Robust Distributed Swarm Learning for Intelligent IoT

Xin Fan[1], Yue Wang[2], Yan Huo[1], and Zhi Tian[2]
[1]School of Electronics and Information Engineering, Beijing Jiaotong University, Beijing, China
[2]Department of Electrical & Computer Engineering, George Mason University, Fairfax, VA, USA
E-mails: {yhuo,fanxin}@bjtu.edu.cn, {ywang56,ztian1}@gmu.edu

***Abstract*—In this paper, we study a communication-efficient distributed learning scheme through a holistic integration of federated learning (FL) and particle swarm optimization, called DSL, which is suitable for the implementation of intelligent IoT applications. Since only one selected optimum from all local devices need to report its local model updates to the parameter server, the communication cost of DSL is much reduced compared to its counterpart of standard FL. However, the DSL is vulnerable to adversarial attackers. To achieve Byzantine-resilient DSL, we propose to introduce a shared dataset for scoring local updates to screen attackers. We further provide the convergence analysis to theoretically demonstrate that CB-DSL is superior than the standard FL. Experiment results show that the learning performance of our proposed CB-DSL outperforms the existing benchmarks with only a small amount of globally shared data. It enjoys higher robustness against Byzantine attacks than the vanilla DSL, and has better communication efficiency than the standard FL[1].**

***Index Terms*—Federated learning, particle swarm optimization, communication efficiency, robustness, convergence analysis**

I. Introduction

With the vigorous development of the Internet of Things (IoT), edge devices have emerged as the main force of computing resources to fuel the development of wireless networks beyond 5G (B5G). A tremendous amount of valuable data collected and stored on these edge devices and the advanced machine learning technologies jointly drive the latest trend in artificial intelligence (AI) at the B5G network edge (edge AI) [1]. Enabling edge AI requires the distributed data can be rapidly and securely access. From either communication, security and privacy, regulatory or economic point of view, it is impractical for a central server to train a satisfactory learning model by collecting raw data from edge devices. Fortunately, federated learning (FL) provides a way for various IoT applications over edge devices in B5G IoT networks, which allows edge learning from distributed local data without compromising their privacy [2]–[5]. In FL, edge devices (local workers) periodically upload their locally trained models to an edge server (parameter server), where the local models are aggregated to update a global model. In this way, FL enables communication-efficient and privacy-preserving distributed learning without raw data exchange.

However, there still exist some remaining challenges in FL especially in its applications for IoT edge networks. For example, when the number of the model parameters is huge, transmission of their updates between edge devices and the parameter server (PS) appears as a major bottleneck to communication-constrained FL implemented over IoT-based edge networks [4], [6]. Besides, gradient descent algorithms are easy to fall into local optimums in solving non-convex problems [7], given the non-convex nature of the cost functions. Last but not the least, FL is vulnerable to Byzantine attacks, meaning that some local workers may behave completely arbitrarily to disrupt cooperative tasks in FL [8]–[10].

To jointly overcome all, we leverage the biological intelligence (BI) and propose a communication-efficient and Byzantine-resilient FL scheme (CB-DSL) by using particle swarm optimization. For the communication challenge, our proposed CB-DSL only requires the worker with the optimal local model to upload its local updates to the PS, which thus reduce communication costs dramatically. For the non-convex problems, CB-DSL takes the advantages of the exploration-and-exploitation mechanism, which enables FL to jump out of the local optimal trap [11], [12]. Since only one worker is selected to upload the optimal local model to the PS, the selected local worker may be a Byzantine attacker to upload an adversarial model. For Byzantine attack issues, a globally shared dataset is used as a globally scoring dataset to test the uploaded optimal local model, and the PS can screen and kick out potential Byzantine attackers if the scoring accuracy at the PS does not match what they reported. Our proposed CB-DSL establishes a new paradigm of efficient and robust edge intelligence through a holistic integration of AI and BI. Our main contributions are summarized as follows.

- We propose a CB-DSL approach to jointly handle the high communication cost, non-convex problem and Byzantine attacks in existing FL. In CB-DSL, local models are evaluated by a globally scoring dataset to select the optimal one. Then only one optimal local model needs to be uploaded to the PS rather than all the local models. Further, the selected optimal local model is verified by the PS to screen potential Byzantine attackers.
- From theoretical point of view, we derive the closed-form expression of the expected convergence rate for our CB-DSL. Our theoretical analysis reflects the impact of different system parameters on the performance of FL methods, and also indicates that our CB-DSL outperforms the standard FL method such as FedAvg.
- We evaluate the proposed CB-DSL in solving image classification problems by using the MNIST dataset.

[1]Our code can be found at:https://github.com/fuanxiyin/CB-DSL.git.

ICC 2023 - IEEE International Conference on Communications | 978-1-5386-7462-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICC45041.2023.10278708

Simulation results show that our proposed CB-DSL outperforms the benchmark methods in terms of higher testing accuracy and robustness.

II. Related Work

Various methods has been proposed in addressing the communication challenges of FL, such as sparsification [13], quantization [14] and infrequent uploading of local updates [15]–[17]. Other than these strategies designed for digital transmission, another promising solution from an aspect of transmissions is analog aggregation based FL, called FL over the air, which exploits the waveform superposition property of the wireless medium to support simultaneous transmission by all the devices [2], [4], [6], [8], [10], [18]. However, all the aforementioned methods require all participating local workers to upload their local updates to the PS, which results in tremendous communication costs in edge networks with massive smart IoT devices. In contrast, we aim to upload only one optimal local model to significantly save the overall communication cost of the massive-IoT edge networks.

Motivated by taking advantage of the swarm biological intelligence of animal flocks, PSO has been developed to solve optimization problems without the assumptions of convexity and differentiability [11], [12]. Recently, a few research efforts have been found in applying PSO algorithms to improve machine learning performance. For example, the authors propose to apply PSO to find the optimal hyperparameters for improving the learning performance of FL in [19]. The mentioned work does not consider to combine the PSO and FL from the algorithm perspective to leverage AI-enabled stochastic gradient descent and BI-enabled particle swarm optimization. To fill such technical gaps, our work proposes a new communication-efficient and Byzantine-resilient FL solution (CB-DSL) with rigorous convergence analysis to demonstrate the advantage of the holistic integration of AI and BI.

III. CB-DSL

In this section, we will start with the models and formulations of standard FL and PSO techniques. Then, we will introduce our communication-efficient and Byzantine-resilient CB-DSL algorithm design.

A. Federated Learning

Consider a distributed computation model with one parameter server (PS) and U local workers. Each local worker stores K data samples in its dataset $\mathfrak{D}_i$. Denote $(\mathbf{x}_{i,k}, y_{i,k})$ as the k-th data 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 point $(\mathbf{x}_{i,k}, y_{i,k})$, where $\mathbf{w} = [w^1, \ldots, w^D]$ of size D consists of the model parameters. The corresponding population loss function is expressed as $F(\mathbf{w}) := \mathbb{E}_{\mathfrak{D}}[f(\mathbf{w}; \mathbf{x}_{i,k}, y_{i,k})]$, where $\mathfrak{D} = \bigcup_i \mathfrak{D}_i$. The PS and local workers collaboratively learn the model parameter vector $\mathbf{w}$ by minimizing

$$\textbf{P1:} \quad \mathbf{w}^* = \arg\min_{\mathbf{w}} \quad F(\mathbf{w}). \tag{1}$$

The minimization of $F(\mathbf{w})$ is typically carried out through the stochastic gradient descent (SGD) algorithm. At the PS, the model parameter $\mathbf{w}_t$ at the t iteration is updated as

$$\text{(Model updating)} \quad \mathbf{w}_t = \mathbf{w}_{t-1} - \alpha \frac{\sum_{i=1}^{U} \mathbf{g}_{i,t}}{U}, \tag{2}$$

where α is the learning rate and $\mathbf{g}_{i,t} = \nabla F_i(\mathbf{w}_{t-1}; \mathbf{x}_{i,k}, y_{i,k}) = \mathbb{E}_{\mathfrak{D}_i}[\frac{\sum_{\mathfrak{B}_i} \nabla f(\mathbf{w}_{t-1}; \mathbf{x}_{i,k}, y_{i,k})}{|\mathfrak{B}_i|}]$ is the local gradient computed at the i-th local worker using its randomly selected mini-batch $\mathfrak{B}_i \subset \mathfrak{D}_i$ with the mini-batch size $|\mathfrak{B}_i|$. The communication overhead for the PS to acquire the sum of local gradients in (2) from local workers in each iteration would be huge especially when D is large.

B. PSO

PSO is a probabilistic approach to solve optimization problem [11], [12], e.g., the problem **P1**. In PSO, the swarm consists of a set of particles, $i = 1, 2..., U$. At the t-th iteration, a particle i holds a particle best solution to the problem represented by a position $\mathbf{w}_{i,t}^p$ in a given search space, and has a updating direction represented by a speed $\mathbf{v}_{i,t}$ for the next step. To find the global optimal value, particles communicate with each other to share their own $\mathbf{w}_{i,t}^p$ variable step-by-step. In this way, each particle is able to set a common $\mathbf{w}_t^g$ (global best) variable from the shared $\mathbf{w}_{i,t}^p$ values that leads to the optimal value of the cost function at the current iteration: $\mathbf{w}_t^g = \arg\min_{i=1,2,\ldots,U} F(\mathbf{w}_{i,t}^p)$. The parameters $\mathbf{w}_t^g$ and $\mathbf{w}_{i,t}^p$ are used for particles to move on to the next step as

$$\mathbf{v}_{i,t+1} = c_0 \mathbf{v}_{i,t} + c_1(\mathbf{w}_{i,t}^p - \mathbf{w}_{i,t}) + c_2(\mathbf{w}_t^g - \mathbf{w}_{i,t}), \tag{3}$$

$$\mathbf{w}_{i,t+1} = \mathbf{w}_{i,t} + \mathbf{v}_{i,t+1}, \tag{4}$$

where c_0 is a positive constant representing the inertia weight, c_1, and c_2 are two random acceleration factors for the particle optimum and the global optimum, which follow the continuous uniform distributions $\mathcal{U}(0, \delta_{c_1})$ and $\mathcal{U}(0, \delta_{c_2})$, respectively.

C. CB-DSL

In CB-DSL, each particle, e.g., i, initiates its starting position $\mathbf{w}_{i,0}^p$ and speed $\mathbf{v}_{i,0}$ for the next step. Given $\mathbf{w}_{i,0}^p$ and its dataset $\mathfrak{D}_i$, each particle calculates its particle cost $F_i(\mathbf{w}_{i,0}^p; \mathfrak{D}_i)$ as its particle optimum $F_{i,0}^p$. At the t-th communication round, each particle sends its particle optimum $F_{i,t}^p$ to the PS, and the PS compares all the received $F_{i,t}^p$'s to select the global optimum by $F_t^g = \min\{F_{i,t}^p\}_i^U$. Then the PS broadcasts the index i_t^p of the selected particle and the selected particle broadcasts its position $\mathbf{w}_{i,t}^p$ as the current global optimal position $\mathbf{w}_t^g$. Given the received $\mathbf{w}_t^g$ and its own $\mathbf{w}_{i,t}^p$, each particle calculates its local gradient $\nabla F_i(\mathbf{w}_{i,t})$ and updates its position and speed as

$$\mathbf{v}_{i,t+1} = c_0 \mathbf{v}_{i,t} + c_1(\mathbf{w}_{i,t}^p - \mathbf{w}_{i,t}) + c_2(\mathbf{w}_t^g - \mathbf{w}_{i,t}) + \alpha \nabla F_i(\mathbf{w}_{i,t}), \tag{5}$$

$$\mathbf{w}_{i,t+1} = \mathbf{w}_{i,t} - \mathbf{v}_{i,t+1}. \tag{6}$$

Then each particle updates its particle optimum $F_{i,t+1}^p$ by $F_{i,t+1}^p = \min\{F_{i,t}^p, F_i(\mathbf{w}_{i,t+1}^p; \mathfrak{D}_i)\}$, which is sent to the PS

for the next iteration. Meanwhile, each particle also updates its particle optimal position as $\mathbf{w}_{i,t+1}^p = \mathbf{w}_{i,t+1}$ if $F_{i,t}^p > F_i(\mathbf{w}_{i,t+1}^p; \mathfrak{D}_i)$; or $\mathbf{w}_{i,t+1}^p = \mathbf{w}_{i,t}^p$ otherwise. The iterations are implemented between the PS and all the particles until convergence.

Since only the model parameter from the local worker with the global optimum score is requested to be reported to the PS, the communication cost at each communication round is reduced significantly in CB-DSL, compared with that required by standard FL. Besides, introducing a speed term into SGD, both the updates of the speed and the gradient in (5) contribute to seek an optimum for individual local workers, which leads to an improvement of SGD. However, the process of collecting $F_{i,t}^p$'s is inherently vulnerable to Byzantine attacks, i.e., a local worker may perform Byzantine attack to send a fake $F_{i,t}^p$ to fool the PS to select its model parameter as the global optimum, which would destroy FL. To solve this problem, we propose to take advantage of a globally shared dataset $\mathfrak{D}_{sc}^G$ to screen Byzantine attackers.

Specifically, we introduce a small dataset $\mathfrak{D}_{sc}^G$ of data which is globally shared between all the local workers and the PS before starting FL[2]. Each particle calculates the particle cost $F_{i,t}^p$ with $\mathfrak{D}_{sc}^G$, i.e., $F_i(\mathbf{w}_{i,t}^p; \mathfrak{D}_{sc}^G)$. After the PS selects the particle, the global optimal model parameter broadcasted by the selected particle can be verified at the PS and all the local workers. If find a Byzantine attack, the attacker would be kicked out to promise a Byzantine-resilient FL.

The detailed steps and operations of our CB-DSL is summarized in **Algorithm 1**.

Algorithm 1 CB-DSL

Initialization:

$\mathbf{w}_{i,0}^p = \mathbf{w}_{i,0}$, $F_{i,t}^p$, for any i and t;

1: **for** each round $t = 1 : T$ **do**
2: **At the workers:**
3: Iteratively update the local model parameter $\mathbf{w}_{i,t}$ and speed $\mathbf{v}_{i,t}$ via (5) and (6);
4: Calculate $F_i(\mathbf{w}_{i,t}; \mathfrak{D}_{sc}^G)$ with the globally shared dataset $\mathfrak{D}_{sc}^G$ and the model parameter $\mathbf{w}_{i,t}$;
5: Set the minimal particle cost $F_{i,t}^p = \min\{F_{i,t-1}^p, F_i(\mathbf{w}_{i,t}^p; \mathfrak{D}_{sc}^G)\}$;
6: **if** $F_{i,t}^p == F_{i,t-1}^p$ **then**
7: Set $\mathbf{w}_{i,t}^p = \mathbf{w}_{i,t-1}^p$;
8: **else**
9: Set $\mathbf{w}_{i,t}^p = \mathbf{w}_{i,t}$;
10: **end if**
11: Send the minimal particle cost $F_{i,t}^p$ to the PS;
12: Upon receiving the index of the selected local worker i_t^p, the i_t^p-th local worker sends $\mathbf{w}_{i,t}^p$ to the PS;
13: **At the PS:**
14: Upon receiving all the $F_{i,t}^p$'s, set $F_t^g = \min\{F_{i,t}^p\}_i^U$ and select the corresponding worker i_t^p;
15: Broadcast i_t^p to local workers;
16: Upon receiving $\mathbf{w}_{i,t}^p$ from the i_t^p-th worker, verify its minimal particle cost $F_{i,t}^p$ by using $\mathfrak{D}_{sc}^G$;
17: If find a attacker, kick it out and repeat line 14 until a legitimate worker is selected.
18: **end for**

IV. Convergence Analysis

In this section, we aim to give the theoretical analysis on the convergence guarantees of CB-DSL. To this end, we firstly make some definitions and assumptions for convergence analysis. Upon these preliminaries, the convergence behaviors of CB-DSL are evaluated and an upper bound on the convergence rate is derived.

A. Assumption and Definition

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

$$\|\nabla F_i(\mathbf{w}_{i,t+1}) - \nabla F_i(\mathbf{w}_{i,t})\| \leq L\|\mathbf{w}_{i,t+1} - \mathbf{w}_{i,t}\|, \forall i, t \quad (7)$$

where $L > 0$ is the Lipschitz constant [5].

The following definitions are made to facilitate analysis. Firstly, we rewrite $\mathbf{w}_{i,t}^p$ and $\mathbf{w}_t^g$ as

$$\mathbf{w}_{i,t}^p = \mathbf{w}_{i,t-1} - \mathbf{v}_{i,t}^p, \quad (8)$$

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

where $\mathbf{v}_{i,t}^p$ and $\mathbf{v}_t^g$ denote the optimal local speed and the optimal global speed at the i-th node in the $(t-1)$-th iteration.

[2]For the implementation point of view, the small amount of globally shared scoring dataset can be either pre-stored in the IoT devices or broadcasted from the PS to all the local workers.

Then the speed $\mathbf{v}_{i,t+1}$ of (5) can be rewritten as

$$\begin{aligned}\mathbf{v}_{i,t+1} &= c_0\mathbf{v}_{i,t} + c_1(-\mathbf{v}_{i,t}^p + \mathbf{v}_{i,t}) \\ &\quad + c_2(-\mathbf{v}_t^g + \mathbf{v}_{i,t}) + \alpha\nabla F_i(\mathbf{w}_{i,t}) \\ &= (c_0 + c_1 + c_2)\mathbf{v}_{i,t} - c_1\mathbf{v}_{i,t}^p \\ &\quad - c_2\mathbf{v}_t^g + \alpha\nabla F_i(\mathbf{w}_{i,t}). \end{aligned} \quad (10)$$

We use $\theta_{i,t}$, $\theta_{i,t}^p$, and θ_t^g to denote the angles between the vectors $\mathbf{v}_{i,t}$ and $\nabla F_i(\mathbf{w}_{i,t})$, $\mathbf{v}_{i,t}^p$ and $\nabla F_i(\mathbf{w}_{i,t})$, $\mathbf{v}_t^g$ and $\nabla F_i(\mathbf{w}_{i,t})$, for any i and t, respectively. Then we have

$$\cos\theta_{i,t} \triangleq \frac{\mathbf{v}_{i,t}\nabla F_i(\mathbf{w}_{i,t})^T}{\|\mathbf{v}_{i,t}\|\|\nabla F_i(\mathbf{w}_{i,t})\|}, \ \forall i, t, \quad (11)$$

$$\cos\theta_{i,t}^p \triangleq \frac{\mathbf{v}_{i,t}^p\nabla F_i(\mathbf{w}_{i,t})^T}{\|\mathbf{v}_{i,t}^p\|\|\nabla F_i(\mathbf{w}_{i,t})\|}, \ \forall i, t, \quad (12)$$

$$\cos\theta_t^g \triangleq \frac{\mathbf{v}_t^g\nabla F_i(\mathbf{w}_{i,t})^T}{\|\mathbf{v}_t^g\|\|\nabla F_i(\mathbf{w}_{i,t})\|}, \ \forall i, t, \quad (13)$$

where we assume

$$\underline{q} \leq \cos\theta_{i,t} \leq \overline{q}, \ \forall i, t \quad (14)$$

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

$$\underline{q}^g \leq \cos\theta_t^g \leq \overline{q}^g, \ \forall i, t. \quad (16)$$

Then we further assume

$$\underline{u} \leq \frac{\|\mathbf{v}_{i,t}\|}{\|\nabla F_i(\mathbf{w}_{i,t})\|} \leq \overline{u}, \ \forall i,t \tag{17}$$

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

$$\underline{u}^g \leq \frac{\|\mathbf{v}_t^g\|}{\|\nabla F_i(\mathbf{w}_{i,t})\|} \leq \overline{u}^g, \ \forall i,t. \tag{19}$$

B. Convergence

With the assumptions and definitions presented in subsection IV.A, the convergence bound of CB-DSL is given by the following **Theorem 1**.

Theorem 1. *For T rounds of communication, the expected convergence rate at each worker is bounded by*

$$\mathbb{E}\left[\sum_{t=1}^{T}\frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right] \leq \frac{F(\mathbf{w}_{i,0}) - F(\mathbf{w}^*)}{T\Phi_E}, \ \forall i, \tag{20}$$

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

Proof: Please refer to Appendix A. ■

The result of **Theorem 1** implies the following order-wise convergence rate

$$\mathbb{E}\left[\sum_{t=1}^{T}\frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right] \leq \mathcal{O}(\frac{1}{T\Phi_E}). \tag{21}$$

The inequality of (21) indicates that the convergence is guaranteed as the number of communication rounds 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, c_1, and c_2 are all set to 0, $\Phi_E = \alpha - 2L\alpha^2$ and CB-DSL degenerates into FedAvg. Thus, when $\Phi_E - (\alpha - 2L\alpha^2) = \frac{2c_0+\delta_{c_1}+\delta_{c_2}}{2}\underline{q}\underline{u} - \frac{\delta_{c_1}}{2}\overline{u}^p\overline{q}^p - \frac{\delta_{c_2}}{2}\overline{u}^g\overline{q}^g - 2L((c_0^2 + \delta_{c_1}c_0 + \delta_{c_2}c_0 + \frac{\delta_{c_1}^2}{3} + \frac{\delta_{c_2}^2}{3} + \frac{\delta_{c_1}\delta_{c_2}}{2})\overline{u}^2 + \frac{\delta_{c_1}^2}{3}(\overline{u}^p)^2 + \frac{\delta_{c_2}^2}{3}(\overline{u}^g)^2) > 0$, CB-DSL converges faster than FedAvg.

Since the datasets over different local workers are non-i.i.d., the learning performance varies with the degree of the dataset heterogeneity. Specifically, the greater the heterogeneity of datasets, the parameters over different local workers will become more diverse, e.g., larger range of the values of $\cos\theta_{i,t}$ among workers. That is, $\underline{q}$ becomes small and $\overline{q}$ becomes large. Intuitively, Φ_E becomes smaller as the heterogeneity of non-i.i.d. datasets increases, which will lead to a worse learning performance veiled by (20) and (21). We theoretically analyze the impact of data heterogeneity on the learning performance of CB-DSL in our journal version [20].

V. Simulation Result

In this section, we demonstrate that a globally shared dataset is beneficial to the learning performance of our CB-DSL.

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)

A. System and Dataset Setting

To evaluate the performance of our CB-DSL compared to the benchmark methods, we perform an empirical simulations by using a handwritten-digit classification task based on the well-known MNIST dataset that consists of 10 classes ranging from digit "0" to "9". In our training procedure, we set a total number of the local workers to be 50, as the IoT devices in an edge network. For the i.i.d. setting, 300 distinct training samples are randomly selected and distributed to each of the local workers as their local datasets, i.e., $K = 300$. The shared scoring dataset consists of 2000 data samples randomly selected from the population training dataset. In addition, we set the relevant parameters as $c_0 = 1$, $\delta_{c_1} = 1$, and $\delta_{c_2} = 1$.

B. Neural Network Setting

As shown in Table I, we use a five-layer CNN as the model architecture. During training process, we use SGD optimizer with learning rate $\alpha = 0.005$ and cross-entropy loss. The batch size is set to 10.

C. Benchmark Setting

We compare our CB-DSL with FedAvg under different settings, including: (1) *FedAvg without a shared dataset*: it is the standard FedAvg. (2) *CB-DSL without a shared dataset*: the local workers use their own scoring dataset to culculate $F_{i,t}^p$ in CB-DSL. (3) *CB-DSL with a shared dataset for local scoring*: the local workers use the shared scoring dataset $\mathfrak{D}_{sc}^G$ to culculate $F_{i,t}^p$ in CB-DSL.

D. Result

Fig. 1 shows the simulation results under the three different settings, respectively. As shown in Fig. 1, CB-DSL is superior than FedAvg under the same settings without any globally shared datasets. A shared scoring dataset can improve the learning performance for CB-DSL. This is because a shared scoring dataset can help the PS to select the global optimum more accurately than that local workers using their own scoring dataset which however makes the loss function $F(\cdot)$ only partially observable at local workers. Besides, the improvement on learning accuracy also indicates that by using

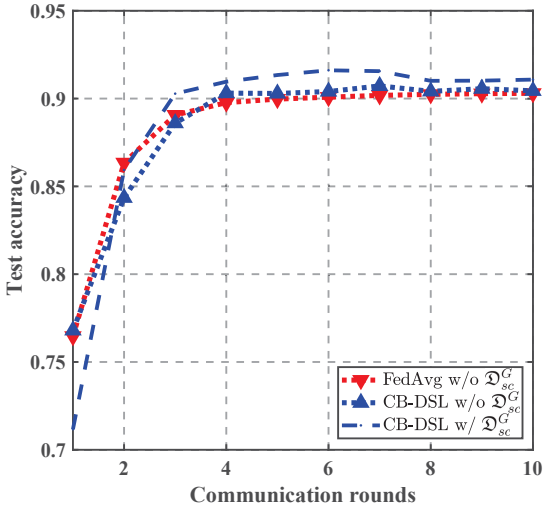

Fig. 1: The performance comparison varies with communication rounds.

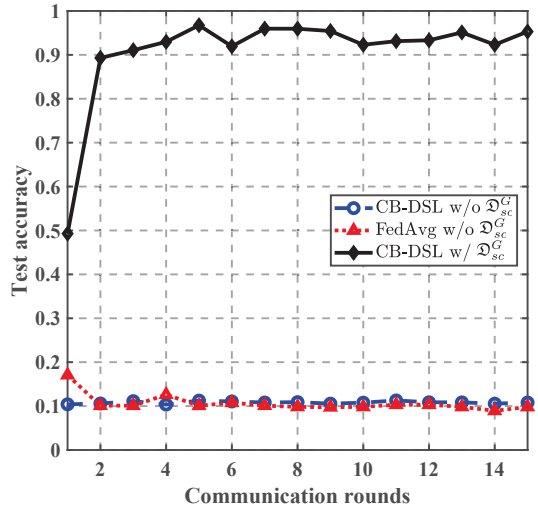

Fig. 2: The performance comparison with a Byzantine attacker under the i.i.d. setting.

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. 2, we provide the performance comparison in the presence of the Byzantine attack. It is obvious that even only one Byzantine attacker can fail FedAvg and CB-DSL without $\mathfrak{D}_{sc}^G$. On the other hand, the CB-DSL with $\mathfrak{D}_{sc}^G$ can effectively defend the Byzantine attack, because the globally shared dataset for scoring $\mathfrak{D}_{sc}^G$ can help identify and screen out the Byzantine attacker as explained in **Algorithm 1**.

In addition, since only one worker is called to send its model parameters to the PS in CB-DSL while all workers need to send their model parameters to the PS in FedAvg, the communication cost in CB-DSL is only $\frac{1}{U}$ of that in FedAvg, given the fact that the communication cost for the transmission of loss values is trivial and thus can be ignored. Our CB-DSL with $\mathfrak{D}_{sc}^G$ 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.

VI. Conclusion

This paper studies the holistic integration of FL and P-SO, named as DSL, which can save communication cost dramatically. However, the vanilla DSL becomes vulnerable to Byzantine attacks. Thus, we propose to use a shared dataset to achieve communication-efficient and Byzantine-resilient DSL (CB-DSL). We provide theoretical analysis of the convergence behavior of CB-DSL, which indicates that our proposed method can achieve better learning performance than FedAvg. Simulation results verify that our proposed solution can improve learning performance compared with the standard FedAvg. Meanwhile, the communication cost of CB-DSL is much reduced communication cost than standard FedAvg. Besides, CB-DSL can effectively defend Byzantine attacks.

Acknowledgments

This work was partly supported by the Science and Technology Innovation Project of Xiongan (Grants #2022XACX0400), the National Science Foundation of the US (Grants #1939553, #2003211,#2128596, #2136202 and #2231209), and the Virginia Research Investment Fund (Commonwealth Cyber Initiative Grant #223996).

Appendix A
Proof of **Theorem 1**

Proof: Because $F_i(\cdot)$ is L-smooth from Assumption 1, according to [21, Lemma 3.4], we have

$$\begin{aligned}
&F_i(\mathbf{w}_{i,t+1}) - F_i(\mathbf{w}_{i,t}) \le (\mathbf{w}_{i,t+1} - \mathbf{w}_{i,t})\nabla F_i(\mathbf{w}_{i,t})^T \\
&+ \frac{L}{2}\|\mathbf{w}_{i,t+1} - \mathbf{w}_{i,t}\|^2 = -\mathbf{v}_{i,t+1}\nabla F_i(\mathbf{w}_{i,t})^T + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2 \\
&= -(c_0 + c_1 + c_2)\mathbf{v}_{i,t}\nabla F_i(\mathbf{w}_{i,t})^T + c_1\mathbf{v}_{i,t}^p\nabla F_i(\mathbf{w}_{i,t})^T \\
&+ c_2\mathbf{v}_t^g\nabla F_i(\mathbf{w}_{i,t})^T - \alpha\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2. \quad (22)
\end{aligned}$$

According to the 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 (14)-(19), for any i and t, we have

$$\begin{aligned}
&\underline{u}\underline{q}\|\nabla F_i(\mathbf{w}_{i,t})\|^2 \le \mathbf{v}_{i,t}\nabla F_i(\mathbf{w}_{i,t})^T \\
&= \|\mathbf{v}_{i,t}\|\|\nabla F_i(\mathbf{w}_{i,t})\|\cos\theta_{i,t} \le \overline{u}\ \overline{q}\|\nabla F_i(\mathbf{w}_{i,t})\|^2, \quad (23) \\
&\underline{q}^p\underline{u}^p\|\nabla F_i(\mathbf{w}_{i,t})\|^2 \le \mathbf{v}_{i,t}^p\nabla F_i(\mathbf{w}_{i,t})^T \\
&= \|\mathbf{v}_{i,t}^p\|\|\nabla F_i(\mathbf{w}_{i,t})\|\cos\theta_{i,t}^p \le \overline{u}^p\overline{q}^p\|\nabla F_i(\mathbf{w}_{i,t})\|^2, \quad (24) \\
&\underline{q}^g\underline{u}^g\|\nabla F_i(\mathbf{w}_{i,t})\|^2 \le \mathbf{v}_t^g\nabla F_i(\mathbf{w}_{i,t})^T \\
&= \|\mathbf{v}_t^g\|\|\nabla F_i(\mathbf{w}_{i,t})\|\cos\theta_t^g \le \overline{u}^g\overline{q}^g\|\nabla F_i(\mathbf{w}_{i,t})\|^2. \quad (25)
\end{aligned}$$

Substituting (23)-(25) to (22), we have

$$F_i(\mathbf{w}_{i,t+1}) - F_i(\mathbf{w}_{i,t}) \leq -(c_0+c_1+c_2)\underline{qu}\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + c_1\overline{u}^p\overline{q}^p\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + c_2\overline{u}^g\overline{q}^g\|\nabla F_i(\mathbf{w}_{i,t})\|^2 + \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2 - \alpha\|\nabla F_i(\mathbf{w}_{i,t})\|^2 = \frac{L}{2}\|\mathbf{v}_{i,t+1}\|^2 + (c_1\overline{u}^p\overline{q}^p + c_2\overline{u}^g\overline{q}^g - (c_0+c_1+c_2)\underline{qu} - \alpha)\|\nabla F_i(\mathbf{w}_{i,t})\|^2. \quad (26)$$

Applying the triangle inequality of norms $\|\mathbf{X}+\mathbf{Y}\| \leq \|\mathbf{X}\|+\|\mathbf{Y}\|$, the submultiplicative property of norms $\|\mathbf{XY}\| \leq \|\mathbf{X}\|\|\mathbf{Y}\|$, and the Jensen's inequality, we have

$$\begin{aligned}\|\mathbf{v}_{i,t+1}\|^2 &= \|(c_0+c_1+c_2)\mathbf{v}_{i,t} - c_1\mathbf{v}_{i,t}^p \\ &\quad - c_2\mathbf{v}_t^g + \alpha\nabla F_i(\mathbf{w}_{i,t})\|^2 \\ &\leq (\|(c_0+c_1+c_2)\mathbf{v}_{i,t}\| + \|c_1\mathbf{v}_{i,t}^p\| \\ &\quad + \|c_2\mathbf{v}_t^g\| + \|\alpha\nabla F_i(\mathbf{w}_{i,t})\|)^2 \\ &\leq 4((c_0+c_1+c_2)^2\|\mathbf{v}_{i,t}\|^2 + c_1^2\|\mathbf{v}_{i,t}^p\|^2 + c_2^2\|\mathbf{v}_t^g\|^2 \\ &\quad + \alpha^2\|\nabla F_i(\mathbf{w}_{i,t})\|^2). \end{aligned} \quad (27)$$

According to the assumptions of $\overline{u}$, $\overline{u}^p$, $\overline{u}^g$ in (17)-(19), for any i and t, we have

$$\|\mathbf{v}_{i,t}\| \leq \overline{u}\|\nabla F_i(\mathbf{w}_{i,t})\|, \quad (28)$$

$$\|\mathbf{v}_{i,t}^p\| \leq \overline{u}^p\|\nabla F_i(\mathbf{w}_{i,t})\|, \quad (29)$$

$$\|\mathbf{v}_t^g\| \leq \overline{u}^g\|\nabla F_i(\mathbf{w}_{i,t})\|. \quad (30)$$

Substituting (28)-(30) to (27), we have

$$\begin{aligned}\|\mathbf{v}_{i,t+1}\|^2 \leq& 4((c_0\overline{u}+c_1\overline{u}+c_2\overline{u})^2 + c_1^2(\overline{u}^p)^2 \\ &+ c_2^2(\overline{u}^g)^2 + \alpha^2)\|\nabla F_i(\mathbf{w}_{i,t})\|^2. \end{aligned} \quad (31)$$

Substituting (31) to (26), we have

$$F_i(\mathbf{w}_{i,t+1}) - F_i(\mathbf{w}_{i,t}) \leq \Phi\|\nabla F_i(\mathbf{w}_{i,t})\|^2, \quad (32)$$

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

Now extend the expectation over randomness in the trajectory, and perform a telescoping sum of (32) over the T iterations:

$$\begin{aligned}F(\mathbf{w}_{i,0}) - F(\mathbf{w}^*) &\geq F(\mathbf{w}_{i,0}) - \mathbb{E}[F(\mathbf{w}_{i,T})] \\ &= \mathbb{E}\left[\sum_{t=1}^{T}(F(\mathbf{w}_{i,t-1}) - F(\mathbf{w}_{i,t}))\right] \\ &\geq \mathbb{E}\left[\sum_{t=1}^{T}\Phi_E\|\nabla F_i(\mathbf{w}_{i,t})\|^2\right], \end{aligned} \quad (33)$$

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

We can rearrange this inequality to yield the rate:

$$\mathbb{E}\left[\sum_{t=1}^{T}\frac{\|\nabla F_i(\mathbf{w}_{i,t})\|^2}{T}\right] \leq \frac{F(\mathbf{w}_{i,0}) - F(\mathbf{w}^*)}{T\Phi_E}. \quad (34)$$

■

REFERENCES

[1] M. Chen, D. Gündüz, K. Huang, W. Saad, M. Bennis, A. V. Feljan, and H. V. Poor, "Distributed learning in wireless networks: Recent progress and future challenges," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3579–3605, 2021.

[2] 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.

[3] Y. Wang, Z. Tian, X. Fan, Y. Huo, C. Nowzari, and K. Zeng, "Distributed swarm learning for internet of things at the edge: Where artificial intelligence meets biological intelligence," *arXiv preprint arXiv:2210.16705*, 2022.

[4] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "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.

[5] 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.

[6] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "1-bit compressive sensing for efficient federated learning over the air," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2022.

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

[8] 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.

[9] G. Zhou, P. Xu, Y. Wang, and Z. Tian, "Robust distributed learning against both distributional shifts and byzantine attacks," *arXiv preprint arXiv:2210.16682*, 2022.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "Joint optimization of communications and federated learning over the air," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 4434–4449, 2022.

[19] 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.

[20] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "Cb-dsl: Communication-efficient and byzantine-robust distributed swarm learning on non-iid data," *arXiv preprint arXiv:2208.05578*, 2022.

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