3b), and the second term is the additional cost of the loss function in training, which is determined by noise. Therefore, the objective loss function under the expectation-based equals adding the regularizer $\sigma_e^2\|\nabla F_j(\mathbf{w})\|^2$. ∎

Remark 1: The penalty over the first-order of the loss function yields a preference for mapping f that are invariant locally at the training points and drop the global model $\mathbf{w}$ into the flat region.

To solve the training problem in (12), we utilize the gradient descent algorithm to find the optimal local model $\mathbf{w}$ for each node, and the details are shown as follows.

In each iteration, the local update at each node is performed based on the previous iteration and the first gradient of the proposed loss function, and the center aggregates the distributed models to find the optimal global model for the next iteration. Therefore, the update rules of the gradient descent can be written as:

$$\text{Center}: \mathbf{w}^{t+1} = \frac{\sum_{j=1}^N D_j\mathbf{w}_j^{t+1}}{D}, \tag{15a}$$

$$\text{Local}: \mathbf{w}_j^{t+1} = \mathbf{w}^t - \eta\nabla F_j^e(\mathbf{w}^t), j = 1, 2, ..., N, \tag{15b}$$

where η is the step size for all nodes. The iteration is executed and it will stop if a specific condition is satisfied. This process is illustrated in Algorithm 1.

To solve the robust problem, we develop the training process by adding the regularizer to approximate the original loss

Algorithm 1 Distributed Gradient Descent Learning Algorithm Under Expectation-based Model

---

Input: η
Output: w
1: number of iteration time $t = 0$.
2: Update **w**:
3: **repeat**
4:     Update $\mathbf{w}_j^{t+1}$, $j = 1, 2, ..., N$ according to $\mathbf{w}_j^{t+1} = \mathbf{w}^t - \eta \nabla F_j^e(\mathbf{w}^t)$
5:     Update $\mathbf{w}^{t+1}$ according to the aggregated rules
6:     $t \leftarrow t + 1$
7: **until** converge

---

function. We transfer the stochastic and non-convex problem into a deterministic and convex problem so that we can utilize the gradient descent method to find the optimal global model **w**. The corresponding performance is shown through simulation in Section VI.

B. Convergence Analysis

In this subsection, we derive the convergence property of the proposed design under the expectation-based model. To obtain the convergence rate of the proposed scheme under the expectation-based model, we first prove that the proposed federated learning is equivalent to a centralized learning, and then derive the corresponding convergence rate.

We start with the essential assumption of the loss function, which can be satisfied normally.

Assumption 1: We assume the following conditions for the loss function of all nodes:
(1) $F_i(\mathbf{w})$ is *convex*,
(2) $F_i(\mathbf{w})$ is $L-Lipschitz$, i.e. $\|F_i(\mathbf{w}) - F_i(\mathbf{w}')\| \leq L\|\mathbf{w} - \mathbf{w}'\|$ for any $\mathbf{w}, \mathbf{w}'$,
(3) $F_i(\mathbf{w})$ is $\beta-smooth$, i.e. $\|\nabla F_i(\mathbf{w}) - \nabla F_i(\mathbf{w}')\| \leq \beta\|\mathbf{w} - \mathbf{w}'\|$ for any $\mathbf{w}, \mathbf{w}'$.

Then, we give a brief definition of centralized learning.

Definition 3 (Centralized learning problem under expectation-based model): Given the proposed local loss function in (13), the global loss function can be written as

$$F^e(\mathbf{w}) = \frac{\sum_{i=1}^N D_i F_i^e(\mathbf{w})}{D},\qquad(16)$$

so that we aim at minimizing $F^e(\mathbf{w})$ at the center by using the same whole datasets. Therefore, the centralized learning problem is to find the optimal global model as

$$\mathrm{P}_4 : \min_{\mathbf{w}} F^e(\mathbf{w}).\qquad(17)$$

The optimization can be easily solved by using the gradient descent, and the center completes the iteration until the specific condition is met. We derive that the proposed federated learning is equivalent to the centralized learning problem under the expectation-based model as follows.

Lemma 2: Given P_1 and P_4 under the expectation-based model, the proposed federated learning is equal to the centralized learning for each iteration t, $t = 0, 1, 2, ...$, which can be written as

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \eta \nabla F^e(\mathbf{w}^t).\qquad(18)$$

Proof: Considering the global aggregation, we can obtain that

$$\begin{aligned}\mathbf{w}^{t+1} &= \frac{\sum_{j=1}^N D_j \mathbf{w}_j^{t+1}}{D}\\ &= \frac{\sum_{j=1}^N D_j(\mathbf{w}^t - \eta F_j^e(\mathbf{w}^t))}{D}\\ &= \frac{\sum_{j=1}^N D_j \mathbf{w}^t}{D} - \eta \frac{\sum_{j=1}^N D_j F_j^e(\mathbf{w}^t)}{D}\\ &= \mathbf{w}^t - \eta \nabla F^e(\mathbf{w}^t).\end{aligned}\qquad(19)$$

$\blacksquare$

To prove the convergence of the proposed distributed learning, we only need to derive that the equivalent centralized learning is convergent.

Lemma 3: Given the original loss function under Assumption 1, there exist constants η and β so that the loss function $f_j(\mathbf{w})$, $j = 1, 2, ..., N$ satisfies that

$$F_i(\mathbf{w}^t) - F_i(\mathbf{w}^*) \leq \|\mathbf{w}^0 - \mathbf{w}^*\|^2 \cdot \frac{1}{\eta\left(1 - \frac{\beta\eta}{2}\right)} \cdot \frac{1}{t},\quad(20)$$

where $\mathbf{w}^0$ is the initialization point of **w**.

Proof: Refer to [41]. $\blacksquare$

Lemma 4: $F(\mathbf{w})$ is $convex$, $L-Lipschitz$ and $\beta-smooth$.

Proof: We can obtain that $F(\mathbf{w})$ is the linear combination of $F_i(\mathbf{w})$ via (16). Straightforwardly from the convexity property, this lemma holds. $\blacksquare$

Proposition 2 (Convergence Under Expectation-based Model): Algorithm 1 yields the following convergence property for the optimization of the global loss function under the expectation-based model

$$F^e(\mathbf{w}^t) - F^e(\mathbf{w}^*) \leq \|\mathbf{w}^0 - \mathbf{w}^*\|^2 \cdot \frac{1}{\eta\left(1 - \frac{(1+\lambda\sigma_e^2)\beta\eta}{2}\right)} \cdot \frac{1}{t},\qquad(21)$$

where $\mathbf{w}^0$ is the initialization point of **w**. It means the convergence rate is $\mathcal{O}(1/t)$.

Proof: The details are shown in Appendix A. $\blacksquare$

Remark 2: The proposed robust design under the expectation-based model converges at $\mathcal{O}(1/t)$. The convergence property as (21) is reduced to the one in (20) as $\sigma_e^2 = 0$, i.e., it is equivalent to the convergence property that is training without noise. The convergence rate will decrease with the increase in σ_e^2 and the proposed design cannot converge when $\left(1 - (1 + \sigma_e^2)\beta\eta/2\right) \leq 0$ specifically. The comparison between the proposed design and the centralized training is simulated specifically in Section VI.

Remark 3: When the proposed loss function does not satisfy the convex assumption, i.e., the loss function is non-convex. The convergence rate of the proposed algorithm is hard to prove in a closed form. However, the authors in [42] show

that the GD algorithm will find a local optimal result. It was called a saddle point where $\nabla F^e(\mathbf{w}^t) = 0$.

V. ROBUST DESIGN USING WORST-CASE MODEL

In this section, we solve the optimization problem using the worst-case model. To solve the uncertainty of noise and the non-convexity problem, we utilize the sampling-based SCA method to represent noise and approximate the objective loss function. We then propose the training process for the robust federated learning and finally derive the convergence property of the proposed design.

A. Proposed Training Algorithm

The training process is proposed to solve the learning problem under the worst-case model. We utilize the sampling-based SCA method to approximate the original objective function, and develop the corresponding updating rules.

The feasible sets of both the local model and the noise are convex sets, and there always exists a saddle point. However, the unavailability of noise results in that the finding of the global minimum point is, in general, NP hard. Therefore, the objective problem faces the main issues: i) the impossibility to estimate accurate value of noise of the worst condition; ii) the non-convexity of the objective functions leading to unavailable optimization.

Considering the uncertainty of noise, it is often possible to obtain a sample of the random noise, either from past data or from computer simulation as shown in [43]. Consequently, one may consider an approximate solution to the problem based on sampling, known as the sample average approximation (SAA) method, and we give a brief introduction as follows.

Lemma 5: The SAA method is to find the optimal x for the stochastic objective in the optimization problem as,

$$x^* = \min \mathbb{E}[f(x;\xi)], \qquad (22)$$

where $f(x;\xi)$ is a given function and affected by the random vector ξ which follows the distribution V. However, the distribution V is unknown, and only sample values of the random vector ξ are available. To solve this problem, the SAA approach approximates the problem by solving

$$\hat{x}^* = \min \frac{1}{N}\sum_{j=1}^{N} f(x;\xi^j), \qquad (23)$$

where ξ^j is the random sample of the random vector ξ, and the collection of N realizations satisfies independent and identically distributed.

Proof: Refer to [43]. ■

Motivated by this method, we consider sampling noise $\Delta\mathbf{w}_j$ in the objective function $F_j(\mathbf{w}+\Delta\mathbf{w}_j)$, and can easily obtain that the worst condition of noise occurs on the boundary. Based on the above consideration, we propose the sampling-based method. At each iteration t of each node, a new realization of the noise $\Delta\mathbf{w}_j^t$ is obtained and the optimization of objective functions is updated via the loss function as follows,

$$F_j(\mathbf{w}+\Delta\mathbf{w}_j) = F_j(\mathbf{w}+\Delta\mathbf{w}_j^t), t = 1, 2, ... \qquad (24)$$

where $\Delta\mathbf{w}_j^t$ satisfies $\|\Delta\mathbf{w}_j^t\|^2 = \sigma_w^2$.

It provides a simple way to approach the objective function under the perfect estimation, but the non-convexity of the objective function is still not resolved. To tackle this challenge, we utilize the SCA scheme to maintain the convexity of the objective functions.

Lemma 6: The SCA algorithm is proposed to approximate an arbitrarily function $f(x)$ by expansion around x^t which is a definite point in the feasible set. It can be simply written as

$$f(x) \approx \tilde{f}(x, x^t) = \rho^t f(x) + (1-\rho^t)\langle x - x^t, g(x^t)\rangle, \qquad (25)$$

where $\rho^t \in (0,1]$ is a sequence, and $g(x^t)$ is the weight average of the first gradient and can be expressed as

$$g(x^t) = (1-\rho^t)g(x^{t-1}) + \rho^t\nabla f(x^t). \qquad (26)$$

Proof: Refer to [44]. ■

With the consideration of SAA and SCA methods, we propose the sampling-based SCA algorithm to solve the robust training problem under the worst-case model of P$_2$ in the following.

Proposition 3 (Robust Training Under Worst-case Model): For the robust training problem under the worst-case model in P$_2$,the optimization problem of each node can be reformulated as

$$\mathrm{P}_5 : \min_{\mathbf{w}} F_j^w(\mathbf{w};\mathbf{w}^t, \Delta\mathbf{w}_j^t) \qquad (27)$$

where $\Delta\mathbf{w}_j^t$ is a sequence by sampling the noise $\Delta\mathbf{w}_j$ satisfying that $\|\Delta\mathbf{w}_j^t\|^2 = \sigma_w^2$, $F_j^w(\mathbf{w};\mathbf{w}^t, \Delta\mathbf{w}_j^t)$ is denoted as the loss function for the node j, and expressed as

$$F_j^w(\mathbf{w};\mathbf{w}^t, \mathbf{w}_j^t) = \rho^t F_j(\mathbf{w}+\Delta\mathbf{w}_j^t) + \lambda\|\mathbf{w}-\mathbf{w}^t\|^2 + (1-\rho^t)\langle \mathbf{w}-\mathbf{w}^t, G_j^{t-1}\rangle, \qquad (28)$$

and G_j^t is an accumulation vector updated recursively according to

$$G_j^t = (1-\rho^t)G_j^{t-1} + \rho^t\nabla_{\mathbf{w}_j}F_j(\mathbf{w}+\Delta\mathbf{w}_j^t), \qquad (29)$$

with $\rho^t \in (0,1]$ being a sequence to be properly chosen ($\rho^0 = 1$), $t = 0, 1, ...$

Proof: The details are shown in Appendix B. ■

The objective function of node j is obtained by replacing the uncertain value with a suitably chosen incremental sample estimate of it. Specifically, a random noise $\Delta\mathbf{w}_j^t$ is realized at iteration t, and the loss function for node j is expressed as follows:

Remark 4: Generally speaking, each node minimizes a sample approximation of the original unstable function. The first term in (28) refers to the sample objective function. The second term refers to the cost which controls the pace for each iteration. The vector G_j^t in the last term represents

Algorithm 2 Distributed Gradient Descent Learning Algorithm Under Worst-case Model

---

Input: $\Delta \mathbf{w}$, γ
Output: $\mathbf{w}$

1: number of iteration time $t = 0$.
2: Update $\mathbf{w}$:
3: **repeat**
4:   Update $\mathbf{w}_j^{t+1}$, $j = 1, 2, ..., N$ according to $\mathbf{w}_j^{t+1} = \mathbf{w}^t + \gamma^{t+1}\left(\mathbf{w}_j^w(\mathbf{w}^t, \Delta \mathbf{w}_j^t) - \mathbf{w}^t\right)$
5:   Update $\mathbf{w}^{t+1}$ according to the aggregated rules
6:   $t \leftarrow t + 1$
7: **until** converge

---

the incremental estimate of the unknown $\nabla_{\mathbf{w}} F_j(\mathbf{w} + \Delta \mathbf{w}_j^t)$ by samples collection over the iterations. Provided that the parameter ρ^t is properly chosen, and the estimation is expected to be more and more accurate as t increases.

Due to the involving of the past optimized model $\mathbf{w}^t$, we consider utilizing the conditional gradient descent method for each node. Similarly, we aggregate the local update at the center and broadcast the new global model for next iteration. The aggregated model $\mathbf{w}^{t+1}$ should be broadcasted to all nodes and it is used to complete the next iteration until meets the specific condition. Given $\mathbf{w}_j^w(\mathbf{w}^t, \Delta \mathbf{w}_j^t)$, the iteration rule is briefly written as follows.

$$\text{Center}: \mathbf{w}^{t+1} = \frac{\sum_{j=1}^N D_j \mathbf{w}_j^{t+1}}{D}, \tag{30a}$$

$$\text{Local}: \mathbf{w}_j^{t+1} = \mathbf{w}^t + \gamma^{t+1}\left(\mathbf{w}_j^w(\mathbf{w}^t, \Delta \mathbf{w}_j^t) - \mathbf{w}^t\right), \tag{30b}$$

where $\gamma^t \in (0, 1]$, $t = 0, 1, 2, ...$. The iteration follows the process illustrated in Algorithm 2.

We develop the training process utilizing the sampling-based SCA algorithm to approximate the training objective function for each node. With the iteration between the conditional gradient descent and the aggregation step, we can obtain the optimal global model $\mathbf{w}$. The corresponding performance is shown through simulations in Section VI.

B. Convergence Analysis

To obtain the convergence rate of the proposed scheme under the worst-case model, we similarly prove that the proposed federated learning is equal to the centralized learning, and then derive the corresponding convergence rate.

Without loss of generality, we first give some assumptions before the further analysis.

Assumption 2: We assume the following for the loss function of all nodes
(1) $\tilde{F}_i(\mathbf{w}, \Delta \mathbf{w})$ is convex,
(2) $\tilde{F}_i(\mathbf{w}, \Delta \mathbf{w})$ is $L - Lipschitz$, i.e., $\|\tilde{F}_i(\mathbf{w}, \Delta \mathbf{w}) - \tilde{F}_i(\mathbf{w}', \Delta \mathbf{w}')\| \leq L\|\mathbf{w} - \mathbf{w}'\|$ for any $\mathbf{w}$, $\mathbf{w}'$ and $\Delta \mathbf{w}$,
(3) $\tilde{F}_i(\tilde{\mathbf{w}}, \Delta \mathbf{w})$ is $\beta - smooth$, i.e., $\|\nabla \tilde{F}_i(\mathbf{w}, \Delta \mathbf{w}) - \nabla \tilde{F}_i(\mathbf{w}', \Delta \mathbf{w})\| \leq \beta\|\mathbf{w} - \mathbf{w}'\|$ for any $\mathbf{w}$, $\mathbf{w}'$ and $\Delta \mathbf{w}$.

We first develop a brief introduction of the optimization problem in centralized learning under worst-case model.

Definition 4 (Centralized learning problem under worst-case model): Given the local loss function in (28), we can obtain that the global loss function in iteration t is

$$F^w(\mathbf{w}; \mathbf{w}^t, \Delta \mathbf{w}^t) = \frac{\sum_{i=1}^N D_i F_i^w(\mathbf{w}; \mathbf{w}^t, \Delta \mathbf{w}^t)}{D}, \tag{31}$$

where $\mathbf{w}^t$ is the global model in last iteration $t-1$, and $\Delta \mathbf{w}^t$ denotes the sampled noise in last iteration $t-1$, which satisfies $\|\Delta \mathbf{w}^t\|^2 \leq \sigma_w^2$.

Due to that we aim at minimizing the global loss function, the centralized learning problem is to find the optimal global parameter in iteration t, i.e.,

$$\mathrm{P}_6 : \min_{\mathbf{w}} F^w(\mathbf{w}; \mathbf{w}^t, \Delta \mathbf{w}^t). \tag{32}$$

The problem can be solved by the SCA algorithm, and the center completes the iteration until meets the specific condition.

In the following, we first prove that the federated learning is equivalent to the centralized learning under the worst-case model. Secondly, we show that the centralized learning under the worst-case model is convergent.

Lemma 7: Given problem under Assumption 2, suppose that $\tau > 0$ and step size γ^t and ρ^t are chosen as $\gamma_i^t = \gamma^t = \frac{1}{t^\alpha}$ and $\rho_i^t = \rho^t = \frac{1}{t^\beta}$, $0.5 < \beta < \alpha < 1$ so that the distributed learning equals to the centralized learning at iteration t, which is expressed as

$$\mathbf{w}^w(\mathbf{w}^t, \Delta \mathbf{w}^t) = \arg \min F^w(\mathbf{w}; \mathbf{w}^t, \Delta \mathbf{w}^t), \tag{33}$$

and the global model aggregation obeys the updating rules as

$$\mathbf{w}^{t+1} = \mathbf{w}^t + \gamma^{t+1}(\mathbf{w}^w(\mathbf{w}^t, \Delta \mathbf{w}^t) - \mathbf{w}^t). \tag{34}$$

Proof: For any iteration $t + 1$, $\mathbf{w}^{t+1}$ satisfies

$$\begin{aligned} \mathbf{w}^{t+1} &= \frac{\sum_{j=1}^N D_j \mathbf{w}_j^{t+1}}{D} \\ &= \frac{\sum_{j=1}^N D_j \left[\mathbf{w}^t + \gamma^{t+1}\left(\mathbf{w}_j^w(\mathbf{w}^t, \Delta \mathbf{w}_j^t) - \mathbf{w}^t\right)\right]}{D} \\ &= \frac{\sum_{j=1}^N D_j \mathbf{w}^t}{D} + \gamma^{t+1} \frac{\sum_{j=1}^N D_j \mathbf{w}_j^w(\mathbf{w}^t, \Delta \mathbf{w}^t)}{D} \\ &\quad - \gamma^{t+1} \frac{\sum_{j=1}^N D_j \mathbf{w}_j^t}{D} \\ &= \mathbf{w}^t + \gamma^{t+1}(\mathbf{w}^w(\mathbf{w}^t, \Delta \mathbf{w}^t) - \mathbf{w}^t). \end{aligned} \tag{35}$$

∎

To prove the convergence of the distributed learning, we only need to prove that the equivalent centralized learning is convergent.

Lemma 8: Given problem under Assumption 2, we can achieve that the global loss function $F^w(\mathbf{w}; \mathbf{w}^t, \Delta \mathbf{w}^t)$ satisfies the Assumption 2.

Proof: According to the aggregation rules, the global loss function $F^w(\mathbf{w}; \mathbf{w}^t, \Delta \mathbf{w}^t)$ is written in (31), which is the

linear combination of the $F_j^w(\mathbf{w}; \mathbf{w}^t, \Delta\mathbf{w}^t)$. Straightforwardly from the convexity property, we can derive the conclusion. ■

Proposition 4 (Convergence Under Worst-case Model): Given problem under Assumption 2, suppose that $\tau > 0$ and step size γ^t and ρ^t are chosen as $\gamma^t = \frac{1}{t^\alpha}$ and $\rho^t = \frac{1}{t^\beta}$, $0.5 < \beta < \alpha < 1$ for the centralized learning. Let $\{\mathbf{w}^t\}$ be the sequence generated by algorithm, $F^w(\mathbf{w}; \mathbf{w}^t, \Delta\mathbf{w}^t)$ be $F^w(\mathbf{w})$ and $\mathbf{w}^w(\mathbf{w}^t, \Delta\mathbf{w}^t)$ be $\mathbf{w}^{w,t}$. The global loss function $F^w(\mathbf{w})$ converges at $\mathcal{O}(\gamma^t)$ so that there exists a constant M satisfying

$$F^w(\mathbf{w}^t) - F^w(\mathbf{w}^*) \le M\gamma^t. \tag{36}$$

Proof: Firstly, we can obtain that $G^t = \sum_{i=1}^N D_i G_i^t / D$, and $\tilde{F}^t = \sum_{i=1}^N D_i \tilde{F}_i^t / D$ via the updating rules. Furthermore, according to lemma, we have that $\tilde{F}^t$ also satisfies the Assumption 2. Invoking the first-order optimality conditions of $F^w(\mathbf{w})$, we have

$$\begin{aligned}
&\rho^t \left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, \nabla\tilde{F}(\mathbf{w}^{w,t}, \Delta\mathbf{w}_j^t) \right\rangle \\
&+ \lambda\|\mathbf{w}^t - \mathbf{w}^{w,t}\|^2 + (1-\rho^t)\left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, G_j^t \right\rangle \\
=&\rho^t \left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, \nabla\tilde{F}(\mathbf{w}^{w,t}, \Delta\mathbf{w}^t) - \nabla\tilde{F}(\mathbf{w}^t, \Delta\mathbf{w}^t) \right\rangle \\
&+ \left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, G^t \right\rangle + \lambda\|\mathbf{w}^t - \mathbf{w}^{w,t}\|^2 \ge 0
\end{aligned} \tag{37}$$

Considering the convexity of the $\tilde{F}(\cdot, \Delta\mathbf{w}^t)$, we can obtain that

$$\left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, G^t \right\rangle \le -\lambda\|\mathbf{w}^t - \mathbf{w}^{w,t}\|^2. \tag{38}$$

Given $F^w(\mathbf{w})$ under the Assumption 2, there will exist a constant L so that

$$\begin{aligned}
F^w(\mathbf{w}^{t+1}) \le{}& F^w(\mathbf{w}^t) + \gamma^{t+1}\left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, \nabla F^w(\mathbf{w}^t) \right\rangle \\
&+ L(\gamma^{t+1})^2\|\mathbf{w}^t - \hat{\mathbf{w}}^t\|^2 \\
={}& F^w(\mathbf{w}^t) + L(\gamma^{t+1})^2\|\mathbf{w}^t - \mathbf{w}^{w,t}\|^2 \\
&+ \gamma^{t+1}\left\langle \mathbf{w}^t - \mathbf{w}^{w,t}, \nabla F^w(\mathbf{w}^t) - G^t + G^t \right\rangle \\
\le{}& F^w(\mathbf{w}^t) - \gamma^{t+1}(\lambda - L\gamma^{t+1})\|\mathbf{w}^t - \mathbf{w}^{w,t}\|^2 \\
&+ \gamma^{t+1}\|\mathbf{w}^t - \mathbf{w}^{w,t}\|\|\nabla F^w(\mathbf{w}^t) - G^t\|
\end{aligned} \tag{39}$$

Suppose that $\lim_{t\to+\infty}\|\mathbf{w}^t - \hat{\mathbf{w}}^t\| \ge W \ge 0$, so that we can derive that

$$\begin{aligned}
F^w(\mathbf{w}^{t+1}) \le{}& F^w(\mathbf{w}^t) - \gamma^{t+1}(\lambda - L\gamma^{t+1})W^2 \\
&+ \gamma^{t+1}\|\nabla F^w(\mathbf{w}^t) - G^t\|W.
\end{aligned} \tag{40}$$

We focus on a realization that $\lim_{t\to+\infty}\|\nabla F^w(\mathbf{w}^t) - G^t\| = 0$. Therefore, there exists a t_0 sufficiently large so that for $\forall t > t_0$

$$\lambda - L\gamma^{t+1} - \frac{1}{W}\|\nabla F^w(\mathbf{w}^t) - G^t\| \ge \tilde{\lambda} > 0. \tag{41}$$

Therefore, the global loss function $F^w(\mathbf{w})$ follows

$$F^w(\mathbf{w}^{t+1}) - F^w(\mathbf{w}^t) \le -\tilde{\lambda}W^2\gamma^{t+1}. \tag{42}$$

We show next that the gap between the $F^w(\mathbf{w}^t)$ and the optimal $F^w(\mathbf{w}^{t^*})$ is

$$\begin{aligned}
F^w(\mathbf{w}^t) - F^w(\mathbf{w}^*) ={}& \left[F^w(\mathbf{w}^t) - F^w(\mathbf{w}^{t_0})\right] \\
&+ \left[F^w(\mathbf{w}^{t_0}) - F^w(\mathbf{w}^{t^*})\right] \\
\le{}& \tilde{\lambda}W^2\left(-\sum_{m=t_0}^t \gamma^m - \sum_{m=t^*}^{t_0}\gamma^m\right) \\
={}& \tilde{\lambda}W^2\left(-\frac{\gamma^{t_0}(1-\gamma^t)}{1-\gamma}\right) \\
&- \tilde{\lambda}W^2\left(\frac{\gamma^{t^*}(1-\gamma^{t_0})}{1-\gamma}\right) \\
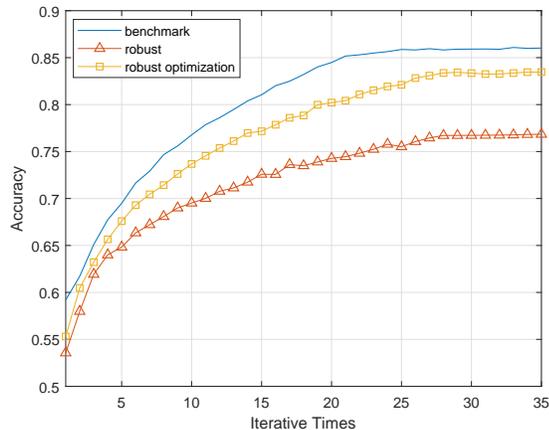\le{}& \tilde{\lambda}W^2\frac{1}{1-\gamma}\gamma^t.
\end{aligned} \tag{43}$$

Let the constant M satisfy $M = \tilde{\lambda}W^2/(1-\gamma)$, and we obtain the convergence rate expression in (36). ■

Remark 5: The proposed robust design under the expectation-based model converges at $\mathcal{O}(\gamma^t)$. The centralized training process converges at $\mathcal{O}(1/t)$, which utilizes the gradient descent under perfect estimation. Compared with the centralized training, the proposed design converges at a higher speed when iteration time t increases. The comparison between the proposed design and the centralized training is simulated specifically in Section VI.
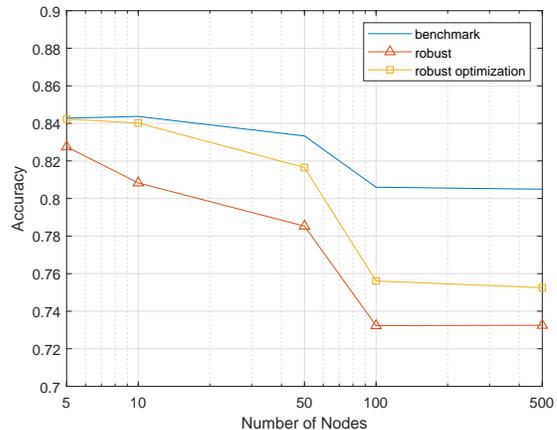
Remark 6: When the loss function under worst-case model is non-convex, the convergence of the proposed loss function is not clear. However, we notice that the authors in [44] first separate the non-convex objective function into the convex part C and the non-convex part C_c, and then add another term in SCA algorithm. Therefore, to solve the nonconvex problem similarly, we need to revise our proposed loss function via adding a new term $\langle \mathbf{w} - \mathbf{w}^t, \nabla F_{C_c}(\mathbf{w}^t + \Delta\mathbf{w}_j^t)\rangle$, where $\nabla F_{C_c}(\mathbf{w}^t + \Delta\mathbf{w}_j^t)$ denotes the non-convex loss function, and the corresponding convergence rate is proved to be $\mathcal{O}(1/t)$.
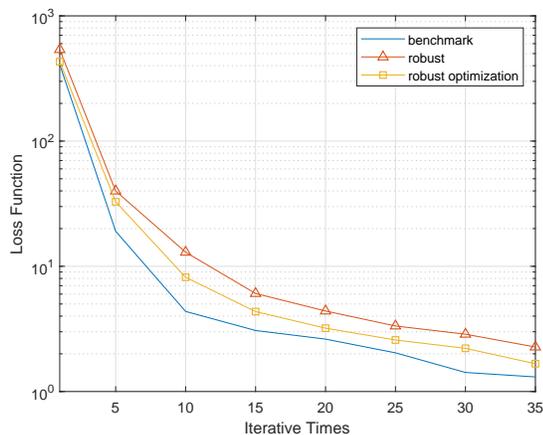
VI. SIMULATION RESULTS

In this section, we evaluate the performance of the proposed algorithms in image classification. The simulation parameters are set as follows unless otherwise specified. For illustration, we consider the learning task of image classification using the well-known MNIST dataset that consists of 10 categories ranging from digit 0 to 9 and a total of 70,000 data (60,000 for training and 10,000 for testing). Besides, we exploit the SVM classification as our loss function for the training process, which outputs a binary label that corresponds to whether the digit is even or odd [45]–[47]. We consider the data partitions as i.i.d. in the distributed nodes, i.e., each data sample is randomly assigned to the nodes. The performance is measured as the prediction accuracy and the values of loss function with respect to the training dataset versus iteration count t.
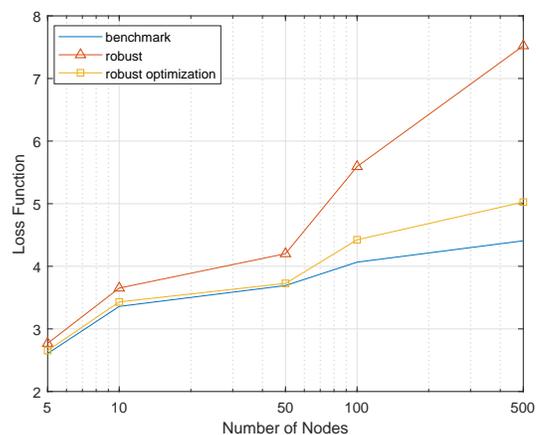
(a) The accuracy performance versus iterative times under expectation-based model.

(b) The loss function performance versus iterative times under expectation-based model.

Fig. 3. The corresponding performance versus iterative times under expectation-based model.

(a) The accuracy performance versus number of nodes under expectation-based model.

(b) The loss function performance versus number of nodes under expectation-based model

Fig. 4. The corresponding performance versus number of nodes under expectation-based model.

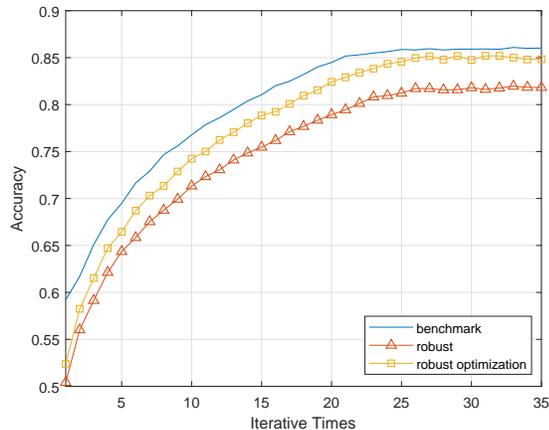For an intuitive comparison, we consider the following baseline approaches:

. Centralized training, where the model is trained via a standard gradient descent procedure and the received value is estimated perfectly.

. Conventional federated training, which is consist of the parameter noise and utilizes the imperfect estimated value to represent the real value for the training process. The model is trained via a standard gradient descent procedure and the loss function is the same as the centralized gradient descent.

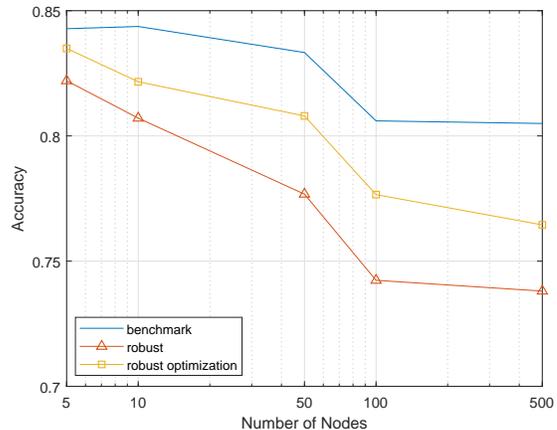### A. Simulation Under Expectation-based Model

We set the noise variance for expectation-based model as $\sigma_e^2 = 1$. We evaluate the prediction accuracy and the loss function values as a function of iterative times in Fig. 3. This agrees with our intuition that the noise has the serious impact on the learning model. The prediction accuracy of the proposed algorithm is obviously higher than the conventional federated training. The performance gap between two schemes increases

with the iteration process as in Fig. 3(a). The result has a profound and refreshing implication that the added regularizer draws the model into a flat region so that the learning model has the ability to resist the noise. In Fig. 3(b), the loss function values of the proposed scheme outperform the conventional method, which reflects that our designed regularizer imposes appropriate punishment.
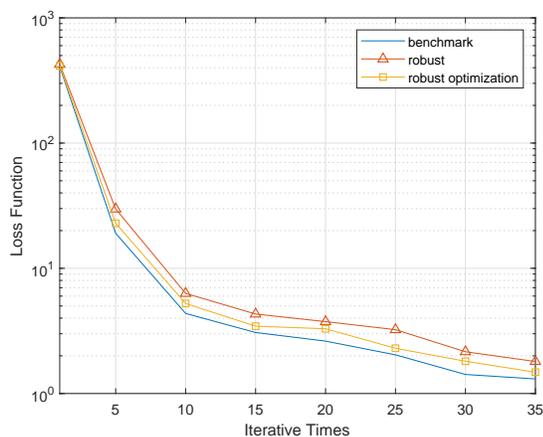
Fig. 4 shows the prediction accuracy and the loss function values for different numbers of nodes. The observations align with our conclusion that the proposed algorithm shows better performance than the conventional scheme on both the prediction accuracy and the loss function values. Furthermore, we notice that the performance of the learning model decreases with the growth of the numbers of nodes, due to that we randomly divide the data samples to all nodes. The division mode makes that the each node has uniform information but not the full information, and it will cause the hardship to find the optimal point in training process. With the growth of the numbers of nodes, each node can only obtain less and less samples and information, which leads to the decrease of the
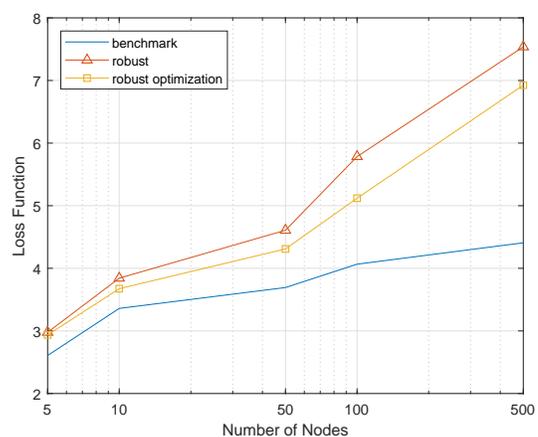
(a) The accuracy performance versus iterative times under worst-case model.

(b) The loss function performance versus iterative times under worst-case model.

Fig. 5. The corresponding performance versus iterative times under worst-case model.

(a) The accuracy performance versus number of nodes under worst-case model.

(b) The loss function performance versus number of nodes under worst-case model.

Fig. 6. The corresponding performance versus number of nodes under worst-case model.

learning performance. As shown in Fig. 4(a), it is interesting to find that accuracy of the proposed design is approaching to the centralized learning with few nodes, which proves the remarkable performance of the proposed design and verifies the direct effects of the proposed regularizer. As illustrated in Fig. 4(b), the loss function values of the proposed design outperform than the conventional scheme with the growth of the nodes number especially.

B. Simulation Under Worst-case Model

We set the spherical region size of the noise for worst-case model as $\sigma_w^2 = 1$, and choose the sample noise sequence as $\|\Delta \mathbf{w}_j^t\|^2 = \sigma_w^2$, $j = 1, 2, ..., N$, $t = 0, 1, 2, ...$ Fig. 5 illustrates the prediction accuracy and the loss function values of the different iterative times. Without consideration of robust design, we notice that the noise reduces both performance of the training processes as shown in Fig. 5(a). The accuracy performance of the proposed scheme is significantly improved, which verifies that the added punishment of the loss function positively affects the noise. With the development of the

iteration process, we can obtain that the performance of the proposed design approaches to the centralized training method. The observations align with our discussions in Remark 3. As shown at Fig. 5(b), the loss function values of all three schemes decrease with the iteration process. It is interesting to see that the proposed scheme shows better performance, which proves the effectiveness of the approximation method.

We show the prediction accuracy and the loss function values with different number of nodes in Fig. 6. With the increase of number of nodes, the prediction accuracy and the loss function values performance of all designs are decreased. However, the robust design of the proposed algorithm performs a remarkable gap in accuracy performance than the conventional design. It is observed that the gap between the conventional training and the proposed design increases with nodes as illustrated in Fig. 6(a). Such accurate learning of proposed design is due to the positive punishment of the loss function and the proper approximation method which makes the global model robust against noise. In Fig. 6(b), the loss function values of the proposed design outperform than the

conventional method, and the gap is still increasing with the growth of the nodes number. The phenomenon verifies that the added punishment and the approximation method behave good effects on the training process.

VII. CONCLUSIONS

In this paper, we have proposed the robust federated learning to resist the noise from wireless communications. Considering the noise in both aggregation and broadcast process, we have formulated the problem with effective noise as parallel optimization problem under the expectation-based model and worst-case model. The corresponding optimization problem under the expectation-based model has been handled via the SAM algorithm, which can transform the effect of noise as the regularizer in the loss function during the training process. We have proposed the sampling based SCA algorithm to solve the optimization problem under the worst-case model. Moreover, the convergent properties of both proposed designs have been derived that proposed designs have acceptable convergence rates. Simulation results have illustrated that both proposed training process under the mentioned models have improved the prediction accuracy and the loss function values due to the proper punishment in the training.

REFERENCES

[1] S. Li, L. Da Xu, and S. Zhao, "5G internet of things: A survey," *J. Ind. Inf. Integr.*, vol. 10, pp. 1–9, 2018.

[2] L. Chen, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, "Communicating or computing over the mac: Function-centric wireless networks," *IEEE Transactions on Communications*, 2019.

[3] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, 2019.

[4] H. Lee, S. H. Lee, T. Q. Quek, and I. Lee, "Deep learning framework for wireless systems: Applications to optical wireless communications," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 35–41, 2019.

[5] H. B. McMahan, E. Moore, D. Ramage, S. Hampson *et al.*, "Communication-efficient learning of deep networks from decentralized data," *in Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 1273–1282, 2017.

[6] J. Konečnỳ, B. McMahan, and D. Ramage, "Federated optimization: Distributed optimization beyond the datacenter," *arXiv preprint arXiv:1511.03575*, 2015.

[7] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[8] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, 2019.

[9] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Federated learning for ultra-reliable low-latency V2V communications," *in Proc. 2018 IEEE Global Communnication Conference (ICC)*, pp. 1–7, 2018.

[10] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge ai: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, 2019.

[11] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and low-latency wireless communication: Tail, risk, and scale," *Proceedings of the IEEE*, vol. 106, no. 10, pp. 1834–1853, 2018.

[12] A. F. Aji and K. Heafield, "Sparse communication for distributed gradient descent," *arXiv preprint arXiv:1704.05021*, 2017.

[13] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," *arXiv preprint arXiv:1712.01887*, 2017.

[14] H. H. Yang, Z. Liu, T. Q. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *arXiv preprint arXiv:1908.06287*, 2019.

[15] T. Chen, G. Giannakis, T. Sun, and W. Yin, "Lag: Lazily aggregated gradient for communication-efficient distributed learning," *in Proc. Conference and Workshop on Neural Information Processing Systems (NIPS)*, pp. 5050–5060, 2018.

[16] J. Chen, X. Pan, R. Monga, S. Bengio, and R. Jozefowicz, "Revisiting distributed synchronous sgd," *arXiv preprint arXiv:1604.00981*, 2016.

[17] M. Kamp, L. Adilova, J. Sicking, F. Hüger, P. Schlicht, T. Wirtz, and S. Wrobel, "Efficient decentralized deep learning by dynamic model averaging," *Mach. Learn. Knowl. Discov. Databases*, pp. 393–409, 2018.

[18] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," *in Proc. 2019 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2019.

[19] N. Agarwal, A. T. Suresh, F. X. X. Yu, S. Kumar, and B. McMahan, "cpsgd: Communication-efficient and differentially-private distributed sgd," *in Proc. Conference and Workshop on Neural Information Processing Systems (NIPS)*, pp. 7564–7575, 2018.

[20] M. R. Sprague, A. Jalalirad, M. Scavuzzo, C. Capota, M. Neun, L. Do, and M. Kopp, "Asynchronous federated learning for geospatial applications," *Mach. Learn. Knowl. Discov. Databases*, pp. 21–28, 2018.

[21] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang *et al.*, "Large scale distributed deep networks," *Adv. Neural Inf. Process Syst.*, pp. 1223–1231, 2012.

[22] Q. Ho, J. Cipar, H. Cui, S. Lee, J. K. Kim, P. B. Gibbons, G. A. Gibson, G. Ganger, and E. P. Xing, "More effective distributed ml via a stale synchronous parallel parameter server," *Adv. Neural Inf. Process Syst.*, pp. 1223–1231, 2013.

[23] Y. Tang and C. Eliasmith, "Deep networks for robust visual recognition," *in Proc. International Conference on Machine Learning (ICML)*, pp. 1055–1062, 2010.

[24] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," *in Proc. International Conference on Machine Learning (ICML)*, pp. 833–840, 2011.

[25] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," *in Proc. International Conference on Machine Learning (ICML)*, pp. 1096–1103, 2008.

[26] C. M. Bishop, "Regularization and complexity control in feed-forward networks," *Aston University, Birmingham, U.K., Tech. Rep. NCRG/95/022*, 1995.

[27] ——, "Training with noise is equivalent to tikhonov regularization," *Neural Compu.*, vol. 7, no. 1, pp. 108–116, 1995.

[28] A. Graves, "Practical variational inference for neural networks," *in Proc. Conference and Workshop on Neural Information Processing Systems (NIPS)*, pp. 2348–2356, 2011.

[29] S. Hochreiter and J. Schmidhuber, "Simplifying neural nets by discovering flat minima," *in Proc. Conference and Workshop on Neural Information Processing Systems (NIPS)*, pp. 529–536, 1995.

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[31] L. Muñoz-González, K. T. Co, and E. C. Lupu, "Byzantine-robust federated machine learning through adaptive model averaging," *arXiv preprint arXiv:1909.05125*, 2019.

[32] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *arXiv preprint arXiv:1903.02891*, 2019.

[33] A. Ghosh, J. Hong, D. Yin, and K. Ramchandran, "Robust federated learning in a heterogeneous environment," *arXiv preprint arXiv:1906.06629*, 2019.

[34] F. Ang and L. Chen, "An efficient training scheme to acquire massive CSI in analog function computation networks," *in Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2019.

[35] S. A. Vorobyov, A. B. Gershman, and Z.-Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: A solution to the signal mismatch problem," *IEEE Trans. Signal Process*, vol. 51, no. 2, pp. 313–324, 2003.

[36] 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, 2019.

[37] T. T. Soong, *Fundamentals of probability and statistics for engineers*. John Wiley & Sons, 2004.

[38] H. Shen, J. Wang, B. Xu, Y. Rong, and C. Zhao, "A worst-case robust MMSE transceiver design for nonregenerative mimo relaying," *IEEE Trans. Wirel. Commun.*, vol. 13, no. 2, pp. 695–709, 2013.

[39] J. Wang, M. Bengtsson, B. Ottersten, and D. P. Palomar, "Robust MIMO precoding for several classes of channel uncertainty," *IEEE Trans. Signal Process.*, vol. 61, no. 12, pp. 3056–3070, 2013.

[40] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT press, 2016.

[41] E. K. Chong and S. H. Zak, *An Introduction to Optimization*. John Wiley & Sons, 2013, vol. 76.

[42] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht, "Gradient descent only converges to minimizers," *Conference on learning theory*, pp. 1246–1257, 2016.

[43] A. J. Kleywegt, A. Shapiro, and T. Homem-de Mello, "The sample average approximation method for stochastic discrete optimization," *SIAM J. Optim.*, vol. 12, no. 2, pp. 479–502, 2002.

[44] Y. Yang, G. Scutari, D. P. Palomar, and M. Pesavento, "A parallel decomposition method for nonconvex stochastic multi-agent optimization problems," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2949–2964, 2016.

[45] E. Osuna, R. Freund, and F. Girosi, "An improved training algorithm for support vector machines," *Neural networks for signal processing VII. Proceedings of the 1997 IEEE signal processing society workshop*, pp. 276–285, 1997.

[46] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu, "IoT security techniques based on machine learning," *arXiv preprint arXiv:1801.06275*, 2018.

[47] L. Xiao, D. Jiang, D. Xu, W. Su, N. An, and D. Wang, "Secure mobile crowdsensing based on deep learning," *China Communications*, vol. 15, no. 10, pp. 1–11, 2018.

APPENDIX A
PROOF OF PROPOSITION 2

The proposed loss function of the node j is

$$F_j^e(\mathbf{w}) = F_j(\mathbf{w}) + \sigma_e^2 \|\nabla F_j(\mathbf{w})\|^2. \tag{44}$$

Taking the derivation of it, we can obtain

$$\begin{aligned}
\nabla F_j^e(\mathbf{w}) &= \nabla F_j(\mathbf{w}) + \sigma_e^2 \nabla \mathrm{tr}(\nabla F_j(\mathbf{w})\nabla F_j(\mathbf{w})^T) \\
&= \nabla F_j(\mathbf{w}) + \sigma_e^2 \nabla F_j(\mathbf{w}) \\
&= (1 + \sigma_e^2)\nabla F_j(\mathbf{w}).
\end{aligned} \tag{45}$$

Following the Lemma 4, we can obtain that the loss function $F_j^e(\mathbf{w})$ of the node j is $\beta - smooth$ with $\tilde{\beta} = (1 + \sigma_e^2)\beta$. Therefore, $F_j^e(\mathbf{w})$ satisfies

$$F_j^e(\mathbf{w}) - F_j^e(\mathbf{w}^*) \leq \|\mathbf{w}^0 - \mathbf{w}^*\|^2 \cdot \frac{1}{\eta\left(1 - \frac{(1+\sigma_e^2)\beta\eta}{2}\right)} \cdot \frac{1}{t}. \tag{46}$$

Furthermore, we can develop the conclusion that $F^e(\mathbf{w})$ is $\beta - smooth$ to satisfy (21). The optimization of the global loss function converges at $\mathcal{O}(1/t)$.

APPENDIX B
PROOF OF PROPOSITION 3

As the efficient solutions of the SCA algorithm, the objective function $F_j(\mathbf{w}, \Delta\mathbf{w}_j^t)$ at the iteration t is determined by the latest updated model $\Delta\mathbf{w}_j^t$ and defined as $F_j^w(\mathbf{w}; \mathbf{w}^t, \Delta\mathbf{w}_j^t)$, which is consist of the original function $F_j(\mathbf{w} + \Delta\mathbf{w}_j^t)$, and the first gradient $\Omega_1(\mathbf{w}) = \langle \mathbf{w} - \mathbf{w}^t, G_j^{t-1}\rangle$. We develop the objective function as follows,

$$F_j^w(\mathbf{w}; \mathbf{w}^t, \Delta\mathbf{w}_j^t) = \rho^t F_j(\mathbf{w}+\Delta\mathbf{w}_j^t)+(1-\rho^t)\langle\mathbf{w}-\mathbf{w}^t, G_j^{t-1}\rangle, \tag{47}$$

and G_j^t is an accumulation vector updated recursively according to

$$G_j^t = (1 - \rho^t)G_j^{t-1} + \rho^t\nabla_{\mathbf{w}}F_j(\mathbf{w} + \Delta\mathbf{w}_j^t), \tag{48}$$

with $\rho^t \in (0, 1]$ being a sequence to be properly chosen ($\rho^0 = 1$) at iterations $t = 0, 1, ...$ respectively.

Notice that the expansion is established only when $\mathbf{w}$ is close to $\mathbf{w}^t$. We add a regularizer as the cost of shrinking the gap between $\mathbf{w}$ and $\mathbf{w}^t$ as:

$$\Omega_2(\mathbf{w}) = \|\mathbf{w} - \mathbf{w}^t\|^2. \tag{49}$$

Therefore, we propose the local loss function as in P_4.

Fan Ang received the B.S. degree in information and communications engineering from Xi'an Jiaotong University, Xian, China, in 2017. She is currently pursuing the M.S. degree with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. Her research interests include wireless IoT communications and federated learning.

Li Chen received the B.E. in electrical and information engineering from Harbin Institute of Technology, Harbin, China, in 2009 and the Ph.D.degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 2014. He is currently a faculty member with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. His research interests include wireless IoT communications and wireless optical communications.

Nan Zhao (S'08-M'11-SM'16) is currently an Associate Professor at Dalian University of Technology, China. He received the B.S. degree in electronics and information engineering in 2005, the M.E. degree in signal and information processing in 2007, and the Ph.D. degree in information and communication engineering in 2011, from Harbin Institute of Technology, Harbin, China. His recent research interests include UAV Communications, Interference Alignment, and Physical Layer Security.

Dr. Zhao is serving or served on the editorial boards of 7 SCI-indexed journals. He received Top Reviewer Award from IEEE Transactions on Vehicular Technology in 2016, and was nominated as an Exemplary Reviewer by IEEE Communications Letters in 2016. He won the best paper awards in IEEE VTC'2017-Spring and MLICOM 2017.

Yunfei Chen (S'02-M'06-SM'10) received his B.E. and M.E. degrees in electronics engineering from Shanghai Jiaotong University, Shanghai, P.R.China, in 1998 and 2001, respectively. He received his Ph.D. degree from the University of Alberta in 2006. He is currently working as an Associate Professor at the University of Warwick, U.K. His research interests include wireless communications, cognitive radios, wireless relaying and energy harvesting.

Weidong Wang received the B.S. degree from Beijing University of Aeronautics and Astronautics, Beijing, China, in 1989 and the M.S. degree from the University of Science and Technology of China, Hefei, China, in 1993. He is currently a Full Professor with the Department of Electronic Engineering and Information Systems, University of Science and Technology of China. His research interests include wireless communication, microwave and millimeter, and radar technology. Dr. Wang is a member of the Committee of Optoelectronic Technology, Chinese Society of Astronautics.

F. Richard Yu (S'00-M'04-SM'08-F'18) received the PhD degree in electrical engineering from the University of British Columbia (UBC) in 2003. From 2002 to 2006, he was with Ericsson (in Lund, Sweden) and a start-up in California, USA. He joined Carleton University in 2007, where he is currently a Professor. He received the IEEE Outstanding Service Award in 2016, IEEE Outstanding Leadership Award in 2013, Carleton Research Achievement Award in 2012. His research interests include wireless cyber-physical systems, connected/autonomous vehicles, security, distributed ledger technology, and deep learning.