This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

CB-DSL: Communication-efficient and Byzantine-robust Distributed Swarm Learning on Non-i.i.d. Data

Xin Fan, *Student Member, IEEE*, Yue Wang, *Senior Member, IEEE*, Yan Huo, *Senior Member, IEEE*, and Zhi Tian, *Fellow, IEEE*

***Abstract*—The valuable data collected by IoT devices together with the resurgence of machine learning (ML) stimulate the latest trend of artificial intelligence (AI) at the edge. However, traditional ML and recent federated learning (FL) face major challenges including communication bottleneck, data heterogeneity, and security concerns in edge IoT. Meanwhile, the swarm nature of IoT systems is overlooked by most existing literature, which calls for new designs of distributed learning algorithms. Inspired by the success of biological intelligence (BI) of gregarious organisms, we propose a novel edge learning approach for swarm IoT, called communication-efficient and Byzantine-robust distributed swarm learning (CB-DSL), through a holistic integration of AI-enabled stochastic gradient descent and BI-enabled particle swarm optimization. To deal with the non-i.i.d. data issues and Byzantine attacks, a small amount of global data samples are introduced in CB-DSL and shared among IoT workers, which alleviates the local data heterogeneity effectively and enables to fully utilize the exploration-exploitation mechanism of swarm intelligence. Our convergence analysis theoretically demonstrates that the CB-DSL is superior to the standard FL with better convergence behavior. We also evaluate the model divergence of CB-DSL by deriving its upper bound, which measures the effectiveness of the introduction of the globally shared dataset.**

***Index Terms*—Distributed swarm learning, federated learning, particle swarm optimization, non-i.i.d. data, convergence analysis, model divergence analysis.**

I. Introduction

With the vigorous growth of Internet of Things (IoT) and Internet of Vehicles (IoV), smart devices are becoming the workhorse at the edge of wireless networks beyond 5G (B5G) [1]. The valuable data directly collected by devices together with the resurgence of machine learning (ML) stimulate the latest trend of artificial intelligence (AI) at the edge of B5G networks, termed as edge learning or edge AI [2], [3]. When conventional ML techniques are applied for edge learning, they are typically deployed in a centralized mode, which hinges on a full collection of distributed local data from the edge IoT devices to a central node. Such a centralized learning approach can obtain high learning accuracy, but the raw data collection process not only consumes huge communication resources but also raises unwilling privacy exposure and severe security concerns [4]. Alternatively, federated learning (FL) has recently attracted great attention and resulted in fruitful attempts for learning-based applications among multiple distributed workers such as personal mobile phones, which allows distributed learning from local data without raw data exchange [5], [6].

Standard FL methods are originally designed for ideal learning settings and wireless environments, which however face several challenges when being adopted for distributed learning among massive edge IoT devices that are usually equipped with limited capability and resources. As the number of model parameters goes very large in deep neural networks, transmission of all the local model updates in FL between IoT devices (working as local workers) and the parameter server (PS) incurs high communication overhead in edge networks. Further, stochastic gradient descent (SGD) is widely applied for model training in FL [6], [7], where independent and identically distributed (i.i.d.) data samples are assumed at local workers and transmission is assumed error-free in order to ensure unbiased estimates and good empirical performances [8], [9]. However, in edge IoT scenarios, local training data samples at different IoT workers turn to be statistically heterogeneous worker-by-worker, giving rise to the non-i.i.d. data issue that may considerably degrades the learning performance of standard FL methods, e.g., Federated Averaging (FedAvg) [10], [11]. In addition, gradient-based algorithms are subject to local optimum traps in solving non-convex problems [12]–[14], such as when training neural networks with nonlinear activations. This issue is aggravated in distributed settings, especially when local workers only collect small-volume data. Last but not the least, standard FL performs well in attack-free network settings, but is vulnerable to Byzantine attacks that may exist in practical edge networks [15]–[18].

Although some of the aforementioned challenges have been recently investigated in the literature of FL for edge networks

Manuscript received October 25, 2022; revised December 28, 2022 and May 19, 2023; accepted August 12, 2023. This work was supported in part by the Science and Technology Innovation Project of Xiongan under Grant 2022XACX0400; in part by the U.S. National Science Foundation under Grants 1939553, 2003211, 2128596, 2136202 and 2231209; and in part by the Virginia Research Investment Fund (Commonwealth Cyber Initiative Grant) under Grant 223996. Parts of this work were presented at the IEEE International Conference on Communications, Rome, Italy, May 2023. *(Corresponding author: Yue Wang.)*

Xin Fan is with the School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China (e-mail: fanxin@bjfu.edu.cn).

Yue Wang is with the Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA (e-mail: ywang182@gsu.edu).

Yan Huo is with the School of Electronics and Information Engineering, Beijing Jiaotong University, Beijing 100044, China (e-mail: yhuo@bjtu.edu.cn).

Zhi Tian is with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA 22030, USA (e-mail: ztian1@gmu.edu).

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

and IoT applications [19]–[21], they mainly focus on the modification and customization of the standard FL techniques, which however largely neglect some important and unique characteristics of IoT devices in edge networks. Such unique characteristics include the large population of devices for many IoT applications, limited communication bandwidth available in edge networks, and non-i.i.d. local data with small data volume at individual IoT workers. By ignoring these characteristics, existing efforts on edge learning fail to consider these limitations in the learning algorithm design for edge IoT systems, which results in learning performance degradation of FL applied to practical IoT edge networks. On the other hand, biological organisms in nature have demonstrated swarm intelligence with superior strength in collectively processing information, making decisions, dealing with uncertainties, adapting to environment changes, and recovering from errors and failures, even though they are individually weak. All these attributes of biological intelligence (BI) are desired by IoT edge learning systems. Notably, bio-inspired swarm optimization techniques are good at collaboratively finding the globally optimal solutions to complex optimization problems thanks to their built-in exploration-exploitation mechanism in swarms, but their convergence speed is typically slow [22], [23].

Motivated to bridge these gaps, this paper leverages both AI and BI to develop a communication-efficient and Byzantine-robust distributed swarm learning (CB-DSL) approach, by reformulating the bio-inspired particle swarm optimization (PSO) problem as a distributed learning problem with non-i.i.d. local data and in the presence of malicious attacks. For non-convex problems, by taking advantage of the exploration-exploitation mechanism of PSO [24], [25], our CB-DSL solutions have an increased chance to jump out of local optimum traps via swarm intelligence. For the communication bottleneck challenge, our CB-DSL only requires the best worker having the minimum loss function value to upload its local model to the PS, which thus dramatically reduces the communication overhead and energy consumption in edge networks. To alleviate the non-i.i.d. data issue, we propose to introduce a small-volume global dataset that is shared among all local workers for dual purposes. A part of this globally shared dataset is used for training, whose effectiveness in relieving the non-i.i.d. problem is evaluated through the model divergence analysis. The other part of the global dataset is used to calculate the fair-value loss for scoring the local models. It helps to identify the per-worker best model for best worker selection, and enables to verify the uploaded local model by which the PS can screen Byzantine attackers. Our main contributions are summarized as follows.

- We propose a new CB-DSL framework by developing a holistic integration of AI-driven SGD and BI-driven PSO, to effectively handle the high communication costs, non-i.i.d. issues, non-convex problems and Byzantine attacks without sacrifice convergence speed, which cannot be achieved by SGD or PSO alone. CB-DSL offers a new paradigm of efficient and robust edge learning tailored for massive smart IoT devices in edge networks, which brings the benefits of swarm intelligence to broad applications of distributed learning.
- From the theoretical point of view, we are the first one to systematically analyze the combination of FL and PSO, by deriving a closed-form expression to quantify the expected convergence rate achieved by our CB-DSL. Our analytical results not only reflect the impact of different settings and parameters of our CB-DSL on the performance of edge learning among distributed workers, but also indicate that our CB-DSL outperforms the standard FL methods such as FedAvg in terms of better convergence rate.
- We further investigate the non-i.i.d. data issue at distributed workers by providing a model divergence analysis to evaluate how the introduction of a globally shared dataset improves the learning performance of our CB-DSL. Our theoretical result reveals that the model divergence is subject to an upper bound, which is decided by the earth mover's distance (EMD) between the data distribution at local workers and the population distribution for the whole datasets.
- Through comprehensive experiments, we test the proposed CB-DSL approach in solving image classification problems by using the MNIST dataset. Simulation results show that our CB-DSL outperforms the benchmark methods in terms of achieving the highest testing accuracy with the fastest convergence under both the i.i.d. and non-i.i.d. cases and even in the presence of Byzantine attacks.

The rest of this paper is organized as follows. Section II reviews the related work. The problem formulation of distributed learning and the framework of CB-DSL technique are systematically presented in Section III, where we develop the CB-DSL algorithm. The expected convergence rate and the model divergence analysis of the CB-DSL technique are studied in Section IV and Section V, respectively. Section VI presents simulation results and comparison of the CB-DSL technique with the benchmark methods, followed by conclusions in Section VII.

Notations: Bold upper and lower case letters denote matrices and vectors, respectively. Euclidean norm of a vector or a matrix is depicted as $\|\cdot\|$. The expectation and the first order derivative are represented by $\mathbb{E}$ and ∇. $\langle\cdot,\cdot\rangle$ calculates the inner product of two vectors. The probability of an event $y=c$ is expressed as $p(y=c)$. The event indicator is symbolled as $\mathbb{1}_{y=c}$, which is equal to 1 when $y=c$, or 0 otherwise.

II. Related Work

Various methods have been proposed in addressing the communication challenges of FL, such as sparsification [26], [27], quantization [28]–[30] and infrequent uploading of local updates [31]–[34]. Theses methods aim to reduce the amount of the communication overhead, by either compressing or dropping some non-informative transmissions. These strategies are investigated predominantly for FL over digital channels based on the orthogonal transmission resource allocation among different local workers. Recently, a promising technique for tackling the communication bandwidth bottleneck emerges in

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

the form of FL over the air [35]–[37], which exploits the fact that the model aggregation operation in FL matches the waveform superposition property of the wireless analog multi-access channels. It can further incorporate other efficiency-enhancing strategies for effectively reducing bandwidth consumption. For instance, communication-efficient FL over the air is developed by combining compression, quantization and concurrent transmission through 1-bit compressive sensing and analog aggregation transmission in [38], [39]. Nevertheless, the aforementioned compression and transmission strategies still require all participating workers to exchange some variants of their local updates, which are not tailored for edge IoT systems and may result in tremendous communication costs and energy consumption in edge networks with massive IoT devices.

To take advantage of the swarm biological intelligence of animal flocks, particle swarm optimization (PSO) has been developed to solve complicated optimization problems without invoking the assumption on convexity [24], [25]. Recently, there are few attempts of applying the PSO ideas to improve machine learning performance. In the centralized setting, PSO is used to optimize the solution and hyperparameters of convolutional neural networks (CNNs) for enhanced recognition accuracy of image classification [40]–[42]. In the distributed setting, two relevant works are found in attempting to integrate PSO into FL to improve FL performance [43], [44]. In [43], FL is used for learning, while PSO is simply applied to search the optimal hyperparameters. Different from [43], our work focuses on the design of distributed training algorithm and model updating strategy for improving the performance and robustness of edge learning systems. In [44], PSO and FL are combined in a simplistic manner for the idealized distributed settings with i.i.d. data and no attacks, which cannot be guaranteed for practical edge IoT systems. Further, the work [44] actually builds on an implicit assumption that a common loss function is available to all distributed workers, which trivializes the assessment of the globally best model. However, in distributed learning problems, loss function is only partially observable at local workers, which is data-dependent and hence different across workers. Thus, the method in [44] is not suitable to edge IoT systems with data of small volume at local workers. More importantly, there has not been any work on theoretical analysis for performance evaluation and convergence guarantee of distributed learning by connecting PSO with FL. To fill these identified technical gaps, in the next sections, we develop a novel efficient and robust edge learning algorithm through a holistic integration of AI-driven SGD and BI-driven PSO and empowered by using a small-volume global dataset, whose advantages are verified by rigorous convergence analysis, model divergence evaluation, and experiments on real data.

To overcome the non-i.i.d. issues in FL, some data-based approaches are proposed to modify the distributions of local datasets by data sharing [11], [45], [46] or data argumentation [47], [48]. Besides, there are some other approaches to addressing the non-i.i.d. data issues in FL by adjusting the algorithm or model structure of FL, such as local fine-tuning [49], personalization layer [50], [51] and so on [52]. Note that these aforementioned approaches can be combined with our CB-DSL as well, if needed.

III. Distributed Swarm Learning

This section starts with the problem statement for distributed learning and the formulations of FL and PSO techniques. Then, the pros and cons of FL and PSO motivate us to bridge distributed learning with swarm optimization techniques to make the best use of both artificial and biological intelligence. In particular, we focus on a systematical integration of FL and PSO for a novel communication-efficient and Byzantine-robust edge learning algorithm with non-i.i.d. local data and in the presence of Byzantine attacks.

Consider a distributed learning model with one parameter server (PS) and U IoT workers, where U is very large but each worker has data of small volume in edge IoT scenarios. Assume that each worker has K_i data samples in its local dataset $\mathfrak{D}_i$, with $|\mathfrak{D}_i| = K_i$, and $i = 1, \ldots, U$. Denote $(\mathbf{x}_{i,k}, y_{i,k})$ as the k-th data sample of the i-th local worker. Let $f(\mathbf{w}; \mathbf{x}_{i,k}, y_{i,k})$ represent the loss function associated with each data sample $(\mathbf{x}_{i,k}, y_{i,k})$, where $\mathbf{w} = [w^1, \ldots, w^D]$ of size D consists of the parameters of a common learning model. The corresponding population loss function for the whole datasets $\mathfrak{D}$ and that for the local dataset $\mathfrak{D}_i$ of the i-th worker are denoted as $F(\mathbf{w}) := \mathbb{E}_{\mathfrak{D}}[f(\mathbf{w}; \mathbf{x}_{i,k}, y_{i,k})]$ and $F_i(\mathbf{w}) := \mathbb{E}_{\mathfrak{D}_i}[f(\mathbf{w}; \mathbf{x}_{i,k}, y_{i,k})]$, respectively, where $\mathfrak{D} = \bigcup_i \mathfrak{D}_i$. For distributed learning, local workers collaboratively learn $\mathbf{w}$ by minimizing

$$\textbf{P1:} \quad \mathbf{w}_i^* = \arg\min_{\mathbf{w}_i} F_i(\mathbf{w}_i), \qquad \text{s.t.,} \quad \mathbf{w}_i = \mathbf{z}, \quad \forall i, \quad (1)$$

where $\mathbf{z}$ is an auxiliary variable to enforce consensus through collaboration among distributed local workers.

A. Federated Learning

For standard FL designed in ideal learning settings and network environments [5], the minimization of $F_i(\mathbf{w})$ is typically carried out by the stochastic gradient descent (SGD) algorithm [5], [6], where local workers iteratively update their local models in FL as

$$\mathbf{w}_{i,t+1} = \mathbf{w}_{i,t} - \frac{\alpha}{U} \sum_{j=1}^{U} \nabla F_j(\mathbf{w}_t; \mathbf{x}_{j,k}, y_{j,k}), \qquad (2)$$

where α is the learning rate and $\nabla F_j(\mathbf{w}_t; \mathbf{x}_{j,k}, y_{j,k}) = \mathbb{E}_{\mathfrak{D}_j}\left[\frac{\sum_{\mathfrak{B}_j} \nabla f(\mathbf{w}_t; \mathbf{x}_{j,k}, y_{j,k})}{|\mathfrak{B}_j|}\right]$ is the local gradient computed at each local worker using its randomly selected mini-batch $\mathfrak{B}_j \subset \mathfrak{D}_j$ with the mini-batch size $|\mathfrak{B}_j|$.

Note that (2) is the mathematical illustration of the iterative local model update, whereas the second term of global gradient averaging therein is typically implemented at the PS and then sent back to local workers. Hence, communications take place in every iteration until convergence, during which the communication overhead to acquire the sum of all U local gradients in (2) would be huge especially when U and D are large. Moreover, for complicated non-convex problems, distributed gradient-based FL solutions may converge to undesired local optima and there is unfortunately a lack of effective mechanisms to escape these traps.

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

B. Particle Swarm Optimization

As a bio-inspired algorithm, PSO is a stochastic optimization approach based on the movement of particles (workers) and the collaboration of swarms to iteratively and cooperatively search for an optimal solution to general optimization problems [24], [25]. Note that PSO is originally designed for optimization problems instead of learning problems with distributed data. In this sense, the loss function in PSO is assumed to be globally common to all particles, i.e., $F_i(\cdot) = F(\cdot), \forall i$ in the problem **P1** in (1). This is however not the case in distributed leaning where $F_i(\cdot)$ is data-dependent and different worker-by-worker, which will be explained in the next subsection.

In PSO, a swarm consists of a large set of particles, $i = 1, 2..., U$. At the current iteration, the position $\mathbf{w}_{i,t}$ of each particle i presents a possible solution to the problem, and meanwhile the velocity $\mathbf{v}_{i,t}$ of each particle i denotes the updating direction for the next step. To find the globally optimal value of $F(\cdot)$, particles collaborate with each other to update their velocities and positions in an iterative manner

$$\mathbf{v}_{i,t+1} = c_0\mathbf{v}_{i,t} + c_1(\mathbf{w}^p_{i,t} - \mathbf{w}_{i,t}) + c_2(\mathbf{w}^g_t - \mathbf{w}_{i,t}), \quad (3)$$

$$\mathbf{w}_{i,t+1} = \mathbf{w}_{i,t} + \mathbf{v}_{i,t+1}, \quad (4)$$

where the velocity is updated as a combination of three sub-directions: inertia $\mathbf{v}_{i,t}$ of the previous updating direction, individual direction towards each particle's own historical best parameter $\mathbf{w}^p_{i,t} = \text{argmin}_{\tau=1,\cdots,t} F(\mathbf{w}_{i,\tau})$, and social direction towards the globally best parameter found by the entire swarm $\mathbf{w}^g_t = \text{argmin}_{i=1,\cdots,U} F(\mathbf{w}^p_{i,t})$. Among the corresponding three weights, the inertia weight c_0 is a positive number, while c_1 and c_2 are positive and random (say, uniformly distributed as $c_1 \sim \mathcal{U}(0, \delta_{c_1})$, and $c_2 \sim \mathcal{U}(0, \delta_{c_2})$) for stochastic optimization. Notably, the weighted combination of the three sub-directions in (3) serves a mechanism for exploration-exploitation trade-offs, where c_0 is set to be linearly decreasing over iterations to tune the solution search process from exploration to exploitation, and c_1 and c_2 indicate the random exploration level at individual particles and the exploitation level in swarm, respectively.

C. Communication-efficient and Byzantine-robust Distributed Swarm Learning

A major challenge from optimization problems to learning problems with distributed data is the lack of a common $F(\cdot)$ for global assessment, which however becomes $F_i(\cdot; \mathfrak{D}_i)$ dependent on local dataset $\mathfrak{D}_i$ in distributed learning. Facing this challenge, we first introduce a very small amount of global dataset[1]: $\mathfrak{D}^G = \mathfrak{D}^G_{tr} \cup \mathfrak{D}^G_{sc}$ to be shared by all workers, and then propose a novel edge learning framework called communication-efficient and Byzantine-robust distributed swarm learning (CB-DSL). The CB-DSL algorithm is implemented in **Algorithm 1**, and schematically illustrated through the following iterative model updating steps.

At the local workers $i = 1, \cdots, U$, the model parameters are updated in a way of integrating BI-enabled PSO with AI-enabled SGD

$$\mathbf{w}_{i,t+1} = \mathbf{w}_{i,t} + \underbrace{c_0\mathbf{v}_{i,t} + c_1(\mathbf{w}^p_{i,t} - \mathbf{w}_{i,t}) + c_2(\mathbf{w}^g_t - \mathbf{w}_{i,t})}_{\textbf{BI}} \underbrace{-\alpha\nabla F_i(\mathbf{w}_{i,t}; \mathfrak{D}_i \cup \mathfrak{D}^G_{tr})}_{\textbf{AI}}, \quad (5)$$

where $\mathfrak{D}^G_{tr}$ is a part of the global dataset $\mathfrak{D}^G$ and used for training to relieve the non-i.i.d. problem. Thanks to the combination of the gradient-free stochastic optimization of the BI term and the gradient-based learning technique of the AI term in (5), the workers are good at searching for the optimal solutions to complex problems with fast convergence.

Then, the local workers calculate their own historical minimum loss function values and maintain their own historical best model parameters

$$\{F^p_{i,t+1}, \mathbf{w}^p_{i,t+1}\} = \arg\min_{\tau=1,\cdots,t+1} F_i(\mathbf{w}_{i,\tau}, \mathfrak{D}^G_{sc}), \quad (6)$$

where $\mathfrak{D}^G_{sc}$ is the other part of the global dataset $\mathfrak{D}^G$ and used to provide fair-value scores of local models for best-worker selection by assessing the per-worker $F^p_{i,t+1}$ that helps to accurately identify $\mathbf{w}^p_{i,t+1}$. Then, all workers report their $F^p_{i,t+1}$ to the PS.

Comparing the received $\{F^p_{i,t+1}\}_i$ from all local workers, the PS selects the best worker $i^\star_{t+1}$ with the global optimum function value

$$\{i^\star_{t+1}, F^g_{t+1}\} = \arg\min_{i=1,\cdots,U} F^p_{i,t+1}. \quad (7)$$

If $F^g_{t+1} < F^g_t$, then the worker with the selected index $i^\star_{t+1}$ is invited to upload its $\mathbf{w}^p_{i^\star_{t+1},t+1}$ to the PS as the globally best model parameter $\mathbf{w}^g_{t+1} = \mathbf{w}^p_{i^\star_{t+1},t+1}$. If $F^g_{t+1} \geq F^g_t$, then no worker is invited to upload local model parameter and the PS simply maintains the globally best model parameter and the globally best loss function value from the previous iteration as $\mathbf{w}^g_{t+1} = \mathbf{w}^g_t$ and $F^g_{t+1} = F^g_t$.

Upon receiving $\mathbf{w}^p_{i^\star_{t+1},t+1}$ from the invited worker, the PS further uses $\mathfrak{D}^G_{sc}$ to verify the reported model parameter $\mathbf{w}^p_{i^\star_{t+1},t+1}$. If a mismatch is detected such as $F(\mathbf{w}^p_{i^\star_{t+1},t+1}, \mathfrak{D}^G_{sc}) \neq F^g_{t+1}$, then a Byzantine attack is identified and the attacker is filtered out; the PS will inquire the next best local worker, until confirmed.

Communication Efficiency. Note that our CB-DSL requires U workers to share their function value $F^p_{i,t+1}$ which is only a scalar, and then invites only one local worker with the global minimum loss function value calculated using $\mathfrak{D}^G_{sc}$ to report its model parameter to the PS. Thus, our CB-DSL can dramatically reduce the overall communication overhead and energy consumption in edge networks during each communication round, compared with that required by standard FL approaches.

Byzantine Robustness. In the process of collecting $F^p_{i,t+1}$'s from local workers, it is inherently vulnerable to Byzantine

[1]For the implementation point of view, a small amount (e.g., 1% of all datasets is adequate as used in our simulations) of globally shared dataset can be generated by a generative adversarial network module for keeping the privacy of workers' own local data [53], which can be either pre-stored in the IoT devices or broadcasted from the PS to all the local workers. The required resources in sharing and local storage are quite low.

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

attacks. For example, a malicious worker may send a fake $\bar{F}^p_{i,t+1}$ $(< F^p_{i,t+1})$ to fool the PS to invite the attacker to upload its fake model parameter as the distorted global optimum, which will undermine edge learning. Thanks to the introduction of $\mathfrak{D}^G_{sc}$ in our CB-DSL, it enables the PS using $\mathfrak{D}^G_{sc}$ to verify the reported model parameter so as to screen and remove the potential Byzantine attackers, resulting our CB-DSL Byzantine robust.

Algorithm 1 CB-DSL

Initialization:

$\mathbf{w}^p_{i,0} = \mathbf{w}_{i,0}$, $F^p_{i,0} = F_i(\mathbf{w}_{i,0}, \mathfrak{D}^G_{sc})$, $\forall i$;

1: **for** each iteration $t = 1 : T$ **do**
2: **at the local workers:**
3: receive $\mathbf{w}^g_t$ from the PS; otherwise maintain $\mathbf{w}^g_t = \mathbf{w}^g_{t-1}$;
4: update the local model parameter $\mathbf{w}_{i,t+1}$ via (5);
5: calculate the per-worker historical minimum loss function value $F^p_{i,t+1}$ and maintain the corresponding per-worker historical best model parameter $\mathbf{w}^p_{i,t+1}$ via (6);
6: send the scalar function value $F^p_{i,t+1}$ to the PS;
7: only the invited local worker sends its $\mathbf{w}^p_{i,t+1}$ to the PS;
8: **at the PS:**
9: compare the received $F^p_{i,t+1}$'s, select the best worker $i^\star_{t+1}$ and identify its function value as F^g_{t+1} via (7);
10: *if* $F^g_{t+1} < F^g_t$, *then* invite the selected worker $i^\star_{t+1}$ to upload its model parameter as the globally best model parameter $\mathbf{w}^g_{t+1} = \mathbf{w}^p_{i^\star_{t+1},t+1}$;
11: *else*, no worker is invited and maintain the globally best model parameter and function value from the previous iteration as $\mathbf{w}^g_{t+1} = \mathbf{w}^g_t$ and $F^g_{t+1} = F^g_t$;
12: given $\mathbf{w}^p_{i^\star_{t+1},t+1}$ received from the invited worker, verify $F(\mathbf{w}^p_{i^\star_{t+1},t+1}, \mathfrak{D}^G_{sc}) == F^g_{t+1}$;
13: *if* an attacker is identified by $F(\mathbf{w}^p_{i^\star_{t+1},t+1}, \mathfrak{D}^G_{sc}) \neq F^g_{t+1}$, remove it and repeat line 8 until a legitimate worker is selected.
14: broadcast $\mathbf{w}^g_{t+1}$ to local workers when a worker is selected; or keep quiet when no worker is invited.
15: **end for**

IV. Convergence Analysis

In this section, we first make some definitions and assumptions for convergence analysis. With these preliminaries, the convergence behavior of our CB-DSL approach is theoretically evaluated by deriving an upper bound of the convergence rate.

A. Assumption and Definition

Assumption 1. (Lipschitz continuity, smoothness): The gradient $\nabla F_i(\mathbf{w})$ of the loss function $F_i(\mathbf{w})$ at node i is uniformly Lipschitz continuous with respect to $\mathbf{w}$, that is,

$$\|\nabla F_i(\mathbf{w}_{i,t+1}) - \nabla F_i(\mathbf{w}_{i,t})\| \leq L\|\mathbf{w}_{i,t+1} - \mathbf{w}_{i,t}\|, \forall\, i, t, \quad (8)$$

where L is a positive constant, referred as the Lipschitz constant for the loss function $F_i(\cdot)$ [54].

To facilitate analyses, we first rewrite $\mathbf{w}^p_{i,t}$ and $\mathbf{w}^g_t$ in (5) as

$$\mathbf{w}^p_{i,t} = \mathbf{w}_{i,t-1} + \mathbf{v}^p_{i,t}, \quad (9)$$

$$\mathbf{w}^g_t = \mathbf{w}_{i,t-1} + \mathbf{v}^g_t, \quad (10)$$

where $\mathbf{v}^p_{i,t}$ and $\mathbf{v}^g_t$ denote the per-worker and globally optimal velocities currently used at the i-th worker.

Then, the DSL velocity update $\mathbf{v}_{i,t+1} = \mathbf{BI} + \mathbf{AI} = \mathbf{w}_{i,t+1} - \mathbf{w}_{i,t}$ in (5) can be rewritten as

$$\begin{aligned}\mathbf{v}_{i,t+1} &= c_0\mathbf{v}_{i,t} + c_1(\mathbf{v}^p_{i,t} - (\mathbf{w}_{i,t} - \mathbf{w}_{i,t-1})) \\ &\quad + c_2(\mathbf{v}^g_t - (\mathbf{w}_{i,t} - \mathbf{w}_{i,t-1})) - \alpha\nabla F_i(\mathbf{w}_{i,t}) \\ &= c_0\mathbf{v}_{i,t} + c_1(\mathbf{v}^p_{i,t} - \mathbf{v}_{i,t}) + c_2(\mathbf{v}^g_t - \mathbf{v}_{i,t}) - \alpha\nabla F_i(\mathbf{w}_{i,t}) \\ &= (c_0 - c_1 - c_2)\mathbf{v}_{i,t} + c_1\mathbf{v}^p_{i,t} + c_2\mathbf{v}^g_t - \alpha\nabla F_i(\mathbf{w}_{i,t}),\end{aligned} \quad (11)$$

where we replace $\nabla F_i(\mathbf{w}_{i,t}; \mathfrak{D}_i \cup \mathfrak{D}^G_{tr})$ by $\nabla F_i(\mathbf{w}_{i,t})$ hereafter for symbol simplicity.

We use $\theta_{i,t}$, $\theta^p_{i,t}$, and θ^g_t to denote the angles between $\mathbf{v}_{i,t}$ and $-\nabla F_i(\mathbf{w}_{i,t})$, between $\mathbf{v}^p_{i,t}$ and $-\nabla F_i(\mathbf{w}_{i,t})$, and between $\mathbf{v}^g_t$ and $-\nabla F_i(\mathbf{w}_{i,t})$, for any i and t, respectively. Then we have

$$\cos\theta_{i,t} \triangleq \frac{\langle \mathbf{v}_{i,t}, -\nabla F_i(\mathbf{w}_{i,t})\rangle}{\|\mathbf{v}_{i,t}\|\|\nabla F_i(\mathbf{w}_{i,t})\|},\ \forall i, t, \quad (12)$$

$$\cos\theta^p_{i,t} \triangleq \frac{\langle \mathbf{v}^p_{i,t}, -\nabla F_i(\mathbf{w}_{i,t})\rangle}{\|\mathbf{v}^p_{i,t}\|\|\nabla F_i(\mathbf{w}_{i,t})\|},\ \forall i, t, \quad (13)$$

$$\cos\theta^g_t \triangleq \frac{\langle \mathbf{v}^g_t, -\nabla F_i(\mathbf{w}_{i,t})\rangle}{\|\mathbf{v}^g_t\|\|\nabla F_i(\mathbf{w}_{i,t})\|},\ \forall i, t. \quad (14)$$

We further assume that the above cosine-similarity measures are bounded, whose lower and upper bounds are denoted as[2]

$$\underline{q} \leq \cos\theta_{i,t} \leq \overline{q},\ \forall i, t \quad (15)$$

$$\underline{q}^p \leq \cos\theta^p_{i,t} \leq \overline{q}^p,\ \forall i, t \quad (16)$$

$$\underline{q}^g \leq \cos\theta^g_t \leq \overline{q}^g,\ \forall i, t, \quad (17)$$

$$\underline{u} \leq \frac{\|\mathbf{v}_{i,t}\|}{\|\nabla F_i(\mathbf{w}_{i,t})\|} \leq \overline{u},\ \forall i, t, \quad (18)$$

$$\underline{u}^p \leq \frac{\|\mathbf{v}^p_{i,t}\|}{\|\nabla F_i(\mathbf{w}_{i,t})\|} \leq \overline{u}^p,\ \forall i, t, \quad (19)$$

$$\underline{u}^g \leq \frac{\|\mathbf{v}^g_t\|}{\|\nabla F_i(\mathbf{w}_{i,t})\|} \leq \overline{u}^g,\ \forall i, t. \quad (20)$$

B. Convergence Bound

We adopt the expected improvement on the gradient in terms of its $\ell 2$ norm, working as an indicator of convergence for non-convex optimization [56], [57]

$$\min_{0,1,\ldots,T} \mathbb{E}[\|\mathbf{g}_t\|^2] \leq \mathbb{E}\left[\sum_{t=1}^{T} \frac{1}{T}\|\mathbf{g}_t\|^2\right], \quad (21)$$

where the norm of the gradient is expected to converge to 0 as T increases to infinity, which means that the solution converges asymptotically.

[2]The velocity update in our CB-DSL can be regarded as a kind of momentum, which is related to the historical gradients. Assumptions similar to (15)-(20) can be found in [55].

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

With the assumptions and definitions presented in Subsection IV.A, the convergence errors of the CB-DSL algorithm developed in Subsection III.C are bounded by the following **Theorem 1**.

Theorem 1. *For T communication rounds, the expected convergence rate at each local worker in CB-DSL is bounded by*

$$\mathbb{E}\left[\sum_{t=1}^{T}\frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right] \leq \frac{F(\mathbf{w}_{i,0})-F(\mathbf{w}^*)}{T\Phi_E},\ \forall i \tag{22}$$

where $\Phi_E = \alpha - \frac{2c_0-\delta_{c_1}-\delta_{c_2}}{2}\overline{q}\,\overline{u} - \frac{\delta_{c_1}}{2}\overline{u}^p\overline{q}^p - \frac{\delta_{c_2}}{2}\overline{u}^g\overline{q}^g - 2L((c_0^2-\delta_{c_1}c_0-\delta_{c_2}c_0+\frac{\delta_{c_1}^2}{3}+\frac{\delta_{c_2}^2}{3}+\frac{\delta_{c_1}\delta_{c_2}}{2})\overline{u}^2+\frac{\delta_{c_1}^2}{3}(\overline{u}^p)^2+\frac{\delta_{c_2}^2}{3}(\overline{u}^g)^2+\alpha^2)$.

Proof. Please refer to Appendix A. □

The result of **Theorem 1** implies the following convergence rate

$$\mathbb{E}\left[\sum_{t=1}^{T}\frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right] \leq \mathcal{O}(\frac{1}{T\Phi_E}). \tag{23}$$

The inequality of (23) indicates that the convergence of the CB-DSL is guaranteed as the number of communication rounds T goes large. That is, as $T\to\infty$, we have $\mathbb{E}\left[\sum_{t=1}^{T}\frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right]\to 0$.

Remark 1. When c_0, δ_{c_1}, and δ_{c_2} are all set to be 0, we have $\Phi_E=\alpha-2L\alpha^2$ in (22) and (23), and CB-DSL degenerates into FedAvg. As $\Phi_E-(\alpha-2L\alpha^2) = \frac{\delta_{c_1}+\delta_{c_2}-2c_0}{2}\overline{q}\,\overline{u} + 2L((\delta_{c_1}c_0+\delta_{c_2}c_0-c_0^2-\frac{\delta_{c_1}^2}{3}-\frac{\delta_{c_2}^2}{3}-\frac{\delta_{c_1}\delta_{c_2}}{2})\overline{u}^2-\frac{\delta_{c_1}^2}{3}(\overline{u}^p)^2-\frac{\delta_{c_2}^2}{3}(\overline{u}^g)^2)-\frac{\delta_{c_1}}{2}\overline{u}^p\overline{q}^p-\frac{\delta_{c_2}}{2}\overline{u}^g\overline{q}^g>0$, CB-DSL converges faster than FedAvg.

V. Model Divergence Analysis for the Case of Non-i.i.d. Data

Intuitively, when the local datasets $\mathfrak{D}_i$ over different local workers are non-i.i.d., the learning performance varies with the degree of the local dataset heterogeneity. Specifically, the greater the heterogeneity of local datasets, the model parameters updated at different local workers will become more diverse, e.g., with a larger range of the values of $\cos\theta_{i,t}$ in (15) among workers. That is, $\underline{q}$ and $\overline{q}$ in (15) go smaller and bigger, respectively. As a result, Φ_E defined in (22) decreases as the heterogeneity of non-i.i.d. datasets increases, which leads to a loose upper bound on the convergence guarantee in (22) and (23) and thus degrades the learning performance with distributed non-i.i.d. datasets. In this section, we provide a statistical analysis to evaluate the impact of the local data heterogeneity on the learning performance of the CB-DSL. We study the model parameter divergence resulted from the distance enlargement between the non-i.i.d. data distributions on local workers and the overall population distribution. We evaluate such a distance by measuring the earth mover's distance (EMD) between these distributions [11], [58].

Consider a C-class classification problem defined over a compact space $\mathcal{X}$ and a label space $\mathcal{Y}$. The k-th data point $(\mathbf{x}_{i,k}, y_{i,k})$ on the i-th local worker distributes over $\mathcal{X}\times\mathcal{Y}$ following the distribution p_i. For the purpose of model divergence analysis, suppose a genie worker who has the population data that reflect the population distribution p of all local workers that may differ from p_i. The genie worker uses such knowledge of p to search for the globally optimal solution to the learning model, which serves as the reference to calibrate the model divergence due to the distributed non-i.i.d. data. Then the original population loss function $F(\mathbf{w}) := \mathbb{E}_{\mathfrak{D}}[f(\mathbf{w};\mathbf{x}_{i,k},y_{i,k})]$ can be rewritten as

$$\begin{aligned}F(\mathbf{w}) &= \mathbb{E}_{\mathbf{x},y\sim p}\left[\sum_{c=1}^{C}\mathbb{1}_{y=c}f_c(\mathbf{x},\mathbf{w})\right]\\ &= \sum_{c=1}^{C}p(y=c)\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w})],\end{aligned} \tag{24}$$

where f_c denotes the probability for the c-th class, $c\in\{1,C\}$.

Then, the learning problem at the genie worker can be formulated as

$$\textbf{P2:}\quad \mathbf{w}^* = \arg\min_{\mathbf{w}}\ \sum_{c=1}^{C}p(y=c)\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w})]. \tag{25}$$

By solving **P2**, the model obtained at the genie worker plays as the globally optimal position in each communication round of CB-DSL. Then according to (11), the velocity at the genie worker in the $(t+1)$-th communication round is updated via

$$\mathbf{v}_{t+1}^g = c_0\mathbf{v}_t^g - \alpha\nabla F(\mathbf{w}_t^g). \tag{26}$$

The model parameter at the genie worker in the $(t+1)$-th communication round is updated as

$$\mathbf{w}_{t+1}^g = \mathbf{w}_t^g + \mathbf{v}_{t+1}^g. \tag{27}$$

Given (5) and (27), the model divergence between the i-th local worker and the genie worker is defined as

$$model\ \ divergence = \frac{\|\mathbf{w}_{i,t+1}-\mathbf{w}_{t+1}^g\|}{\|\mathbf{w}_{t+1}^g\|}. \tag{28}$$

Next, we provide **Theorem 2** to evaluate the model divergence by deriving its upper bound theoretically.

Theorem 2. *Under the assumption that $\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w})]$ is L_c-Lipschitz for each class $c\in\{1,C\}$, we have the following inequality for the model divergence after $(t+1)$ communication rounds*

$$\begin{aligned}\|\mathbf{w}_{i,t+1}-\mathbf{w}_{t+1}^g\| \leq\ & \beta^{t+1}\|\mathbf{w}_{i,0}-\mathbf{w}_0^g\|\\ &+|c_0-c_1-c_2|\sum_{j=0}^{t}\beta^{t-j}\|\mathbf{v}_{i,j}-\mathbf{v}_j^g\|\\ &+\alpha\sum_{c=1}^{C}|p_i(y=c)-p(y=c)|\sum_{j=0}^{t}f_{max}(\mathbf{w}_j^g),\end{aligned} \tag{29}$$

where $\beta = 1+\alpha\sum_{c=1}^{C}p_i(y=c)L_c$ *and* $f_{max}(\mathbf{w}_j^g) = \max\{\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_j^g)]\}_{c=1}^{C}$.

Proof. Please refer to Appendix B. □

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

Remark 2. Our theoretical result of **Theorem 2** indicates that the model divergence can be upper bounded in (29) after $(t+1)$ communication rounds, which mainly comes from three parts, including the initial model divergence, i.e., $\|\mathbf{w}_{i,0} - \mathbf{w}_0^g\|$, the velocity divergence after t communication rounds, i.e., $\|\mathbf{v}_{i,j} - \mathbf{v}_j^g\|$, and the model divergence induced by the probability distance between the data distribution on the i-th local worker and the ground truth distribution for the whole population as on the genie worker, i.e., $\sum_{c=1}^{C} |p_i(y = c) - p(y = c)|$.

Remark 3. In (29), the initial model divergence (first term) and the velocity divergence (second term) after $(t+1)$ communication rounds are iteratively amplified by β. Since $\beta > 1$, if different local workers start from different initial model parameters in the standard FL, then the model divergence will still be enlarged, even though the local workers have i.i.d. data.

Remark 4. In (29), the third term $\sum_{c=1}^{C} |p_i(y = c) - p(y = c)|$ is the EMD between the data distribution on the i-th local worker and the population distribution [58], when the distance metric is defined as $|p_i(y = c) - p(y = c)|$. The impact of EMD is affected by the learning rate α, the number of communication rounds t, and the class-wise maximum gradient $f_{max}(\mathbf{w}_j)$.

VI. Experimental Results

This section demonstrates that our CB-DSL with a small amount of globally shared dataset outperforms the benchmark methods, with better learning performance and faster convergence speed, on both the i.i.d. and non-i.i.d. settings, even in the presence of Byzantine attacks.

A. System and Dataset Setting

To evaluate the learning performance of our CB-DSL, we perform empirical simulations[3] by conducting a handwritten-digit classification task based on the widely-used MNIST dataset[4] that consists of 10 classes ranging from digit "0" to "9". In the MNIST dataset, there are 60000 training samples and 10000 testing samples. In the training procedure, we set the total number of local workers to be $U = 50$, as the IoT devices in an edge network. For each local worker in the i.i.d. setting, 300 distinct training samples are randomly selected as its local datasets, i.e., $K_i = 300, \forall i$. To build the non-i.i.d. data setting upon the MNIST dataset, we first sort all the 60000 training samples based on the classification labels. Then we divide the 60000 training samples into 200 shards, each of which consists 300 samples, that are highly non-i.i.d. shard by shard [6]. We randomly allocate two shards to each local worker for the edge learning problem. The globally shared scoring dataset $\mathfrak{D}_{sc}^G$ consists of 2000 data samples, and the globally shared training dataset $\mathfrak{D}_{tr}^G$ consists of 150 data samples for the i.i.d. setting and 600 data samples for the non-i.i.d. setting. In addition, we set $c_0 = 1$, $\delta_{c_1} = 1$, and $\delta_{c_2} = 1$.

[3]Our code is available at: https://github.com/fuanxiyin/CB-DSL.git.

[4]http://yann.lecun.com/exdb/mnist/

TABLE I: Model architecture of the experiment.

Layer	Details
1	Conv2D(1, 6, 5) ReLU, MaxPool2D(2, 2)
2	Conv2D(6, 16, 5) ReLU, MaxPool2D(2, 2)
3	FC(16 * 4 * 4, 120) ReLU
4	FC(120, 84) ReLU
5	FC(84,10)

B. Neural Network Setting

For the learning model architecture, we use a five-layer Convolutional Neural Network (CNN) whose detailed hyper-parameter settings are listed in Table I. For the convolutional layers (Conv2D), we list the sizes of the parameters with sequence of input and output dimensions, and kernel size. For the max pooling layers (MaxPool2D), we list kernel and stride sizes. For the fully-connected layers (FC), we list input and output dimensions. During the training process, we use the SGD optimizer with learning rate $\alpha = 0.005$ and the cross-entropy loss. The batch size is set as $\|\mathfrak{B}_i\| = 10, \forall i$ for the mini-batch SGD [6], [7].

C. Different Approaches

We compare the proposed CB-DSL with FedAvg [5], given either i.i.d. or non-i.i.d. data[5], for different cases of globally shared dataset (without any shared dataset, with shared dataset for scoring, with shared dataset for training, with shared dataset for both scoring and training), including:

1) *FedAvg without any globally shared dataset* $\mathfrak{D}^G$: it is the standard FedAvg [5].
2) *CB-DSL without any globally shared dataset* $\mathfrak{D}^G$: the local workers use their own local dataset to calculate $F_{i,t}^p$.
3) *CB-DSL with a globally shared dataset for scoring* $\mathfrak{D}_{sc}^G$: the local workers use the globally shared scoring dataset to calculate $F_{i,t}^p$ in CB-DSL.
4) *FedAvg with a globally shared dataset for training* $\mathfrak{D}_{tr}^G$: the local workers use both their own local dataset and the globally shared training dataset to train their local models in standard FedAvg [5].
5) *CB-DSL with a globally shared dataset for both training* $\mathfrak{D}_{tr}^G$ *and scoring* $\mathfrak{D}_{sc}^G$: the local workers use both their own local dataset and the globally shared training dataset to train their local models and then use the globally shared scoring dataset to calculate $F_{i,t}^p$ in CB-DSL.

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

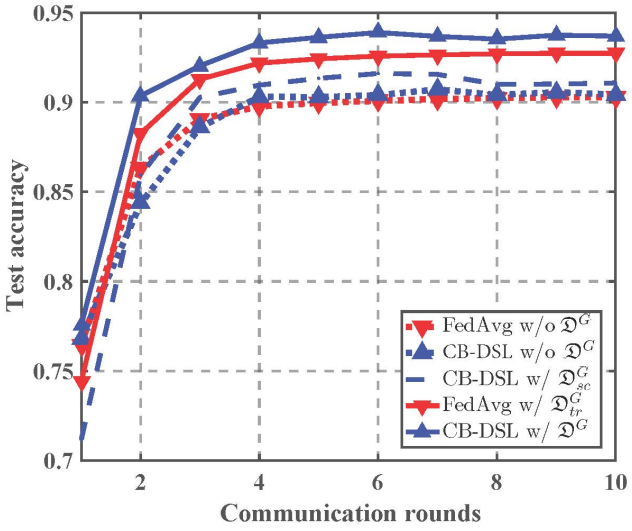


Fig. 1: The performance comparison under the i.i.d. setting.

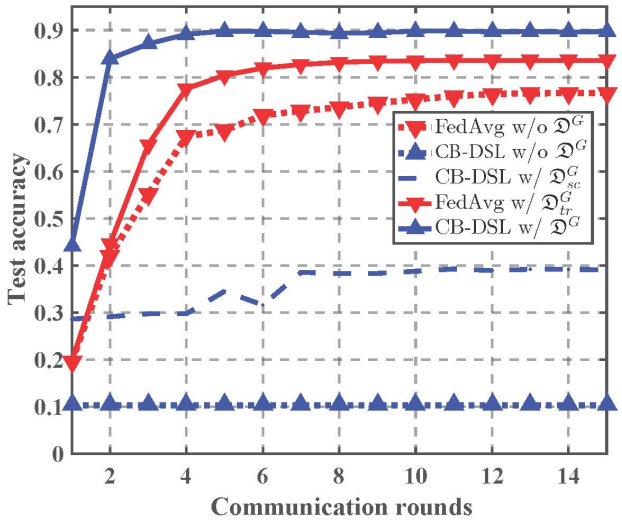


Fig. 2: The performance comparison under the non-i.i.d. setting.

D. Evaluation and Comparison

Fig. 1 and Fig. 2 show the simulation results for the five cases under the i.i.d. and the non-i.i.d. settings, respectively. As shown in Fig. 1, CB-DSL without $\mathfrak{D}^G$ is slightly better than FedAvg under the same learning settings for the i.i.d. case. A globally shared scoring dataset $\mathfrak{D}^G_{sc}$ introduced in CB-DSL can improve the learning performance of CB-DSL without any globally shared dataset. This is because $\mathfrak{D}^G_{sc}$ can help to select the global optimum more accurately than that based on local workers simply using their own dataset which however makes the loss function $F(\cdot)$ only partially observable at local workers. In addition, a globally shared training dataset $\mathfrak{D}^G_{tr}$ can further improve the learning performance of FedAvg and CB-DSL, since the data samples are increased for

[5]In this work, we mainly focus on evaluating the basic concept and general methodology of the proposed CB-DSL framework and algorithm design compared with the vanilla FedAvg, while other existing techniques for solving the non-i.i.d. issues can also be equipped with our CB-DSL for implementation in practice, such as the momentum-based methods [59] and regularization strategies [60].

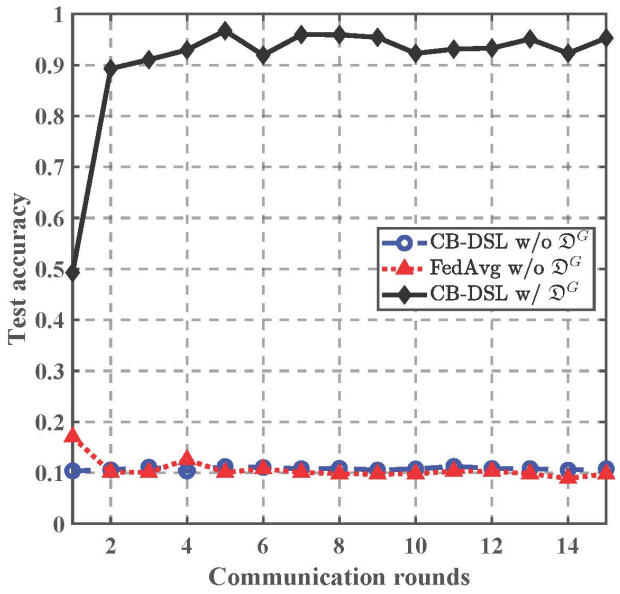


Fig. 3: The performance comparison with a Byzantine attacker under the i.i.d. setting.

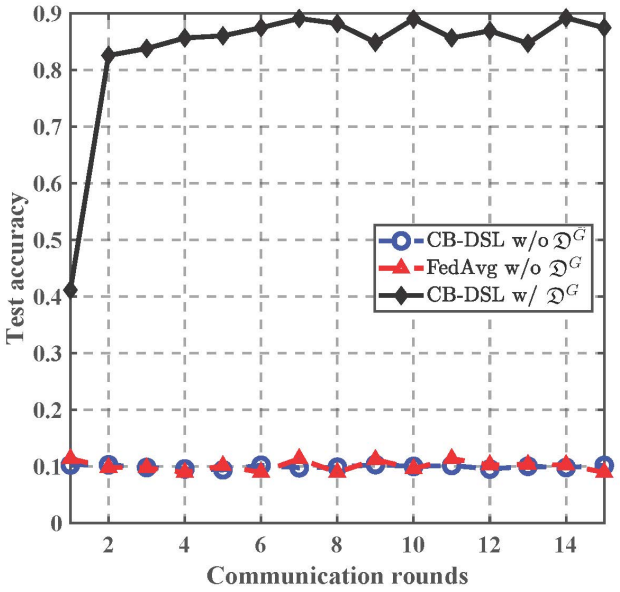


Fig. 4: The performance comparison with a Byzantine attacker under the non-i.i.d. setting.

training. Meanwhile, CB-DSL is superior thanks to its benefits by leveraging the exploration-exploitation gains from the BI component and the fast convergence characteristics from the AI component.

In Fig. 2, when CB-DSL runs without globally shared dataset for training $\mathfrak{D}^G_{tr}$, it cannot work properly in the non-i.i.d. setting. This is because CB-DSL hinges on single best worker selection which however may not hold the optimum model at all due to the model divergence from the ground truth population distribution point of view in the non-i.i.d. setting. Although using a globally shared scoring dataset $\mathfrak{D}^G_{sc}$ can slightly improve the learning performance of CB-DSL, it is still worse than FedAvg where all workers with non-i.i.d. data contribute to model average at the cost of high communication

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

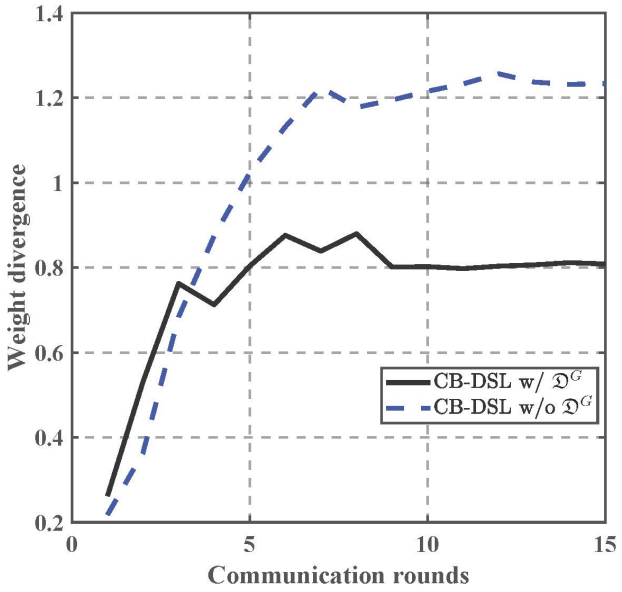


Fig. 5: The comparison of the weight divergences under the non-i.i.d. setting.

cost. When both a globally shared training dataset and scoring dataset are used as $\mathfrak{D}^G = \mathfrak{D}^G_{tr} \cup \mathfrak{D}^G_{sc}$, CB-DSL turns to outperform FedAvg. This is because $\mathfrak{D}^G_{tr}$ helps to relieve the local data heterogeneity issue by making the local datasets to become more i.i.d., which decreases the EMD between the data distributions on local workers and the population distribution as revealed by our model divergence analysis in Section V. Besides, the improvement on learning accuracy also indicates that by using the exploration-exploitation mechanism of PSO, our CB-DSL solutions have an increased chance to jump out of local optimum traps via the swarm intelligence.

In Fig. 3 and Fig. 4, we provide the performance comparison in the presence of the Byzantine attack for both the i.i.d. and the non-i.i.d. settings, respectively. In the simulations, the Byzantine attacker can send any information in order to fool the PS. From Fig. 3 and Fig. 4, it is obvious that even only one Byzantine attacker can fail FedAvg and CB-DSL without $\mathfrak{D}^G$. On the other hand, the CB-DSL with $\mathfrak{D}^G$ can effectively defend the Byzantine attack, because the globally shared dataset for scoring $\mathfrak{D}^G_{sc}$ can help identify and screen out the Byzantine attacker as explained in **Algorithm 1**.

In Fig. 5, we further evaluate the weight divergences effects under the non-i.i.d. setting. As the communication rounds increase, the weight divergences of CB-DSL with or without $\mathfrak{D}^G$ first increase and then flatten out after several communication rounds. The final steady-state weight divergence of the CB-DSL with $\mathfrak{D}^G$ is much less than that of the CB-DSL without $\mathfrak{D}^G$, as depicted by the gap between the two curves in Fig. 5. Such a nontrivial gap confirms the theoretical results of **Theorem 2**: (1) the model divergence will be enlarged as the communication rounds increase (this is because that the initial model divergence is iteratively amplified by β, as explained in *Remark 3*); (2) the use of global data $\mathfrak{D}^G$ can reduce the weight divergence (this is because that the use of $\mathfrak{D}^G$ decreases the EMD between the data distributions on local workers and the population distribution, as explained in *Remark 4*).

Note that only one local worker is selected and invited to send its model parameter to the PS in CB-DSL, while all workers need to send their model parameters to the PS in FedAvg. Therefore, the communication cost consumed in CB-DSL is only $\frac{1}{U}$ of that in FedAvg, given the fact that the communication cost for the transmission of loss function values as a scalar is relatively trivial to the transmission of the model parameter vector and thus can be ignored. In addition, we can see from Fig. 1 and Fig. 2 that our CB-DSL with $\mathfrak{D}^G$ uses fewer communication rounds than FedAvg to achieve the same learning accuracy. As a result, our CB-DSL is communication-efficient with less communication rounds and less communication overhead per round in practical applications.

VII. Conclusion

This work studies a novel communication-efficient and Byzantine-robust distributed swarm learning (CB-DSL) approach for edge IoT systems, as a holistic integration of the AI-enabled SGD and the BI-enabled PSO. We propose to introduce a globally shared dataset to overcome the major challenging issues in edge learning including: the partially observability of loss function in distributed learning problems, the non-i.i.d. local data issues, and the potential Byzantine attacks. We provide theoretical analysis of the convergence behavior of the proposed CB-DSL, which indicates that our method can achieve better learning performance than existing distributed learning methods. Further, we provide the model divergence evaluation of the proposed CB-DSL in the non-i.i.d. settings, which quantifies how a globally shared dataset can improve the learning performance of the CB-DSL in the non-i.i.d. setting. Simulation results verify that our proposed CB-DSL solution can improve learning performance in both the i.i.d. and non-i.i.d. settings, compared with the standard FedAvg. Meanwhile, the communication saving by the CB-DSL inherits the advantage of the bio-inspired PSO techniques with much reduced communication cost than standard FedAvg.

Appendix A
Proof of **Theorem 1**

Proof. Because $F_i(\cdot)$ is L-smooth from Assumption 1, according to [61, Lemma 3.4] and velocity update in (11), we have

$$\begin{aligned}
&F_i(\mathbf{w}_{i,t+1}) - F_i(\mathbf{w}_{i,t}) \\
&\leq (\mathbf{w}_{i,t+1} - \mathbf{w}_{i,t})^T \nabla F_i(\mathbf{w}_{i,t}) + \frac{L}{2}\|\mathbf{w}_{i,t+1} - \mathbf{w}_{i,t}\|^2 \\
&= \mathbf{v}_{i,t+1}^T \nabla F_i(\mathbf{w}_{i,t}) + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2 \\
&= (c_0 - c_1 - c_2)\mathbf{v}_{i,t}^T \nabla F_i(\mathbf{w}_{i,t}) + c_1(\mathbf{v}_{i,t}^p)^T \nabla F_i(\mathbf{w}_{i,t}) \\
&\quad + c_2(\mathbf{v}_t^g)^T \nabla F_i(\mathbf{w}_{i,t}) - \alpha\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2.
\end{aligned} \tag{30}$$

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

According to the definitions and assumptions of $\overline{q}$, $\overline{q}^p$, $\overline{q}^g$, $\underline{q}$, $\underline{q}^p$, $\underline{q}^g$, $\overline{u}$, $\overline{u}^p$, $\overline{u}^g$, $\underline{u}$, $\underline{u}^p$, $\underline{u}^g$ in (15)-(20), for any i and t, we have

$$\begin{aligned}\underline{u}\,\underline{q}\|\nabla F_i(\mathbf{w}_{i,t})\|^2 &\le \mathbf{v}_{i,t}^T\nabla F_i(\mathbf{w}_{i,t})\\ &= \|\mathbf{v}_{i,t}\|\|\nabla F_i(\mathbf{w}_{i,t})\|\cos\theta_{i,t}\\ &\le \overline{u}\,\overline{q}\|\nabla F_i(\mathbf{w}_{i,t})\|^2, \qquad (31)\\ \underline{q}^p\underline{u}^p\|\nabla F_i(\mathbf{w}_{i,t})\|^2 &\le (\mathbf{v}_{i,t}^p)^T\nabla F_i(\mathbf{w}_{i,t})\\ &= \|\mathbf{v}_{i,t}^p\|\|\nabla F_i(\mathbf{w}_{i,t})\|\cos\theta_{i,t}^p\\ &\le \overline{u}^p\overline{q}^p\|\nabla F_i(\mathbf{w}_{i,t})\|^2, \qquad (32)\\ \underline{q}^g\underline{u}^g\|\nabla F_i(\mathbf{w}_{i,t})\|^2 &\le (\mathbf{v}_t^g)^T\nabla F_i(\mathbf{w}_{i,t})\\ &= \|\mathbf{v}_t^g\|\|\nabla F_i(\mathbf{w}_{i,t})\|\cos\theta_t^g\\ &\le \overline{u}^g\overline{q}^g\|\nabla F_i(\mathbf{w}_{i,t})\|^2. \qquad (33)\end{aligned}$$

Substituting (31)-(33) to (30), we have

$$\begin{aligned}&F_i(\mathbf{w}_{i,t+1}) - F_i(\mathbf{w}_{i,t})\\ &\le (c_0 - c_1 - c_2)\overline{q}\,\overline{u}\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + c_1\overline{u}^p\overline{q}^p\|\nabla F_i(\mathbf{w}_{i,t})\|^2\\ &\quad + c_2\overline{u}^g\overline{q}^g\|\nabla F_i(\mathbf{w}_{i,t})\|^2 - \alpha\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2\\ &= (c_1\overline{u}^p\overline{q}^p + c_2\overline{u}^g\overline{q}^g + (c_0 - c_1 - c_2)\overline{q}\,\overline{u} - \alpha)\|\nabla F_i(\mathbf{w}_{i,t})\|^2\\ &\quad + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2. \qquad (34)\end{aligned}$$

Applying the triangle inequality of norms $\|\mathbf{X}+\mathbf{Y}\| \le \|\mathbf{X}\| + \|\mathbf{Y}\|$, the submultiplicative property of norms $\|\mathbf{XY}\| \le \|\mathbf{X}\|\|\mathbf{Y}\|$, and the Jensens inequality $(\sum_{i=1}^n a_i)^2 \le n\sum_{i=1}^n a_i^2$, we have

$$\begin{aligned}&\|\mathbf{v}_{i,t+1}\|^2 = \|(c_0 - c_1 - c_2)\mathbf{v}_{i,t} + c_1\mathbf{v}_{i,t}^p + c_2\mathbf{v}_t^g - \alpha\nabla F_i(\mathbf{w}_{i,t})\|^2\\ &\le (\|(c_0 - c_1 - c_2)\mathbf{v}_{i,t}\| + \|c_1\mathbf{v}_{i,t}^p\| + \|c_2\mathbf{v}_t^g\| + \|\alpha\nabla F_i(\mathbf{w}_{i,t})\|)^2\\ &\le 4((c_0 - c_1 - c_2)^2\|\mathbf{v}_{i,t}\|^2 + c_1^2\|\mathbf{v}_{i,t}^p\|^2 + c_2^2\|\mathbf{v}_t^g\|^2 + \alpha^2\|\nabla F_i(\mathbf{w}_{i,t})\|^2). \qquad (35)\end{aligned}$$

According to the assumptions of $\overline{u}$, $\overline{u}^p$, $\overline{u}^g$ in (18)-(20), for any i and t, we have

$$\begin{aligned}\|\mathbf{v}_{i,t}\| &\le \overline{u}\|\nabla F_i(\mathbf{w}_{i,t})\|, \qquad (36)\\ \|\mathbf{v}_{i,t}^p\| &\le \overline{u}^p\|\nabla F_i(\mathbf{w}_{i,t})\|, \qquad (37)\\ \|\mathbf{v}_t^g\| &\le \overline{u}^g\|\nabla F_i(\mathbf{w}_{i,t})\|. \qquad (38)\end{aligned}$$

Substituting (36)-(38) to (35), we have

$$\begin{aligned}&\|\mathbf{v}_{i,t+1}\|^2\\ &\le 4((c_0\overline{u} - c_1\overline{u} - c_2\overline{u})^2\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + c_1^2(\overline{u}^p)^2\|\nabla F_i(\mathbf{w}_{i,t})\|^2\\ &\quad + c_2^2(\overline{u}^g)^2\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + \alpha^2\|\nabla F_i(\mathbf{w}_{i,t})\|^2)\\ &= 4((c_0\overline{u} - c_1\overline{u} - c_2\overline{u})^2 + c_1^2(\overline{u}^p)^2 + c_2^2(\overline{u}^g)^2 + \alpha^2)\|\nabla F_i(\mathbf{w}_{i,t})\|^2. \qquad (39)\end{aligned}$$

Substituting (39) to (34), we have

$$\begin{aligned}&F_i(\mathbf{w}_{i,t+1}) - F_i(\mathbf{w}_{i,t})\\ &\le (c_1\overline{u}^p\overline{q}^p + c_2\overline{u}^g\overline{q}^g + (c_0 - c_1 - c_2)\overline{q}\,\overline{u} - \alpha)\|\nabla F_i(\mathbf{w}_{i,t})\|^2\\ &+ 2L((c_0\overline{u} - c_1\overline{u} - c_2\overline{u})^2 + c_1^2(\overline{u}^p)^2 + c_2^2(\overline{u}^g)^2 + \alpha^2)\|\nabla F_i(\mathbf{w}_{i,t})\|^2\\ &= \Phi\|\nabla F_i(\mathbf{w}_{i,t})\|^2, \qquad (40)\end{aligned}$$

where $\Phi = c_1\overline{u}^p\overline{q}^p + c_2\overline{u}^g\overline{q}^g + (c_0 - c_1 - c_2)\overline{q}\,\overline{u} - \alpha + 2L((c_0\overline{u} - c_1\overline{u} - c_2\overline{u})^2 + c_1^2(\overline{u}^p)^2 + c_2^2(\overline{u}^g)^2 + \alpha^2)$.

Then we extend the expectation over randomness introduced by CB-DSL and mini-batch training data in the trajectory of iterations, and perform a telescoping sum of (40) over the T iterations

$$\begin{aligned}F(\mathbf{w}_{i,0}) - F(\mathbf{w}^*) &\ge F(\mathbf{w}_{i,0}) - \mathbb{E}[F(\mathbf{w}_{i,T})]\\ &= \mathbb{E}\left[\sum_{t=1}^T (F(\mathbf{w}_{i,t-1}) - F(\mathbf{w}_{i,t}))\right]\\ &\ge \mathbb{E}\left[\sum_{t=1}^T \Phi_E\|\nabla F_i(\mathbf{w}_{i,t})\|^2\right], \qquad (41)\end{aligned}$$

where $\Phi_E = \mathbb{E}[-\Phi] = -\frac{\delta_{c_1}}{2}\overline{u}^p\overline{q}^p - \frac{\delta_{c_2}}{2}\overline{u}^g\overline{q}^g - \frac{2c_0 - \delta_{c_1} - \delta_{c_2}}{2}\overline{q}\,\overline{u} + \alpha - 2L((c_0^2 - \delta_{c_1}c_0 - \delta_{c_2}c_0 + \frac{\delta_{c_1}^2}{3} + \frac{\delta_{c_2}^2}{3} + \frac{\delta_{c_1}\delta_{c_2}}{2})\overline{u}^2 + \frac{\delta_{c_1}^2}{3}(\overline{u}^p)^2 + \frac{\delta_{c_2}^2}{3}(\overline{u}^g)^2 + \alpha^2)$.

Finally, we can rearrange the inequality of (41) to yield the convergence rate

$$\mathbb{E}\left[\sum_{t=1}^T \frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right] \le \frac{F(\mathbf{w}_{i,0}) - F(\mathbf{w}^*)}{T\Phi_E}. \qquad (42)$$

Hence, the proof is completed. □

APPENDIX B
PROOF OF THEOREM 2

Proof. Based on the definitions of $\mathbf{w}_{i,t+1}$ and $\mathbf{w}_{t+1}^g$ in (5) and (27), we have

$$\begin{aligned}\|\mathbf{w}_{i,t+1} - \mathbf{w}_{t+1}^g\| &= \|\mathbf{w}_{i,t} - \mathbf{w}_t^g + \mathbf{v}_{i,t+1} - \mathbf{v}_{t+1}^g\|\\ &\le \|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| + \|\mathbf{v}_{i,t+1} - \mathbf{v}_{t+1}^g\|. \qquad (43)\end{aligned}$$

Then based on the definitions of $\mathbf{v}_{i,t+1}$ and $\mathbf{v}_{t+1}^g$ in (11) and (26), we get

$$\begin{aligned}&\|\mathbf{v}_{i,t+1} - \mathbf{v}_{t+1}^g\|\\ &= \|(c_0 - c_1 - c_2)\mathbf{v}_{i,t} + c_1\mathbf{v}_{i,t}^p + (c_2 - c_0)\mathbf{v}_t^g\\ &\quad - \alpha\nabla F_i(\mathbf{w}_{i,t}) + \alpha\nabla F(\mathbf{w}_t^g)\|\\ &\le \|(c_0 - c_1 - c_2)\mathbf{v}_{i,t} + c_1\mathbf{v}_{i,t}^p + (c_2 - c_0)\mathbf{v}_t^g\|\\ &\quad + \|\alpha\nabla F_i(\mathbf{w}_{i,t}) - \alpha\nabla F(\mathbf{w}_t^g)\|\\ &\le \|(c_0 - c_1 - c_2)\mathbf{v}_{i,t} + c_1\mathbf{v}_t^g + (c_2 - c_0)\mathbf{v}_t^g\|\\ &\quad + \|\alpha\nabla F_i(\mathbf{w}_{i,t}) - \alpha\nabla F(\mathbf{w}_t^g)\|\\ &= |c_0 - c_1 - c_2|\|\mathbf{v}_{i,t} - \mathbf{v}_t^g\| + \alpha\|\nabla F_i(\mathbf{w}_{i,t}) - \nabla F(\mathbf{w}_t^g)\|. \qquad (44)\end{aligned}$$

Given the definitions of gradients at local workers and the genie worker $\nabla F_i(\mathbf{w}_{i,t}) = \sum_{c=1}^C p_i(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x}, \mathbf{w}_{i,t})]$

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

and $\nabla F(\mathbf{w}_t^g) = \sum_{c=1}^{C} p(y{=}c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)]$, respectively, we have

$$\begin{aligned}
&\|\nabla F_i(\mathbf{w}_{i,t}) - \nabla F(\mathbf{w}_t^g)\| \\
&= \Bigg\| \sum_{c=1}^{C} p_i(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_{i,t})] \\
&\qquad - \sum_{c=1}^{C} p(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)] \Bigg\| \\
&= \Bigg\| \sum_{c=1}^{C} p_i(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_{i,t})] \\
&\qquad - \sum_{c=1}^{C} p_i(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)] \\
&\qquad + \sum_{c=1}^{C} p_i(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)] \\
&\qquad - \sum_{c=1}^{C} p(y=c)\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)] \Bigg\| \\
&\le \Bigg\| \sum_{c=1}^{C} p_i(y{=}c)(\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_{i,t})] - \nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)]) \Bigg\| \\
&\quad + \Bigg\| \sum_{c=1}^{C} (p_i(y{=}c) - p(y{=}c))\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)] \Bigg\|. \qquad (45)
\end{aligned}$$

Letting $f_{max}(\mathbf{w}_t^g) = \max\{\nabla\mathbb{E}_{\mathbf{x}|y=c}[f_c(\mathbf{x},\mathbf{w}_t^g)]\}_{c=1}^{C}$, and applying the Lipschitz continuity, the equality of (45) can be further rewritten as

$$\begin{aligned}
&\|\nabla F_i(\mathbf{w}_{i,t}) - \nabla F(\mathbf{w}_t^g)\| \\
&\qquad \le \sum_{c=1}^{C} p_i(y=c)L_c\|\mathbf{w}_{i,t} - \mathbf{w}_t\| \\
&\qquad\quad + f_{max}(\mathbf{w}_t^g)\sum_{c=1}^{C} |(p_i(y=c) - p(y=c))|. \qquad (46)
\end{aligned}$$

Combining (43), (44), and (45), we have

$$\begin{aligned}
&\|\mathbf{w}_{i,t+1} - \mathbf{w}_{t+1}^g\| \\
&\le \|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| + \|\mathbf{v}_{i,t+1} - \mathbf{v}_{t+1}^g\| \\
&\le \|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| + |c_0{-}c_1{-}c_2|\|\mathbf{v}_{i,t} - \mathbf{v}_t^g\| \\
&\quad + \alpha\|\nabla F_i(\mathbf{w}_{i,t}) - \nabla F(\mathbf{w}_t^g)\| \\
&\le \|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| + |c_0{-}c_1{-}c_2|\|\mathbf{v}_{i,t} - \mathbf{v}_t^g\| \\
&\quad + \alpha\sum_{c=1}^{C} p_i(y{=}c)L_c\|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| \\
&\quad + \alpha f_{max}(\mathbf{w}_t^g)\sum_{c=1}^{C} |p_i(y=c) - p(y=c)| \\
&= (1 + \alpha\sum_{c=1}^{C} p_i(y=c)L_c)\|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| \\
&\quad + |c_0 - c_1 - c_2|\,\|\mathbf{v}_{i,t}{-}\mathbf{v}_t^g\| \\
&\quad + \alpha f_{max}(\mathbf{w}_t^g)\sum_{c=1}^{C} |p_i(y=c) - p(y=c)|. \qquad (47)
\end{aligned}$$

Letting $\beta = 1 + \alpha\sum_{c=1}^{C} p_i(y=c)L_c$, we rewrite (47) as

$$\begin{aligned}
&\|\mathbf{w}_{i,t+1} - \mathbf{w}_{t+1}^g\| \\
&\le \beta\|\mathbf{w}_{i,t} - \mathbf{w}_t^g\| + |c_0 - c_1 - c_2|\|\mathbf{v}_{i,t} - \mathbf{v}_t^g\| \\
&\quad + \alpha f_{max}(\mathbf{w}_t^g)\sum_{c=1}^{C} |p_i(y=c) - p(y=c)| \\
&\le \beta^2\|\mathbf{w}_{i,t-1} - \mathbf{w}_{t-1}^g\| + \beta|c_0 - c_1 - c_2|\|\mathbf{v}_{i,t-1} - \mathbf{v}_{t-1}^g\| \\
&\quad + \beta\alpha f_{max}(\mathbf{w}_t^g)\sum_{c=1}^{C}|p_i(y{=}c){-}p(y{=}c)| + |c_0{-}c_1{-}c_2|\|\mathbf{v}_{i,t}{-}\mathbf{v}_t^g\| \\
&\quad + \alpha f_{max}(\mathbf{w}_t^g)\sum_{c=1}^{C} |p_i(y=c) - p(y=c)| \\
&\le \beta^{t+1}\|\mathbf{w}_{i,0} - \mathbf{w}_0^g\| + |c_0 - c_1 - c_2|\sum_{j=0}^{t}\beta^{t-j}\|\mathbf{v}_{i,j} - \mathbf{v}_j^g\| \\
&\quad + \alpha\sum_{c=1}^{C} |p_i(y=c){-}p(y=c)|\sum_{j=0}^{t} f_{max}(\mathbf{w}_j^g). \qquad (48)
\end{aligned}$$

Hence, the proof is completed. □

REFERENCES

[1] Y. Wang and Z. Tian, *Big Data in 5G*. Encyclopedia of Wireless Networks, Springer International Publishing, 2018.

[2] 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*, vol. 33, no. 5, pp. 156–165, 2019.

[3] Y. Shi, K. Yang, T. Jiang, J. Zhang, and K. B. Letaief, "Communication-efficient edge ai: Algorithms and systems," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2167–2191, 2020.

[4] C. Perera, R. Ranjan, L. Wang, S. U. Khan, and A. Y. Zomaya, "Big data privacy in the internet of things era," *IT Professional*, vol. 17, no. 3, pp. 32–39, 2015.

[5] 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.

[6] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[7] Q. Qian, R. Jin, J. Yi, L. Zhang, and S. Zhu, "Efficient distance metric learning by adaptive sampling and mini-batch stochastic gradient descent (sgd)," *Machine Learning*, vol. 99, no. 3, pp. 353–372, 2015.

[8] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik, "Sgd: General analysis and improved rates," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5200–5209.

[9] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[10] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[11] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018.

[12] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[13] Z. Huo and H. Huang, "Asynchronous stochastic gradient descent with variance reduction for non-convex optimization," *arXiv preprint arXiv:1604.03584*, 2016.

[14] S. Vlaski and A. H. Sayed, "Second-order guarantees of stochastic gradient descent in non-convex optimization," *IEEE Transactions on Automatic Control*, 2021.

[15] Z. Yang, A. Gang, and W. U. Bajwa, "Adversary-resilient distributed and decentralized statistical inference and machine learning: An overview of recent advances under the byzantine threat model," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 146–159, 2020.

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

[16] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "Bev-sgd: Best effort voting sgd against byzantine attacks for analog-aggregation-based federated learning over the air," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18 946–18 959, 2022.

[17] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5650–5659.

[18] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "Best effort voting power control for byzantine-resilient federated learning over the air," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2022, pp. 1–6.

[19] A. Khanna and S. Kaur, "Internet of Things (IoT), applications and challenges: A comprehensive review," *Wireless Personal Communications*, vol. 114, no. 2, pp. 1687–1762, 2020.

[20] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.

[21] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine learning in IoT security: Current solutions and future challenges," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1686–1721, 2020.

[22] C. Selvaraj, R. S. Kumar, and M. Karnan, "A survey on application of bio-inspired algorithms," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 1, pp. 366–70, 2014.

[23] S. Almufti, R. Marqas, and V. Ashqi, "Taxonomy of bio-inspired optimization algorithms," *Journal Of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 23, 2019.

[24] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the sixth international symposium on micro machine and human science*. Ieee, 1995, pp. 39–43.

[25] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[26] A. F. Aji and K. Heafield, "Sparse communication for distributed gradient descent," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 440–445.

[27] Y. Lin, S. Han, H. Mao, Y. Wang, and B. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *International Conference on Learning Representations*, 2018.

[28] Y. Liu, K. Yuan, G. Wu, Z. Tian, and Q. Ling, "Decentralized dynamic admm with quantized and censored communications," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2019, pp. 1496–1500.

[29] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns," in *Fifteenth Annual Conference of the International Speech Communication Association*. Citeseer, 2014.

[30] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "Qsgd: Communication-efficient sgd via gradient quantization and encoding," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[31] Y. Liu, W. Xu, G. Wu, Z. Tian, and Q. Ling, "Communication-censored admm for decentralized consensus optimization," *IEEE Transactions on Signal Processing*, vol. 67, no. 10, pp. 2565–2579, 2019.

[32] P. Xu, Z. Tian, Z. Zhang, and Y. Wang, "Coke: Communication-censored kernel learning via random features," in *2019 IEEE Data Science Workshop (DSW)*. IEEE, 2019, pp. 32–36.

[33] P. Xu, Z. Tian, and Y. Wang, "An energy-efficient distributed average consensus scheme via infrequent communication," in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2018, pp. 648–652.

[34] P. Xu, Y. Wang, X. Chen, and Z. Tian, "Coke: Communication-censored decentralized kernel learning," *Journal of Machine Learning Research*, vol. 22, no. 196, pp. 1–35, 2021.

[35] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2022–2035, 2020.

[36] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "Joint optimization for federated learning over the air," in *2022 IEEE International Conference on Communications (ICC 2022)*. IEEE, 2022, pp. 1–6.

[37] ——, "Joint optimization of communications and federated learning over the air," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 4434–4449, 2022.

[38] ——, "Communication-efficient federated learning through 1-bit compressive sensing and analog aggregation," in *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2021, pp. 1–6.

[39] ——, "1-bit compressive sensing for efficient federated learning over the air," *IEEE Transactions on Wireless Communications*, 2022.

[40] A. R. Syulistyo, D. M. J. Purnomo, M. F. Rachmadi, and A. Wibowo, "Particle swarm optimization (pso) for training optimization on convolutional neural network (cnn)," *Jurnal Ilmu Komputer dan Informasi*, vol. 9, no. 1, pp. 52–58, 2016.

[41] F. E. F. Junior and G. G. Yen, "Particle swarm optimization of deep neural networks architectures for image classification," *Swarm and Evolutionary Computation*, vol. 49, pp. 62–74, 2019.

[42] T. Serizawa and H. Fujita, "Optimization of convolutional neural network using the linearly decreasing weight particle swarm optimization," *arXiv preprint arXiv:2001.05670*, 2020.

[43] B. Qolomany, K. Ahmad, A. Al-Fuqaha, and J. Qadir, "Particle swarm optimized federated learning for industrial iot and smart city services," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 2020, pp. 1–6.

[44] S. Park, Y. Suh, and J. Lee, "Fedpso: federated learning using particle swarm optimization to reduce communication costs," *Sensors*, vol. 21, no. 2, p. 600, 2021.

[45] N. Yoshida, T. Nishio, M. Morikura, K. Yamamoto, and R. Yonetani, "Hybrid-fl for wireless networks: Cooperative learning mechanism using non-iid data," in *ICC 2020-2020 IEEE International Conference On Communications (ICC)*. IEEE, 2020, pp. 1–7.

[46] T. Tuor, S. Wang, B. J. Ko, C. Liu, and K. K. Leung, "Overcoming noisy and irrelevant data in federated learning," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 5020–5027.

[47] M. Duan, D. Liu, X. Chen, Y. Tan, J. Ren, L. Qiao, and L. Liang, "Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications," in *2019 IEEE 37th international conference on computer design (ICCD)*. IEEE, 2019, pp. 246–254.

[48] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data," *arXiv preprint arXiv:1811.11479*, 2018.

[49] C. T Dinh, N. Tran, and J. Nguyen, "Personalized federated learning with moreau envelopes," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 394–21 405, 2020.

[50] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," *arXiv preprint arXiv:1912.00818*, 2019.

[51] X.-C. Li, L. Gan, D.-C. Zhan, Y. Shao, B. Li, and S. Song, "Aggregate or not? exploring where to privatize in dnn based federated learning under different non-iid scenes," *arXiv preprint arXiv:2107.11954*, 2021.

[52] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *Neurocomputing*, vol. 465, pp. 371–390, 2021.

[53] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.

[54] 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 Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2020.

[55] W. Liu, L. Chen, Y. Chen, and W. Zhang, "Accelerating federated learning via momentum gradient descent," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 8, pp. 1754–1766, 2020.

[56] J. Wang and G. Joshi, "Cooperative sgd: A unified framework for the design and analysis of local-update sgd algorithms," *Journal of Machine Learning Research*, vol. 22, no. 213, pp. 1–50, 2021.

[57] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "signsgd: Compressed optimisation for non-convex problems," in *International Conference on Machine Learning*. PMLR, 2018, pp. 560–569.

[58] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision*, vol. 40, no. 2, pp. 99–121, 2000.

[59] A. Xu and H. Huang, "Coordinating momenta for cross-silo federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 8, 2022, pp. 8735–8743.

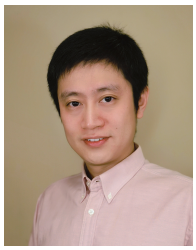
[60] V. Saligrama, D. A. E. Acar, P. N. Whatmough, R. Matas, M. Mattina, and Y. Zhao, "Federated learning based on dynamic regularization," *arXiv preprint arXiv:2111.04263*, 2022.

[61] S. Bubeck *et al.*, "Convex optimization: Algorithms and complexity," *Foundations and Trends in Machine Learning*, vol. 8, no. 3-4, pp. 231–357, 2015.

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2023.3312345

Xin Fan (Student Member, IEEE) received the B.E., M.E. and Ph.D. degrees from the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China, in 2016, 2018 and 2023, respectively. He was a visiting Ph.D. student in the Electrical and Computer Engineering Department of George Mason University, Fairfax, VA, USA, from 2020 to 2022. He is currently an Assistant Professor with the School of Information Science and Technology, Beijing Forestry University, Beijing, China. His current research interests lie in the areas of wireless communications, machine learning, security and privacy, optimization, statistical signal processing, and blockchain.



Yue Wang (Senior Member, IEEE) received the Ph.D. degree in communication and information system from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2011. Currently, he is an Assistant Professor with the Department of Computer Science, Georgia State University, Atlanta, GA, USA. Previously, he was a Research Assistant Professor with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA, USA. His general interests include machine learning, signal processing, wireless communications, and their applications in cyber-physical systems. His specific research focuses on compressive sensing, massive MIMO, millimeter-wave communications, wideband spectrum sensing, cognitive radios, direction of arrival estimation, high-dimensional data analysis, and distributed optimization and learning.

Yan Huo (Senior Member, IEEE) received the B.E. and Ph.D. degrees in communication and information system from Beijing Jiaotong University, Beijing, China, in 2004 and 2009, respectively. He was a Visiting Scholar with the Department of Computer Science, George Washington University, from 2015 to 2016. He is currently a Professor with the School of Electronics and Information Engineering, Beijing Jiaotong University. His research focuses on wireless communications, security and privacy, and the Internet of Things. It involves building and simulating prototype systems and conducting real experiments and measurements. He has served as Associate Editor for IEEE Access and a Reviewer for a number of journals, including IEEE Journal on Selected Areas in Communications, IEEE Wireless Communications, IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, IEEE Transactions on Vehicular Technology, and IEEE Transactions on Industrial Informatics.

Zhi Tian (Fellow, IEEE) is currently a Professor with the Electrical and Computer Engineering Department of George Mason University, Fairfax, VA, USA, since 2015. Prior to that, she was on the Faculty of Michigan Technological University, Houghton, MI, USA, from 2000 to 2014. She served as a Program Director at the U.S. National Science Foundation from 2012 to 2014. Her research interest lies in the areas of statistical signal processing, wireless communications, machine learning, and estimation and detection theory. Her current research focuses on compressed sensing for random processes, statistical inference of network data, distributed network optimization and learning, and millimeter-wave communications. She was an IEEE Distinguished Lecturer for both the IEEE Communications Society and the IEEE Vehicular Technology Society. She served as Associate Editor for IEEE Transactions on Wireless Communications and IEEE Transactions on Signal Processing. She received the IEEE Communications Society TCCN Publication Award in 2018. She was a Member-at-Large of the Board of Governors of the IEEE Signal Processing Society for the term of 2019-2021.

Authorized licensed use limited to: George Mason University. Downloaded on December 03,2023 at 19:21:04 UTC from IEEE Xplore. Restrictions apply.
© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.