This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

Device scheduling for secure aggregation in wireless federated learning

Na Yan, Kezhi Wang, Kangda Zhi, Cunhua Pan, Kok Keong Chai,
and H. Vincent Poor, *Life Fellow, IEEE*

***Abstract*—Federated learning (FL) has been widely investigated in academic and industrial fields to resolve the issue of data isolation in the distributed Internet of Things (IoT) while maintaining privacy. However, challenges persist in ensuring adequate privacy and security during the aggregation process. In this paper, we investigate device scheduling strategies that ensure the security and privacy of wireless FL. Specifically, we measure the privacy leakage of user data using differential privacy (DP) and assess the security level of the system through mean square error security (MSE-security). We commence by deriving analytical results that reveal the impact of device scheduling on privacy and security protection, as well as on the learning process. Drawing from these analytical findings, we propose three scheduling policies that can achieve secure aggregation of wireless FL under different cases of channel noise. In particular, we formulate an integer nonlinear fractional programming problem to improve the learning performance while guaranteeing privacy and security of wireless FL. We provide an insightful solution in the closed form to the optimization problem when the model has a high dimension. For the general case, we propose a secure and private aggregation (SPA) algorithm based on the branch-and-bound (BnB) method, which can obtain the optimal solution with low complexity. The effectiveness of the proposed schemes for device selection is validated through simulations.**

***Index Terms*—Federated learning (FL), device scheduling, branch-and-bound (BnB), integer nonlinear fractional programming.**

I. Introduction

With the rapid development of distributed Internet of Things (IoT), considerable volumes of data have been produced at the edge of networks. By employing specific algorithms, machine learning (ML) can reveal concealed patterns within vast datasets through comprehensive data analysis. However, the limited local data at a signal IoT device results in poor ML performance, which prompts the introduction of federated learning (FL) [1] to solve data islands and achieve efficient resource utilization. It enables edge devices to train a model locally with the help of a central server, such as a base station (BS). Specifically, the edge devices begin by downloading the latest global model from the BS. Subsequently, they calculate gradients or locally update the model using their datasets. Following this, the computed gradients or updated models are transmitted to the BS, which then updates the global model. This process continues until convergence. By training models locally, FL not only fully utilizes the computational resources of the edge devices but also efficiently reduces power consumption, latency, and privacy risks in comparison to uploading extensive raw data to the BS in centralized ML approaches. Despite these benefits of FL on edge data processing, its practical deployment over wireless networks, often referred to as wireless FL [2], [3], faces significant challenges regarding privacy and security issues. Although security and privacy terms are often used interchangeably in the existing literature, it is essential to underscore their distinction. In the context of FL, privacy specifically refers to safeguarding individual data throughout the collaborative training process, with a focus on protecting sensitive information stored on local devices. On the other hand, security in FL is a more comprehensive concept, encompassing the overall robustness of the FL system. It includes safeguarding against various threats, such as model poisoning attacks, data injection attacks, and communication security.

It has been demonstrated that keeping the training data on edge devices fails to provide sufficient privacy protection when some attacks [4]–[6] are applied on the uploaded local updates, leading to the potential leakage of information about the training data [7]. To address these issues, several privacy-preserving aggregation protocols based on secure multiparty computation (SMC) [8], differential privacy (DP) [9]–[12] and functional encryption [13] were proposed aiming to prevent the aggregator from analyzing the local updates. The work [14] proposed a double-masking protocol to guarantee the confidentiality of users' local gradients where the central server was required to provide the "Proof" about the correctness of its aggregated results to each edge device. However, these approaches in [8], [13], [14] rely on a trusted third party. Besides, the substantial computational costs result in the inefficiency of these methods presented in [15], [16].

In addition, the unreliability of wireless communication and the malicious behaviors of third parties can lead to incorrect updates. These challenges have the potential to yield adverse effects on the FL process, and in some instances, they can even result in the divergence of the FL process [17].

Na Yan, Kangda Zhi and Kok Keong Chai are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, E1 4NS, U.K. (e-mail: n.yan, k.zhi, michael.chai@qmul.ac.uk). Kezhi Wang is with Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, U.K. (email: kezhi.wang@brunel.ac.uk). Cunhua Pan is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China. (email: cpan@seu.edu.cn). H. Vincent Poor is with the Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ 08544, USA. (email: poor@princeton.edu)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

The authors of [18], [19] adopted a covert communication (CC) technique with which a friendly jammer transmits jamming signals to prevent an eavesdropper from detecting the update transmission of the local model from mobile devices in FL. The work of [20] utilized power control to improve the security of FL in the internet of drones (IoD) networks where security rate was employed to measure the security of wireless communications. Nonetheless, limited attention has been devoted to investigating the impact of device scheduling on the security and privacy of wireless FL. Given the diverse communication conditions among the devices, scheduling decisions [21] play a pivotal role not only in ensuring the quality of the aggregated gradient at the BS but also in reducing the risk of privacy and security, and therefore deserves deep investigation.

This paper addresses this research gap by delving into various device scheduling schemes across different levels of channel noise, addressing concerns related to privacy leakage and eavesdropping, which often serve as the initial vulnerability for malicious third parties to launch security attacks. Particularly, in the case that channel noise is not sufficient to provide protection of privacy and security, the BS schedules some of the devices to send artificial noise to aid channel noise against attacks on privacy and security. Our main contributions are summarized as follows.

- We quantify the privacy leakage and the security level by conducting theoretical analyses using DP and mean square error security (MSE-security) [22], respectively. The results reveal the impact of device scheduling on the protection of privacy and security.
- Building upon the insights derived from the analytical results, we propose scheduling policies to ensure user privacy and system security in three cases: (1) channel noise is sufficient for protecting privacy and security with all device participation; (2) channel noise is sufficient for protecting privacy and security with partial device participation; (3) channel noise is insufficient for protecting privacy and security with any device participation.
- We formulate an integer nonlinear fractional optimization problem in the case of insufficient channel noise. For the special case where the model is high-dimensional, a closed-form solution is obtained and useful insights are drawn. In the general case, a branch-and-bound (BnB) based algorithm is proposed, which addresses this problem with low computational complexity.

II. System Model and Preliminaries

We consider an over-the-air FL (OTA-FL) system where N edge devices, denoted by $\mathcal{N} = \{1, 2, ..., N\}$, collaboratively train a model with the help of a BS. The devices and the BS communicate through a shared multiple access channel (MAC) where all devices transmit their gradients simultaneously. Specifically, the BS is assumed to be "honest but curious" and may attempt to learn the personal information from the received gradients, which is regarded as a privacy threat. Additionally, an eavesdropper (Eve) in the system tries to eavesdrop up the gradients, which is regarded as a security issue. In this paper, we employ channel noise and artificial noise as protection to prevent privacy leakage and eavesdropping. In the case that channel noise is insufficient for security and privacy protection, we assume that the BS schedules some of the devices as participants involved in the training, and some of the remaining devices to send artificial noise to enhance system security and protect privacy.

A. Wireless FL

Assume that each device of index $n \in \mathcal{N}$ has a local dataset $\mathcal{D}_n$ which contains D_n pairs of training samples $(\boldsymbol{u}, v)$ where $\boldsymbol{u}$ is the raw data and v is the corresponding label. For simplicity, we assume that $D_1 = \cdots = D_N$. The purpose of the FL task is to obtain a model parameter that minimizes the loss function, i.e.,

$$\min_{\boldsymbol{m}} L(\boldsymbol{m}) = \frac{1}{N} \sum_{n=1}^{N} L_n(\boldsymbol{m}), \tag{1}$$

where $\boldsymbol{m} \in \mathbb{R}^d$ is the model parameter to be optimized. $L_n(\boldsymbol{m})$ denotes the objective function of device n and is defined by

$$L_n(\boldsymbol{m}) = \frac{1}{D_n} \sum_{(\boldsymbol{u},v)\in\mathcal{D}_n} l(\boldsymbol{m}; (\boldsymbol{u}, v)), \tag{2}$$

where $l(\boldsymbol{m}; (\boldsymbol{u}, v))$ is an empirical loss function defined by the learning task, quantifying the loss of $\boldsymbol{m}$ at sample $(\boldsymbol{u}, v)$.

To solve problem (1), an iterative approach based on stochastic gradient descent (SGD) is typically applied. The main procedure of basic SGD applied in FL is given as follows:

- **Step 1**: *Parameter broadcasting*: At the beginning of round t, the BS first broadcasts the latest global model parameter $\boldsymbol{m}^t$ to the scheduled devices [1].
- **Step 2**: *Local training*: (1) Each participant performs the initialization of the local model by setting the received global model parameter as the local model parameter, i.e., $\boldsymbol{m}_n^t = \boldsymbol{m}^t$. (2) Each device randomly selects a batch of data $\mathcal{B}_n$ of size B_n from $\mathcal{D}_n$ and computes the stochastic gradient based on $\mathcal{B}_n$. More specifically, the stochastic gradient is given by

$$\boldsymbol{g}_n^t \triangleq \nabla L_n\left(\boldsymbol{m}_n^t; \mathcal{B}_n\right) = \frac{1}{B_n} \sum_{(\boldsymbol{u},v)\in\mathcal{B}_n} \nabla l\left(\boldsymbol{m}_n^t; (\boldsymbol{u}, v)\right). \tag{3}$$

  By contrast, the full gradient based on $\mathcal{D}_n$ is given by

$$\nabla L_n\left(\boldsymbol{m}_n^t\right) = \frac{1}{D_n} \sum_{(\boldsymbol{u},v)\in\mathcal{D}_n} \nabla l\left(\boldsymbol{m}_n^t; (\boldsymbol{u}, v)\right). \tag{4}$$

- **Step 3**: *Gradient aggregation*[2]: (1) Denote the set of training participants at round t by $\mathcal{U}^t \subseteq \mathcal{N}$ and the set

[1] In the first iteration, the global model is randomly initialized at the BS.

[2] We simplify the theoretical analysis by considering a basic approach involving aggregation after a single local training round. The proposed schemes are also applicable to the federated averaging algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

of the devices scheduled to send artificial noise at round t by $\mathcal{M}^t \subseteq \mathcal{N}/\mathcal{U}^t$. Assuming that the gradient $\|\boldsymbol{g}_n^t\|_2$ is bounded by G, the signal transmitted from device n is given by

$$\boldsymbol{x}_n^t = \frac{\sqrt{P_n}}{G}\boldsymbol{g}_n^t, n \in \mathcal{U}^t, \text{ and} \tag{5}$$

$$\boldsymbol{x}_n^t = \sqrt{\frac{P_n}{d}}\boldsymbol{e}_n^t, n \in \mathcal{M}^t, \tag{6}$$

where P_n is the maximum transmission power of device n and $\boldsymbol{e}_n^t \sim \mathcal{N}(0, \mathbf{I}_d)$ is artificial Gaussian noise. Assuming that the channel gain coefficient between device n and the BS is $h_{n,B}^t$, the receiver signal at the BS can be expressed as

$$\boldsymbol{y}^t = \sum_{n\in\mathcal{U}^t} \frac{h_{n,B}^t\sqrt{P_n}}{G}\boldsymbol{g}_n^t + \sum_{n\in\mathcal{M}^t} h_{n,B}^t\sqrt{\frac{P_n}{d}}\boldsymbol{e}_n^t + \boldsymbol{r}_B^t, \tag{7}$$

where $\boldsymbol{r}_B^t \sim \mathcal{N}(0, \sigma_B\mathbf{I}_d)$ is the received noise at the BS. In order to incorporate the impact of the diverse channel conditions of devices on the learning process, the BS performs the post-processing using the sum of the channel conditions of the participants[3], i.e.,

$$\tilde{\boldsymbol{g}}^t = \frac{G}{\sum_{n\in\mathcal{U}^t} h_{n,B}^t\sqrt{P_n}}\boldsymbol{y}^t. \tag{8}$$

- **Step 4**: *Model update*: The BS leverages the noisy gradient estimate, incorporating noise to prevent inference of sensitive information from the recovered gradient, to perform the global model update, i.e.,

$$\boldsymbol{m}^{t+1} = \boldsymbol{m}^t - \tau^t\tilde{\boldsymbol{g}}^t, \tag{9}$$

where τ^t is the learning rate (also termed the step size in SGD).

The above iteration steps are repeated until a certain training termination condition is met.

In order to formally quantify the privacy leakage and the security level of the system, we introduce the DP and MSE-security concepts in the following.

B. DP and MSE-Security

DP [23] is defined on the conception of the adjacent dataset, which guarantees that the contribution of any individual data sample to the model remains statistically indistinguishable. More specifically, DP quantifies information leakage in FL by measuring the sensitivity of the disclosed statistics (i.e., the gradients) to the change of a single data point in the input dataset. The basic definition of (ϵ, ζ)-DP is given as follows.

Definition 1. *(ϵ, ζ)-DP [23]: A randomized mechanism $\mathcal{O}$ guarantees (ϵ, ζ)-DP if for two adjacent datasets $\mathcal{D}, \mathcal{D}'$ differing in one sample, and measurable output space $\mathcal{Q}$ of $\mathcal{O}$, it satisfies,*

$$\Pr[\mathcal{O}(\mathcal{D}) \in \mathcal{Q}] \leqslant e^{\epsilon}\Pr[\mathcal{O}(\mathcal{D}') \in \mathcal{Q}] + \zeta. \tag{10}$$

The additive term ζ allows for breaching ϵ-DP with probability ζ while ϵ denotes the protection level and a smaller ϵ means a higher privacy level. DP can be achieved by adding random noise to data. Specifically, the Gaussian DP mechanism which guarantees privacy by adding artificial Gaussian noise is introduced as follows.

Definition 2. *Gaussian mechanism [23]: A mechanism $\mathcal{O}$ is called as a Gaussian mechanism if it alters the output of another algorithm $\mathcal{L} : \mathcal{D} \to \mathcal{Q}$ by adding Gaussian noise, i.e.,*

$$\mathcal{O}(\mathcal{D}) = \mathcal{L}(\mathcal{D}) + \mathcal{N}\left(0, \sigma^2\mathbf{I}_d\right). \tag{11}$$

A Gaussian mechanism $\mathcal{O}$ guarantees (ϵ, ζ)-DP with

$$\epsilon = \frac{\Delta S}{\sigma}\sqrt{2\ln\left(\frac{1.25}{\zeta}\right)}, \tag{12}$$

where $\Delta S \triangleq \max_{\mathcal{D},\mathcal{D}'} \|\mathcal{L}(\mathcal{D}) - \mathcal{L}(\mathcal{D}')\|_2$ stands for the sensitivity of the algorithm $\mathcal{L}$ signifying the extent to which the algorithm's output varies when a single data point is altered.

According to the Gaussian mechanism described above, privacy leakage depends both on the sensitivity of the algorithm $\mathcal{L}$ and on the power of the added Gaussian noise.

MSE-security was proposed in [22] to measure the security of analog messages and is introduced as follows.

Definition 3. *$(\mathcal{E}, \phi)$-MSE-security [22]: A uniform distributed mechanism $\mathcal{E} : \mathcal{G} \to \mathcal{Y}$, where $\mathcal{Y}$ is a measureable and bounded output space, guarantees $(\mathcal{E}, \phi)$-MSE-security if under a uniform distribution of $\mathcal{E}\left(\{\boldsymbol{g}_n^t\}_{n\in\mathcal{U}^t}\right)$, for any Eve's estimator $e : \mathcal{Z} \to \mathcal{Y}$, there is a real number $\phi \geqslant 0$ satisfies $\mathbb{E}\left[\left(e(\boldsymbol{z}^t) - \mathcal{E}\left(\{\boldsymbol{g}_n^t\}_{n\in\mathcal{U}^t}\right)\right)^2\right] \geqslant \phi$.*

In statistical terms, a scheme guaranteeing $(\mathcal{E}, \phi)$-MSE-security means that all estimators that the eavesdropper can apply have MSE at least ϕ.

III. Device scheduling for secure aggregation

In this section, we conduct theoretical analyses and propose device scheduling policies for guaranteeing the privacy and security of OTA-FL. For ease of presentation, we define $p_{n,B}^t = h_{n,B}^t\sqrt{P_n}$ in the rest of this paper.

A. Privacy, Security and Convergence Analysis

We first conduct theoretical analyses to demonstrate the impact of device scheduling on privacy and security protection, as well as its influence on learning performance.

[3]If the goal is to explore the influence of dataset size on aggregation within an unbalanced dataset scenario, a similar post-processing mechanism can be employed. However, for the sake of simplicity and to narrow our focus to understanding the influence of channel conditions, we opt to use identical dataset sizes in this paper.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

1) Assumptions: For analysis, we provide the following assumptions first[4].

Assumption 1. *Assumptions on gradients: (1) The stochastic gradient is an unbiased estimate of the full gradient at that device, i.e.,* $\mathbb{E}\left[\boldsymbol{g}_n^t\right] = \nabla L_n\left(\boldsymbol{m}_n^t\right)$. *(2) The variance of the stochastic gradients at each device is bounded:* $\mathbb{E}\left[\left\|\boldsymbol{g}_n^t - \nabla L_n\left(\boldsymbol{m}_n^t\right)\right\|_2^2\right] \leqslant \vartheta^2$, *where the bound does not depend on* n *or* t. *(3) The expected squared norm of the stochastic gradients at each device is bounded:* $\mathbb{E}\left[\left\|\boldsymbol{g}_n^t\right\|_2\right] \leqslant G$, *which can be gauranteened by gradient clipping [26], [27].*

Assumption 2. *For each* n, $L_n(\cdot)$ *is* θ*-smooth, where* θ *does not depend on* n, *i.e.,* $L_n\left(\boldsymbol{\iota}'\right) - L_n\left(\boldsymbol{\iota}\right) \leqslant \left(\boldsymbol{\iota}' - \boldsymbol{\iota}\right)^{\mathrm{T}} \nabla L_n\left(\boldsymbol{\iota}\right) + \frac{\theta}{2}\left\|\boldsymbol{\iota}' - \boldsymbol{\iota}\right\|_2^2$.

Assumption 3. *For each* n, $L_n(\cdot)$ *is* ρ*-strongly convex, where* ρ *does not depend on* n, *i.e.,* $L_n\left(\boldsymbol{\iota}'\right) - L_n\left(\boldsymbol{\iota}\right) \geqslant \left(\boldsymbol{\iota}' - \boldsymbol{\iota}\right)^{\mathrm{T}} \nabla L_n\left(\boldsymbol{\iota}\right) + \frac{\rho}{2}\left\|\boldsymbol{\iota}' - \boldsymbol{\iota}\right\|_2^2$.

2) Privacy analysis: Even though the gradients are already aggregated before reaching the BS in OTA-FL, where the BS cannot access the individual gradients and models in common scenarios, there still exists a potential for privacy leakage in OTA-FL in some specific scenarios. Firstly, in those training rounds where only one device is scheduled to upload its gradient [28], the individual gradient cannot be hidden and protected by over-the-air aggregation. Furthermore, privacy can be compromised in specific circumstances where the gradients from other devices remain fixed while only the gradient from a particular device is updated, as considered in most differentially private OTA-FL studies [10]–[12], [29]. This scenario creates ideal conditions for malicious attackers to intercept the information. Therefore, the analysis results obtained under this scenario establish an upper bound of privacy leakage, ensuring stronger privacy protection. Based on the assumptions, we next present the privacy analysis.

Lemma 1. *OTA-FL achieves* (ϵ_n^t, ζ)*-DP for device* $n, n \in \mathcal{U}^t$, *where*

$$\epsilon_n^t = \frac{2\kappa p_{n,B}^t}{\sqrt{\sigma_{B,Tot}^t}}, \text{ with } \sigma_{B,Tot}^t = \sum_{n\in\mathcal{M}^t} \frac{p_{n,B}^t}{\sqrt{d}} + \sigma_B. \quad (13)$$

Proof: : Please refer to Appendix A. ■

Lemma 1 reveals that devices with better channel quality are more prone to privacy disclosure. Therefore, for reducing the privacy leakage in the system, one can either increase the power of noise or select devices with smaller channel condition coefficients $p_{n,B}^t$ to participate in training.

3) Security analysis: The received signal at the eavesdropper is given by

$$\boldsymbol{z}^t = \sum_{n\in\mathcal{U}^t} \frac{h_{n,E}^t\sqrt{P_n}}{G}\boldsymbol{g}_n^t + \sum_{n\in\mathcal{M}^t} h_{n,E}^t\sqrt{\frac{P_n}{d}}\boldsymbol{e}_n^t + \boldsymbol{r}_E^t, \quad (14)$$

[4]These assumptions are necessary for theoretical analysis and are widely adopted in [10], [12], [24], [25]

where $h_{n,E}^t$ is the channel gain coefficient between device n and the eavesdropper, and $\boldsymbol{r}_E^t \sim \mathcal{N}\left(0, \sigma_E\mathbf{I}_d\right)$ is the received noise at the eavesdropper.

Assume that the goal of the eavesdropper is to recover an averaging estimate of the gradients, denoted by $\boldsymbol{g}_{ave}^t = \frac{1}{|\mathcal{U}^t|}\sum_{n\in\mathcal{U}^t}\boldsymbol{g}_n^t$. By defining $p_{n,E}^t = h_{n,E}^t\sqrt{P_n}$ and $\Lambda^t = \max_{n\in\mathcal{U}^t}\left\{p_{n,B}^t\right\}$, the security analysis is given as follows.

Lemma 2. *Assume that the elements of* $\boldsymbol{g}_n^t$ *are distributed uniformly in [a, b]. The aggregation mechanism* $\mathcal{E}^t : \left(\boldsymbol{g}_n^t\right)_{n\in\mathcal{U}^t} \to \boldsymbol{z}^t \in \mathcal{Z}$ *guarantees* $\left(\mathcal{E}^t, \gamma_E^t\Xi\left(\frac{b-a}{\sqrt{\gamma_E^t}}\right)\right)$*-MSE-security. Specifically,*

$$\gamma_E^t = \frac{G^2}{|\mathcal{U}^t|\left(\Lambda^t\right)^2}\left(\sum_{n\in\mathcal{M}^t}\frac{\left(p_{n,E}^t\right)^2}{d} + \sigma_E\right), \quad (15)$$

where

$$\Xi(t) = \int_0^t\int_{-\infty}^{+\infty}\left(v + \frac{\varphi_N(-v) - \varphi_N(t-v)}{\Phi_N(t-v) - \Phi_N(-v)} - u\right)^2 \cdot \frac{1}{t}\varphi_N(u-v)\,dvdu \quad (16)$$

with $\varphi_N(\cdot)$ *and* $\Phi_N(\cdot)$ *denote the probability density function and the cumulative distribution function of the standard normal distribution, respectively.*

Proof: Please refer to Appendix B. ■

As $\gamma_E^t\Xi\left(\frac{b-a}{\sqrt{\gamma_E^t}}\right)$ increases with γ_E^t [22], a larger γ_E^t means greater system security. We use γ_E^t to indicate the security level of the system, referred to as the security coefficient. Upon observing (15), it is evident that in order to attain a larger security coefficient, one can either increase the aggregated noise at Eve or choose devices that contribute to a smaller Λ^t.

4) Convergence analysis: To illustrate the impact of device scheduling on the learning process, we present the convergence analysis in the following.

Theorem 1. *Assume that* $L_n\left(\boldsymbol{m}^*\right) - L_n\left(\boldsymbol{m}_n^*\right) \leq \Gamma$ *for all* n *and* $\frac{1}{\varrho} \leqslant \tau^t \leqslant \frac{1}{\theta}$ *with* ϱ *a constant. The gap between* $\boldsymbol{m}^{t+1}$ *and* $\boldsymbol{m}^*$ *is given by*

$$\mathbb{E}\left[\left\|\boldsymbol{m}^{t+1} - \boldsymbol{m}^*\right\|_2^2\right] \leqslant \left(1 - \rho\tau^t\right)\mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right] + \left(\tau^t\right)^2\left(2\varrho\Gamma + \vartheta^2 + G^2\Psi^t\right), \quad (17)$$

where

$$\Psi^t = \frac{N\sum_{n\in\mathcal{M}^t}\left(p_{n,B}^t\right)^2 + d\sigma_B}{\left(\sum_{n\in\mathcal{U}^t}p_{n,B}^t\right)^2}, \quad (18)$$

which characterizes the impact of the device scheduling in training round t.

Proof: Please refer to Appendix C. ■

According to Lemma 1 and Lemma 2, a larger power of noise and a lower power of gradient contribute to security

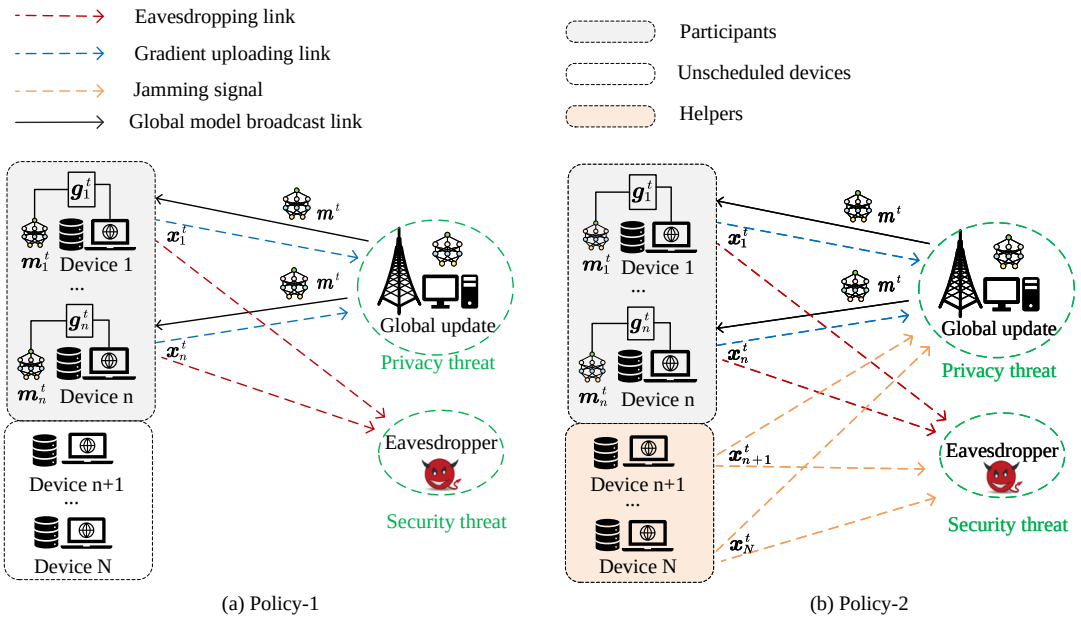


Fig. 1: Illustration of different scheduling policies.

and privacy protection. However, these conditions also result in a larger optimality gap, as shown in (17), thereby negatively impacting the learning process. The scale of the noise and the power of the uploaded gradients depend on the scheduling of the devices. Therefore, device scheduling is crucial for secure and private OTA-FL.

B. Device Scheduling Policies

In this subsection, we propose different device scheduling strategies to guarantee privacy and security. Assume that the privacy constraint of each device is ϵ and Υ is the required security coefficient. We omit the index t and define $p_M = \max_n \{p_{n,B}\}$ and $p_m = \min_n \{p_{n,B}\}$ for ease of presentation. Following Lemma 1 and Lemma 2, devices with poor channel conditions, i.e., smaller $p_{n,B}$ and $p_{n,E}$, have less risk at privacy leakage and security attack. Therefore, if the channel condition coefficients of all the devices in this system are lower than a certain critical point, the channel noise at the BS and Eve are enough to prevent privacy leakage and security attacks. Specifically, we can obtain the critical point $\hat{p} = \min\left\{\frac{\epsilon\sqrt{\sigma_B}}{2\kappa}, \frac{G\sqrt{\sigma_E}}{N\sqrt{\Upsilon}}\right\}$ by solving $\frac{2\hat{p}\kappa}{\sqrt{\sigma_B}} \leqslant \epsilon$ and $\frac{G\sqrt{\sigma_E}}{N\hat{p}} \geqslant \sqrt{\Upsilon}$ (according to Lemma 1 and Lemma 2) where only the channel noise is considered for guaranteeing privacy and security. We replaced the $|\mathcal{U}^t|$ in (15) with N for simplicity, which offers a stronger security guarantee. Then, we propose the following policies in different cases of channel noise.

1) Sufficient channel noise for all device participation: In the case that $p_M \leqslant \hat{p}$, the received noise at the BS and Eve is sufficient for the device with the best channel condition to participate in training while satisfying the privacy and security constraints. Therefore, all the devices should be selected as participants to achieve better learning performance without the need for any devices to transmit artificial noise.

2) Sufficient channel noise for partial device participation: In the case that $p_m \leqslant \hat{p} \leqslant p_M$, if no device is selected as a helper that sends Gaussian artificial noise to increase the power of the aggregated noise, the received noise at the BS and Eve can only provide qualified privacy and security protection when the devices satisfying $p_n \leqslant \hat{p}$ are selected to participate in the training process. In such cases, there are two scheduling schemes to realize secure aggregation.

- Policy-1: Select those devices with $p_n \leqslant \hat{p}$ as participants to engage in the training, and other devices will be absent in this round, as shown in Fig. 1 (a).
- Policy-2: Select some devices as participants and others are selected as helpers to send artificial noise, as shown in Fig. 1 (b).

3) Insufficient channel noise for any device participation: In the case that $\hat{p} \leqslant p_m$, the channel noise at the BS and Eve cannot guarantee qualified privacy and security for any device as a participant if no extra artificial noise is added. Therefore, some devices need to be selected as helpers to send artificial noise to degrade the SNR of the eavesdropper. Consequently, Policy-1 will no longer be applicable, and the only solution is Policy-2.

Then, how to choose devices that can ensure privacy and security while having a minimal negative impact on learning performance is a tradeoff problem.

C. Optimized Device Scheduling for the Insufficient Channel Noise Case

We formulate an optimization problem aiming to minimize the impact of device scheduling on the training with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

the consideration of privacy and security constraints. For tractability, we consider that a device is either selected as a participant or as a helper to send artificial noise. We introduce a vector $\boldsymbol{a} = [a_1, ...a_N]$ to denote the role of devices in each round. Specifically, $a_n = 1$ indicates that device n is selected as a participant, otherwise, device n plays the role of a helper. Then, the optimization problem can be formulated as follows:

$$\mathbf{P1}. \quad \min_{\boldsymbol{a}} \ \Psi = \frac{N\sum_{n=1}^{N}(1-a_n)\,p_{n,B}^2 + d\sigma_B}{\left(\sum_{n=1}^{N} a_n p_{n,B}\right)^2} \tag{19}$$

$$\textbf{s.t.} \quad a_n \in \{0,1\}, \forall n \in \mathcal{N}, \tag{19a}$$

$$\frac{2\kappa a_n p_{n,B}}{\sqrt{\sum_{n=1}^{N}\frac{(1-a_n)p_{n,B}^2}{d} + \sigma_B}} \leqslant \epsilon, \forall n \in \mathcal{N}, \tag{19b}$$

$$\frac{G^2\left(\sum_{n=1}^{N}\frac{(1-a_n)p_{n,E}^2}{d} + \sigma_E\right)}{\left(\sum_{n=1}^{N} a_n\right)^2 \max_n\left\{a_n p_{n,B}^2\right\}} \geqslant \Upsilon. \tag{19c}$$

The objective of this problem is to minimize the adverse effects of noise on the learning process while ensuring privacy and security. Constraint (19b) ensures the desired privacy level and constraint (19c) ensures the desired security level. To help understand the properties of this optimization, we first consider a special but useful case with high-dimensional learning models.

1) Closed-form solution for high-dimensional models: In practical scenarios, the learning model typically is high-dimensional in order to guarantee the prediction accuracy. In this case, we are able to simplify the optimization problem and therefore propose closed-form optimal solutions, which can provide useful insights for practical FL systems. Specifically, assuming that $d \to \infty$, problem $\mathbf{P1}$ can be recast as,

$$\mathbf{P}2. \quad \max_{\boldsymbol{a}} \ \sum_{n=1}^{N} a_n p_{n,B} \tag{20}$$

$$\mathbf{s.t.} \quad a_n \in \{0,1\}, \forall n \in \mathcal{N}, \tag{20a}$$

$$2\kappa a_n p_{n,B} \leqslant \epsilon\sqrt{\sigma_B}, \forall n \in \mathcal{N}, \tag{20b}$$

$$\left(\sum_{n=1}^{N} a_n\right)\max_n\{a_n p_{n,B}\} \leq G\sqrt{\frac{\sigma_E}{\Upsilon}}. \tag{20c}$$

Assume that the elements in $\boldsymbol{p}_B = [p_{1,B}, ..., p_{n,B}, ..., p_{N,B}]$ are sorted in descending order. Then, we have the following result.

Lemma 3. *Assume that $p_{i,B}$ is the largest element of $\boldsymbol{p}_B$ that satisfies (20b). Then, there are only $N-i+1$ closed-form solutions which may be the globally optimal solution to $\mathbf{P}2$. The x-th, $1 \leq x \leq N-i+1$, possible solution $\boldsymbol{a}^x$ is given by*

$$[\boldsymbol{a}^x]_n = \begin{cases} 1, & \text{if } i+x-1 \leq n \leq i+x+K_x-2 \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

where $K_x = \min\left\{N-i-x+2, \lfloor\frac{G\sqrt{\sigma_E}}{p_{i+x-1,B}\sqrt{\Upsilon}}\rfloor\right\}$.

Proof: Firstly, it can be shown that a larger number of variables that are equal to one yields a larger objective value. Therefore, we need to identify at most how many variables a_n can be set to one and and what their indices n are. From constraint (20b), we know that a variable a_n corresponding to a large $p_{n,B}$ cannot be one since this would violate the constraint. Under the given assumption that $p_{i,B}$ is the largest element of $\boldsymbol{p}_B$ that satisfies (20b), it is only feasible to let $a_i, ..., a_N$ equal to one. By analyzing (20c), we can find that there are only $N-i+1$ solutions that may achieve the best performance. Specifically, the x-th solution corresponds to the setting that $a_1 = ... = a_{i+x-2} = 0$ and $a_{i+x-1} = 1$. In this case, from (20c), we have $\max\{a_n p_{n,B}\} = p_{i+x-1,B}$ and then the maximal number of variable a_n that could be equal to one is $K_x = \min\left\{N-i-x+2, \lfloor\frac{G\sqrt{\sigma_E}}{p_{i+x-1,B}\sqrt{\Upsilon}}\rfloor\right\}$. Then, we need to decide which K_x variables are equal to one. Based on the objective function (20), clearly, the optimal allocation is to set $a_1 = ... = a_{i+x-2} = 0$, $a_{i+x-1} = ... = a_{i+x-1+K_x-1} = 1$ and $a_{i+x-1+K_i} = ... = a_N = 0$. ■

Based on Lemma 3, we can use a one-dimenstional search to obtain the optimal solution. I.e., the optimal solution for Problem P2 is $\boldsymbol{a}^y$ where

$$y = \arg\max_x \left\{\sum_{n=1}^{N}[\boldsymbol{a}^x]_n p_{n,B}\right\}. \tag{22}$$

The solution in (21) indicates that only some of the variables in the middle can be set to one, which means that some devices with best and worst channel conditions cannot be selected as participants. This validates the use of artificial noise to effect a tradeoff between achieving privacy and security and guaranteeing learning performance. In particular, choosing devices with the best channel conditions could result in a high risk of privacy leakage and security breaches, while choosing devices with the worst channel conditions would detract from the learning performance. Based on this insight, we next propose a low-complexity heuristic algorithm based on BnB, referred to as the secure and private aggregatoin (SPA) algorithm, to solve Problem $\mathbf{P1}$.

2) BnB-based SPA Algorithm for Problem $\mathbf{P1}$*:* In the proposed algorithm, we utilize the idea of BnB to quickly cut down branches of infeasible solutions by checking the constraints.

We now assume, alternatively to the above analysis, that the elements in $\boldsymbol{p}_B = [p_{1,B}, ..., p_{n,B}, ..., p_{N,B}]$ are sorted in ascending order. It is clear that when $\sum_{n=1}^{N} a_n p_{n,B}$ is small, the value of the objective function is large. By contrast, it can be observed from constraints (19b) and (19c) that the fewer the number of $a_n = 1, n \in \mathcal{N}$ and the smaller the $p_{n,B}$ are, the easier the constraints can be satisfied. More specifically, if $\boldsymbol{a} = [1, 0, ..., 0, ..., 0]$ cannot satisfy constraints (19b) and (19c), any other solutions $\boldsymbol{a} = [1, a_2, ..., a_n, ..., a_N]$ cannot meet constraints (19b) and (19c) either. Under this circumstance, all the solutions with $a_1 = 1$ are infeasible and should be discarded. Following this idea, we can delete half of the solution space of the subproblem in each branch-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

Algorithm 1 BnB-based SPA Algorithm for Solving Problem $\mathbf{P1}$

Input: Given ϵ and Υ and initialize $\Psi^* = +\infty$.
Output: $\boldsymbol{a}^*$.

for $iter \in [1, N]$ **do**
  Let $\boldsymbol{a}^{(iter)} = [0, ..., 0, ..., 0]$.
  for $index \in [iter, N]$ **do**
    Let $a_{index}^{(iter)} = 1$.
    if $\boldsymbol{a}^{(iter)}$ is infesiable for (19b) and (19c) **then**
      Set $a_{index}^{(iter)} = 0$.
    end if
  end for
  Compute the objective value $\Psi\left(\boldsymbol{a}^{(iter)}\right)$.
  if $\Psi\left(\boldsymbol{a}^{(iter)}\right) \leqslant \Psi^*$ **then**
    $\Psi^* = \Psi\left(\boldsymbol{a}^{(iter)}\right)$ and $\boldsymbol{a}^* = \boldsymbol{a}^{(iter)}$.
  end if
end for

and-bound round, and therefore, we can keep narrowing the search space effectively. Given the property of the objective function, we try to to find a solution with more variables equal to 1 while satisfying the constraints. Further, to introduce more diversity to the solutions, we will branch and bound starting from different indices of the nodes, i.e., from $a_n, \forall n$. Specifically, one round of the detailed branch-and-bound process from a_n is described as follows:

- Branching: Select the current node a_n that has not been branched yet. We branch it into two nodes: one is to set it as the participant, and the other is to set it as the helper.
- Bounding: Check if $\boldsymbol{a} = [0, ..., 0, a_n = 1, 0, ..., 0]$ meets the constraints (19b) and (19c).
- Pruning: If $\boldsymbol{a} = [0, ..., 0, a_n = 1, 0, ..., 0]$ satisfies constraints (19b) and (19c), the node is selected as a participant since this selection scheme would definitely lead to a better objective value than selecting this node as a helper. In this case, the branch with $a_n = 0$ is cut off. Otherwise, this node is selected as a helper and the branch with $a_n = 1$ is cut off.

By iteratively conducting the branch-and-bound process, the overall algorithm for solving Problem $\mathbf{P1}$ is formally presented in Algorithm 1.

The proposed BnB-based algorithm operates at a complexity of $\mathcal{O}\left(N^2\right)$, which can efficiently address Problem $\mathbf{P1}$.

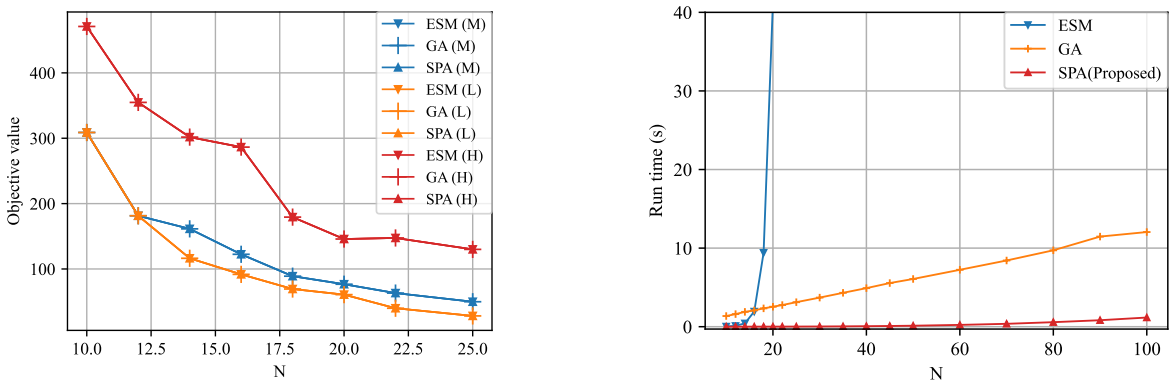
IV. Simulation Results

In this section, we evaluate the performance of the proposed SPA algorithm and the scheduling policies. In the case that channel noise is sufficient for the private and secure participation of full devices, we just schedule all the devices to participate in training. Therefore, we do not consider the simulation of this case. We evaluate our proposed scheme by training a convolutional neural network (CNN) and multi-layer perceptron (MLP) models on two image classification tasks using MNIST and CIFAR10 datasets, both of them have 10 classes of data sample. In particular, the CNN consists of two 5×5 convolution layers with rectified linear unit (ReLU) activation. The two convolution layers have 10 and 20 channels respectively, and each layer has 2×2 max pooling, a fully connected layer with 50 units and ReLU activation, finally a 10 units fully connected layer with a log-softmax output layer. For regularization, dropout is applied. The MLP model consists of an input layer followed by two fully connected hidden layers with ReLU activation and finally an output layer. The first hidden layer contains 256 units, and the second hidden layer contains 64 units. Finally, there is an output layer with 10 units and a softmax activation function. The learning rate is set as $\eta = 0.1$. The transmit power budgets at all devices are assumed to be the same and are set to $P_n = 5$W. The powers of the additive Gaussian noise at both the BS and Eve are set to $\sigma_B = \sigma_E = 1$W.

A. Evaluation of the SPA Algorithm

In this section, we evaluate the performance of the proposed BnB-based SPA algorithm in solving the optimization problem in Policy-2 by comparing it with the exhaustive search method (ESM), and genetic algorithm (GA) under three cases of privacy and security requirements: low (L) level privacy and security requirement, i.e, $\Upsilon = 0.1, \epsilon = 20$; medium (M) level, i.e, $\Upsilon = 0.5, \epsilon = 12$; high (H) level, i.e., $\Upsilon = 8, \epsilon = 0.7$; Specifically, ESM, also known as brute-force search, which is a very general problem-solving technique and algorithmic paradigm. By fully checking all possible solutions, ESM can guarantee the optimal solution to the problem, which can be used to validate the effectiveness of the GA and the proposed SPA by observing the performance gap between them and the optimal results. However, due to the prohibitive computational complexity, ESM is only employed in the case with a smaller number of participants. GA is a metaheuristic algorithm inspired by the process of natural selection, and it is commonly used to generate high-quality solutions to optimization and search problems by relying on biologically inspired operators such as mutation, crossover and selection [30]. To implement GA, we utilize the Python tool geatpy, where the "Encoding" is set to "BG", "NIND" is set to "100", the maximum generation is set to "180", and the crossover probabilityis set to "0.7".

Fig. 2(a) illustrates the objective value of Problem 1 computed through ESM, GA, and our proposed SPA algorithm. ESM ensures optimal solutions by exhaustively examining all potential solutions, providing a benchmark to assess GA and the proposed SPA's performance. Notably, from Fig. 2, we observe that the proposed BnB-based SPA algorithm and GA yield identical results to ESM, showcasing the effectiveness of our SPA. In Fig. 2 (b), we plot the execution time of the proposed SPA algorithm, GA, and ESM where the security and privacy coefficients are set as $\Upsilon = 1.5$ and $\epsilon = 12$, respectively. The results reveal that in the case of relatively small N, i.e., $N = 10, 12, 14$, the execution time



(a) The objective value comparison.

(b) Time comparison.

Fig. 2: The comparison between ESM, GA and SPA.

of ESM is similar to that of the SPA algorithm and less than that of GA. However, as N increases from $N = 16$, the time consumed by ESM increases sharply; therefore, the time consumption of ESM is omitted when N is large than 20. By contrast, the proposed SPA algorithm always consumes the least amount of time and the growth rate is slow as N increases. Therefore, the proposed SPA algorithm can achieve the same performance as GA while maintaining a very low complexity, indicating that it is promising for application in large-scale FL systems.

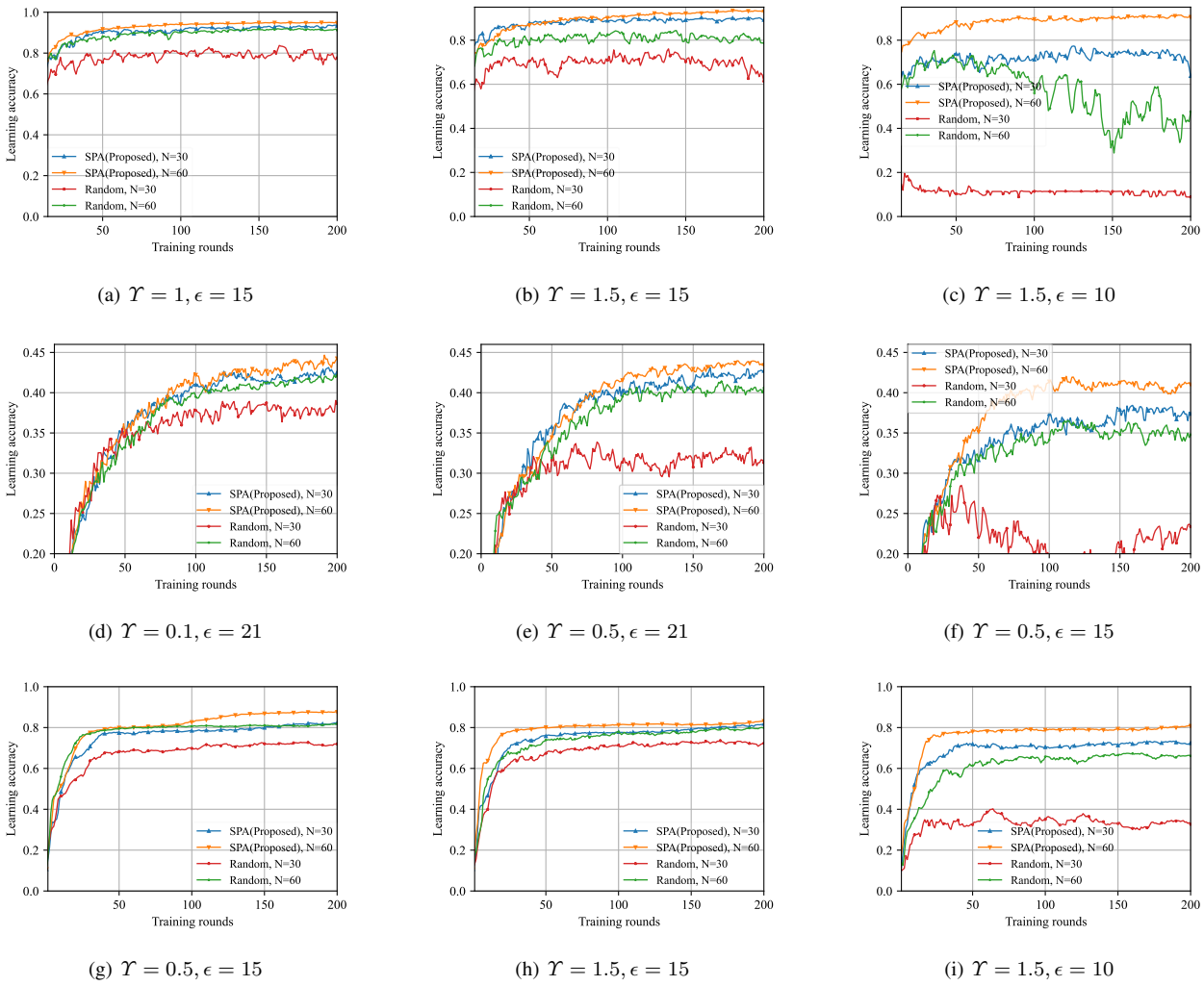
In Fig. 3, we illustrate the learning accuracy of training the CNN and MLP models on the MNIST and CIFAR10 datasets, utilizing the SPA scheduling scheme and a random scheduling method across various privacy and security requirements. Specifically, Fig. 3 (a)-(c) depict the accuracy of the CNN on the MNIST dataset; Fig. 3(d)-(f) illustrate the accuracy of the CNN on the CIFAR10 dataset; and Fig. 3(g)-(i) show the accuracy of the MLP on the MNIST dataset. Fig. 3(b), Fig. 3(e), and Fig. 3(h) illustrate cases with increased security requirements compared to Fig. 3(a), Fig. 3(d), and Fig. 3(g), respectively. On the other hand, Fig. 3 (c), Fig. 3(f), and Fig. 3(i) represent scenarios with increased privacy requirements compared to Fig. 3 (b), Fig. 3(e), and Fig. 3(h), respectively. The results reveal that the SPA algorithm consistently achieves comparable performance across various privacy and security requirements, whereas random scheduling methods experience significant performance deterioration with increased privacy and security demands. Additionally, the number of devices noticeably influences SPA performance under heightened privacy and security requirements, as illustrated in Fig. 3(c), Fig. 3(f), and Fig. 3(i). This is attributed to the fact that stricter privacy and security requirements necessitate greater noise. In this case, the more devices participating, the distortion of gradient averaging diminishes accordingly.

B. Evaluation of Policy-1 and Policy-2

In this subsection, we compare the performance of Policy-1 and Policy-2 by training CNN on the CIFAR10 dataset, as illustrated in Fig. 4 (a)-(c), and the MNIST dataset, as depicted in Fig. 4 (d)-(f). From both cases, we observe that as the number of devices increases, Policy-1 gradually outperforms Policy-2. It is well-known that having more participants and less noise distortion can lead to a more accurate model. Under Policy-1, only devices that can be protected by channel noise are selected to participate in the training process. In contrast, Policy-2 has the capability to include a larger number of devices in the training process, benefiting from the amplified noise power provided by the helpers. Consequently, Policy-1 involves fewer devices in the training process compared to Policy-2. Therefore, in scenarios where only a limited number of devices are eligible to participate in the training process under Policy-1, Policy-2 exhibits superior performance. As the total number of devices increases, more devices become eligible for selection as participants in Policy-1, where the noise is limited to channel noise. In contrast, although Policy-2 selects more devices to participate in training, the noise power is also greater than that in Policy-1. Therefore, when the number of devices is larger, Policy-1 outperforms Policy-2. Nonetheless, Policy-2 is indispensable in scenarios where channel noise fails to protect any device for participation in the training process. Consequently, each policy offers unique advantages and is irreplaceable in specific scenarios.

V. Conclusion

In this paper, we have studied the secure aggregation of wireless OTA-FL leveraging channel noise and employing scheduling policies designed for three cases of channel noise. Specifically, we have formulated an optimization problem for scenarios where channel noise is insufficient to ensure security and privacy. In such cases, we have selected some devices for participation in training while choosing others



(a) $\Upsilon = 1, \epsilon = 15$ (b) $\Upsilon = 1.5, \epsilon = 15$ (c) $\Upsilon = 1.5, \epsilon = 10$

(d) $\Upsilon = 0.1, \epsilon = 21$ (e) $\Upsilon = 0.5, \epsilon = 21$ (f) $\Upsilon = 0.5, \epsilon = 15$

(g) $\Upsilon = 0.5, \epsilon = 15$ (h) $\Upsilon = 1.5, \epsilon = 15$ (i) $\Upsilon = 1.5, \epsilon = 10$

Fig. 3: The learning performance of the proposed SPA algorithm of (a)-(c) CNN on MNIST dataset; (d-f) CNN on CIFAR10 dataset; and (g-i) MLP on MNIST dataset.

as helpers, transmitting artificial Gaussian noise. We have derived a closed-form solution for cases involving high-dimensional models. Additionally, we have introduced an algorithm named SPA based on BnB, which effectively addresses the problem with low computational complexity.

Appendix A
Proof of Lemma 1

Here we use index k to denote a particular device, instead of n, to avoid confusion between the specific index of device n and the notation n in the summation. Assume that $\mathcal{B}_k$ and $\mathcal{B}_k^{'}$ are two adjacent datasets differing in one sample. $(\boldsymbol{y}^t)'$ is the received signal based on $\mathcal{B}_k^{'}$ at the BS, which only differs in one gradient from $\boldsymbol{y}^t$. The gradient $(\boldsymbol{g}_k^t)'$ from participant k in $(\boldsymbol{y}^t)'$ is obtained based on $\mathcal{B}_k^{'}$. Based on the definition of sensitivity and Assumption 1, one has

$$\begin{aligned}\Delta S_k^t &\triangleq \max_{\mathcal{B}_k, \mathcal{B}_k^{'}} \left\| \boldsymbol{y}^t - (\boldsymbol{y}^t)' \right\|_2 \\ &= \max_{\mathcal{B}_n, \mathcal{B}_k^{'}} \left\| \frac{h_{k,B}^t \sqrt{P_k}}{G} \left(\boldsymbol{g}_k^t - (\boldsymbol{g}_k^t)' \right) \right\|_2 \\ &= \frac{h_{k,B}^t \sqrt{P_k}}{G} \left\| \boldsymbol{g}_k^t - (\boldsymbol{g}_k^t)' \right\|_2 \overset{(a)}{\leqslant} 2p_{k,B}^t, \end{aligned} \tag{23}$$

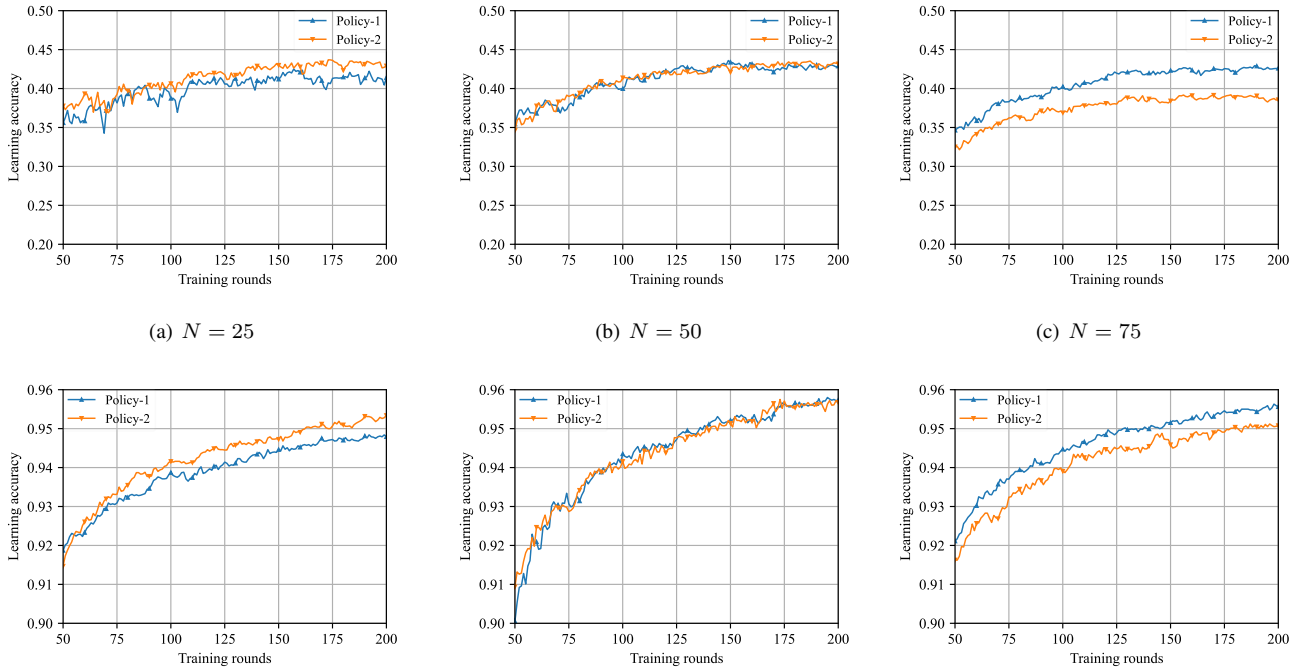
where (a) is from the triangle inequality and Assumption 1. In accordance with the Gaussian mechanism of DP and the above result, this completes the proof of Lemma 1.

Appendix B
Proof of Lemma 2

Firstly, since the elements in $\boldsymbol{g}_n^t$ are uniformly distributed in $[a, b]$, the $\boldsymbol{g}_{ave}^t$ follows the same distribution in $[a, b]$. For analysis, we define $\tilde{\mathcal{E}}^t : (\boldsymbol{g}_n^t)_{n \in \mathcal{U}^t} \to \tilde{\boldsymbol{z}}^t \in \tilde{\mathcal{Z}}$ where

$$\tilde{\boldsymbol{z}}^t = \sum_{n \in \mathcal{U}^t} \frac{\Lambda^t}{G} \boldsymbol{g}_n^t + \boldsymbol{r}_{E,Tot}^t, \tag{24}$$

where $\boldsymbol{r}_{E,Tot}^t = \sum_{n \in \mathcal{M}^t} h_{n,E}^t \sqrt{\frac{P_n}{d}} \boldsymbol{e}_n^t + \boldsymbol{r}_E^t$. Assume that the variance of $\tilde{\boldsymbol{z}}^t$ is σ. Following Lemma 3 and Lemma 4



(a) $N = 25$ (b) $N = 50$ (c) $N = 75$

(d) $N = 25$ (e) $N = 50$ (f) $N = 75$

Fig. 4: The performance of Policy-1 and Policy-2 of training (a)-(c) CNN on CIFAR10 dataset; (d)-(f) CNN on MNIST dataset.

in [22], the minimum MSE estimator $e\left(\tilde{\boldsymbol{z}}^t\right)$ for estimating $\boldsymbol{g}_{ave}^t$ from the observations $\tilde{\boldsymbol{z}}^t$ satisfies

$$\mathbb{E}\left[\left(\boldsymbol{g}_{ave}^t - e\left(\tilde{\boldsymbol{z}}^t\right)\right)^2\right] = \sigma \Xi\left(\frac{b-a}{\sqrt{\sigma}}\right). \tag{25}$$

The lowest-variance unbiased estimator is

$$e\left(\tilde{\boldsymbol{z}}^t\right) = \boldsymbol{g}_{ave}^t + \frac{G}{\left|\mathcal{U}^t\right|^t \Lambda^t} \boldsymbol{r}_{E,Tot}^t, \tag{26}$$

with the variance given by

$$\gamma_E^t = \frac{G^2}{\left(\left|\mathcal{U}^t\right|^t \Lambda^t\right)^2}\left(\sum_{n\in\mathcal{M}^t} \frac{\left(h_{n,E}^t\right)^2 P_n}{d} + \sigma_E\right), \tag{27}$$

where $\Lambda^t = \max\limits_{n\in\mathcal{U}^t}\left\{h_{n,B}^t\sqrt{P_n}\right\}$. It thus follows from (30) and Definition 3 that $\tilde{\mathcal{E}}^t$ guarantees $\left(\tilde{\mathcal{E}}^t, \gamma_E^t \Xi\left(\frac{b-a}{\sqrt{\gamma_E^t}}\right)\right)$.

On the other hand, one has

$$\mathbb{E}\left[\left\|e\left(\tilde{\boldsymbol{z}}^t\right) - \boldsymbol{g}_{ave}^t\right\|^2\right] = \left(\frac{G}{\left|\mathcal{U}^t\right|^t \Lambda^t}\right)^2 \mathbb{E}\left[\left\|\boldsymbol{r}_{E,Tot}^t\right\|^2\right]. \tag{28}$$

Similarly, we also have

$$\begin{aligned}
&\mathbb{E}\left[\left\|e\left(\boldsymbol{z}^t\right) - \boldsymbol{g}_{ave}^t\right\|^2\right] \\
=&\mathbb{E}\left[\left\|\sum_{n\in\mathcal{U}^t}\left(\frac{h_{n,E}^t\sqrt{\lambda_n^t}}{\Lambda^t} - 1\right)\boldsymbol{g}_n^t + \frac{G}{\Lambda^t}\boldsymbol{r}_{E,Tot}^t\right\|^2\right] \\
\overset{(a)}{=}&\frac{1}{\left|\mathcal{U}^t\right|^2}\mathbb{E}\left[\left\|\sum_{n\in\mathcal{U}^t}\left(\frac{h_{n,E}^t\sqrt{\lambda_n^t}}{\Lambda^t} - 1\right)\boldsymbol{g}_n^t\right\|^2\right] \\
&+\left(\frac{G}{\left|\mathcal{U}^t\right|^t \Lambda^t}\right)^2 \mathbb{E}\left[\left\|\boldsymbol{r}_{E,Tot}^t\right\|^2\right],
\end{aligned} \tag{29}$$

where (a) comes from $\mathbb{E}\left[\boldsymbol{r}_{E,Tot}^t\right] = 0$. Obviously, $\mathbb{E}\left[\left\|e\left(\tilde{\boldsymbol{z}}^t\right) - \boldsymbol{g}_{ave}^t\right\|^2\right]$ is smaller than $\mathbb{E}\left[\left\|e\left(\boldsymbol{z}^t\right) - \boldsymbol{g}_{ave}^t\right\|^2\right]$, and therefore, $e\left(\tilde{\boldsymbol{z}}^t\right)$ is a closer estimate of $\boldsymbol{g}_{ave}^t$. Then, $e\left(\boldsymbol{z}^t\right)$ has a larger variance and can achieve at least $\left(\tilde{\mathcal{E}}^t, \gamma_E^t \Xi\left(\frac{b-a}{\sqrt{\gamma_E^t}}\right)\right)$-MSE-security.

Alternatively, from the communication point of view, one can also argue that $\tilde{\boldsymbol{z}}^t$ could have a better recovery of gradient than $\boldsymbol{z}^t$ because of a higher SNR as $\Lambda^t = \max\limits_{n\in\mathcal{U}^t}\left\{h_{n,B}^t\sqrt{P_n}\right\}$. Therefore, if $\tilde{\boldsymbol{z}}^t$ can guarantee at least $\left(\tilde{\mathcal{E}}^t, \gamma_E^t \Xi\left(\frac{b-a}{\sqrt{\gamma_E^t}}\right)\right)$-MSE-security, then so can $\boldsymbol{z}^t$. This completes the proof of Lemma 2.

Appendix C
Proof of Theorem 1

Before delving into the proof of Theorem 1, by applying Assumption 1, first provide the following results:

$$\mathbb{E}\left[\hat{\boldsymbol{g}}^t - \bar{\boldsymbol{g}}^t\right] = \sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t}\mathbb{E}\left[\boldsymbol{g}_n^t - \nabla L_n\left(\boldsymbol{m}_n^t\right)\right] = 0, \tag{30}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

and

$$\begin{aligned}&\mathbb{E}\left[\left\|\hat{\boldsymbol{g}}^t-\bar{\boldsymbol{g}}^t\right\|_2^2\right]\\=&\mathbb{E}\left[\left\|\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\left(\boldsymbol{g}_n^t-\nabla L_n\left(\boldsymbol{m}_n^t\right)\right)\right\|_2^2\right]\\\overset{(a)}{\leqslant}&\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\mathbb{E}\left[\left\|\boldsymbol{g}_n^t-\nabla L_n\left(\boldsymbol{m}_n^t\right)\right\|_2^2\right]\overset{(b)}{\leqslant}\vartheta^2,\end{aligned}\tag{31}$$

where (a) is obtained by using Jensen's inequality and (b) is from Assumption 1. For ease of presentation, we define

$$\hat{\boldsymbol{g}}^t=\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\boldsymbol{g}_n^t,\tag{32}$$

$$\bar{\boldsymbol{g}}^t=\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\nabla L_n\left(\boldsymbol{m}_n^t\right),\tag{33}$$

$$\boldsymbol{r}_{B,Tot}^t=\sum_{n\in\mathcal{M}^t}h_{n,B}^t\sqrt{\frac{P_n}{d}}\boldsymbol{e}_n^t+\boldsymbol{r}_B^t,\tag{34}$$

and

$$\boldsymbol{r}_{E,Tot}^t=\sum_{n\in\mathcal{M}^t}h_{n,E}^t\sqrt{\frac{P_n}{d}}\boldsymbol{e}_n^t+\boldsymbol{r}_E^t.\tag{35}$$

Then, the update of the global model in (9) can be re-expressed as

$$\boldsymbol{m}^{t+1}=\boldsymbol{m}^t-\tau^t\left(\hat{\boldsymbol{g}}^t+\frac{G}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\boldsymbol{r}_{B,Tot}^t\right).\tag{36}$$

Consequently, the gap between the global model parameter $\boldsymbol{m}^{t+1}$ and the optimal global model parameter $\boldsymbol{m}^*$ can be expressed as,

$$\begin{aligned}&\mathbb{E}\left[\left\|\boldsymbol{m}^{t+1}-\boldsymbol{m}^*\right\|_2^2\right]\\=&\mathbb{E}\left[\left\|\boldsymbol{m}^t-\tau^t\hat{\boldsymbol{g}}^t-\tau^t\frac{G}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\boldsymbol{r}_{B,Tot}^t-\boldsymbol{m}^*\right\|_2^2\right]\\=&\mathbb{E}\left[\left\|\boldsymbol{m}^t-\tau^t\hat{\boldsymbol{g}}^t+\tau^t\bar{\boldsymbol{g}}^t-\tau^t\bar{\boldsymbol{g}}^t\right.\right.\\&\left.\left.-\tau^t\frac{G}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\boldsymbol{r}_{B,Tot}^t-\boldsymbol{m}^*\right\|_2^2\right]\\\overset{(a)}{=}&\mathbb{E}\left[\left\|\boldsymbol{m}^t-\tau^t\bar{\boldsymbol{g}}^t-\tau^t\frac{G}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\boldsymbol{r}_{B,Tot}^t-\boldsymbol{m}^*\right\|_2^2\right]\\&+\left(\tau^t\right)^2\mathbb{E}\left[\left\|\hat{\boldsymbol{g}}^t-\bar{\boldsymbol{g}}^t\right\|_2^2\right]\\\overset{(b)}{=}&\mathbb{E}\left[\left\|\boldsymbol{m}^t-\tau^t\bar{\boldsymbol{g}}^t-\boldsymbol{m}^*\right\|_2^2\right]\\&+\left(\frac{\tau^tG}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\right)^2\mathbb{E}\left[\left\|\boldsymbol{r}_{B,Tot}^t\right\|_2^2\right]\\&+\left(\tau^t\right)^2\mathbb{E}\left[\left\|\hat{\boldsymbol{g}}^t-\bar{\boldsymbol{g}}^t\right\|_2^2\right]\\\overset{(c)}{=}&\mathbb{E}\left[\left\|\boldsymbol{m}^t-\boldsymbol{m}^*\right\|_2^2\right]+\underbrace{\left(\tau^t\right)^2\mathbb{E}\left[\left\|\bar{\boldsymbol{g}}^t\right\|_2^2\right]}_{A}\\&\underbrace{-2\tau^t\left\langle\boldsymbol{m}^t-\boldsymbol{m}^*,\bar{\boldsymbol{g}}^t\right\rangle}_{B}\\&+\left(\tau^t\right)^2\vartheta^2+\left(\frac{\tau^tG}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\right)^2\underbrace{\mathbb{E}\left[\left\|\boldsymbol{r}_{B,Tot}^t\right\|_2^2\right]}_{C},\end{aligned}\tag{37}$$

where (a) and (c) are obtained by applying (30) and (31), ands (b) is from the fact that $\mathbb{E}\left[\boldsymbol{r}_{B,Tot}^t\right]=0$.

Next, we obtain upper bounds on each term in (37), separately. Firstly, we have an upper bound on the term A in (37) as follows:

$$\begin{aligned}&\left(\tau^t\right)^2\mathbb{E}\left[\left\|\bar{\boldsymbol{g}}^t\right\|_2^2\right]\\=&\left(\tau^t\right)^2\mathbb{E}\left[\left\|\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\nabla L_n\left(\boldsymbol{m}_n^t\right)\right\|_2^2\right]\\\overset{(a)}{\leqslant}&\left(\tau^t\right)^2\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\mathbb{E}\left[\left\|\nabla L_n\left(\boldsymbol{m}_n^t\right)\right\|_2^2\right]\\\leqslant&2\theta\left(\tau^t\right)^2\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\mathbb{E}\left[L_n\left(\boldsymbol{m}_n^t\right)-L_n\left(\boldsymbol{m}_n^*\right)\right],\end{aligned}\tag{38}$$

where (a) is obtained by applying Jensen's inequality and the θ-smooth property that

$$\left\|\nabla L_n\left(\boldsymbol{m}_n^t\right)\right\|_2^2\leqslant2\theta\left[L\left(\boldsymbol{m}_n^t\right)-L\left(\boldsymbol{m}_n^*\right)\right],\tag{39}$$

is applied in the last inequality. Then, an upper bound on the term B in (37) is given by

$$\begin{aligned}&-2\tau^t\left\langle\boldsymbol{m}^t-\boldsymbol{m}^*,\bar{\boldsymbol{g}}^t\right\rangle\\=&2\tau^t\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\mathbb{E}\left[\left\langle\boldsymbol{m}^*-\boldsymbol{m}_n^t,\nabla L_n\left(\boldsymbol{m}^t\right)\right\rangle\right]\\\overset{(a)}{\leqslant}&2\tau^t\sum_{n\in\mathcal{U}^t}\frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t}p_{n,B}^t}\mathbb{E}\left[L_n\left(\boldsymbol{m}^*\right)-L_n\left(\boldsymbol{m}_n^t\right)\right.\\&\left.-\frac{\rho}{2}\left\|\boldsymbol{m}^t-\boldsymbol{m}^*\right\|_2^2\right],\end{aligned}\tag{40}$$

where (a) is from Assumption 3. By combining (38) with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/JIOT.2024.3405855, IEEE Internet of Things Journal

(40), we obtain an upper bound on the term $A+B$ as follows:

$$\begin{aligned}
&\left(\tau^t\right)^2 \mathbb{E}\left[\left\|\bar{\boldsymbol{g}}^t\right\|_2^2\right] - 2\tau^t\left\langle \boldsymbol{m}^t - \boldsymbol{m}^*, \bar{\boldsymbol{g}}^t\right\rangle \\
=&2\theta\left(\tau^t\right)^2 \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}_n^t\right) - L_n\left(\boldsymbol{m}_n^*\right)\right] \\
&- 2\tau^t \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}_n^t\right) - L_n\left(\boldsymbol{m}^*\right)\right] \\
&- \rho\tau^t \mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right] \\
=&-2\tau^t\left(1-\theta\tau^t\right) \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}_n^t\right) - L_n\left(\boldsymbol{m}^*\right)\right] \\
&+ 2\theta\left(\tau^t\right)^2 \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}^*\right) - L_n\left(\boldsymbol{m}_n^*\right)\right] \\
&- \rho\tau^t \mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right] \\
=&\underbrace{-2\tau^t\left(1-\theta\tau^t\right) \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}_n^t\right) - L_n\left(\boldsymbol{m}^*\right)\right]}_{D} \\
&+ 2\theta\left(\tau^t\right)^2 \Gamma - \rho\tau^t \mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right].
\end{aligned} \tag{41}$$

To obtain an upper bound on the term D in (41), we have

$$\begin{aligned}
&-2\tau^t\left(1-\theta\tau^t\right) \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}_n^t\right) - L_n\left(\boldsymbol{m}^*\right)\right] \\
=&2\tau^t\left(1-\theta\tau^t\right) \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \left[\mathbb{E}\left[L_n\left(\boldsymbol{m}_n^*\right) - L_n\left(\boldsymbol{m}_n^t\right)\right]\right. \\
&\left.+\mathbb{E}\left[L_n\left(\boldsymbol{m}^*\right) - L_n\left(\boldsymbol{m}_n^*\right)\right]\right] \\
\overset{(a)}{\leqslant}&2\tau^t\left(1-\theta\tau^t\right) \sum_{n\in\mathcal{U}^t} \frac{p_{n,B}^t}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t} \mathbb{E}\left[L_n\left(\boldsymbol{m}^*\right) - L_n\left(\boldsymbol{m}_n^*\right)\right] \\
=&2\tau^t\left(1-\theta\tau^t\right)\Gamma,
\end{aligned} \tag{42}$$

where (a) comes from that $2\tau^t\left(1-\theta\tau^t\right) \geqslant 0$ and $L_n\left(\boldsymbol{m}_n^*\right) - L_n\left(\boldsymbol{m}_n^t\right) \leqslant 0$. Substituting (42) back into (41), we finally get an upper bound on the term $A+B$ in (37) as follows:

$$\begin{aligned}
&\left(\tau^t\right)^2 \mathbb{E}\left[\left\|\bar{\boldsymbol{g}}^t\right\|_2^2\right] - 2\tau^t\left\langle \boldsymbol{m}^t - \boldsymbol{m}^*, \bar{\boldsymbol{g}}^t\right\rangle \\
\leqslant&2\tau^t\left(1-\theta\tau^t\right)\Gamma + 2\theta\left(\tau^t\right)^2\Gamma - \rho\tau^t\mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right] \\
\leqslant&2\tau^t\Gamma - \rho\tau^t\mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right] \\
\overset{(a)}{\leqslant}&2\varrho\left(\tau^t\right)^2\Gamma - \rho\tau^t\mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right],
\end{aligned} \tag{43}$$

where (a) is from $\frac{1}{\varrho} \leqslant \tau^t$. Then, an upper bound on the term C in (37) is given by

$$\begin{aligned}
&\mathbb{E}\left[\left\|\boldsymbol{r}_{B,Tot}^t\right\|_2^2\right] \\
=&\mathbb{E}\left[\left\|\sum_{n\in\mathcal{M}^t} \frac{p_{n,B}^t}{\sqrt{d}} \boldsymbol{e}_n^t + \boldsymbol{r}_B^t\right\|_2^2\right] \\
\overset{(a)}{\leqslant}&\mathbb{E}\left[\left\|\sum_{n\in\mathcal{M}^t} \frac{p_{n,B}^t}{\sqrt{d}} \boldsymbol{e}_n^t\right\|_2^2\right] + \mathbb{E}\left[\left\|\boldsymbol{r}_B^t\right\|_2^2\right] \\
\overset{(b)}{\leqslant}&\left|\mathcal{M}^t\right| \sum_{n\in\mathcal{M}^t} \frac{\left(p_{n,B}^t\right)^2}{d} \mathbb{E}\left[\left\|\boldsymbol{e}_n^t\right\|_2^2\right] + d\sigma_B \\
\leqslant&N \sum_{n\in\mathcal{M}^t} \left(p_{n,B}^t\right)^2 + d\sigma_B,
\end{aligned} \tag{44}$$

where (a) is from the fact that $\mathbb{E}\left[\boldsymbol{e}_n^t\right] = \mathbb{E}\left[\boldsymbol{r}_B^t\right] = 0$ and step (b) is obtained by applying Jensen's inequality. Finally, by substituting (43), (44) into (37), we get an upper bound on $\mathbb{E}\left[\left\|\boldsymbol{m}^{t+1} - \boldsymbol{m}^*\right\|_2^2\right]$ as follows:

$$\begin{aligned}
\mathbb{E}\left[\left\|\boldsymbol{m}^{t+1} - \boldsymbol{m}^*\right\|_2^2\right] \leqslant& \left(1-\rho\tau^t\right)\mathbb{E}\left[\left\|\boldsymbol{m}^t - \boldsymbol{m}^*\right\|_2^2\right] \\
&+ 2\varrho\left(\tau^t\right)^2\Gamma + \left(\tau^t\right)^2\vartheta^2 \\
&+ \left(\frac{\tau^t G}{\sum_{n\in\mathcal{U}^t} p_{n,B}^t}\right)^2 \left(N \sum_{n\in\mathcal{M}^t} \left(p_{n,B}^t\right)^2 + d\sigma_B\right).
\end{aligned} \tag{45}$$

This completes the proof of Theorem 1.

REFERENCES

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

[2] Z. Chen, W. Yi, Y. Liu, and A. Nallanathan, "Knowledge-aided federated learning for energy-limited wireless networks," *IEEE Trans. Commun.*, 2023.

[3] Z. Chen, W. Yi, and A. Nallanathan, "Exploring representativity in device scheduling for wireless federated learning," *IEEE Trans. Wireless Commun.*, 2023.

[4] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Security*, 2015, pp. 1322–1333.

[5] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proc. IEEE Symp. Security Privacy (SP)*, 2019, pp. 691–706.

[6] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proc. IEEE Symp. Security Privacy (SP)*, 2017, pp. 3–18.

[7] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning," in *Proc. IEEE Symp. Security Privacy (SP)*, 2018, pp. 1–15.

[8] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, "A hybrid approach to privacy-preserving federated learning," in *Proc. ACM Workshop Artif. Intell. Security*, 2019, pp. 1–11.

[9] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. Vincent Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3454–3469, 2020.

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

[11] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Differentially private AirComp federated learning with power adaptation harnessing receiver noise," in *Proc. IEEE Global Commun. Conf*, 2020, pp. 1–6.

[12] D. Liu and O. Simeone, "Privacy for free: Wireless federated learning via uncoded transmission with adaptive power control," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 170–185, 2020.
[13] R. Xu, N. Baracaldo, Y. Zhou, A. Anwar, and H. Ludwig, "Hybridalpha: An efficient approach for privacy-preserving federated learning," in *Proc. ACM Workshop Artif. Intell. Security*, 2019, pp. 13–23.
[14] G. Xu, H. Li, S. Liu, K. Yang, and X. Lin, "Verifynet: Secure and verifiable federated learning," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 911–926, 2019.
[15] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2017, pp. 1175–1191.
[16] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, "Secure single-server aggregation with (poly) logarithmic overhead," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2020, pp. 1253–1269.
[17] M. Shayan, C. Fung, C. J. Yoon, and I. Beschastnikh, "Biscotti: A ledger for private and secure peer-to-peer machine learning," arXiv:1811.09904, 2018.
[18] N. T. T. Van, N. C. Luong, H. T. Nguyen, F. Shaohan, D. Niyato, and D. I. Kim, "Latency minimization in covert communication-enabled federated learning network," *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 13 447–13 452, 2021.
[19] Y.-A. Xie, J. Kang, D. Niyato, N. T. T. Van, N. C. Luong, Z. Liu, and H. Yu, "Securing federated learning: A covert communication-based approach," *IEEE Netw.*, vol. 37, no. 1, pp. 118–124, 2023.
[20] J. Yao and N. Ansari, "Secure federated learning by power control for internet of drones," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 4, pp. 1021–1031, 2021.
[21] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 317–333, 2020.
[22] M. Frey, I. Bjelaković, and S. Stańczak, "Towards secure over-the-air computation," arXiv:2001.03174, 2020.
[23] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, pp. 211–407, 2014.
[24] S. Wan, J. Lu, P. Fan, Y. Shao, C. Peng, and K. B. Letaief, "Convergence analysis and system design for federated learning over wireless networks," *IEEE J. Sel. Areas Commun.,*, vol. 39, no. 12, pp. 3622–3639, 2021.
[25] J. Ren, W. Ni, and H. Tian, "Toward communication-learning tradeoff for federated learning at the network edge," *IEEE Commun. Lett.*, vol. 26, no. 8, pp. 1858–1862, 2022.
[26] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3454–3469, 2020.
[27] K. Wei, J. Li, C. Ma, M. Ding, W. Chen, J. Wu, M. Tao, and H. V. Poor, "Personalized federated learning with differential privacy and convergence guarantee," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 4488–4503, 2023.
[28] C. Xie, S. Koyejo, and I. Gupta, "Asynchronous federated optimization," arXiv:1903.03934, 2019.
[29] S. Chen, D. Yu, Y. Zou, J. Yu, and X. Cheng, "Decentralized wireless federated learning with differential privacy," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6273–6282, 2022.
[30] M. Mitchell, *An Introduction to Genetic Algorithms*. MIT Press, 1998.