Federated Learning with Differential Privacy: Algorithms and Performance Analysis

Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farokhi Farhad,
Shi Jin, Tony Q. S. Quek, H. Vincent Poor

***Abstract*—Federated learning (FL), as a manner of distributed machine learning, is capable of significantly preserving clients' private data from being exposed to external adversaries. Nevertheless, private information can still be divulged by analyzing on the differences of uploaded parameters from clients, e.g., weights trained in deep neural networks. In this paper, to effectively prevent information leakage, we propose a novel framework based on the concept of differential privacy (DP), in which artificial noises are added to the parameters at the clients side before aggregating, namely, noising before model aggregation FL (NbAFL). First, we prove that the NbAFL can satisfy DP under distinct protection levels by properly adapting different variances of artificial noises. Then we develop a theoretical convergence bound of the loss function of the trained FL model in the NbAFL. Specifically, the theoretical bound reveals the following three key properties: 1) There is a tradeoff between the convergence performance and privacy protection levels, i.e., a better convergence performance leads to a lower protection level; 2) Given a fixed privacy protection level, increasing the number N of overall clients participating in FL can improve the convergence performance; 3) There is an optimal number of maximum aggregation times (communication rounds) in terms of convergence performance for a given protection level. Furthermore, we propose a K-random scheduling strategy, where K $(1 < K < N)$ clients are randomly selected from the N overall clients to participate in each aggregation. We also develop the corresponding convergence bound of the loss function in this case and the K-random scheduling strategy can also retain the above three properties. Moreover, we find that there is an optimal K that achieves the best convergence performance at a fixed privacy level. Evaluations demonstrate that our theoretical results are consistent with simulations, thereby facilitating the designs on various privacy-preserving FL algorithms with different tradeoff requirements on convergence performance and privacy levels.**

***Index Terms*—Federated learning, differential privacy, convergence performance, information leakage, client selection**

I. Introduction

With AlphaGo's glorious success, it is expected that the big data-driven artificial intelligence (AI) will soon be applied in all aspects of our daily life, including medical care, food and agriculture, intelligent transportation systems, etc. At the same time, the rapid proliferations of Internet of Things (IoTs) call for data mining and learning securely and reliably in distributed systems [1]–[3]. When integrating AI in a variety of IoT applications, distributed machine learning (ML) are remarkably effective for many data processing tasks by defining parameterized functions from inputs to outputs as compositions of basic building blocks [4], [5]. Federated learning (FL), as a recent advance of distributed ML, was proposed, in which data are acquired and processed locally at the clients side, and then the updated ML parameters are transmitted to a central server for aggregating, i.e., averaging on these parameters [6]–[8]. Typically, clients in FL are distributed devices such as sensors, wearable devices, or mobile phones. The goal of FL is to fit a model generated by an empirical risk minimization (ERM) objective. However, FL also poses several key challenges, such as private information leakage, expensive communication costs between servers and clients, and device variability [9]–[14].

Generally, distributed stochastic gradient descent (SGD) is adopted in FL for training ML models. In [15], [16], bounds for FL convergence performance were developed based on distributed SGD, with a one-step local update before global aggregations. The work in [17] considered partially global aggregations, where after each local update step, parameter aggregation is performed over a non-empty subset of the clients set. In order to analyze the convergence more effectively, federated proximal (FedProx) was proposed [18] by adding regularization on each local loss function. The work in [19] obtained the convergence bound of SGD based FL that incorporates non-independent-and-identically-distributed (non-*i.i.d.*) data distributions among clients.

At the same time, with the ever increasing awareness of data security of personal information, privacy preservation has become a worldwide and significant issue, especially for the big data applications and distributed learning systems. One prominent advantage of FL is that it enables local training without personal data exchange between the server and clients, thereby protecting clients' data from being eavesdropped by hidden adversaries. Nevertheless, private information can still be divulged to some extent from adversaries' analyzing on the differences of related parameters trained and uploaded by the clients, e.g., weights trained in neural networks [20]–[22].

A natural approach to preventing information leakage is to add artificial noises, known as differentially private (DP) techniques [23], [24]. Existing works on DP based learning algorithms include local DP (LDP) [25]–[27], DP based distributed SGD [28], [29] and DP meta learning [30]. In the

Kang Wei, Jun Li, and Chuan Ma are with School of Electrical and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail:{wei.kang, jun.li, chuan.ma}@njust.edu.cn).

Ming Ding and Farokhi Farhad are with Data61, CSIRO, Sydney, NSW 2015, Australia (e-mail:{ming.ding, Farhad.Farokhi}@data61.csiro.au).

Howard Hao Yang and Tony Q. S. Quek are with the Information System Technology and Design Pillar, Singapore University of Technology and Design, Singapore (e-mail:{howard yang, tonyquek}@sutd.edu.sg).

Shi Jin is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: jinshi@seu.edu.cn).

H. Vincent Poor is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: poor@princeton.edu).

LDP, each client perturbs its information locally and only sends a randomized version to a server, thereby protecting both the clients and server against private information leakage. The work in [26] proposed solutions to building up a LDP-compliant SGD, which powers a variety of important ML tasks. The work in [27] considered the distribution estimation at the server over uploaded data from clients while providing protections on these data with LDP. The work in [28] improved the computational efficiency of DP based SGD by tracking detailed information of the privacy loss, and obtained accurate estimates on the overall privacy loss. The work in [29] proposed novel DP based SGD algorithms and analyzed their performance bounds which are shown to be related to privacy levels and the sizes of datasets. Also, the work in [30] focused on the class of gradient-based parameter-transfer methods and developed a DP based meta learning algorithm that not only satisfies the privacy requirement but also retains provable learning performance in convex settings.

More specifically, DP based FL approaches are usually devoted to capturing the tradeoff between privacy and convergence performance in the training process. The work in [31] proposed a FL algorithm with the consideration on preserving clients' privacy. This algorithm can achieve a good training performance at a given privacy level, especially when there is a sufficiently large number of participating clients. The work in [32] presented an alternative approach that utilizes both DP and secure multiparty computation (SMC) to prevent differential attacks. However, the above two works on DP-based FL design have not taken into account the privacy protection during the parameter uploading stage, i.e., the clients' private information can be potentially intercepted by hidden adversaries when uploading the training results to the server. Moreover, these two works only showed empirical results by simulations, but lacked theoretical analysis on the FL system, such as tradeoff between privacy, convergence performance, and convergence rate. Up to now, the theoretical analysis on convergence behavior of FL with privacy-preserving noise perturbations has not yet been detailed in existing literatures, which will be the major focus of our work in this paper.

In this paper, to effectively prevent information leakage, we propose a novel framework based on the concept of differential privacy (DP), in which each client perturbs its trained parameters locally by purposely adding noises before uploading them to the server for aggregation, namely, noising before model aggregation FL (NbAFL). To the best of authors' knowledge, this is the first piece of work of its kind that theoretically analyzes the convergence property of differentially private FL algorithms. First, we prove that the proposed NbAFL scheme satisfies the requirement of DP in terms of global data under a certain noise perturbation level with Gaussian noises by properly adapting their variances. Then, we develop theoretically a convergence bound of the loss function of the trained FL model in the NbAFL with artificial Gaussian noises. Our developed bound reveals the following three key properties: 1) There is a tradeoff between the convergence performance and privacy protection levels, i.e., a better convergence performance leads to a lower protection level; 2) Increasing the number N of overall clients participating in FL can improve the convergence performance, given a fixed privacy protection level; 3) There is an optimal number of maximum aggregation times in terms of convergence performance for a given protection level. Furthermore, we propose a K-random scheduling strategy, where K ($1 < K < N$) clients are randomly selected from the N overall clients to participate in each aggregation. We also develop the corresponding convergence bound of the loss function in this case. From our analysis, the K-random scheduling strategy can retain the above three properties. Also, we find that there exists an optimal value of K that achieves the best convergence performance at a fixed privacy level. Evaluations demonstrate that our theoretical results are consistent with simulations. Therefore, our analytical results are helpful for the design on privacy-preserving FL architectures with different tradeoff requirements on convergence performance and privacy levels.

The remainder of this paper is organized as follows. In Section II, we introduce backgrounds on FL, DP and a conventional DP-based FL algorithm. In Section III, we detail the proposed NbAFL and analyze the privacy performance based on DP. In Section IV, we analyze the convergence bound of NbAFL and reveal the relationship between privacy levels, convergence performance, the number of clients, and the number of global aggregations. In Section V, we propose the K-random scheduling scheme and develop the convergence bound. We show the analytical results and simulations in Section VI. We conclude the paper in Section VII. A summary of basic concepts and notations is provided in Tab. I.

Table I: Summary of Main Notations

$\mathcal{M}$	A randomized mechanism for DP
x, x'	Adjacent databases
ϵ, δ	The parameters related to DP
$\mathcal{C}_i$	The i-th client
$\mathcal{D}_i$	The database held by the owner $\mathcal{C}_i$
$\mathcal{D}$	The database held by all the clients
$\lvert \cdot \rvert$	The cardinality of a set
N	Total number of all clients
K	The number of chosen clients ($1 < K < N$)
t	The index of the t-th aggregation
T	The number of aggregation times
$\mathbf{w}$	The vector of model parameters
$F(\mathbf{w})$	Global loss function
$F_i(\mathbf{w})$	Local loss function from the i-th client
μ	A presetting constant of the proximal term
$\mathbf{w}_i^{(t)}$	Local uploading parameters of the i-th client
$\mathbf{w}^{(0)}$	Initial parameters of the global model
$\mathbf{w}^{(t)}$	Global parameters generated from all local parameters at the t-th aggregation
$\mathbf{v}^{(t)}$	Global parameters generated from K clients' parameters at the t-th aggregation
$\mathbf{w}^*$	True optimal model parameters that minimize $F(\mathbf{w})$
$\widetilde{\mathbf{W}}$	The set of all local parameters with pertubation

II. Preliminaries

In this section, we will present preliminaries and related background knowledge on FL and DP. Also, we introduce a conventional DP-based FL algorithm that will be discussed in our following analysis as a benchmark.

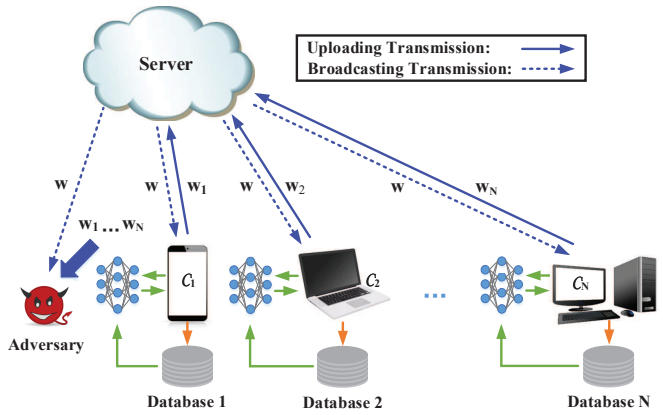

Figure 1: A FL training model with hidden adversaries who can eavesdrop trained parameters from both the clients and the server.

A. Federated Learning

Let us consider a general FL system consisting of one server and N clients, as depicted in Fig. 1. Let $\mathcal{D}_i$ denote the local database held by the client $\mathcal{C}_i$, where $i \in \{1, 2, \ldots, N\}$. At the server, the goal is to learn a model over data that resides at the N associated clients. An active client, participating in the local training, needs to find a vector $\mathbf{w}$ of an AI model to minimize a certain loss function. Formally, the server aggregates the weights sent from the N clients as

$$\mathbf{w} = \sum_{i=1}^{N} p_i \mathbf{w}_i, \tag{1}$$

where $\mathbf{w}_i$ is the parameter vector trained at the i-th client, $\mathbf{w}$ is the parameter vector after aggregating at the server, N is the number of clients, $p_i = \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \geq 0$ with $\sum_{i=1}^{N} p_i = 1$, and $|\mathcal{D}| = \sum_{i=1}^{N} |\mathcal{D}_i|$ is the total size of all data samples. Such an optimization problem can be formulated as

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_{i=1}^{N} p_i F_i(\mathbf{w}), \tag{2}$$

where $F_i(\cdot)$ is the local loss function of the i-th client. Generally, the local loss function $F_i(\cdot)$ is given by local empirical risks. The training process of such a FL system usually contains the following four steps:

- **Step 1:** *Local training:* All active clients locally compute training gradients or parameters and send locally trained ML parameters to the server;
- **Step 2:** *Model aggregating:* The server performs secure aggregation over the uploaded parameters from N clients without learning local information;
- **Step 3:** *Parameters broadcasting:* The server broadcasts the aggregated parameters to the N clients;
- **Step 4:** *Model updating:* All clients update their respective models with the aggregated parameters and test the performance of the updated models.

In the FL process, the N clients with the same data structure collaboratively learn a ML model with the help of a cloud server. After a sufficient number of local training and update exchanges between the server and its associated clients, the solution to the optimization problem (2) is able to converge to that of the global optimal learning model.

B. Threat Model

The server in this paper is assumed to be honest. However, there are external adversaries targeting at clients' private information. Although the individual dataset $\mathcal{D}_i$ of the i-th client is kept locally in FL, the intermediate parameter $\mathbf{w}_i$ needs to be shared with the server, which may reveal the clients' private information as demonstrated by model inversion attacks. For example, authors in [33] demonstrated a model-inversion attack that recovers images from a facial recognition system. In addition, the privacy leakage can also happen in the broadcasting (through downlink channels) phase by analyzing the global parameter $\mathbf{w}$.

We also assume that uplink channels are more secure than downlink broadcasting channels, since clients can be assigned to different channels (e.g., time slots, frequency bands) dynamically in each uploading time, while downlink channels are broadcasting. Hence, we assume that there are at most L ($L \leq T$) exposures of uploaded parameters from each client in the uplink[1] and T exposures of aggregated parameters in the downlink, where T is the number of aggregation times.

C. Differential Privacy

(ϵ, δ)-DP provides a strong criterion for privacy preservation of distributed data processing systems. Here, $\epsilon > 0$ is the distinguishable bound of all outputs on neighboring datasets $\mathcal{D}_i, \mathcal{D}_i'$ in a database, and δ represents the event that the ratio of the probabilities for two adjacent datasets $\mathcal{D}_i, \mathcal{D}_i'$ cannot be bounded by e^{ϵ} after adding a privacy preserving mechanism. With an arbitrarily given δ, a privacy preserving mechanism with a larger ϵ gives a clearer distinguishability of neighboring datasets and hence a higher risk of privacy violation. Now, we will formally define DP as follows.

Definition 1: *((ϵ, δ)-DP [23]): A randomized mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{R}$ with domain $\mathcal{X}$ and range $\mathcal{R}$ satisfies (ϵ, δ)-DP, if for all measurable sets $\mathcal{S} \subseteq \mathcal{R}$ and for any two adjacent databases $\mathcal{D}_i, \mathcal{D}_i' \in \mathcal{X}$,*

$$\Pr[\mathcal{M}(\mathcal{D}_i) \in \mathcal{S}] \leq e^{\epsilon} \Pr[\mathcal{M}(\mathcal{D}_i') \in \mathcal{S}] + \delta. \tag{3}$$

For numerical data, a Gaussian mechanism defined in [23] can be used to guarantee (ϵ, δ)-DP. According to [23], we present the following DP mechanism by adding artificial Gaussian noises.

In order to ensure that the given noise distribution $n \sim \mathcal{N}(0, \sigma^2)$ preserves (ϵ, δ)-DP, where $\mathcal{N}$ represents the Gaussian distribution, we choose noise scale $\sigma \geq c\Delta s/\epsilon$ and the constant $c \geq \sqrt{2\ln(1.25/\delta)}$ for $\epsilon \in (0, 1)$. In this result, n is the value of an additive noise sample for a data in the dateset, Δs is the sensitivity of the function s given by $\Delta s = \max_{\mathcal{D}_i, \mathcal{D}_i'} \|s(\mathcal{D}_i) - s(\mathcal{D}_i')\|$, and s is a real-valued function.

[1]Here we assume that the adversary cannot know where the parameters come from.

Considering the above DP mechanism, choosing an appropriate level of noise remains a significant research problem, which will affect the privacy guarantee of clients and the convergence rate of the FL process.

III. Federated Learning with Differential Privacy

In this section, we first introduce the concept of global DP and analyze the DP performance in the context of FL. Then we propose the NbAFL scheme that can satisfy the DP requirement by adding proper noisy perturbations at both the clients and the server.

A. Global Differential Privacy

Here, we define a global (ϵ, δ)-DP requirement for both uplink and downlink channels. From the uplink perspective, using a clipping technique, we can ensure that $\|\mathbf{w}_i\| \leq C$, where $\mathbf{w}_i$ denotes training parameters from the i-th client without perturbation and C is a clipping threshold for bounding $\mathbf{w}_i$. We assume that the batch size in the local training is equal to the number of training samples and then define local training process in the i-th client by

$$s_{\mathrm{U}}^{\mathcal{D}_i} \triangleq \mathbf{w}_i = \arg\min_{\mathbf{w}} F_i(\mathbf{w}, \mathcal{D}_i) = \frac{1}{|\mathcal{D}_i|} \sum_{j=1}^{|\mathcal{D}_i|} \arg\min_{\mathbf{w}} F_i(\mathbf{w}, \mathcal{D}_{i,j}), \quad (4)$$

where $\mathcal{D}_i$ is the i-th client's database and $\mathcal{D}_{i,j}$ is the j-th sample in $\mathcal{D}_i$. Thus, the sensitivity of $s_{\mathrm{U}}^{\mathcal{D}_i}$ can be expressed as

$$\Delta s_{\mathrm{U}}^{\mathcal{D}_i} = \max_{\mathcal{D}_i, \mathcal{D}_i'} \|s_{\mathrm{U}}^{\mathcal{D}_i} - s_{\mathrm{U}}^{\mathcal{D}_i'}\| = \max_{\mathcal{D}_i, \mathcal{D}_i'} \left\| \frac{1}{|\mathcal{D}_i|} \sum_{j=1}^{|\mathcal{D}_i|} \arg\min_{\mathbf{w}} F_i(\mathbf{w}, \mathcal{D}_{i,j}) - \frac{1}{|\mathcal{D}_i'|} \sum_{j=1}^{|\mathcal{D}_i'|} \arg\min_{\mathbf{w}} F_i(\mathbf{w}, \mathcal{D}_{i,j}') \right\| = \frac{2C}{|\mathcal{D}_i|}, \quad (5)$$

where $\mathcal{D}_i'$ is an adjacent dataset to $\mathcal{D}_i$ which has the same size but only differ by one sample, and $\mathcal{D}_{i,j}'$ is the j-th sample in $\mathcal{D}_i'$. From the above result, a global sensitivity in the uplink channel can be defined by

$$\Delta s_{\mathrm{U}} \triangleq \max\left\{\Delta s_{\mathrm{U}}^{\mathcal{D}_i}\right\}, \quad \forall i. \quad (6)$$

To achieve a small global sensitivity, the ideal condition is that all the clients use sufficient local datasets for training. Hence, we define the minimum size of the local datasets by m and then obtain $\Delta s_{\mathrm{U}} = \frac{2C}{m}$. To ensure (ϵ, δ)-DP for each client in the uplink in one exposure, we set the noise scale, represented by the standard deviation of the additive Gaussian noise, as $\sigma_{\mathrm{U}} = c\Delta s_{\mathrm{U}}/\epsilon$. Considering L exposures of local parameters, we need to set $\sigma_{\mathrm{U}} = cL\Delta s_{\mathrm{U}}/\epsilon$ due to the linear relation between ϵ and σ_{U} in the Gaussian mechanism.

From the downlink perspective, the aggregation operation for $\mathcal{D}_i$ can be expressed as

$$s_{\mathrm{D}}^{\mathcal{D}_i} \triangleq \mathbf{w} = p_1\mathbf{w}_1 + \ldots + p_i\mathbf{w}_i + \ldots + p_N\mathbf{w}_N, \quad (7)$$

where $1 \leq i \leq N$ and $\mathbf{w}$ is the aggregated parameters at the server to be broadcast to the clients. Regarding the sensitivity of $s_{\mathrm{D}}^{\mathcal{D}_i}$, i.e., $\Delta s_{\mathrm{D}}^{\mathcal{D}_i}$, we have the following lemma.

Lemma 1 (Sensitivity after the aggregation operation): *In FL training process, the sensitivity for $\mathcal{D}_i$ after the aggregation operation $s_{\mathrm{D}}^{\mathcal{D}_i}$ is given by*

$$\Delta s_{\mathrm{D}}^{\mathcal{D}_i} = \frac{2Cp_i}{m}. \quad (8)$$

Proof 1: *See Appendix A.*

Algorithm 1: Noising before Aggregation FL

Data: T, $\mathbf{w}^{(0)}$, μ, ϵ and δ
Initialization: $t = 1$ and $\mathbf{w}_i^{(0)} = \mathbf{w}^{(0)}$, $\forall i$
while $t \leq T$ **do**
  Local training process:
  while $\mathcal{C}_i \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_N\}$ **do**
    Update the local parameters $\mathbf{w}_i^{(t)}$ as
    $\mathbf{w}_i^{(t)} = \arg\min_{\mathbf{w}_i} \left(F_i(\mathbf{w}_i) + \frac{\mu}{2}\|\mathbf{w}_i - \mathbf{w}^{(t-1)}\|^2\right)$
    Clip the local parameters
    $\mathbf{w}_i^{(t)} = \mathbf{w}_i^{(t)} / \max\left(1, \frac{\|\mathbf{w}_i^{(t)}\|}{C}\right)$
    Add noise and upload parameters
    $\widetilde{\mathbf{w}}_i^{(t)} = \mathbf{w}_i^{(t)} + \mathbf{n}_i^{(t)}$
  Model aggregating process:
  Update the global parameters $\mathbf{w}^{(t)}$ as
    $\mathbf{w}^{(t)} = \sum_{i=1}^{N} p_i \widetilde{\mathbf{w}}_i^{(t)}$
  The server broadcasts global noised parameters
    $\widetilde{\mathbf{w}}^{(t)} = \mathbf{w}^{(t)} + \mathbf{n}_{\mathrm{D}}^{(t)}$
  Local testing process:
  while $\mathcal{C}_i \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_N\}$ **do**
    Test the aggregating parameters $\widetilde{\mathbf{w}}^{(t)}$ using local dataset
  $t \leftarrow t + 1$
Result: $\widetilde{\mathbf{w}}^{(T)}$

Remark 1: *From the above lemma, to achieve a small global sensitivity in the downlin channel which is defined by*

$$\Delta s_{\mathrm{D}} \triangleq \max\left\{\Delta s_{\mathrm{D}}^{\mathcal{D}_i}\right\} = \max\left\{\frac{2Cp_i}{m}\right\}, \quad \forall i, \quad (9)$$

the ideal condition is that all the clients should use the same size of local datasets for training, i.e., $p_i = 1/N$.

From the above remark, when setting $p_i = 1/N$, $\forall i$, we can obtain the optimal value of the sensitivity Δs_{D}. So here we should add noise at the client side first and then decide whether or not to add noises at server to satisfy the (ϵ, δ)-DP criterion in the downlink channel.

Theorem 1 (DP guarantee for downlink channels): *To ensure (ϵ, δ)-DP in the downlink channels with T aggregations,*

the standard deviation of Gaussian noises $\mathbf{n}_{\mathrm{D}}$ *that are added to the aggregated parameter* $\boldsymbol{w}$ *by the server can be given as*

$$\sigma_{\mathrm{D}} = \begin{cases} \frac{2cC\sqrt{T^2-L^2N}}{mN\epsilon} & T > L\sqrt{N}, \\ 0 & T \leq L\sqrt{N}. \end{cases} \tag{10}$$

Proof 2: *See Appendix B.*

Theorem 1 shows that to satisfy a (ϵ, δ)-DP requirement for the downlink channels, additional noises $\mathbf{n}_{\mathrm{D}}$ need to be added by the server. With a certain L, the standard deviation of additional noises is depending on the relationship between the number of aggregation times T and the number of clients N. The intuition is that a larger T can lead to a higher chance of information leakage, while a larger number of clients is helpful for hiding their private information. This theorem also provides the variance value of the noises that should be added to the aggregated parameters. Based on the above results, we propose the following NbAFL algorithm.

B. Proposed NbAFL

Algorithm 1 outlines our NbAFL for training an effective model with a global (ϵ, δ)-DP requirement. We denote by μ the presetting constant of the proximal term and by $\mathbf{w}^{(0)}$ the initiate global parameter. At the beginning of this algorithm, the server broadcasts the required privacy level parameters (ϵ, δ) are set and the initiate global parameter $\mathbf{w}^{(0)}$ are sent to clients. In the t-th aggregation, N active clients respectively train the parameters by using local databases with preset termination conditions. After completing the local training, the i-th client, $\forall i$, will add noises to the trained parameters $\mathbf{w}_i^{(t)}$, and upload the noised parameters $\widetilde{\mathbf{w}}_i^{(t)}$ to the server for aggregation.

Then the server update the global parameters $\mathbf{w}^{(t)}$ by aggregating the local parameters integrated with different weights. Additive noises $\mathbf{n}_{\mathrm{D}}^{(t)}$ are added to this $\mathbf{w}^{(t)}$ according to **Theorem 1** before being broadcast to the clients. Based on the received global parameters $\widetilde{\mathbf{w}}^{(t)}$, each client will estimate the accuracy by using local testing databases and start the next round of training process based on these received parameters. The FL process completes after the aggregation time reaches a preset number T and the algorithm returns $\widetilde{\mathbf{w}}^{(T)}$.

Now, let us focus on the privacy preservation performance of the NbAFL. First, the set of all local parameters, denoted by $\widetilde{\mathbf{W}} = \{\widetilde{\mathbf{w}}_1, \ldots, \widetilde{\mathbf{w}}_N\}$, are received by the server. Owing to the local perturbations in the NbAFL, it will be difficult for malicious adversaries to infer the information at the i-client from its uploaded parameters $\widetilde{\mathbf{w}}_i$. After the model aggregation, the aggregated parameters $\mathbf{w}$ will be sent back to clients via broadcast channels. This poses threats on clients's privacy as potential adversaries may reveal sensitive information about individual clients from $\mathbf{w}$. In this case, additive noises may be posed to $\mathbf{w}$ based on **Theorem 1**.

IV. Convergence Analysis on NbAFL

In this section, we are ready to analyze the convergence performance of the proposed NbAFL. First, we analyze the expected increment of adjacent aggregations in the loss function with Gaussian noises. Then, we focus on deriving the convergence property under the global (ϵ, δ)-DP requirement.

For the convenience of the analysis, we make the following assumptions on the loss function and network parameters.

Assumption 1: *We make assumptions on the global loss function* $F(\cdot)$ *defined by* $F(\cdot) \triangleq \sum_i^N p_i F_i(\cdot)$*, and the* i*-th local loss function* $F_i(\cdot)$ *as follows:*

1) $F_i(\mathbf{w})$ *is convex;*
2) $F_i(\mathbf{w})$ *satisfies the Polyak-Lojasiewicz condition with the positive parameter* l*, which implies that* $F(\mathbf{w}) - F(\mathbf{w}^*) \leq \frac{1}{2l}\|\nabla F(\mathbf{w})\|^2$*, where* $\mathbf{w}^*$ *is the optimal result;*
3) $F(\mathbf{w}^{(0)}) - F(\mathbf{w}^*) = \Theta$*;*
4) $F_i(\mathbf{w})$ *is* β*-Lipschitz, i.e.,* $\|F_i(\mathbf{w}) - F_i(\mathbf{w}')\| \leq \beta\|\mathbf{w} - \mathbf{w}'\|$*, for any* $\mathbf{w}$*,* $\mathbf{w}'$*;*
5) $F_i(\mathbf{w})$ *is* ρ*-Lipschitz smooth, i.e.,* $\|\nabla F_i(\mathbf{w}) - \nabla F_i(\mathbf{w}')\| \leq \rho\|\mathbf{w} - \mathbf{w}'\|$*, for any* $\mathbf{w}$*,* $\mathbf{w}'$*, where* ρ *is a constant determined by the practical loss function;*
6) *For any* i *and* $\mathbf{w}$*,* $\|\nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w})\| \leq \varepsilon_i$*, where* ε_i *is the divergence metric.*

Similar to the gradient divergence, the divergence metric ε_i is the metric to capture the divergence between the gradients of the local loss functions and that of the aggregated loss function, which is essential for analyzing SGD. The divergence is related to how the data is distributed at different nodes. Using **Assumption 1**, we then have the following lemma.

Lemma 2 (B-dissimilarity of various clients): *For a given ML parameter* $\mathbf{w}$*, there exists* B *satisfying*

$$\mathbb{E}\left\{\|\nabla F_i(\mathbf{w})\|^2\right\} \leq \|\nabla F(\mathbf{w})\|^2 B^2, \quad \forall i. \tag{11}$$

Proof 3: *See Appendix C.*

Lemma 2 comes from the assumption of the divergence metric and demonstrates the statistical heterogeneity of all clients. As mentioned earlier, the values of ρ and $B(\mathbf{w})$ are determined by the specific global loss function $F(\mathbf{w})$ in practice and the training parameters $\mathbf{w}$. With the above preparation, we are now ready to analyze the convergence property of NbAFL. First, we present the following lemma to derive an expected increment bound on the loss function during each iteration of parameters with artificial noises.

Lemma 3 (Expected increment in the loss function): *After receiving updates, from the* t*-th to the* $(t+1)$*-th aggregation, the expected difference in the loss function can be upper-bounded by*

$$\begin{aligned} \mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t+1)}) - F(\widetilde{\mathbf{w}}^{(t)})\} &\leq \lambda_2 \mathbb{E}\{\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|^2\} \\ &+ \lambda_1 \mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|\} + \lambda_0 \mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}, \end{aligned} \tag{12}$$

where

$$\lambda_0 = \frac{\rho}{2}, \ \lambda_1 = \frac{1}{\mu} + \frac{\rho B}{\mu}, \tag{13}$$

$$\lambda_2 = -\frac{1}{\mu} + \frac{\rho B}{\mu^2} + \frac{\rho B^2}{2\mu^2}, \tag{14}$$

and $\mathbf{n}^{(t)}$ *are the equivalent noises imposed on the parameters after the* t*-th aggregation, given by*

$$\mathbf{n}^{(t)} = \sum_{i=1}^{N} p_i \mathbf{n}_i^{(t)} + \mathbf{n}_{\mathrm{D}}^{(t)}. \tag{15}$$

Proof 4: *See Appendix D.*

In this lemma, the value of an additive noise sample n in vector $\mathbf{n}^{(t)}$ satisfies the following Gaussian distribution $n \sim \mathcal{N}(0, \sigma_{\mathrm{A}}^2)$. Also, we can obtain $\sigma_{\mathrm{A}} = \sqrt{\sigma_{\mathrm{D}}^2 + \sigma_{\mathrm{U}}^2/N}$ from Section III. From the right hand side (RHS) of the above inequality, we can see that it is crucial to select a proper proximal term μ to achieve a low upper-bound. It is clear that artificial noises with a large σ_{A} may improve the DP performance in terms privacy protection. However, from the RHS of (12), a large σ_{A} may enlarge the expected difference of the loss function between two consecutive aggregations, leading to a deterioration of convergence performance.

Furthermore, to satisfy the global (ϵ, δ)-DP, by using **Theorem 1**, we have

$$\sigma_{\mathrm{A}} = \begin{cases} \frac{cT\Delta s_{\mathrm{D}}}{\epsilon} & T > L\sqrt{N}, \\ \frac{cL\Delta s_{\mathrm{U}}}{\sqrt{N}\epsilon} & T \leq L\sqrt{N}. \end{cases} \tag{16}$$

Next, we will analyze the convergence property of NbAFL with the (ϵ, δ)-DP requirement.

Theorem 2 (Convergence upper bound of the NbAFL): *With required protection level* ϵ*, the convergence upper bound of* **Algorithm 1** *after* T *aggregations is given by*

$$\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(T)}) - F(\mathbf{w}^*)\} \leq P^T \Theta + \left(\frac{\kappa_1 T}{\epsilon} + \frac{\kappa_0 T^2}{\epsilon^2}\right)\left(1 - P^T\right), \tag{17}$$

where

$$P = 1 + 2l\lambda_2,\ \kappa_1 = \frac{\lambda_1 \beta c}{m(1-P)}\sqrt{\frac{2}{N\pi}} \tag{18}$$

and

$$\kappa_0 = \frac{\lambda_0 c^2}{m^2(1-P)N}. \tag{19}$$

Proof 5: *See Appendix D.*

Theorem 2 reveals an important relationship between privacy and utility by taking into account the protection level ϵ and the number of aggregation times T. As the number of aggregation times T increases, the first term of the upper bound decreases but the second term increases. Furthermore, By viewing T as a continuous variable and by writing the RHS of (17) as $h(T)$, we have

$$\frac{d^2 h(T)}{d^2 T} = \left(\Theta - \frac{\kappa_1 T}{\epsilon} - \frac{\kappa_0 T^2}{\epsilon^2}\right) P^T \ln^2 P - 2\left(\frac{\kappa_1}{\epsilon} + \frac{2\kappa_0 T}{\epsilon^2}\right) P^T \ln P + \frac{2\kappa_0}{\epsilon^2}\left(1 - P^T\right). \tag{20}$$

It can be seen that the second term and third term of on the RHS of (20) are always positive. When N and ϵ are set to be large enough, we can see that κ_1 and κ_0 are small, and thus the first term can also be positive. In this case, we have $d^2 h(T)/d^2 T > 0$ and the upper bound is convex for T.

Remark 2: *As can be seen from this theorem, expected gap between the achieved loss function* $F(\widetilde{\mathbf{w}}^{(T)})$ *and the minimum one* $F(\mathbf{w}^*)$ *is a decreasing function of* ϵ*. By increasing* ϵ*, i.e., relaxing the privacy protection level, the performance of NbAFL algorithm will improve. This is reasonable because the variance of artificial noises decreases, thereby improving the convergence performance.*

Remark 3: *The number of clients* N *will also affect its iterative convergence performance, i.e., a larger* N *would achieve a better convergence performance. This is because a lager* N *leads to a lower variance of the artificial noises.*

Remark 4: *There is an optimal number of maximum aggregation times* T *in terms of convergence performance for given* ϵ *and* N*. In more detail, a larger* T *may lead to a higher variance of artificial noises, and thus pose a negative impact on convergence performance. On the other hand, more iterations can generally boost the convergence performance if noises are not large enough. In this sense, there is a tradeoff on choosing a proper* T*.*

V. K-Client Random Scheduling Policy

In this section, we consider the case where only $K (K < N)$ clients are selected to participate in the aggregation process, namelly K-random scheduling.

We now discuss how to add artificial noises in the K-random scheduling to satisfy a global (ϵ, δ)-DP. It is nature that in the uplink channels, each of the K scheduled clients should add noises with scale $\sigma_{\mathrm{U}} = cL\Delta s_{\mathrm{U}}/\epsilon$ for achieving (ϵ, δ)-DP. This is equivalent to the noise scale in the all-clients selection case in **Section III**, since each client only considers its own privacy for uplink channels in both cases. However, the derivation of the noise scale in the downlink will be different for the K-random scheduling. As an extension of **Theorem 1**, we present the following lemma in the case of K-random scheduling on how to obtain σ_{D}.

Lemma 4 (DP guarantee in K-random scheduling): *In the NbAFL algorithm with* K*-random scheduling, to satisfy a global* (ϵ, δ)*-DP, and the standard deviation* σ_{D} *of additive Gaussian noises for downlink channels should be set as*

$$\sigma_{\mathrm{D}} = \begin{cases} \frac{2cC\sqrt{\frac{T^2}{b^2} - L^2 K}}{mK\epsilon} & T > \frac{\epsilon}{\gamma}, \\ 0 & T \leq \frac{\epsilon}{\gamma}, \end{cases} \tag{21}$$

where

$$b = -\frac{T}{\epsilon}\ln\left(1 - \frac{N}{K} + \frac{N}{K}e^{\frac{-\epsilon}{T}}\right), \quad \gamma = -\ln\left(1 - \frac{K}{N} + \frac{K}{N}e^{\frac{-\epsilon}{L\sqrt{K}}}\right). \tag{22}$$

Proof: See Appendix F. ∎

Lemma 4 recalculates σ_{D} by considering the number of chosen clients K. Generally, the number of clients N is fixed, we thus focus on the effect of K. Based on the DP analysis in **Lemma 4**, we can obtain the following theorem.

Theorem 3 (Convergence under K-random scheduling): *With required protection level ϵ and the number of chosen clients K, for any $\Theta > 0$, the convergence upper bound after T aggregation times is given by*

$$\mathbb{E}\{F(\widetilde{\mathbf{v}}^T) - F(\mathbf{w}^*)\} \leq Q^T\Theta + \frac{1-Q^T}{1-Q}\left(\frac{c\alpha_1\beta}{-mK\ln\left(1-\frac{N}{K}+\frac{N}{K}e^{-\frac{\epsilon}{T}}\right)}\sqrt{\frac{2}{\pi}} + \frac{c^2\alpha_0}{m^2K^2\ln^2\left(1-\frac{N}{K}+\frac{N}{K}e^{-\frac{\epsilon}{T}}\right)}\right). \quad (23)$$

where

$$Q = 1 + \frac{2l}{\mu^2}\left(\frac{\rho B^2}{2} + \rho B + \frac{\rho B^2}{K} + \frac{2\rho B^2}{\sqrt{K}} + \frac{\mu B}{\sqrt{K}} - \mu\right), \quad (24)$$

$$\alpha_0 = \frac{2\rho K}{N} + \rho,\ \alpha_1 = 1 + \frac{2\rho B}{\mu} + \frac{2\rho B\sqrt{K}}{\mu N}, \quad (25)$$

and

$$\widetilde{\mathbf{v}}^{(T)} = \sum_{i=1}^{K} p_i\left(\mathbf{w}_i^{(T)} + \mathbf{n}_i^{(T)}\right) + \mathbf{n}_{\mathrm{D}}^{(T)}. \quad (26)$$

Proof 6: *See Appendix G.*

The above theorem provides the convergence upper bound between $F(\widetilde{\mathbf{v}}^T)$ and $F(\mathbf{w}^*)$ under K-random scheduling. Using K-random scheduling, we can obtain an important relationship between privacy and utility by taking into account the protection level ϵ, the number of aggregation times T and the number of chosen clients K.

Remark 5: *From the bound derived in* **Theorem 3**, *we conclude that there is an optimal K in between 0 and N that achieves the optimal convergence performance. That is, by finding a proper K, the K-random scheduling policy is superior to the one that all N clients participate in the FL aggregations.*

VI. Simulation Results

In this section, we evaluate the proposed NbAFL by using multi-layer perception (MLP) and real-world federated datasets. In order to characterize the convergence property of NbAFL, we conduct experiments by varying the protection levels of ϵ, the number of clients N, the number of maximum aggregation times T and the number of chosen clients K.

We conduct experiments on the standard MNIST dataset for handwritten digit recognition consisting of 60000 training examples and 10000 testing examples [34]. Each example is a 28×28 size gray-level image. Our baseline model uses a a MLP network with a single hidden layer containing 256 hidden units. In this feed-forward neural network, we use a ReLU units and softmax of 10 classes (corresponding to the 10 digits) with the cross-entropy loss function. For the optimizer of networks, we set the learning rate to 0.002. The values of ρ, β, l and B are determined by the specific loss function, and we will use estimated values in our simulations [19].

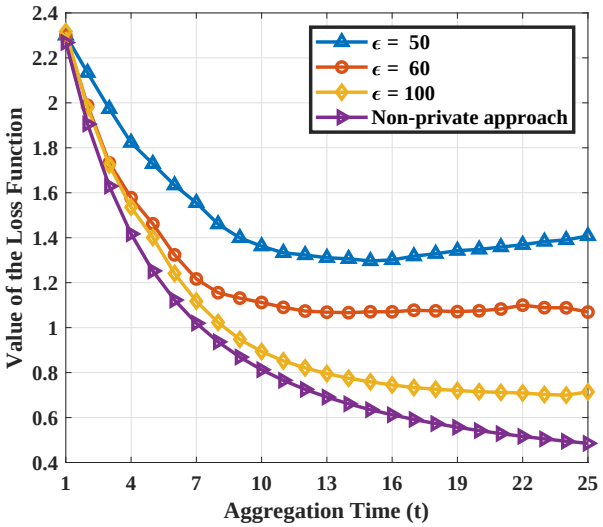

Figure 2: The comparison of training loss with various protection levels for 50 clients using $\epsilon = 50$, $\epsilon = 60$ and $\epsilon = 100$, respectively.

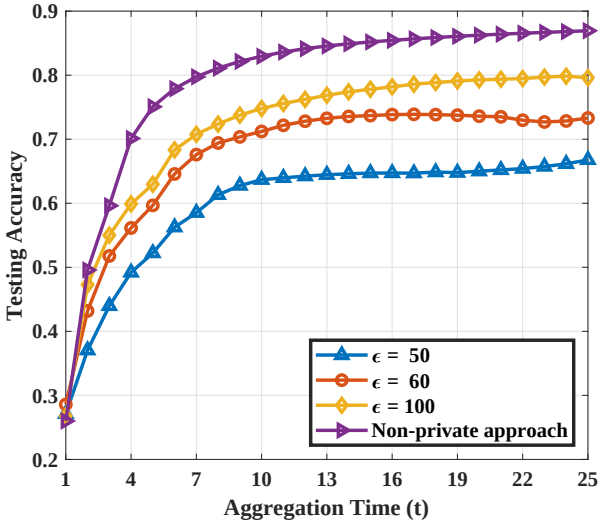

Figure 3: The comparison of training accuracy with various protection levels for 50 clients using $\epsilon = 50$, $\epsilon = 60$ and $\epsilon = 100$, respectively.

A. Performance Evaluation on Protection Levels

In Fig. 2 and Fig. 3, we choose various protection levels $\epsilon = 50$, $\epsilon = 60$ and $\epsilon = 100$ to show the results of the loss function and testing accuracies in NbAFL. Furthermore, we also include a non-private approach to compare with our NbAFL. In this experiment, we set $N = 50$, $T = 25$ and $\delta = 0.01$, and compute the values of the loss function as a function of the aggregation times T. As shown in Fig. 2, values of the loss function in NbAFL are decreasing as we relax the privacy guarantees (increasing ϵ). Meanwhile, in Fig. 3, testing accuracies are also increasing as the privacy parameter reduces. Such observation results are in line with **Remark 2**.

Considering the K-client random scheduling, in Fig. 4 and Fig. 5, we investigate the performances with various protection levels $\epsilon = 50$, $\epsilon = 60$ and $\epsilon = 100$. For simulation parameters, we set $N = 50$, $K = 20$, $T = 25$, and $\delta = 0.01$. As shown

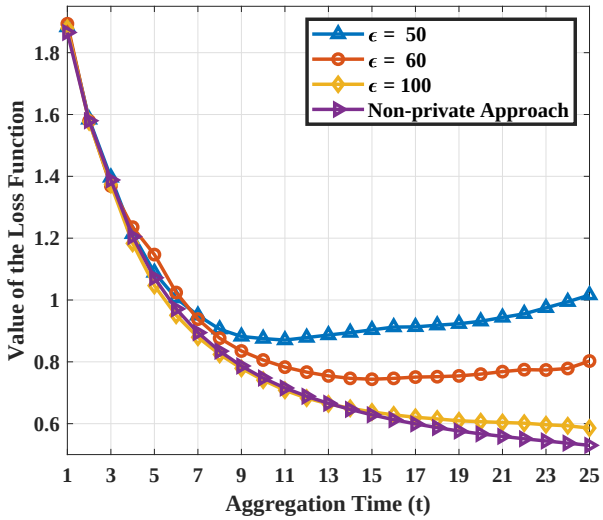

Figure 4: The comparison of training loss with various privacy levels for 50 clients using $\epsilon = 50$, $\epsilon = 60$ and $\epsilon = 100$, respectively.

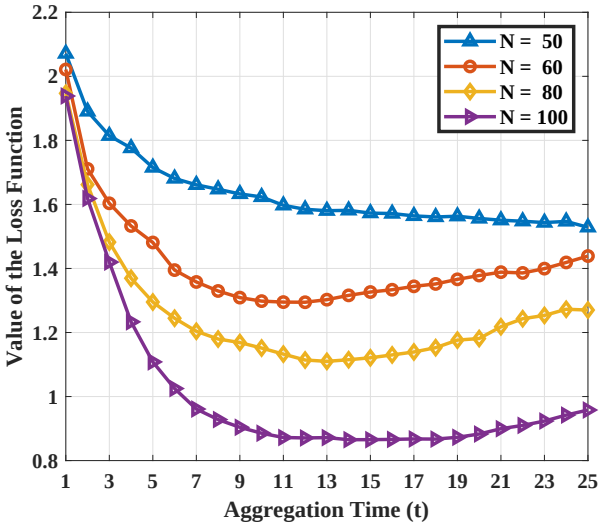

Figure 6: The value of the loss function with various numbers of clients under $\epsilon = 60$ under NbAFL Algorithm with 50 clients.

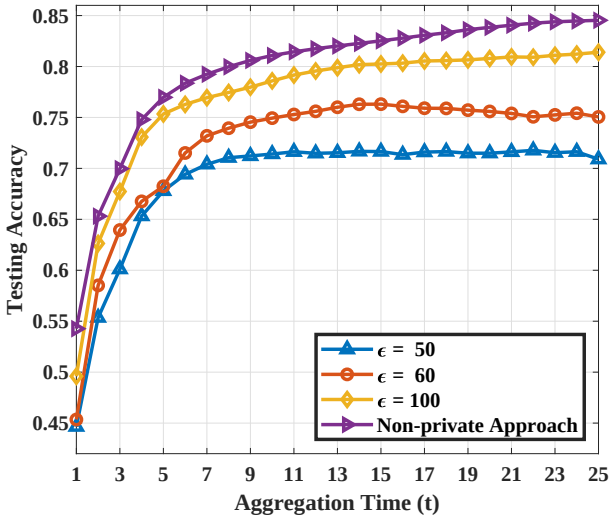

Figure 5: The comparison of training accuracy with various privacy levels for 50 clients using $\epsilon = 50$, $\epsilon = 60$ and $\epsilon = 100$, respectively.

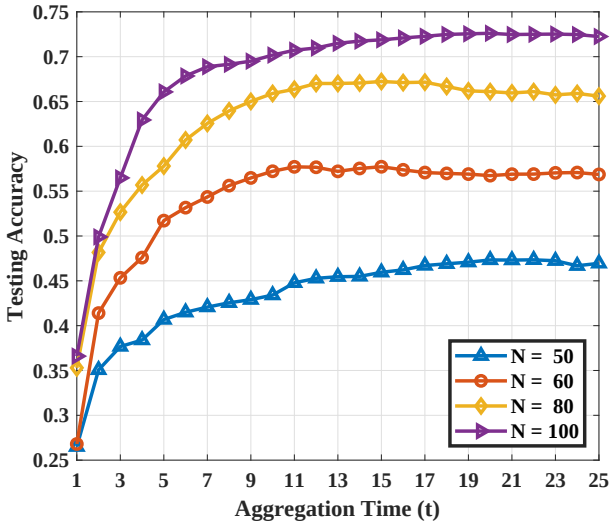

Figure 7: The value of the loss function with various numbers of clients under $\epsilon = 60$ under NbAFL Algorithm with 50 clients.

in Fig. 4 and Fig. 5, the convergence performance under the K-client random scheduling is improved with an increasing ϵ.

B. Impact of the number of clients N

Fig. 6 and Fig. 7 compare the convergence performance of NbAFL under required protection level $\epsilon = 60$ and $\delta = 10^{-2}$ as a function of clients' number, N. In this experiment, we set $N = 50$, $N = 60$, $N = 80$ and $N = 100$. We notice that the performance among different numbers of clients is governed by **Remark 3**. This is because that more clients not only provide larger global datasets for training, but also bring down the of standard deviation additive noises due to the aggregation.

C. Impact of the number of maximum aggregation times T

In Fig. 8, we show the theoretical upper bound of training loss as a function of maximum aggregation times with various privacy levels $\epsilon = 50$, 60 and 100 under NbAFL algorithm. Fig. 9 compares the theoretical upper bound using the dotted line and experimental results using the solid line with $\epsilon = 60$ and 100. Fig. 8 and Fig. 9 reveal that under a low privacy level (a large ϵ), NbAFL gives a large improvement in terms of the convergence performance. This observation is in line with **Remark 4**, and the reason comes from the fact that a lower privacy level decreases the standard deviation of additive noises and the server can obtain better quality ML model parameters from the clients. Fig. 8 also implies that an optimal number of maximum aggregation times increases almost with respect to the increasing ϵ.

Fig. 10 compares the normal NbAFL and K-random

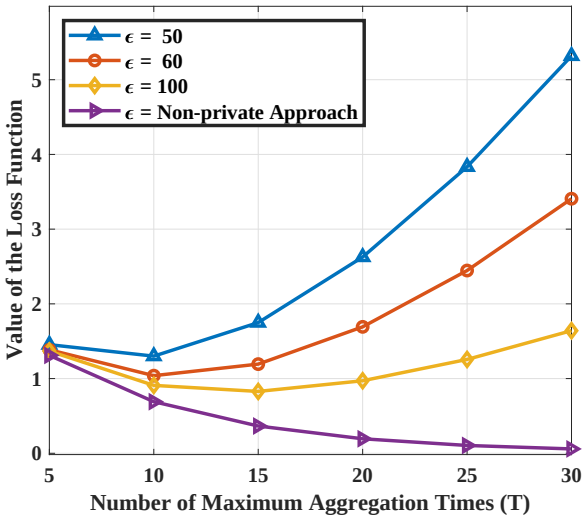

Figure 8: The convergence upper bounds with various privacy levels $\epsilon = 50$, 60 and 100 under 50-clients' NbAFL algorithm.

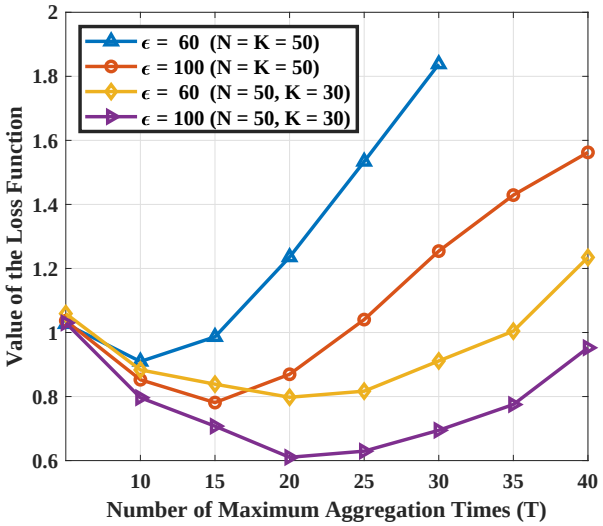

Figure 10: The value of the loss function with various privacy levels $\epsilon = 60$ and $\epsilon = 80$ under NbAFL Algorithm with 50 clients.

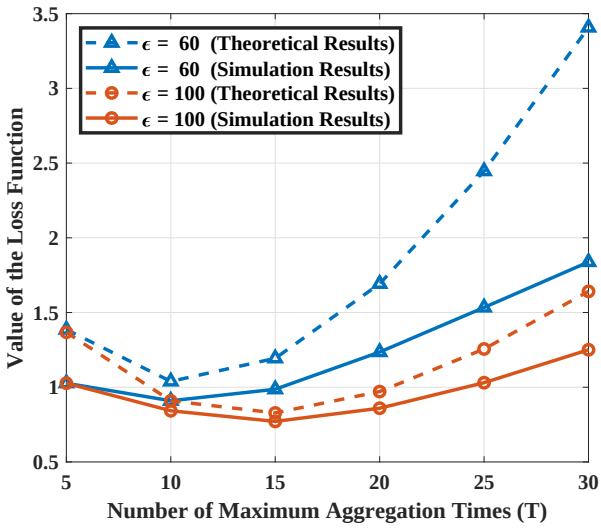

Figure 9: The comparison of the loss function between experimental and theoretical results with the various aggregation times under NbAFL Algorithm with 50 clients.

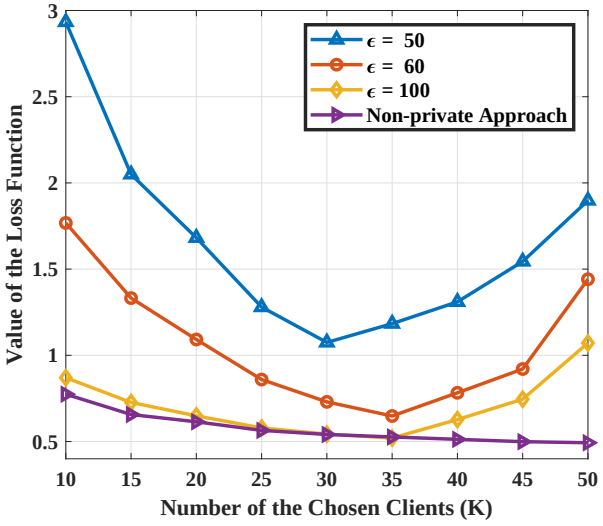

Figure 11: The value of the loss function with various numbers of chosen clients under $\epsilon = 50, 60, 100$ under NbAFL Algorithm and non-private approach with 50 clients.

scheduling based NbAFL for a given protection level. In Fig. 10, we plot the values of the loss function in NbAFL with various numbers of maximum aggregation times. This figure shows that the value of loss function is a convex function of maximum aggregation times for a given protection leavel under NbAFL algorithm, which validates **Remark 4**. From Fig. 10, we can also see that for a given ϵ, K-random scheduling based NbAFL algorithm has a better convergence performance than the normalized NbAFL algorithm for a larger T. This is because that K-random scheduling can bring down the variance of artificial noises with little performance loss.

D. Impact of the number of chosen clients K

In Fig. 11, we plot values of the loss function with various numbers of chosen clients K under the random scheduling policy in NbAFL. The number of clients is $N = 50$, and K clients are randomly chosen to participate in training and aggregation in each iteration. In this experiment, we set $\epsilon = 50$, $\epsilon = 60$, $\epsilon = 80$ and $\delta = 0.01$. Meanwhile, we also exhibit the performance of the non-private approach with various numbers of chosen clients K. Note that an optimal K which further improves the convergence performance exists for various protection levels, due to a trade-off between enhance privacy protection and involving larger global training datasets in each model updating round. This observation is in line with **Remark 5**. The figure shows that in NbAFL, for a given protection level ϵ, the K-random scheduling can obtain a better tradeoff than the normal selection policy.

VII. Conclusions

In this paper, we have focused on differential attacks in SGD based FL. We first define a global (ϵ, δ)-DP requirement for

both uplink and downlink channels, and develop variances of artificial noises at clients and server sides. Then, we propose a novel framework based on the concept of global (ϵ, δ)-DP, named NbAFL. We develop theoretically a convergence bound of the loss function of the trained FL model in the NbAFL. From theoretical convergence bounds, we obatin the following results: 1) There is a tradeoff between the convergence performance and privacy protection levels, i.e., a better convergence performance leads to a lower protection level; 2) Increasing the number N of overall clients participating in FL can improve the convergence performance, given a fixed privacy protection level; 3) There is an optimal number of maximum aggregation times in terms of convergence performance for a given protection level. Furthermore, we propose a K-random scheduling strategy and also develop the corresponding convergence bound of the loss function in this case. In addition to above three properties. we find that there exists an optimal value of K that achieves the best convergence performance at a fixed privacy level. Extensive simulation results confirm the correctness of our analysis. Therefore, our analytical results are helpful for the design on privacy-preserving FL architectures with different tradeoff requirements on convergence performance and privacy levels.

Appendix A
Proof of Lemma 1

From the downlink perspective, for all $\mathcal{D}_i$ and $\mathcal{D}_i'$ which differ in a signal entry, the sensitivity can be expressed as

$$\Delta s_{\mathrm{D}}^{\mathcal{D}_i} = \max_{\mathcal{D}_i, \mathcal{D}_i'} \| s_{\mathrm{D}}^{\mathcal{D}_i} - s_{\mathrm{D}}^{\mathcal{D}_i'} \|. \tag{27}$$

Based on (4) and (7), we have

$$s_{\mathrm{D}}^{\mathcal{D}_i} = p_1 \mathbf{w}_1(\mathcal{D}_1) + \ldots + p_i \mathbf{w}_i(\mathcal{D}_i) + \ldots + p_N \mathbf{w}_N(\mathcal{D}_N) \tag{28}$$

and

$$s_{\mathrm{D}}^{\mathcal{D}_i'} = p_1 \mathbf{w}_1(\mathcal{D}_1) + \ldots + p_i \mathbf{w}_i(\mathcal{D}_i') + \ldots + p_N \mathbf{w}_N(\mathcal{D}_N), \tag{29}$$

Furthermore, the sensitivity can be given as

$$\begin{aligned} \Delta s_{\mathrm{D}}^{\mathcal{D}_i} &= \max_{\mathcal{D}_i, \mathcal{D}_i'} \| p_i \mathbf{w}_i(\mathcal{D}_i) - p_i \mathbf{w}_i(\mathcal{D}_i') \| \\ & p_i \max_{\mathcal{D}_i, \mathcal{D}_i'} \| \mathbf{w}_i(\mathcal{D}_i) - \mathbf{w}_i(\mathcal{D}_i') \| = p_i \Delta s_{\mathrm{U}}^{\mathcal{D}_i} \leq \frac{2 C p_i}{m}. \end{aligned} \tag{30}$$

Hence, we can set $\Delta s_{\mathrm{D}}^{\mathcal{D}_i} = \frac{2 C p_i}{m}$. This completes the proof. □

Appendix B
Proof of Theorem 1

To ensure a global (ϵ, δ)-DP in the uplink channels, the standard deviation of additive noises in client sides can be set to $\sigma_{\mathrm{U}} = cL\Delta s_{\mathrm{U}}/\epsilon$ due to the linear relation between ϵ and σ_{U} with Gaussian mechanism, where $\Delta s_{\mathrm{U}} = \frac{2C}{m}$ is the sensitivity for the aggregation operation and m is the data size of each client. We then set the sample in the i-th local noise vector to a same distribution $n_i \sim \varphi(n)$ (i.i.d for all i) because each client is coincident with the same global (ϵ, δ)-DP. The aggregation process with artificial noises added by clients can be expressed as

$$\widetilde{\mathbf{w}} = \sum_{i=1}^{N} p_i \left(\mathbf{w}_i + \mathbf{n}_i \right) = \sum_{i=1}^{N} p_i \mathbf{w}_i + \sum_{i=1}^{N} p_i \mathbf{n}_i. \tag{31}$$

The distribution $\phi_N(n)$ of $\sum_{i=1}^{N} p_i n_i$ can be expressed as

$$\phi_N(n) = \bigotimes_{i=1}^{N} \varphi_i(n), \tag{32}$$

where $p_i n_i \sim \varphi_i(n)$, and $\bigotimes$ is convolutional operation.

When we use Gaussian mechanism for n_i with noise scale σ_{U}, the distribution of $p_i n_i$ is also Gaussian distribution. To obtain a small sensitivity Δs_{D}, we set $p_i = 1/N$. Furthermore, the noise scale $\sigma_{\mathrm{U}}/\sqrt{N}$ of the Gaussian distribution $\phi_N(n)$ can be calculated. To ensure a global (ϵ, δ)-DP in downlink channels, we know the standard deviation of additive noises can be set to $\sigma_{\mathrm{A}} = cT\Delta s_{\mathrm{D}}/\epsilon$, where $\Delta s_{\mathrm{D}} = 2C/mN$. Hence, we can obtain the standard deviation of additive noises at the server as

$$\sigma_{\mathrm{D}} = \sqrt{\sigma_{\mathrm{A}}^2 - \frac{\sigma_{\mathrm{U}}^2}{N}} = \begin{cases} \frac{2cC\sqrt{T^2 - L^2 N}}{mN\epsilon} & T > L\sqrt{N}, \\ 0 & T \leq L\sqrt{N}. \end{cases} \tag{33}$$

Hence, **Theorem 1** has been proved. □

Appendix C
Proof of Lemma 2

Due to **Assumption 1**, we have

$$\mathbb{E}\left\{ \| \nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w}) \|^2 \right\} \leq \mathbb{E}\{\varepsilon_i^2\} \tag{34}$$

and

$$\begin{aligned} &\mathbb{E}\left\{ \| \nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w}) \|^2 \right\} \\ &\quad = \mathbb{E}\left\{ \| \nabla F_i(\mathbf{w}) \|^2 \right\} - 2\mathbb{E}\left\{ \nabla F_i(\mathbf{w})^{\top} \right\} \nabla F(\mathbf{w}) \\ &\quad + \| \nabla F(\mathbf{w}) \|^2 = \mathbb{E}\left\{ \| \nabla F_i(\mathbf{w}) \|^2 \right\} - \| \nabla F(\mathbf{w}) \|^2. \end{aligned} \tag{35}$$

Considering (34), (35) and $\nabla F(\mathbf{w}) = \mathbb{E}\{\nabla F_i(\mathbf{w})\}$, we have

$$\begin{aligned} \mathbb{E}\left\{ \| \nabla F_i(\mathbf{w}) \|^2 \right\} &\leq \| \nabla F(\mathbf{w}) \|^2 + \mathbb{E}\{\varepsilon_i^2\} \\ &= \| \nabla F(\mathbf{w}) \|^2 B(\mathbf{w})^2. \end{aligned} \tag{36}$$

Note that when $\| \nabla F(\mathbf{w}) \|^2 \neq 0$, there exists

$$B(\mathbf{w}) = \sqrt{1 + \frac{\mathbb{E}\{\varepsilon_i^2\}}{\| \nabla F(\mathbf{w}) \|^2}} \geq 1, \tag{37}$$

which satisfies the equation. We can notice that a smaller value of $B(\mathbf{w})$ implies that the local loss functions are more locally similar. When all the local loss functions are the same, then $B(\mathbf{w}) = 1$, for all $\mathbf{w}$. Therefore, we can have

$$\mathbb{E}\left\{ \| \nabla F_i(\mathbf{w}) \|^2 \right\} \leq \| \nabla F(\mathbf{w}) \|^2 B^2, \quad \forall i, \tag{38}$$

where B is the upper bound of $B(\mathbf{w})$. This completes the proof. □

APPENDIX D
PROOF OF LEMMA 3

Considering the aggregation process with artificial noises added by clients and the server in the $(t+1)$-th aggregation, we have

$$\widetilde{\mathbf{w}}^{(t+1)} = \sum_{i=1}^{N} p_i \mathbf{w}_i^{(t+1)} + \mathbf{n}^{(t+1)}, \tag{39}$$

where

$$\mathbf{n}^{(t)} = \sum_{i=1}^{N} p_i \mathbf{n}_i^{(t)} + \mathbf{n}_{\mathrm{D}}^{(t)}. \tag{40}$$

Because $F_i(\cdot)$ is ρ-Lipschitz smooth, we know

$$F_i(\widetilde{\mathbf{w}}^{(t+1)}) \leq F_i(\widetilde{\mathbf{w}}^{(t)}) + \nabla F_i(\widetilde{\mathbf{w}}^{(t)})^\top (\widetilde{\mathbf{w}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}) + \frac{\rho}{2}\|\widetilde{\mathbf{w}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|^2, \tag{41}$$

for all $\widetilde{\mathbf{w}}^{(t+1)}$, $\widetilde{\mathbf{w}}^{(t)}$. Combining $F(\widetilde{\mathbf{w}}^{(t)}) = \mathbb{E}\{F_i(\widetilde{\mathbf{w}}^{(t)})\}$ and $\nabla F(\widetilde{\mathbf{w}}^{(t)}) = \mathbb{E}\{\nabla F_i(\widetilde{\mathbf{w}}^{(t)})\}$, we have

$$\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t+1)}) - F(\widetilde{\mathbf{w}}^{(t)})\} \leq \mathbb{E}\{\nabla F(\widetilde{\mathbf{w}}^{(t)})^\top (\widetilde{\mathbf{w}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)})\} + \frac{\rho}{2}\mathbb{E}\{\|\widetilde{\mathbf{w}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|^2\}. \tag{42}$$

We define

$$J(\mathbf{w}_i^{(t+1)}; \widetilde{\mathbf{w}}^{(t)}) \triangleq F_i(\mathbf{w}_i^{(t+1)}) + \frac{\mu}{2}\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|^2. \tag{43}$$

Then, we know

$$\nabla J(\mathbf{w}_i^{(t+1)}; \widetilde{\mathbf{w}}^{(t)}) = \nabla F_i(\mathbf{w}_i^{(t+1)}) + \mu\left(\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\right) \tag{44}$$

and

$$\begin{aligned}\widetilde{\mathbf{w}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)} &= \sum_{i=1}^{N}\left(\mathbf{w}_i^{(t+1)} + \mathbf{n}_i^{(t+1)}\right) + \mathbf{n}_{\mathrm{D}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)} \\ &= \frac{1}{\mu}\mathbb{E}\{\nabla J(\mathbf{w}_i^{(t+1)}; \widetilde{\mathbf{w}}^{(t)}) - \nabla F_i(\mathbf{w}_i^{(t+1)})\} + \mathbf{n}^{(t+1)}.\end{aligned} \tag{45}$$

Because $F_i(\cdot)$ is ρ-Lipschitz smooth, we can obtain

$$\begin{aligned}\mathbb{E}\{\nabla F_i(\mathbf{w}_i^{(t+1)})\} &\leq \mathbb{E}\{\nabla F_i(\widetilde{\mathbf{w}}^{(t)}) + \rho\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|\} \\ &= \nabla F(\widetilde{\mathbf{w}}^{(t)}) + \rho\mathbb{E}\{\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|\}.\end{aligned} \tag{46}$$

Now, let us bound $\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|$. We know

$$\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\| \leq \|\mathbf{w}_i^{(t+1)} - \widehat{\mathbf{w}}_i^{(t+1)}\| + \|\widehat{\mathbf{w}}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|, \tag{47}$$

where $\widehat{\mathbf{w}}_i^{(t+1)} = \arg\min_{\mathbf{w}} J_i(\mathbf{w}; \widetilde{\mathbf{w}}^{(t)})$. Let us define $\overline{\mu} = \mu - \rho > 0$, then we know $J_i(\mathbf{w}; \widetilde{\mathbf{w}}^{(t)})$ is $\overline{\mu}$-convexity. Based on this, we can obtain

$$\|\widehat{\mathbf{w}}_i^{(t+1)} - \mathbf{w}_i^{(t+1)}\| \leq \frac{\theta}{\mu}\|\nabla F_i(\widetilde{\mathbf{w}}^{(t)})\| \tag{48}$$

and

$$\|\widehat{\mathbf{w}}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\| \leq \frac{1}{\overline{\mu}}\|\nabla F_i(\widetilde{\mathbf{w}}^{(t)})\|, \tag{49}$$

where θ denotes a θ solution of $\min_{\mathbf{w}} J_i(\mathbf{w}; \widetilde{\mathbf{w}}^{(t)})$ [18]. Now, we can use the inequality (48) and (49) to obtain

$$\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\| \leq \frac{1+\theta}{\overline{\mu}}\|\nabla F_i(\widetilde{\mathbf{w}}^{(t)})\|. \tag{50}$$

Therefore,

$$\begin{aligned}\|\widetilde{\mathbf{w}}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\| &\leq \|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\| + \|\mathbf{n}^{(t+1)}\| \\ &\leq \mathbb{E}\{\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|\} + \|\mathbf{n}^{(t+1)}\| \\ &\leq \frac{1+\theta}{\overline{\mu}}\mathbb{E}\{\|\nabla F_i(\widetilde{\mathbf{w}}^{(t)})\|\} + \|\mathbf{n}^{(t+1)}\| \\ &\leq \frac{B(1+\theta)}{\overline{\mu}}\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\| + \|\mathbf{n}^{(t+1)}\|.\end{aligned} \tag{51}$$

Using (46) and (47), we know

$$\begin{aligned}&\|\mathbb{E}\{\nabla F_i(\mathbf{w}_i^{(t+1)})\} - \nabla F(\widetilde{\mathbf{w}}^{(t)}) - \mathbb{E}\{\nabla J(\mathbf{w}_i^{(t+1)}; \widetilde{\mathbf{w}}^{(t)})\}\| \\ &\quad\leq \rho\mathbb{E}\{\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{w}}^{(t)}\|\} + \mathbb{E}\{\nabla J(\mathbf{w}_i^{(t+1)}; \widetilde{\mathbf{w}}^{(t)})\} \\ &\quad\leq \frac{\rho B(1+\theta)}{\overline{\mu}}\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\| + B\theta\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|.\end{aligned} \tag{52}$$

Substituting (46), (51) and (52) into (42), we know

$$\begin{aligned}&\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t+1)}) - F(\widetilde{\mathbf{w}}^{(t)})\} \\ &\quad\leq \mathbb{E}\left\{\nabla F(\widetilde{\mathbf{w}}^{(t)})^\top\left(-\frac{1}{\mu}\nabla F(\widetilde{\mathbf{w}}^{(t)}) + \frac{1}{\mu}\mathbf{n}^{(t+1)} + \left(\frac{\rho B(1+\theta)}{\mu\overline{\mu}} + \frac{B\theta}{\mu}\right)\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|\right)\right\} \\ &\quad+ \frac{\rho}{2}\mathbb{E}\left\{\left[\frac{B(1+\theta)}{\overline{\mu}}\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\| + \|\mathbf{n}^{(t+1)}\|\right]^2\right\}.\end{aligned} \tag{53}$$

Then, using triangle inequation, we can obtain

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t+1)}) - F(\widetilde{\mathbf{w}}^{(t)})\} &\leq \lambda_2\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|^2 \\ &+ \lambda_1\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\}\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\| + \lambda_0\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}.\end{aligned} \tag{54}$$

where

$$\lambda_2 = -\frac{1}{\mu} + \frac{B}{\mu}\left[\frac{\rho(1+\theta)}{\overline{\mu}} + \theta\right] + \frac{\rho B^2(1+\theta)^2}{2\overline{\mu}^2}, \tag{55}$$

$$\lambda_1 = \frac{1}{\mu} + \frac{\rho B(1+\theta)}{\overline{\mu}} \text{ and } \lambda_0 = \frac{\rho}{2}. \tag{56}$$

In this convex case, where $\overline{\mu} = \mu$, if $\theta = 0$, all subproblems are solved accurately. We know $\lambda_2 = -\frac{1}{\mu} + \frac{\rho B}{\mu^2} + \frac{\rho B^2}{2\mu^2}$, $\lambda_1 = \frac{1}{\mu} + \frac{\rho B}{\mu}$ and $\lambda_0 = \frac{\rho}{2}$. This completes the proof. □

APPENDIX E
PROOF OF THEOREM 2

We assume that F satisfies the Polyak-Lojasiewicz inequality [35] with positive parameter l, which implies that

$$\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t)}) - F(\mathbf{w}^*)\} \leq \frac{1}{2l}\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|^2. \tag{57}$$

Moreover, subtract $\mathbb{E}\{F(\mathbf{w}^*)\}$ in both sides of (54), we know

$$\begin{aligned}&\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t+1)}) - F(\mathbf{w}^*)\} \\ &\quad\leq \mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t)}) - F(\mathbf{w}^*)\} + \lambda_2\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\|^2 \\ &\quad+ \lambda_1\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\}\|\nabla F(\widetilde{\mathbf{w}}^{(t)})\| + \lambda_0\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}.\end{aligned} \tag{58}$$

Considering $\|\nabla F(\mathbf{w}^{(t)})\| \leq \beta$ and (57), we have

$$\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t+1)}) - F(\mathbf{w}^*)\} \leq (1+2l\lambda_2)\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(t)}) - F(\mathbf{w}^*)\} + \lambda_1\beta\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\} + \lambda_0\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}, \quad (59)$$

where $F(\mathbf{w}^*)$ is the loss function corresponding to the optimal parameters $\mathbf{w}^*$. Considering the same and independent distribution of additive noises, we define $\mathbb{E}\{\|\mathbf{n}^{(t)}\|\} = \mathbb{E}\{\|\mathbf{n}\|\}$ and $\mathbb{E}\{\|\mathbf{n}^{(t)}\|^2\} = \mathbb{E}\{\|\mathbf{n}\|^2\}$, for $0 \leq t \leq T$. Applying (59) recursively, we have

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(T)}) - F(\mathbf{w}^*)\} &\leq (1+2l\lambda_2)^T\mathbb{E}\{F(\mathbf{w}^{(0)}) - F(\mathbf{w}^*)\} \\ &+ \left(\lambda_1\beta\mathbb{E}\{\|\mathbf{n}\|\} + \lambda_0\mathbb{E}\{\|\mathbf{n}\|^2\}\right)\sum_{t=0}^{T-1}(1+2l\lambda_2)^t \\ &= (1+2l\lambda_2)^T\mathbb{E}\{F(\mathbf{w}^{(0)}) - F(\mathbf{w}^*)\} \\ &+ \left(\lambda_1\beta\mathbb{E}\{\|\mathbf{n}\|\} + \lambda_0\mathbb{E}\{\|\mathbf{n}\|^2\}\right)\frac{(1+2l\lambda_2)^T - 1}{2l\lambda_2}. \quad (60)\end{aligned}$$

If $T \leq L\sqrt{N}$ and then $\sigma_{\mathrm{D}} = 0$, this case is special. Hence, we will consider the condition that $T > L\sqrt{N}$. Based on (16), we have $\sigma_{\mathrm{A}} = \Delta s_{\mathrm{D}} Tc/\epsilon$. Hence, we can obtain

$$\mathbb{E}\{\|\mathbf{n}\|\} = \frac{\Delta s_{\mathrm{D}} Tc}{\epsilon}\sqrt{\frac{2N}{\pi}} \text{ and } \mathbb{E}\{\|\mathbf{n}\|^2\} = \frac{\Delta s_{\mathrm{D}}^2 T^2 c^2 N}{\epsilon^2}. \quad (61)$$

Substituting (100) into (60), setting $\Delta s_{\mathrm{D}} = 1/N$ and $F(\mathbf{w}^{(0)}) - F(\mathbf{w}^*) = \Theta$, we have

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(T)}) - F(\mathbf{w}^*)\} &\leq (1+2l\lambda_2)^T\Theta \\ &+ \left(\frac{\lambda_1 T\beta c}{\epsilon}\sqrt{\frac{2}{N\pi}} + \frac{\lambda_0 T^2 c^2}{\epsilon^2 N}\right)\frac{(1+2l\lambda_2)^T - 1}{2l\lambda_2} \quad (62) \\ &= P^T\Theta + \left(\frac{\kappa_1 T}{\epsilon} + \frac{\kappa_0 T^2}{\epsilon^2}\right)\left(1 - P^T\right),\end{aligned}$$

where $P = 1 + 2l\lambda_2$, $\kappa_1 = \frac{\lambda_1\beta c}{m(P-1)}\sqrt{\frac{2}{N\pi}}$ and $\kappa_0 = \frac{\lambda_0 c^2}{m^2(P-1)N}$. This completes the proof. □

Appendix F
Proof of Lemma 4

We define the sampling parameter $q \triangleq K/N$ to represent the probability of being selected by the server for each client in an aggregation. Let $\mathcal{M}_{1:T}$ denote $(\mathcal{M}_1, \ldots, \mathcal{M}_T)$ and similarly let $o_{1:T}$ denote a sequence of outcomes $(o_1, \ldots, o_T)$. Considering a global (ϵ, δ)-DP in the downlinks channels, we use σ_{A} to represent the standard deviation of aggregated Gaussian noises. With neighboring datasets $\mathcal{D}_i$ and $\mathcal{D}_i'$, we are looking at

$$\begin{aligned}&\left|\ln\frac{\Pr[\mathcal{M}_{1:T}(\mathcal{D}'_{i,1:T}) = o_{1:T}]}{\Pr[\mathcal{M}_{1:T}(\mathcal{D}_{i,1:T}) = o_{1:T}]}\right| \\ &= \left|\sum_{i=1}^{T}\ln\frac{(1-q)e^{-\frac{\|n\|^2}{2\sigma_{\mathrm{A}}^2}} + qe^{-\frac{\|n+\Delta s_{\mathrm{D}}\|^2}{2\sigma_{\mathrm{A}}^2}}}{e^{-\frac{\|n\|^2}{2\sigma_{\mathrm{A}}^2}}}\right| \\ &= \left|\sum_{i=1}^{T}\ln\left(1 - q + qe^{-\frac{2n\Delta s_{\mathrm{D}} + \Delta s_{\mathrm{D}}^2}{2\sigma_{\mathrm{A}}^2}}\right)\right| \\ &= \left|\ln\prod_{i=1}^{T}\left(1 - q + qe^{-\frac{2n\Delta s_{\mathrm{D}} + \Delta s_{\mathrm{D}}^2}{2\sigma_{\mathrm{A}}^2}}\right)\right|.\end{aligned} \quad (63)$$

This quantity is bounded by ϵ, we require

$$\left|\ln\frac{\Pr[\mathcal{M}_{1:T}(\mathcal{D}'_{i,1:T}) = o_{1:T}]}{\Pr[\mathcal{M}_{1:T}(\mathcal{D}_{i,1:T}) = o_{1:T}]}\right| \leq \epsilon. \quad (64)$$

Considering the independence of adding noises, we know

$$T\ln\left(1 - q + qe^{-\frac{2n\Delta s_{\mathrm{D}} + \|\Delta s_{\mathrm{D}}\|^2}{2\sigma_{\mathrm{A}}^2}}\right) \geq -\epsilon. \quad (65)$$

We can obtain the result

$$n \leq -\frac{\sigma_{\mathrm{A}}^2}{\Delta s_{\mathrm{D}}}\ln\left(\frac{\exp(-\frac{\epsilon}{T})}{q} - \frac{1}{q} + 1\right) - \frac{\Delta s_{\mathrm{D}}}{2}. \quad (66)$$

We set

$$b = -\frac{T}{\epsilon}\ln\left(\frac{\exp(-\epsilon/T) - 1}{q} + 1\right). \quad (67)$$

Hence,

$$\ln\left(\frac{\exp(-\epsilon/T) - 1}{q} + 1\right) = -\frac{b\epsilon}{T}. \quad (68)$$

Note that ϵ and T should satisfy

$$\epsilon < -T\ln(1-q) \text{ or } T > \frac{-\epsilon}{\ln(1-q)}. \quad (69)$$

Then,

$$n \leq \frac{\sigma_{\mathrm{A}}^2 b\epsilon}{T\Delta s_{\mathrm{D}}} - \frac{\Delta s_{\mathrm{D}}}{2}. \quad (70)$$

Using the tail bound $\Pr[n > \eta] \leq \frac{\sigma_{\mathrm{A}}}{\sqrt{2\pi}}\frac{1}{\eta}e^{-\eta^2/2\sigma_{\mathrm{A}}^2}$, we can obtain

$$\ln\left(\frac{\eta}{\sigma_{\mathrm{A}}}\right) + \frac{\eta^2}{2\sigma_{\mathrm{A}}^2} > \ln\left(\sqrt{\frac{2}{\pi}}\frac{1}{\delta}\right). \quad (71)$$

Let us set $\sigma_{\mathrm{A}} = c\Delta s_{\mathrm{D}} T/b\epsilon$, if $b\epsilon/T \in (0,1)$, the inequation (71) can be solved as

$$c^2 \geq 2\ln\left(\frac{1.25}{\delta}\right). \quad (72)$$

Meanwhile, ϵ and T should satisfy

$$\epsilon < -T\ln\left(1 - q + \frac{q}{e}\right) \text{ or } T > \frac{-\epsilon}{\ln\left(1 - q + \frac{q}{e}\right)}. \quad (73)$$

If $b\epsilon/T > 1$, we can also obtain $\sigma_{\mathrm{A}} = c\Delta s_{\mathrm{D}} T/b\epsilon$ by adjusting the value of c. The standard deviation of requiring noises is given as

$$\sigma_{\mathrm{A}} \geq \frac{c\Delta s_{\mathrm{D}} T}{b\epsilon}. \quad (74)$$

Hence, if Gaussian noises are added at the client sides, we can obtain the additive noise scale in the server as

$$\sigma_{\mathrm{D}} = \sqrt{\left(\frac{c\Delta s_{\mathrm{D}} T}{b\epsilon}\right)^2 - \frac{c^2 L^2 \Delta s_{\mathrm{U}}^2}{K\epsilon^2}} = \begin{cases} \frac{2cC\sqrt{\frac{T^2}{b^2} - L^2 K}}{mK\epsilon} & T > bL\sqrt{K}, \\ 0 & T \leq bL\sqrt{K}. \end{cases} \quad (75)$$

Furthermore, considering (69), we can obtain

$$\sigma_{\mathrm{D}} = \begin{cases} \frac{2cC\sqrt{\frac{T^2}{b^2} - L^2 K}}{mK\epsilon} & T > \frac{\epsilon}{\gamma}, \\ 0 & T \leq \frac{\epsilon}{\gamma}, \end{cases} \quad (76)$$

where

$$\gamma = -\ln\left(1 - q + qe^{\frac{-\epsilon}{L\sqrt{K}}}\right). \quad (77)$$

This completes the proof. □

Appendix G
Proof of Theorem 3

Here we define

$$\mathbf{v}^{(t)} = \sum_{i=1}^{K} p_i \mathbf{w}_i^{(t)}, \quad (78)$$

$$\widetilde{\mathbf{v}}^{(t)} = \sum_{i=1}^{K} p_i \left(\mathbf{w}_i^{(t)} + \mathbf{n}_i^{(t)}\right) + \mathbf{n}_{\mathrm{D}}^{(t)} \quad (79)$$

and

$$\mathbf{n}^{(t+1)} = \sum_{i=1}^{K} p_i \mathbf{n}_i^{(t+1)} + \mathbf{n}_{\mathrm{D}}^{(t)}. \quad (80)$$

which considers the aggregated parameters under K-random scheduling. Because $F_i(\cdot)$ and $F(\cdot)$ are β-Lipschitz, we obtain that

$$\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t+1)})\} - F(\mathbf{w}^{(t+1)}) \leq \beta \|\widetilde{\mathbf{v}}^{(t+1)} - \mathbf{w}^{(t+1)}\|. \quad (81)$$

Because β is the Lipchitz continuity constant of function F, we have

$$\beta \leq \|\nabla F(\widetilde{\mathbf{v}}^{(t)})\| + \rho\left(\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t)}\| + \|\mathbf{v}^{(t+1)} - \widetilde{\mathbf{v}}^{(t)}\|\right). \quad (82)$$

From (51), we know

$$\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t)}\| \leq \frac{B(1+\theta)}{\overline{\mu}} \|\nabla F(\widetilde{\mathbf{v}}^{(t)})\|. \quad (83)$$

Then, we have

$$\mathbb{E}\{\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t+1)}\|^2\} = \|\mathbf{w}^{(t+1)}\|^2 - 2[\mathbf{w}^{(t+1)}]^\top \mathbb{E}\{\widetilde{\mathbf{v}}^{(t+1)}\} + \mathbb{E}\{\|\widetilde{\mathbf{v}}^{(t+1)}\|^2\}. \quad (84)$$

Furthermore, we can obtain

$$\mathbb{E}\{\widetilde{\mathbf{v}}^{(t+1)}\} = \frac{1}{\binom{N}{K}} \frac{\binom{N}{K}}{N} K \sum_{i=1}^{N} p_i \mathbf{w}_i^{(t+1)} + \mathbf{n}^{(t+1)} = \mathbb{E}\{\mathbf{w}_i^{(t+1)}\} + \mathbf{n}^{(t+1)} = \mathbf{w}^{(t+1)} + \mathbf{n}^{(t+1)} \quad (85)$$

and

$$\mathbb{E}\{\|\widetilde{\mathbf{v}}^{(t+1)}\|^2\} = \mathbb{E}\left\{\left\|\sum_{i=1}^{K}\left(p_i \mathbf{w}_i^{(t+1)} + p_i \mathbf{n}_i^{(t+1)}\right)\right\|^2\right\} = \mathbb{E}\left\{\left\|\sum_{i=1}^{K} p_i \mathbf{w}_i^{(t+1)}\right\|^2\right\} + \mathbb{E}\left\{\left\|\sum_{i=1}^{K} p_i \mathbf{n}_i^{(t+1)}\right\|^2\right\} + 2\mathbb{E}\left\{\left[\sum_{i=1}^{K} p_i \mathbf{w}_i^{(t+1)}\right]^\top \mathbf{n}^{(t+1)}\right\}. \quad (86)$$

Due the independence between $\mathbf{w}_i^{(t+1)}$ and $\mathbf{n}_i^{(t+1)}$, we know

$$\mathbb{E}\left\{\left\|\sum_{i=1}^{K} p_i \mathbf{w}_i^{(t+1)}\right\|^2\right\} = \mathbb{E}\left\{\sum_{i=1}^{K}\left\|p_i \mathbf{w}_i^{(t+1)}\right\|^2\right\}. \quad (87)$$

Note that we set $p_i = D_i / \sum_{i=1}^{K} D_i = 1/K$ in K-random scheduling in order to a small sensitivity Δs_{D}. We have

$$\mathbb{E}\left\{\left\|\sum_{i=1}^{K} p_i \mathbf{w}_i^{(t+1)}\right\|^2\right\} = \frac{1}{NK}\sum_{i=1}^{N}\left\|\mathbf{w}_i^{(t+1)}\right\|^2 + \frac{(K-1)}{NK(N-1)}\sum_{i=1}^{N}\sum_{j=1\bigcup j\neq i}^{N}\left[\mathbf{w}_i^{(t+1)}\right]^\top \mathbf{w}_j^{(t+1)} \leq \frac{1}{K^2}\sum_{i=1}^{K}\|\mathbf{w}_i^{(t+1)}\|^2 + \frac{K-1}{K}\|\mathbf{w}^{(t+1)}\|^2 \quad (88)$$

and

$$\mathbb{E}\{\|\widetilde{\mathbf{v}}^{(t+1)}\|^2\} \leq \frac{1}{K^2}\sum_{i=1}^{K}\|\mathbf{w}_i^{(t+1)}\|^2 + \frac{K-1}{K}\|\mathbf{w}^{(t+1)}\|^2 + \|\mathbf{n}^{(t+1)}\|^2 + 2[\mathbf{w}^{(t+1)}]^\top \mathbf{n}^{(t+1)}. \quad (89)$$

Combining (84) and (89), we can obtain

$$\mathbb{E}\{\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t+1)}\|^2\} \leq \frac{1}{K^2}\sum_{i=1}^{K}\|\mathbf{w}_i^{(t+1)}\|^2 - \frac{1}{K}\|\mathbf{w}^{(t+1)}\|^2 + \|\mathbf{n}^{(t+1)}\|^2 \leq \frac{1}{K^2}\sum_{i=1}^{K}\|\mathbf{w}_i^{(t+1)} - \widetilde{\mathbf{v}}^{(t)}\|^2 + \|\mathbf{n}^{(t+1)}\|^2. \quad (90)$$

Using (50), we know

$$\mathbb{E}\{\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t+1)}\|^2\} \leq \|\mathbf{n}^{(t+1)}\|^2 + \frac{B^2(1+\theta)^2}{K\overline{\mu}^2}\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\|^2. \quad (91)$$

Moreover,

$$\mathbb{E}\{\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t+1)}\|\} \leq \|\mathbf{n}^{(t+1)}\| + \frac{B(1+\theta)}{\overline{\mu}\sqrt{K}}\|\nabla F(\widetilde{\mathbf{v}}^{t})\|. \quad (92)$$

Substituting (54), (82) and (92) into (81), setting $\theta = 0$ and $\overline{\mu} = \mu$, we can obtain

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t+1)})\} - F(\widetilde{\mathbf{v}}^{(t)}) &\leq F(\mathbf{w}^{(t+1)}) - F(\widetilde{\mathbf{v}}^{(t)}) \\ \left(\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\| + 2\rho\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t)}\|\right)\mathbb{E}\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t+1)}\| &\\ + \rho\mathbb{E}\{\|\mathbf{w}^{(t+1)} - \widetilde{\mathbf{v}}^{(t+1)}\|^2\} = \alpha_2\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\|^2 &\\ + \alpha_1\|\mathbf{n}^{(t+1)}\|\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\| + \alpha_0\|\mathbf{n}^{(t+1)}\|^2, \quad (93)\end{aligned}$$

where

$$\alpha_2 = \frac{1}{\mu^2}\left(\frac{\rho B^2}{2} + \rho B + \frac{\rho B^2}{K} + \frac{2\rho B^2}{\sqrt{K}} + \frac{\mu B}{\sqrt{K}} - \mu\right), \quad (94)$$

$$\alpha_1 \;=\; 1 + \frac{2\rho B}{\mu} + \frac{2\rho B\sqrt{K}}{\mu N} \text{ and } \alpha_0 \;=\; \frac{2\rho K}{N} + \rho. \quad (95)$$

In this case, we take expectation $\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t+1)}) - F(\widetilde{\mathbf{v}}^{(t)})\}$ as follows,

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t+1)}) - F(\widetilde{\mathbf{v}}^{(t)})\} &\leq \alpha_2\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\|^2 \\ + \alpha_1\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\}\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\| &+ \alpha_0\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}. \quad (96)\end{aligned}$$

For $\Theta > 0$ and $f(\mathbf{v}^{(0)}) - f(\mathbf{w}^*) = \Theta$, we can obtain

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t+1)}) - F(\mathbf{w}^*)\} &\\ \leq \mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t)}) - F(\mathbf{w}^*)\} + \alpha_2\|\nabla F(\widetilde{\mathbf{v}}^{(t)})\|^2 &\\ + \alpha_1\beta\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\} + \alpha_0\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}. \quad (97)\end{aligned}$$

If we select the penalty parameter μ to make $\alpha_2 < 0$ and using (57), we know

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t+1)}) - F(\mathbf{w}^*)\} &\leq (1+2l\alpha_2)\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(t)}) - F(\mathbf{w}^*)\} \\ + \alpha_1\beta\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|\} &+ \alpha_0\mathbb{E}\{\|\mathbf{n}^{(t+1)}\|^2\}. \quad (98)\end{aligned}$$

Considering independence of additive noises and applying (98) recursively, we have

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{v}}^{(T)}) - F(\mathbf{w}^*)\} &\leq (1+2l\alpha_2)^T\mathbb{E}\{F(\mathbf{v}^{(0)}) - F(\mathbf{w}^*)\} \\ &+ \frac{1-(1+2l\alpha_2)^T}{2l\alpha_2}\left(\alpha_1\beta\mathbb{E}\{\|\mathbf{n}\|\} + \alpha_0\mathbb{E}\{\|\mathbf{n}\|^2\}\right) \\ &= Q^T\Theta + \frac{1-Q^T}{1-Q}\left(\alpha_1\beta\mathbb{E}\{\|\mathbf{n}\|\} + \alpha_0\mathbb{E}\{\|\mathbf{n}\|^2\}\right), \quad (99)\end{aligned}$$

where $Q = 1+2l\alpha_2$. Substituting (74) into (99), we can obtain

$$\mathbb{E}\{\|\mathbf{n}\|\} = \frac{\Delta s_{\mathrm{D}} Tc}{b\epsilon}\sqrt{\frac{2N}{\pi}}, \mathbb{E}\{\|\mathbf{n}\|^2\} = \frac{\Delta s_{\mathrm{D}}^2 T^2 c^2 N}{b^2\epsilon^2} \quad (100)$$

and

$$\begin{aligned}\mathbb{E}\{F(\widetilde{\mathbf{v}}^{T}) - F(\mathbf{w}^*)\} &\leq Q^T\Theta \\ &+ \frac{1-Q^T}{1-Q}\left(\frac{c\alpha_1\beta}{-mK\ln\left(1-\frac{N}{K}+\frac{N}{K}e^{-\frac{\epsilon}{T}}\right)}\sqrt{\frac{2}{\pi}}\right. \quad (101) \\ &\left. + \frac{c^2\alpha_0}{m^2K^2\ln^2\left(1-\frac{N}{K}+\frac{N}{K}e^{-\frac{\epsilon}{T}}\right)}\right).\end{aligned}$$

This completes the proof. □

REFERENCES

[1] J. Li, S. Chu, F. Shu, J. Wu, and D. N. K. Jayakody, "Contract-Based Small-Cell Caching for Data Disseminations in Ultra-Dense Cellular Networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 5, pp. 1042–1053, May 2019.

[2] Z. Ma, M. Xiao, Y. Xiao, Z. Pang, H. V. Poor, and B. Vucetic, "High-Reliability and Low-Latency Wireless Communication for Internet of Things: Challenges, Fundamentals, and Enabling Technologies," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7946–7970, Oct. 2019.

[3] H. Lee, S. H. Lee, and T. Q. S. Quek, "Deep Learning for Distributed Optimization: Applications to Wireless Resource Management," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2251–2266, Oct. 2019.

[4] W. Sun, J. Liu, and Y. Yue, "AI-Enhanced Offloading in Edge Computing: When Machine Learning Meets Industrial IoT," *IEEE Netw.*, vol. 33, no. 5, pp. 68–74, Sep. 2019.

[5] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep Learning for IoT Big Data and Streaming Analytics: A Survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2923–2960, Jun. 2018.

[6] H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, "Federated Learning of Deep Networks using Model Averaging," *arXiv*, 2016. [Online]. Available: http://arxiv.org/abs/1602.05629

[7] J. Konecný *et al.*, "Federated Learning: Strategies for Improving Communication Efficiency," *arXiv*, 2016. [Online]. Available: http://arxiv.org/abs/1610.05492

[8] U. Mohammad and S. Sorour, "Adaptive Task Allocation for Asynchronous Federated Mobile Edge Learning," *arXiv*, 2019. [Online]. Available: http://arxiv.org/abs/1905.01656

[9] X. Wang *et al.*, "In-Edge AI: Intelligentizing Mobile Edge Computing, Caching and Communication by Federated Learning," *IEEE Network*, vol. 33, no. 5, pp. 156–165, Sep. 2019.

[10] Y. Qiang, L. Yang, C. Tianjian, and T. Yongxin, "Federated Machine Learning: Concept and Applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 12:1–12:19, Jan. 2019.

[11] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," *arXiv*, 2019. [Online]. Available: https://arxiv.org/abs/1908.07873

[12] N. H. Tran, W. Bao, A. Zomaya, N. Minh N.H., and C. S. Hong, "Federated Learning over Wireless Networks: Optimization Model Design and Analysis," in *Proc. IEEE INFOCOM*, Apr. 2019, pp. 1387–1395.

[13] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling Policies for Federated Learning in Wireless Networks," *IEEE Trans. on Commun.*, pp. 1–1, to appear 2019.

[14] M. Hao, H. Li, G. Xu, S. Liu, and H. Yang, "Towards Efficient and Privacy-Preserving Federated Deep Learning," in *Proc. IEEE ICC*, Paris, France, May 2019, pp. 1–6.

[15] A. Alekh and D. J. C, "Distributed Delayed Stochastic Optimization," in *Proc. IEEE CDC*, Maui, HI, USA, Dec. 2012.

[16] L. Xiangru, H. Yijun, L. Yuncheng, and L. Ji, "Asynchronous Parallel Stochastic Gradient for Nonconvex Optimization," in *Proc. ACM NIPS*, Montreal, Canada, Dec. 2015, pp. 2737–2745.

[17] X. Lian *et al.*, "Can Decentralized Algorithms Outperform Centralized Algorithms? A Case Study for Decentralized Parallel Stochastic Gradient Descent," in *Proc. ACM NIPS*, Long Beach, California, USA, Dec. 2017, pp. 5336–5346.

[18] T. Li *et al.*, "On the Convergence of Federated Optimization in Heterogeneous Networks," *arXiv*, 2018. [Online]. Available: http://arxiv.org/abs/1812.06127

[19] S. Wang *et al.*, "Adaptive Federated Learning in Resource Constrained Edge Computing Systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[20] R. Shokri and V. Shmatikov, "Privacy-Preserving Deep Learning," in *Proc. ACM CCS*, Denver, Colorado, USA, Oct. 2015, pp. 1310–1321.

[21] Z. Wang *et al.*, "Beyond Inferring Class Representatives: User-Level Privacy Leakage From Federated Learning," in *Proc. IEEE INFOCOM*, Paris, France, Apr. 2019, pp. 2512–2520.

[22] C. Ma, J. Li, M. Ding, H. Hao Yang, F. Shu, T. Q. S. Quek, and H. V. Poor, "On Safeguarding Privacy and Security in the Framework of Federated Learning," *arXiv*, 2019. [Online]. Available: https://arxiv.org/abs/1909.06512

[23] C. Dwork and A. Roth, "The Algorithmic Foundations of Differential Privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2014.

[24] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical Privacy: The SuLQ Framework," in *Proc. ACM PODS*, Baltimore, Maryland, Jun. 2005, pp. 128–138.

[25] Úlfar Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response," in *Proc. ACM CCS*, Scottsdale, Arizona, USA, Nov. 2014, pp. 1054–1067.
[26] N. Wang *et al.*, "Collecting and Analyzing Multidimensional Data with Local Differential Privacy," in *Proc. IEEE ICDE*, Macao, China, Apr. 2019, pp. 638–649.
[27] S. Wang, L. Huang, Y. Nie, X. Zhang, P. Wang, H. Xu, and W. Yang, "Local Differential Private Data Aggregation for Discrete Distribution Estimation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 9, pp. 2046–2059, Sep. 2019.
[28] A. Martin *et al.*, "Deep Learning with Differential Privacy," in *Proc. ACM CCS*, Vienna, Austria, Oct. 2016, pp. 308–318.
[29] N. Wu, F. Farokhi, D. Smith, and M. A. Kâafar, "The Value of Collaboration in Convex Machine Learning with Differential Privacy," *arXiv*, 2019. [Online]. Available: http://arxiv.org/abs/1906.09679
[30] J. Li, M. Khodak, S. Caldas, and A. Talwalkar, "Differentially Private Meta-Learning," *arXiv*, 2019. [Online]. Available: https://arxiv.org/abs/1909.05830
[31] R. C. Geyer, T. Klein, and M. Nabi, "Differentially Private Federated Learning: A Client Level Perspective," *arXiv*, 2017. [Online]. Available: http://arxiv.org/abs/1712.07557
[32] S. Truex *et al.*, "A Hybrid Approach to Privacy-Preserving Federated Learning," *arXiv*, 2018. [Online]. Available: http://arxiv.org/abs/1812.03224
[33] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. ACM CCS*, New York, NY, USA, 2015, pp. 1322–1333.
[34] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[35] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, 1st ed. Springer Publishing Company, Incorporated, 2014.