Truthful Incentive Mechanism for Federated Learning with Crowdsourced Data Labeling

Yuxi Zhao, Xiaowen Gong, Shiwen Mao
Department of Electrical and Computer Engineering
Auburn University, Auburn, AL 36849
Email: {yzz0171,xgong}@auburn.edu, smao@ieee.org

IEEE INFOCOM 2023 - IEEE Conference on Computer Communications | 979-8-3503-3414-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/INFOCOM53939.2023.10228923

***Abstract*—Federated learning (FL) has recently emerged as a promising paradigm that trains machine learning (ML) models on clients' devices in a distributed manner without the need of transmitting clients' data to the FL server. In many applications of ML (e.g., image classification), the labels of training data need to be generated manually by human agents (e.g., recognizing and annotating objects in an image), which are usually costly and error-prone. In this paper, we study FL with crowdsourced data labeling where the local data of each participating client of FL are labeled manually by the client. We consider the strategic behavior of clients who may not make desired effort in their local data labeling and local model computation (quantified by the mini-batch size used in the stochastic gradient computation), and may misreport their local models to the FL server. We first characterize the performance bounds on the training loss as a function of clients' data labeling effort, local computation effort, and reported local models, which reveal the impacts of these factors on the training loss. With these insights, we devise Labeling and Computation Effort and local Model Elicitation (LCEME) mechanisms which incentivize strategic clients to make truthful efforts as desired by the server in local data labeling and local model computation, and also report true local models to the server. The truthful design of the LCEME mechanism exploits the non-trivial dependence of the training loss on clients' hidden efforts and private local models, and overcomes the intricate coupling in the joint elicitation of clients' efforts and local models. Under the LCEME mechanism, we characterize the server's optimal local computation effort assignments and analyze their performance. We evaluate the proposed FL algorithms with crowdsourced data labeling and the LCEME mechanism for the MNIST-based hand-written digit classification. The results corroborate the improved learning accuracy and cost-effectiveness of the proposed approaches.**

***Index Terms*—Federated Learning, Crowdsourcing, Incentive Mechanism**

I. Introduction

Federated learning (FL) [1] is an emerging and promising ML paradigm, which performs the training of ML models in a distributed manner. Instead of transmitting data from a potentially large number of devices to a central server in the edge or cloud for training, FL allows the data to remain at devices (such as smartphone), and trains a global ML model on the server by collecting and aggregating model updates locally computed on each device based on her local data. One significant advantage of using FL is to preserve the privacy of individual device's data. Moreover, since only local ML model updates, instead of local data, are sent to the server, the communication costs can be greatly reduced. Furthermore, FL can exploit the substantial computation capabilities of ubiquitous smart devices, which are often under-utilized. As a result, FL can achieve collaborative intelligence, which can enable many AI applications based on networked systems, such as connected and autonomous vehicles, collaborative robots, multi-user virtual/mixed reality.

Recent studies on FL typically focus on supervised learning, which requires a large amount of training data with data labels in the learning process. In many applications of ML, data labels have to be generated manually by human users. For example, for image classification, the object in an image should be recognized and annotated by a human user as the label of the image data. Therefore, as FL does not allow a client to share her local data with the server or other clients, to participate in FL, a client needs to manually label her local data (e.g., images), before she can compute local model updates from her locally labeled data.

However, data labels generated by human clients of FL are subject to errors. For example, a client may misclassify a dog as a cat. As a result, this incorrect data label will lead to error in the local model, and thus error in the global model obtained by the FL server. Moreover, the labeling error rate of a client generally varies for different clients, depending on the client's knowledge level of the labeling task. For example, a client who is familiar with dogs will have a lower labeling error rate than another client who is not. Furthermore, the accuracy of data labels is also affected by a client's effort made in the data labeling task. The data label error rate will be low when the client makes much effort in labeling the data, and otherwise is high when the client makes little or no effort. For example, a client may make no effort in image classification by randomly guessing the object in an image without actually recognizing it.

While a client's effort impacts the accuracy of her data labels, the effort can be her hidden action that is only known by the client herself and cannot be observed by the FL server. Due to the inaccurate nature of data labels, a strategic client may label her local data arbitrarily without making effort in data labeling, while the server will not be able to verify whether effort is actually made or not. Moreover, the effort

The work of Y. Zhao and X. Gong was supported by the startup fund of X. Gong from Auburn University, and U.S. NSF grant ECCS-2121215. The work of S. Mao was supported in part by U.S. NSF grant CNS-2107190.

made by a client in computing her local model update, which can be quantified by the mini-batch size used by the client in stochastic gradient descent, can also be the client's hidden action that cannot be verified by the server. As a result, a client may have incentive to compute her local update with a small mini-batch size so as to reduce her resources used in local computation. Furthermore, the local model computed by a client from her local data can also be her private information that she can manipulate in favor of herself, e.g., a client may increase or decrease her true local model and report it to the server.

In the presence of such strategic clients with hidden data labeling and local computation efforts and private local models, our goal is to incentivize the clients to make truthful efforts as desired by the FL server and reveal their true local models. Such a truthful incentive mechanism is desirable as it eliminates the possibility of manipulation, which would encourage clients to participate in FL. More importantly, the truthful elicitation of clients' efforts and local models ensures that the FL server can obtain a global model with high and guaranteed accuracy from the learning process, which is a key performance metric of FL.

The joint elicitation of data labeling effort, local computation effort, and local models for FL calls for a new design that is very different from existing truthful mechanisms. First, the training loss of the global model obtained from FL has a non-trivial dependence on clients' exerted efforts and reported models. As a result, existing incentive mechanisms for effort and data elicitation do not work for the problem here. Second, due to the complex relationship between the impacts of labeling effort, computation effort, and local models on the training loss, the joint elicitation of effort and models needs to overcome the coupling therein. Third, given the truthful incentive mechanism for effort and model elicitation, the FL server needs to determine how much effort should be made by each client, in order to maximize the server's payoff.

The main contributions of this paper are as follows.

- We propose an FL framework with crowdsourced data labeling based on a truthful incentive mechanism, where the labels of a client's local training data for FL are manually generated by the human client and are subject to errors. We consider strategic clients whose actual efforts in data labeling and local model computation as well as actual local models cannot be verified by the FL server.
- We first characterize the performance bounds on the training loss as a function of clients' data labeling effort, local computation effort (quantified by the mini-batch size), and reported local models. It shows that the labeling and computation efforts as well as the reported models have non-trivial impacts on the training loss. Based on the obtained insights, we develop the Labeling and Computation Effort and Local model Elicitation (LCEME) mechanism which incentivize clients to truthfully make efforts in data labeling and local computation, and report local models. The truthful design of the LCEME mechanism overcomes the intricate coupling in the joint elicitation of labeling effort, computation effort, and local models. Based on the LCEME mechanism, we then characterize the optimal computation effort assignment for maximizing the FL server's payoff.
- We evaluate the proposed FL with crowdsourced data labeling for the MNIST-based hand-written digit recognition. The results demonstrate that the proposed algorithms outperform the methods that do not consider data labeling errors or do not use an incentive mechanism.

The remainder of this paper is organized as follows. Section II reviews the related work. In Section III, we describe the system model and formulate the problem of incentive mechanism design. In Section IV, we study the performance bound on the training loss. In Section V, we devise the LCEME mechanism and the server's optimal effort allocation. Simulation results are presented in Section VI. Section VII concludes this paper.

II. Related Work

Incentive Mechanism for Federated Learning. FL has emerged as a disruptive computing paradigm for ML by democratizing the learning process to potentially many individual devices. Most existing studies on FL have focused on algorithm design for FL, such as for reducing the local model drifts across non-IID clients and participating clients selection. Meanwhile, there have been several recent works on computation and communication resource allocation for FL [2]–[9]. On the other hand, a few recent works studied incentive mechanisms [10]–[20] for FL that take into account participating clients' strategic behavior. In particular, most of these works considered compensating clients' communication and computation costs with an economic approach, such as Stackelberg game [14], auction theory [15], cooperative game [16], [17], and contract theory [18], [19]. However, all these prior works have focused on either incentivizing clients' participation via cost compensation, or truthfully eliciting clients' participation costs. [20] proposed VCG-based mechanisms that incentivize clients to truthfully report their local models. In contrast, this paper studies incentive mechanisms for truthful elicitation of clients' local models as well as their efforts in data labeling and local computation.

Truthful Incentive Mechanism for Effort and Data Elicitation. There have been lots of research on incentive mechanisms for various applications of data collection and processing, particularly for data crowdsourcing [21]–[30]. Many incentive mechanisms incentivize agents to truthfully reveal their participating cost, where the cost is considered to be private for an agent that may not be revealed truthfully without appropriate incentive. There have been studies on truthful mechanism design for hidden efforts in economics literature [31], which is concerned with strategic agents that can make hidden efforts not desired by a principal who recruits the agents to work on a task. A few recent works have studied this problem in the context of crowdsourcing [27], [32]–[35]. Mechanism design for truthful elicitation of strategic agents' data (e.g., opinions) has been extensively studied in various applications (e.g., [36]), more recently for crowdsourcing [27], [32], [34], [35], [37]. The data of an agent can be private

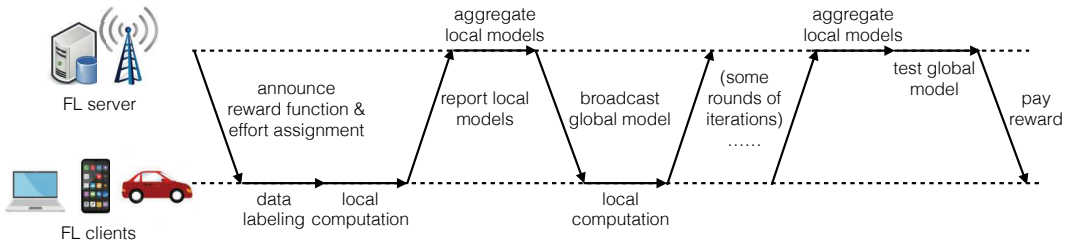

Fig. 1. Schedule of FL with crowdsourced data labeling based on a truthful incentive mechanism.

information that the agent can manipulate in favor of her benefit. Different from existing works, in this paper, we focus on FL and aim to design truthful mechanisms that jointly elicit clients' hidden efforts in data labeling and local computation and private local models. The truthful mechanism design is non-trivially different from existing works, due to 1) complex dependence of the training loss on clients' data labeling and local computation efforts and local models; 2) intricate coupling in joint elicitation of the clients' efforts and models.

III. System Model and Problem Formulation

In this section, we first describe a FL system with crowdsourced data labeling based on a truthful incentive mechanism (as illustrated in Fig. 1), and then present the design objectives for truthful incentive mechanisms.

A. FL with Crowdsourced Data Labeling

Consider the following FL problem:

$$\min_{\mathbf{w}} \; F(\mathbf{w}) \triangleq \sum_{i=1}^{N} p_i F_i(\mathbf{w}), \tag{1}$$

where $F_i(\mathbf{w})$ is defined by

$$F_i(\mathbf{w}) \triangleq \frac{1}{\tilde{D}_i} \sum_{m=1}^{\tilde{D}_i} f(\mathbf{w}; \xi_m^i),$$

$f(\cdot)$ is the per-sample loss function of client i, $N \triangleq |\mathcal{N}|$ is the number of clients, p_i is the weight of client i, $\sum_{i\in\mathcal{N}} p_i \triangleq 1$, $\mathcal{D}_i = \{\xi_1^i, \xi_2^i, \ldots, \xi_{\tilde{D}_i}^i\}$ is client i's local dataset for updating the model parameter, and $D \triangleq \sum_{i=1}^{N} \tilde{D}_i$. Without loss of generality, for ease of exposition, we assume that all clients have the same per-sample loss function $f(\cdot)$.

Data Labeling. To participate in FL, each client needs to have a labeled local dataset $\mathcal{D}_i$. In this paper, we assume the clients collaboratively train for classification tasks, where each client needs to label her local dataset (i.e., classify the local data samples based on the features of data). After finding the classification labels, each client $i \in \mathcal{N}$ obtains the local dataset $\mathcal{D}_i$.

For simplicity, we assume that each client has two strategies for the labeling effort $e_i \in \{0, 1\}$, where $e_i = 1$ and $e_i = 0$ indicate making and not making effort, respectively. If client i makes effort, then the labels in her dataset are correct; otherwise, the labels are randomly selected from all possible classes without considering the corresponding features. We know that an ML model trained on a correctly labeled dataset is more likely to make useful predictions than a model trained on incorrectly labeled data. Therefore, making effort $e_i = 1$ means higher accuracy of the trained model than not making effort (We prove this intuition in Section IV.). We assume that every client can fully control the amount of effort they make, and the server does not have such information.

Local Model Computation. In each round of FL, clients communicate their local updates to the server and receive the updated global model from the server. In round t,[1] each client i receives the global model $\mathbf{w}_{t-1}$ from the server, sets $\mathbf{w}_{t,0}^i = \mathbf{w}_{t-1}$, and then performs H local iterations of SGD. In the hth local iteration, client i computes the average gradient $g_{t,h-1}^i$ of the loss function using a mini-batch of D_i data samples randomly drawn from her local dataset $\mathcal{D}_i$. Then client i updates her local model as

$$\mathbf{w}_{t,h}^i = \mathbf{w}_{t,h-1}^i - \eta g_{t,h-1}^i,$$

where

$$g_{t,h-1}^i \triangleq \frac{1}{D_i} \sum_{j=1}^{D_i} \nabla f(\mathbf{w}, \xi_{t,h}^{i,j}),$$

η is the step size, and $\xi_{t,h}^{i,j}$ is the jth data sample randomly drawn from client i's local dataset $\mathcal{D}_i$. After H local iterations, client i sends her local update $\mathbf{w}_{t,H}^i$ for round t to the server.

The computation effort D_i represents the mini-batch size client i uses to update her local model in each round. Due to the randomness of data sampling for computing the update in SGD, the computed gradient of a client could deviate from the expected gradient, and thus slow down the convergence of the FL global model. It has been proved that the larger the mini-batch size D_i, the lower the variance of her local update [38]. Thus, a local update computed with a larger mini-batch size benefits the FL training.

At the end of round t, the server aggregates clients' local models and updates the global model as

$$\mathbf{w}_t = \sum_{i=1}^{N} p_i \mathbf{w}_{t,H}^i.$$

Effort Assignment. Before data labeling and local computation, the server assigns the labeling effort e_i' and computation effort D_i' to each client i and notifies client i of e_i' and D_i'. The labeling effort $e_i' \in \{0, 1\}$ indicates whether the server desires client i to make effort in labeling, and the computation effort D_i' indicates the mini-batch size that the server desires client i to use to update her local model in each round. Clients' effort assignments generally vary for different clients due to

[1] We use t and h as the index of communication rounds and local iterations, respectively. The subscript (t, h) denotes the hth local iteration in round t.

their diverse characteristics (e.g., weight in model aggregation, computation capability).

After being assigned effort e_i', each client i generates labels for the local dataset by making actual effort e_i. Since e_i is a hidden action of client i, it is possible that client i manipulates e_i against the assignment e_i' to her own advantage such that $e_i \neq e_i'$.

Furthermore, a client incurs a computation cost (measured by the computation time, energy consumption, etc.) for computing a local model update, which depends on the computation capability of the client and the mini-batch size used to compute the update. Thus, client i may also have incentive to manipulate D_i against the assignment D_i' to her own advantage such that $D_i \neq D_i'$.

Local Model Reporting. When reporting the local model to the server, a client i may have incentive to misreport her local model to her own advantage, i.e.,

$$\mathbf{w}_t^i = \mathbf{w}_{t-1} - \gamma_i \eta g_{t-1}^i,$$

where $\gamma_i \geq 0$, $\forall i \in \mathcal{N}$ is the model reporting coefficient, which is the multiple of the gradient client i uses to update her local model[2]. When $\gamma_i = 1$, client i reports the actual local model to the server, which is desired by the FL server. When $\gamma_i \neq 1$, the gradient is reduced or increased. In this case, the trained model of FL will be affected, and thus the training loss $F(\mathbf{w})$. It is possible that client i manipulates γ_i to her own advantage such that $\gamma_i \neq 1$.

B. Truthful Incentive Mechanism for FL

After the training process, the FL server tests the trained global model of FL to a data sample ξ randomly drawn from a testing dataset $\mathcal{D}_0$. It is commonly assumed in existing studies that the FL server can test the trained FL model (e.g., [18], [39]). Then the server can determine each client's reward based on the testing loss $f(\mathbf{w}_T, \xi)$ observed for the testing data sample ξ. Note that the server only needs to test the trained model to a single random data sample from $\mathcal{D}_0$. For example, the testing can be performed when the server applies the trained model to an unseen data sample for inference/prediction and observes its true label later.

Based on the testing loss $f(\mathbf{w}_T, \xi)$, the server pays a reward r_i to each client i, according to a certain reward function:

$$r_i(e_i', \boldsymbol{e}_{-i}', D_i', \boldsymbol{D}_{-i}', \gamma_i', \boldsymbol{\gamma}_{-i}', f(\mathbf{w}_T, \xi)), \tag{2}$$

where $\boldsymbol{e}_{-i}'$, $\boldsymbol{D}_{-i}'$, and $\boldsymbol{\gamma}_{-i}'$ are other clients' assigned data labeling and computation effort, and model reporting coefficient, respectively. The reward function is pre-defined by the server and announced to all clients before they participate in FL. We can see that the reward depends on not only the assigned efforts and model reporting coefficient but also the testing loss of the final global model.

Each client i's payoff is the difference between the reward paid by the server and her cost in data labeling and computing her local model, given by

$$u_i(e_i, \boldsymbol{e}', D_i, \boldsymbol{D}', \gamma_i, \boldsymbol{\gamma}') \triangleq$$
$$r_i(e_i', \boldsymbol{e}_{-i}', D_i', \boldsymbol{D}_{-i}', \gamma_i', \boldsymbol{\gamma}_{-i}', f(\mathbf{w}_T, \xi)) - c_l^i e_i - \sum_{t=1}^{T} c_p^i D_i,$$

where $\boldsymbol{e}'$, $\boldsymbol{D}'$, and $\boldsymbol{\gamma}'$ are clients' assigned data labeling effort, computation effort, and model reporting coefficient, respectively. The data labeling cost coefficient c_l^i captures the resources consumed by client i if she makes an effort, i.e., $e_i = 1$, in data labeling, and the computation cost coefficient c_p^i is client i's cost of computing her local update using one data sample. If client i makes no effort in data labeling, i.e., $e_i = 0$, there incurs no data labeling cost. Here we assume that clients have the same data labeling cost coefficient (i.e., $c_l = c_l^i$, $\forall i \in \mathcal{N}$), and the labeling and computation cost coefficients are known to the server. This assumption is reasonable when the costs of labeling a client's dataset and computing using a data sample are determined by uniform market prices (e.g., in Amazon Mechanical Turk, a usual reward for labeling an image is \$0.1). A client's computation cost is affected by her computation cost coefficient c_p^i and computation effort $\boldsymbol{D}'$. We can also relax the restriction of the uniform labeling cost coefficient. Since a client i can only affect the training loss through her actual e_i, D_i, and γ_i, we omit the loss function $f(\mathbf{w}_T, \xi)$ in the expression of client i's utility u_i. The detailed reward function design will be given in Section V.

The server's payoff u_0 is the negative training loss minus the total reward paid to the clients, i.e.,

$$u_0(\boldsymbol{e}', \boldsymbol{D}', \boldsymbol{\gamma}', f(\mathbf{w}_T, \xi)) \triangleq -f(\mathbf{w}_T, \xi) - \sum_{i \in \mathcal{N}} r_i. \tag{3}$$

Since clients may manipulate their actual efforts and report untruthful local models, the global model may be different from that when clients do not behave truthfully, i.e.,

$$\mathbf{w}_T|_{\boldsymbol{e}', \boldsymbol{D}', \boldsymbol{\gamma}'} \neq \mathbf{w}_T|_{\boldsymbol{e}, \boldsymbol{D}, \boldsymbol{\gamma}}.$$

This means that the final global model obtained with efforts and reported local model manipulation cannot solve the FL problem given in (1). Nevertheless, the training loss of FL is also affected, i.e.,

$$F(\mathbf{w}_T)|_{\boldsymbol{e}', \boldsymbol{D}', \boldsymbol{\gamma}'} \neq F(\mathbf{w}_T)|_{\boldsymbol{e}, \boldsymbol{D}, \boldsymbol{\gamma}}.$$

Furthermore, some clients' manipulation would discourage other clients to participate in FL. For the reasons discussed above, here we aim to design a mechanism that can incentivize clients to make data labeling and computation efforts as the server desired and upload their actual local models. This can be achieved by properly defining the reward function r_i. The truthful mechanism should have the following features:

Definition 1: A mechanism achieves truthful strategies of all clients as a *Nash equilibrium* (NE) if, given that all other clients truthfully make data labeling and computation effort as the server desired and upload their actual local models, the best strategy for client i to maximize her payoff is to behave truthfully, i.e.,

$$E[u_i(e_i', \boldsymbol{e}_{-i}', D_i', \boldsymbol{D}_{-i}', \gamma_i', \boldsymbol{\gamma}_{-i}')] \geq$$
$$E[u_i(e_i, \boldsymbol{e}_{-i}', D_i, \boldsymbol{D}_{-i}', \gamma_i, \boldsymbol{\gamma}_{-i}')], \forall e_i, D_i, \gamma_i. \tag{4}$$

We should also notice that the payoff of each client i should

[2]In this paper, we assume that clients' strategies do not change over time in FL training.

be non-negative so that the client will have the incentive to participate. This property is known as individual rationality.

Definition 2: A mechanism is *individually rational* (IR) if for each client i, its expected payoff is non-negative if she behaves truthfully, i.e.,

$$E[u_i(e_i', \boldsymbol{e}'_{-i}, D_i', \boldsymbol{D}'_{-i}, \gamma_i', \boldsymbol{\gamma}'_{-i})] \geq 0, \forall e_i, D_i, \gamma_i. \quad (5)$$

IV. Training Loss Analysis

In this section, we characterize the performance bounds on the training loss as a function of clients' data labeling effort, local computation effort, and reported local models, which reveal the impacts of these factors on the training loss.

We first make the following general assumptions on the loss functions $F_1, \ldots, F_N$, $\forall i \in \mathcal{N}$.

Assumption 1: $F_1, \ldots, F_N$ are L-smooth.

Assumption 2: $F_1, \ldots, F_N$ are μ-strongly convex.

Assumption 3: The variance of the stochastic gradient of a data sample in a device is bounded: $E\left\|\nabla f\left(\mathbf{w}_t, \xi_m^i\right) - \nabla F_i\left(\mathbf{w}_t\right)\right\|^2 \leq \sigma_i^2$, $\forall i \in \mathcal{N}$, $\forall t$.

Assumption 4: The variance of the stochastic gradient of a data sample when the client makes no effort on labeling is bounded: $E\left\|\nabla f\left(\mathbf{w}_t, \xi_m^i\right) - \nabla f\left(\mathbf{w}_t, \xi_m^{i\,\prime}\right)\right\|^2 \leq \beta$, $\forall i \in \mathcal{N}$, $\forall t$.

Assumption 5: The expected squared norm of stochastic gradients is bounded: $E\left\|\nabla F_i\left(\mathbf{w}_t\right)\right\|^2 \leq G^2$, $\forall i \in \mathcal{N}$, $\forall t$.

In Assumption 4, we assume that the variance of the stochastic gradient of a data sample when the client makes no labeling effort is upper bounded, and the bound β is known by the server. The server can calculate the bound using the loss function and the range of data's value. Next, we use a simple example to demonstrate how to obtain the bound β. We use a simple linear regression model to illustrate the convergence problem. Assume that the loss function is given by

$$f\left(\mathbf{w}, \xi_m^i\right) = \frac{1}{2}\|\boldsymbol{x}_m^i \boldsymbol{w} - y_m^i\|^2, \quad \forall i \in \mathcal{N}.$$

A data sample with correct and incorrect labels are denoted as $\xi_m^i = (\boldsymbol{x}_m^i, y_m^i)$ and $\xi_m^{i\,\prime} = (\boldsymbol{x}_m^i, y_m^{i\,\prime})$, respectively.

The variance of the stochastic gradient of a data sample is

$$\begin{aligned} & E\left\|\nabla f\left(\mathbf{w}, \xi_m^i\right) - \nabla f\left(\mathbf{w}, \xi_m^{i\,\prime}\right)\right\|^2 \\ =& \|(\boldsymbol{x}_m^i \boldsymbol{w} - y_m^i)\boldsymbol{x}_m^i - (\boldsymbol{x}_m^i \boldsymbol{w} - y_m^{i\,\prime})\boldsymbol{x}_m^i\|^2 \\ =& \|(y_m^{i\,\prime} - y_m^i)\boldsymbol{x}_m^i\|^2 \leq 2YX, \end{aligned}$$

where ${y_m^i}^2 \leq Y$ and $\|\boldsymbol{x}_m^i\|^2 \leq X$. Then we have $\beta = 2YX$.

Theorem 1: Suppose Assumptions 1 to 5 hold, and the step size $\eta \leq \frac{1}{2L}$. Then the FL training loss is bounded above by:

$$\begin{aligned} & E[F(\mathbf{w}_T) - F(\mathbf{w}^*)] \leq L(1-\mu\eta)^{TH} E\left\|\mathbf{w}_0 - \mathbf{w}^*\right\|^2 \\ & + 2L\eta^2 \sum_{t=1}^{T}\sum_{h=1}^{H}(1-\mu\eta)^{TH-(t-1)H-h} \\ & \sum_{i\in\mathcal{N}}\left({p_i}^2\frac{\sigma_i^2}{D_i} + 6Lp_i d_i + p_i(1-e_i)\beta\right. \\ & \left. + 2p_i\left((\gamma_i - 1)^2 + (H-1)^2\right)\left(G^2 + \frac{\sigma_i^2}{D_i} + (1-e_i)\beta\right)\right), \quad (6) \end{aligned}$$

where $d_i \triangleq E[F_i(\mathbf{w}^*)] - E[F_i(\mathbf{w}_i^*)]$ quantifies the heterogeneity degree of the data held by client i [40].

The proof is given in Appendix A.

Remark 1: The first term of the training loss bound decreases geometrically with the total number of local iterations TH, and is due to that SGD in expectation makes progress towards the optimal solution. The bound is also affected by other factors, i.e., the randomness of data sampling in SGD for computing local updates $p_i^2\frac{\sigma_i^2}{D_i}$, the data heterogeneity of clients' data $p_i d_i$, the data labeling effort level of each client $p_i(1-e_i)\beta$, the local model misreporting γ_i, and the number of local iterations per round H. We can see that any $\gamma_i \neq 1$, i.e., any client untruthfully reports her local model, increases the training loss bound. Thus, it is desired that all clients report their actual local model (i.e., $\gamma_i = 1$, $\forall i \in \mathcal{N}$) to minimize the training loss. Moreover, as the coefficients in the training loss bound depend on the aggregation weight p_i, a client with a higher weight p_i has a larger impact on the training loss than that with a lower weight p_i.

Remark 2: The randomness of data sampling in SGD for computing local updates affects the training loss, which depends on each client's mini-batch size D_i in each iteration (i.e., computation effort). We can observe that a larger mini-batch size D_i reduces the training loss. The terms involving e_i depend on the data labeling effort of each client. If client i makes effort in data labeling, these terms equal 0; otherwise, if client i makes no effort in data labeling, these terms equal $p_i\beta$. Thus, it is desirable that all clients make data labeling effort (i.e., $e_i = 1$, $\forall i \in \mathcal{N}$) to minimize the training loss.

V. Truthful Incentive Mechanisms for Data Labeling Effort, Local Computation Effort, and Local Model Elicitation

In this section, we propose the LCEME mechanism that satisfies the truthful and IR properties to incentivize clients to make efforts as the server desired and report actual local models. Then, we find the optimal computation effort assignment under the LCEME mechanism that maximizes the server's payoff.

A. LCEME Mechanism Design

We first present the design of the LCEME mechanism.

Definition 3: Given the data labeling effort assignment $e_i' = 1$, the model reporting coefficient assignment $\gamma_i' = 1$, and any computation effort assignment D_i', the LCEME mechanism's reward function for client i, $\forall i \in \mathcal{N}$, is given by

$$\begin{aligned} & r_i(e_i', \boldsymbol{e}'_{-i}, D_i', \boldsymbol{D}'_{-i}, \gamma_i', \boldsymbol{\gamma}'_{-i}, f(\mathbf{w}_T, \xi)) \\ & = \Omega(\boldsymbol{D}') - \Phi(D_i') f(\mathbf{w}_T, \xi) + c_l, \end{aligned} \quad (7)$$

where

$$\begin{aligned} & \Omega(\boldsymbol{D}') = \Phi(D_i')\left(L(1-\mu\eta)^{TH} E\left\|\mathbf{w}_0 - \mathbf{w}^*\right\|^2 + \right. \\ & \left. A\sum_{i\in\mathcal{N}}(6Lp_i d_i + {p_i}^2\frac{\sigma_i^2}{D_i'} + 2p_i(H-1)^2(G^2 + \frac{\sigma_i^2}{D_i'}))\right) + Tc_p^i D_i', \end{aligned}$$

$\boldsymbol{e}' = \mathbf{1}^{1\times N}$, $\boldsymbol{\gamma}' = \mathbf{1}^{1\times N}$, $\Phi(D_i') = \frac{{D_i'}^2 c_p^i T}{A\sigma_i^2 p_i(p_i + 2(H-1)^2)}$, $A = 2L\eta\frac{1-(1-\mu\eta)^{TH}}{\mu}$, and the assigned computation effort D_i' satisfies $D_i' \geq \sigma_i\sqrt{\frac{c_l p_i(p_i+2(H-1)^2)}{\beta c_p^i T(1+2(H-1)^2)}}$.

Note that the reward function depends on the testing loss which is observed by the server. In this paper, for ease of exposition, we assume that the expected testing loss is equal to the training loss. This assumption is reasonable: in practice, the entire training dataset of FL (i.e., $\cup_{i=1}^{N}\mathcal{D}_i$) is often a good representation of the testing dataset $\mathcal{D}_0$, so that the expected testing loss is well approximated by the training loss. Based on this assumption, the expected payoff of client i is given by:

$$\begin{aligned}&E_\xi[u_i(e_i,\boldsymbol{e}'_{-i},D_i,\boldsymbol{D}'_{-i},\gamma_i,\boldsymbol{\gamma}'_{-i})]\\&=E_\xi[r_i(e'_i,\boldsymbol{e}'_{-i},D'_i,\boldsymbol{D}'_{-i},\gamma'_i,\boldsymbol{\gamma}'_{-i},f(\mathbf{w}_T,\xi))]-c_le_i-Tc_p^iD_i\\&=\Omega(\boldsymbol{D}')-\Phi(D'_i)F(\mathbf{w}_T)+c_l-c_le_i-Tc_p^iD_i\end{aligned}\tag{8}$$

where ξ is a random data sample drawn from the testing dataset $\mathcal{D}_0$.

Next, based on Theorem 1, we approximate the expected training loss $F(\mathbf{w}_T)$ in terms of the optimal training loss $F(\mathbf{w}^*)$ plus the upper bound on the training loss gap given in the right-hand-side of (6). Then we assume that each client uses $\hat{u}_i$ as her expected payoff function, where $\hat{u}_i$ is defined as (8) with $F(\mathbf{w}_T)$ replaced by the right-hand-side of (6) (the optimal training loss term $F(\mathbf{w}^*)$ is omitted as it does not affect the truthful mechanism design). This is a reasonable assumption since 1) a client cannot find the expected training loss $F(\mathbf{w}_T)$, but can find the upper bound in (6); 2) using the upper bound on the training loss gap can capture the worst case of the client's expected payoff. Therefore, in the rest of this paper, each client determines her strategic behavior for maximizing the payoff function $\hat{u}_i$.

Next, we use two theorems to prove that the LCEME mechanism satisfies the truthful and IR properties, with respect to the clients' payoff functions $\hat{u}_i$.

Theorem 2: The LCEME mechanism is truthful.

We show how the LCEME mechanism achieves the truthful property using three lemmas.

Lemma 1: Under the LCEME mechanism, given that client i makes any data labeling effort e_i and computation effort D_i, her optimal reported local model is her true local model, i.e., $\gamma_i=1$.

It can be shown that the expected payoff of client i is a convex function of γ_i. We can obtain the result of Lemma 1 by calculating the partial derivative of the expected payoff of client i with respect to γ_i and letting the derivative equal 0.

Using Lemma 1, we can express client i's approximated expected payoff $\hat{u}_i$ as

$$\begin{aligned}\hat{u}_i(e_i,D_i,D'_i)&=\Phi(D'_i)A({p_i}^2\frac{\sigma_i^2}{D'_i}+2p_i(H-1)^2\frac{\sigma_i^2}{D'_i})+Tc_p^iD'_i\\&-\Phi(D'_i)A\left(p_i^2\frac{\sigma_i^2}{D_i}+p_i(1-e_i)\beta\right.\\&\left.+2p_i(H-1)^2(\frac{\sigma_i^2}{D_i}+(1-e_i)\beta)\right)+c_l-c_le_i-Tc_p^iD_i.\end{aligned}$$

Lemma 2: Under the LCEME mechanism, given that clients report their optimal local models $\gamma_i=1,\forall i\in\mathcal{N}$, and client i makes any computation effort, client i's optimal actual effort is the desired effort, i.e., $e_i=1$.

Then, we show that, when client i makes any labeling effort, her expected payoff is always lower than that when she makes effort:

$$\begin{aligned}&\hat{u}_i(1,D_i,D'_i)-\hat{u}_i(e_i,D_i,D'_i)\\=&\frac{{D'_i}^2c_p^iT(1+2(H-1)^2)}{\sigma_i^2p_i^2(p_i+2(H-1)^2)}p_i(1-e_i)\beta-c_l+c_le_i\\=&(\frac{{D'_i}^2c_p^iT(1+2(H-1)^2)\beta}{\sigma_i^2p_i(p_i+2(H-1)^2)}-c)(1-e_i)\\\geq&(c-c)(1-e_i)\geq 0,\end{aligned}$$

where the inequality follows from the constraint on D'_i.

Using Lemma 1 and Lemma 2, we can express client i's approximated expected payoff $\hat{u}_i$ as

$$\begin{aligned}\hat{u}_i(D_i,D'_i)=&-\Phi(D'_i)A(p_i^2\frac{\sigma_i^2}{D_i}+2p_i(H-1)^2\frac{\sigma_i^2}{D_i})-Tc_p^iD_i\\&+\Phi(D'_i)A({p_i}^2\frac{\sigma_i^2}{D'_i}+2p_i(H-1)^2\frac{\sigma_i^2}{D'_i})+Tc_p^iD'_i.\end{aligned}$$

Lemma 3: Given that clients report their optimal local models $\gamma_i=1$ and make effort in data labeling $e_i=1$, $\forall i\in\mathcal{N}$, client i's optimal actual computation effort is the desired computation effort, i.e., $D_i=D'_i$.

Now that the expected payoff is a convex function of client i's actual computation effort D_i, we can obtain client i's optimal actual computation effort D_i by calculating the partial derivative of the expected payoff of client i with respect to D_i and letting the derivative equal to 0, which is the desired computation effort D'_i.

Given the definition of truthful mechanisms (Definition 1), the LCEME mechanism is truthful. □

Theorem 3: The LCEME mechanism is IR.

The proof is given in Appendix B.

Remark 3: Here we discuss the rationale of the LCEME mechanism. The server's goal is to incentivize clients to make actual data labeling and computation effort as desired by the server and report their true local models. Thus, client i's reward function r_i should be a function of her actual efforts (e_i and D_i) and model report coefficient (γ_i). Otherwise, clients can deceive the server to gain more rewards. Thus, we design the reward function as a function of the training loss, which has been proved to be determined by clients' actual efforts and model reporting strategies in Theorem 1. In the refined reward function, client i's optimal strategy to maximize her expected payoff is to make data labeling and computation efforts as desired by the server and report her actual local model.

B. Optimal Computation Effort Assignment

A desirable objective for the server is to find the optimal assignment that maximizes her expected payoff.

Definition 4: The server's optimal assignment $\boldsymbol{D}^*$ for LCEME mechanism is the assignment function $\boldsymbol{D}'$ that maximizes the server's payoff, i.e.,

$$\begin{aligned}&\boldsymbol{D}^*\triangleq\arg\max_{\boldsymbol{D}'}E[u_0(\boldsymbol{D}',f(\mathbf{w}_T,\xi))]\\&s.t.\quad D_i^*\geq\sqrt{\frac{c_l\sigma_i^2p_i(p_i+2(H-1)^2)}{\beta c_p^iT(1+2(H-1)^2)}},\forall i\in\mathcal{N}.\end{aligned}\tag{9}$$

The constraint in (9) is to make sure that the LCEME mechanism is truthful.

The problem given in (9) is equivalent to the problem:

$$\begin{aligned} \boldsymbol{D}^* \triangleq \arg\min_{\boldsymbol{D}'} \left\{ F(\mathbf{w}_T) - F(\mathbf{w}^*) + \sum_{i\in\mathcal{N}} r_i \right\}, \\ s.t.\ \ D_i^* \geq \sqrt{\frac{c_l \sigma_i^2 p_i (p_i + 2(H-1)^2)}{\beta c_p^i T (1 + 2(H-1)^2)}}, \forall i \in \mathcal{N}, \end{aligned} \tag{10}$$

where $F(\mathbf{w}^*)$ can be seen as a constant.

From the above problem formulation, we observe that there exists a tradeoff between the FL training loss and the server's payment to clients. We know that the training loss reduces when clients use larger mini-batch sizes to compute their local updates from Theorem 1. However, using larger mini-batch sizes increases the server's payment. Therefore, we aim to find the optimal computation effort (in the form of mini-batch size) assignment for each client to maximize the server's payoff.

Theorem 4: The server's optimal computation effort allocation is given by

$$D_i^* = \max\left\{ \sqrt{\frac{A(p_i^2\sigma_i^2 + 2p_i(H-1)^2)}{c_p^i T}}, \sqrt{\frac{c_l \sigma_i^2 p_i (p_i + 2(H-1)^2)}{\beta c_p^i T (1 + 2(H-1)^2)}} \right\}, \forall i \in \mathcal{N}.$$

The proof is given in Appendix C.

Remark 4: From Theorem 4, we can see that the server's optimal computation effort for a client i increases with her weight p_i and gradient variance σ_i^2. This is because when client i has a larger p_i and/or σ_i^2, the effect of the randomness of her SGD computation per data sample on the global model will be larger. From Theorem 1, we know that a larger mini-batch size D_i reduces the randomness of data sampling in SGD. Thus, assigning a larger computation effort for client i can reduce the training loss. We also see that D_i^* decreases as client i's computation cost c_p^i increases. This is because a larger computation cost increases the reward paid by the server. When a client's computation cost is large, the server prefers to allocate a smaller mini-batch size to the client to reduce the payment. We can also show that a client's optimal mini-batch size increases as the number of local iterations H increases. This is because a local update's quality can be improved by using a larger mini-batch size, and thus reduce the error caused by performing multiple local iterations.

VI. Simulation Results

In this section, we conduct real data based simulations to validate the theoretical findings and evaluate the LCEME mechanism. We first describe the simulation setups, and then we present the evaluation results and analyses.

We implement a simulated system consisting of a server and 10 clients. We use the widely used MNIST dataset [41] for simulations in Matlab. Each training element is a handwritten digit picture that represents numbers from 0 to 9. Each client conducts one layer of CNN for one local iteration in each round ($H = 1$). We denote the heterogeneity degree of a client's dataset as the percentage of data with labels the same as the last digit of the client's index. For the remaining data of the client, we uniformly draw the training data samples from the entire training set. Unless otherwise specified, client i's heterogeneity degree is 0.4, and the mini-batch size is $D_i = 50$.

A. Impact of Clients' Strategies on Training Loss

We first compare the training loss while clients' data labeling and computation efforts changes. From Figs. 2 and 3, we can see that the training loss decreases and the model accuracy increases as D_i increases. We also observe that when there exist clients who make no effort in data labeling, the training loss increases, and the model accuracy decreases. The observations conform to our theoretical result in Theorem 1. We also compare the training loss while clients report local models with different model reporting coefficients and truthfully make efforts. We observe from Figs. 4 and 5 that the training loss is minimized when all clients report their actual local model. When there exist clients report local model untruthfully, the training loss increases, and the model accuracy decreases. This conforms to the result in Theorem 1 that the more clients truthfully report local models, the lower the training loss. We also observe that, although the training loss bounds are the same when $\gamma_i = 0$ and $\gamma_i = 1$, the training loss is lower when $\gamma = 0$. Figs. 2, 3, 4, and 5 demonstrate that, when clients truthfully make efforts and report local models, the training loss is minimized and the model accuracy is maximized.

B. Impact of Truthfulness on Clients' Payoff

We compare a client's payoff while making the desired data labeling effort $e_1 = 1$ or not $e_1 = 0$, and reporting the actual local model $\gamma_1 = 1$ or not $\gamma_1 \neq 1$, as the computation effort D_1 changes. The assigned computation effort $D_1' = 60$. We let other clients behave truthfully. We observe from Fig. 6 that a client's payoff, when she makes data labeling and computation effort as the server desired and reports actual local model, is always higher than that when her behavior is untruthful. Furthermore, we also observe that the client's payoff is positive when she behaves truthfully. The simulation results demonstrate that the LCEME mechanism is truthful and achieves the IR property.

C. Server's Payoff

We compare the server's payoff while clients make different computation efforts. From Fig. 7, we can see that the server's payoff is maximized when clients make the server's optimal computation effort. When clients do not make the optimal computation effort, the server's payoff is lower even if the total computation effort of clients is the same as the optimal computation effort allocation. This is because, in the former case, the computation effort allocation does not care about clients' heterogeneous computation cost and thus causes higher computation costs. We also simulate the case where clients' computation effort $D_i = 100$ is always higher than the optimal computation effort. We observe that among three

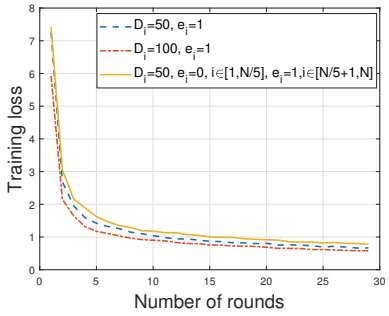

Fig. 2. Impact of effort level on the training loss.

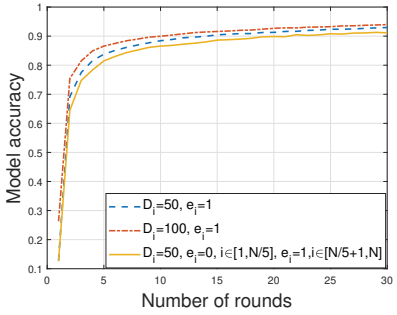

Fig. 3. Impact of effort level on the model accuracy.

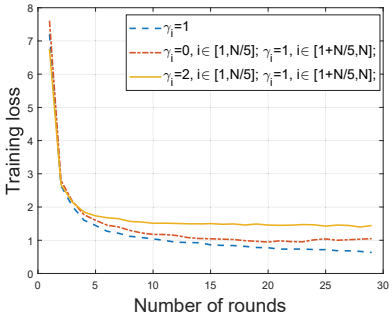

Fig. 4. Impact of model reporting coefficient on the training loss.

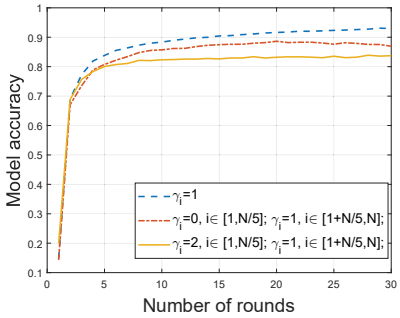

Fig. 5. Impact of model reporting coefficient on the model accuracy.

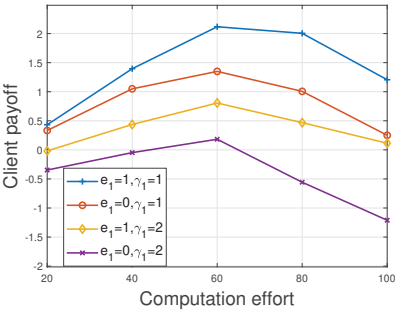

Fig. 6. Impact of clients' behavior on the payoff.

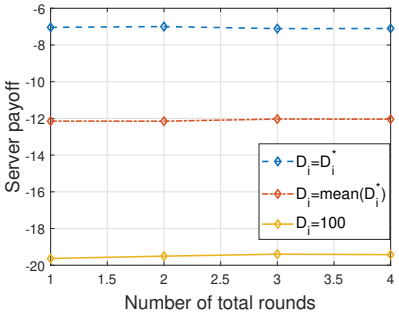

Fig. 7. Impact of computation effort allocation on server's payoff.

cases, this case results in the lowest server's payoff. This is because clients' computation costs are ignored when assigning D_i, resulting in an increase in the server cost.

VII. Conclusion and Future Work

In this paper, we studied FL with crowdsourced data labels, where the local data of each participating client are labeled manually by the client. We characterized the performance bounds on the training loss as a function of clients' data labeling effort, local computation effort, and reported local models. We then devised truthful incentive mechanisms which motivate strategic clients to make truthful efforts as desired by the server in data labeling and local model computation, and also report true local models to the server based on the derived performance bound. Simulations based on real data demonstrated the efficacy of the proposed algorithms.

For future work, we will extend our study to more general settings. In this paper, we studied truthful mechanism design under the assumption that clients' costs are known to the server. The mechanism design problem where clients' costs are also private is more practical but challenging. Another direction is to consider partial participation of clients. In this case, the truthful mechanism design and the optimal labeling effort assignment will be different.

Appendix

A. Proof of Theorem 1

We define a virtual sequence $\bar{\mathbf{w}}_{t,h}$, given by $\bar{\mathbf{w}}_{t,h} = \sum_{i\in\mathcal{N}} p_i \mathbf{w}^i_{t,h}, \forall t, h$. Note that $\bar{\mathbf{w}}_{t,h}$ is not accessible when clients have not completed H local iterations (i.e., $h < H$), and $\mathbf{w}_t = \bar{\mathbf{w}}_{t,H}$.

$$\|\bar{\mathbf{w}}_{t,H} - \mathbf{w}^*\|^2 = \left\|\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^* - \eta \sum_{i\in\mathcal{N}} \gamma_i p_i g^{i}_{t,H-1}{}'\right\|^2 \leq$$
$$2\underbrace{\|\bar{\mathbf{w}}_{t,H-1} - \eta\bar{g}_{t,H-1} - \mathbf{w}^*\|^2}_{A_1} + 2\underbrace{\left\|\eta\bar{g}_{t,H-1} - \eta\sum_{i\in\mathcal{N}} \gamma_i p_i g^{i}_{t,H-1}{}'\right\|^2}_{A_2} \tag{11}$$

where $g^{i}_{t,h}{}'$ is the gradient when client i makes any data labeling effort, $\bar{g}_{t,h} \triangleq \sum_{i\in\mathcal{N}} p_i \bar{g}^i_{t,h} \triangleq \sum_{i\in\mathcal{N}} p_i E[g^i_{t,h}]$, and $g^i_{t,h}$ is the gradient when client i makes data labeling effort.

$$A_1 = \|\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^*\|^2 + \underbrace{\eta^2\|\bar{g}_{t,H-1}\|^2}_{B_1} - \underbrace{2\eta\langle\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^*, \bar{g}_{t,H-1}\rangle}_{B_2}. \tag{12}$$

For B_2, we have

$$B_2 = -2\eta\sum_{i\in\mathcal{N}} p_i\langle\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^i_{t,H-1}, \bar{g}^i_{t,H-1}\rangle - 2\eta\sum_{i\in\mathcal{N}} p_i\langle\mathbf{w}^i_{t,H-1} - \mathbf{w}^*, \bar{g}^i_{t,H-1}\rangle.$$

We use the convexity of $\|\cdot\|^2$ and the L-smoothness of F_i to bound B_1, the Cauchy-Schwarz inequality and AM-GM inequality to bound the first term of B_2, and the μ-strong convexity of F_i to bound the second term of B_2. We have

$$A_1 \leq \|\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^*\|^2 + 2L\eta^2\sum_{i\in\mathcal{N}} p_i(F_i(\mathbf{w}^i_{t,H-1}) - F_i(\mathbf{w}^*_i)) + \sum_{i\in\mathcal{N}} p_i\left(\left\|\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^i_{t,H-1}\right\|^2 + \eta^2\left\|\bar{g}^i_{t,H-1}\right\|^2\right)$$

$$
\begin{aligned}
&-2\eta \sum_{i\in\mathcal{N}} p_i \left(F_i(\mathbf{w}^i_{t,H-1}) - F_i(\mathbf{w}^*) + \frac{\mu}{2}\left\|\mathbf{w}^i_{t,H-1} - \mathbf{w}^*\right\|^2\right)\\
\le& (1-\mu\eta)\left\|\mathbf{w}_{t,H-1} - \mathbf{w}^*\right\|^2 + \sum_{i\in\mathcal{N}} p_i \left\|\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^i_{t,H-1}\right\|^2\\
&+ 4L\eta^2 \sum_{i\in\mathcal{N}} p_i (F_i(\mathbf{w}^i_{t,H-1}) - F_i(\mathbf{w}^*_i))\\
&-2\eta \sum_{i\in\mathcal{N}} p_i \left(F_i(\mathbf{w}^i_{t,H-1}) - F_i(\mathbf{w}^*)\right),
\end{aligned}
$$

in which we denote the last two lines as C_1.

$$
\begin{aligned}
C_1 =& 4L\eta^2 \sum_{i\in\mathcal{N}} p_i (F_i(\mathbf{w}^*) - F_i(\mathbf{w}^*_i))\\
&- 2\eta(1-2L\eta) \sum_{i\in\mathcal{N}} p_i (F_i(\mathbf{w}^i_{t,H-1}) - F_i(\mathbf{w}^*))\\
\le& 4L\eta^2 \sum_{i\in\mathcal{N}} p_i d_i - 2\eta(1-2L\eta)\left(-\sum_{i\in\mathcal{N}} p_i\right.\\
&(\eta L\left(F_i(\bar{\mathbf{w}}_{t,H-1} - F_i(\mathbf{w}^*_i)\right) + \frac{1}{2\eta}\left\|\mathbf{w}^i_{t,H-1} - \bar{\mathbf{w}}_{t,H-1}\right\|^2\\
&\left.+F_i(\bar{\mathbf{w}}_{t,H-1}) - F_i(\mathbf{w}^*))\right).\\
\le& 2\eta(1-2L\eta)(\eta L - 1)\sum_{i\in\mathcal{N}} p_i \left(F_i(\bar{\mathbf{w}}_{t,H-1} - F_i(\mathbf{w}^*_i)\right)\\
&+ (4L\eta^2 + 2L\eta^2(1-2L\eta))\sum_{i\in\mathcal{N}} p_i d_i\\
&+ (1-2L\eta)\sum_{i\in\mathcal{N}} p_i \left\|\mathbf{w}^i_{t,H-1} - \bar{\mathbf{w}}_{t,H-1}\right\|^2\\
\le& 6L\eta^2 \sum_{i\in\mathcal{N}} p_i d_i + \sum_{i\in\mathcal{N}} p_i \left\|\mathbf{w}^i_{t,H-1} - \bar{\mathbf{w}}_{t,H-1}\right\|^2.
\end{aligned}
$$

Thus we can further bound A_1 as

$$
\begin{aligned}
E[A_1] \le& (1-\mu\eta)\left\|\bar{\mathbf{w}}_{t,H-1} - \mathbf{w}^*\right\|^2\\
&+ 6L\eta^2 \sum_{i\in\mathcal{N}} p_i d_i + 2\sum_{i\in\mathcal{N}} p_i \left\|\mathbf{w}^i_{t,H-1} - \bar{\mathbf{w}}_{t,H-1}\right\|^2. \qquad (13)
\end{aligned}
$$

Next, we bound

$$
\begin{aligned}
&\sum_{i\in\mathcal{N}} p_i E\left\|\bar{\mathbf{w}}_{t,h} - \mathbf{w}^i_{t,h}\right\|^2 \le \sum_{i\in\mathcal{N}} p_i E\left\|\mathbf{w}^i_{t,h} - \mathbf{w}_{t,1}\right\|^2\\
\le& \eta^2 \sum_{i\in\mathcal{N}} p_i E\| \sum_{h=1}^{H-1} {g^i_{t,h}}'\|^2 \le \eta^2(H-1)\sum_{i\in\mathcal{N}} p_i \sum_{h=1}^{H-1} E\|{g^i_{t,h}}'\|^2. \qquad (14)
\end{aligned}
$$

Using Assumption 4, we have

$$
\begin{aligned}
&E\|g^i_{t,h} - {g^i_{t,h}}'\|^2\\
=& E\|\frac{1}{D_i}\sum_j (\nabla f_i(\mathbf{w}_{t,h}, \xi^{i,j}_t) - \nabla f_i(\mathbf{w}_{t,h}, \xi^{i,j'}_t))\|^2\\
\le& \frac{1}{D_i}\sum_j E_{\xi^{i,j'}_t | \xi^{i,j}_t}\left[\|(\nabla f_i(\mathbf{w}_{t,h}, \xi^{i,j}_t) - \nabla f_i(\mathbf{w}_{t,h}, \xi^{i,j'}_t))\|^2\right]\\
\le& (1-e_i)\beta. \qquad (15)
\end{aligned}
$$

From [38], we have

$$
E\left\|\bar{g}^i_{t,h} - g^i_{t,h}\right\|^2 \le \frac{\sigma_i^2}{D_i}. \qquad (16)
$$

From (15), (16), and Assumption 5, we have

$$
\begin{aligned}
E\left\|{g^i_{t,h}}'\right\|^2 =& E\left\|{g^i_{t,h}}' - g^i_{t,h} + g^i_{t,h} - \bar{g}^i_{t,h} + \bar{g}^i_{t,h}\right\|^2\\
\le& 2E\left\|{g^i_{t,h}}' - g^i_{t,h}\right\|^2 + 2E\left\|g^i_{t,h} - \bar{g}^i_{t,h}\right\|^2 + 2E\left\|\bar{g}^i_{t,h}\right\|^2\\
\le& 2(1-e_i)\beta + \frac{2\sigma_i^2}{D_i} + 2G^2. \qquad (17)
\end{aligned}
$$

Thus we can bound (14) as

$$
\begin{aligned}
&\sum_{i\in\mathcal{N}} p_i E\left\|\bar{\mathbf{w}}_{t,h} - \mathbf{w}^i_{t,h}\right\|^2\\
\le& 2\eta^2(H-1)^2 \sum_{i\in\mathcal{N}} p_i((1-e_i)\beta + \frac{\sigma_i^2}{D_i} + G^2). \qquad (18)
\end{aligned}
$$

Next, we bound A_2. From (16) and (17), we have

$$
\begin{aligned}
E[A_2] =& \left\|\eta\bar{g}_{t,H-1} - \eta\sum_{i\in\mathcal{N}} \gamma_i p_i {g^i_{t,H-1}}'\right\|^2\\
=& \eta^2 E\left\|\bar{g}_{t,H-1} - g_{t,H-1} + g_{t,H-1} + \sum_{i\in\mathcal{N}} \gamma_i p_i {g^i_{t,H-1}}'\right\|^2\\
\le& 2\eta^2 E\left\|\bar{g}_{t,H-1} - g_{t,H-1}\right\|^2 + 2\eta^2 E\left\|g'_{t,H-1} - g_{t,H-1}\right\|^2\\
&+ 2\eta^2 \sum_{i\in\mathcal{N}} p_i(\gamma_i - 1)^2 E\left\|{g^i_{t,H-1}}'\right\|^2\\
\le& 2\eta^2 \sum_{i\in\mathcal{N}} ({p_i}^2 \frac{\sigma_i^2}{D_i} + p_i(1-e_i)\beta\\
&+ 2p_i(\gamma_i - 1)^2(G^2 + \frac{\sigma_i^2}{D_i} + (1-e_i)\beta)). \qquad (19)
\end{aligned}
$$

Combining (11), (13), (18), and (19), we have

$$
\begin{aligned}
&E\left\|\mathbf{w}_{T,H} - \mathbf{w}^*\right\|^2\\
\le& 2(1-\mu\eta)\left\|\mathbf{w}_{T,H-1} - \mathbf{w}^*\right\|^2 + 12L\eta^2 \sum_{i\in\mathcal{N}} p_i d_i\\
&+ 4\eta^2 \sum_{i\in\mathcal{N}} ({p_i}^2 \frac{\sigma_i^2}{D_i} + p_i(1-e_i)\beta)\\
&+ 4\eta^2 \sum_{i\in\mathcal{N}} p_i((\gamma_i - 1)^2 + 2(H-1)^2)(G^2 + \frac{\sigma_i^2}{D_i} + (1-e_i)\beta).
\end{aligned}
$$

Using induction and the smoothness of F, we have (6).

B. Proof of Theorem 3

Given that all users behave truthfully, the expected payoff of user i, $\forall i$ is given by

$$
\begin{aligned}
E[u_i] =& \Omega(\boldsymbol{D}') - \Phi(D'_i)F(\mathbf{w}_T) + c_l - c_l e'_i - T c^i_p D'_i.\\
\ge& \Phi(D'_i)(F(\mathbf{w}_T) - F(\mathbf{w}^*)) + T c^i_p D'_i\\
&- \Phi(D'_i)(F(\mathbf{w}_T) - F(\mathbf{w}^*)) + c_l - c_l e'_i - T c^i_p D'_i = 0.
\end{aligned}
$$

C. Proof of Theorem 4

The total expected reward paid by the server is bounded by $\sum_{i\in\mathcal{N}} r_i \ge \sum_{i\in\mathcal{N}}(c_l + T c^i_p D_i)$. Using (6), we have

$$
\begin{aligned}
&F(\mathbf{w}_T) - F(\mathbf{w}^*) + \sum_{i\in\mathcal{N}} r_i \le L(1-\mu\eta)^{TH} E\left\|\mathbf{w}_0 - \mathbf{w}^*\right\|^2\\
&+ \sum_{i\in\mathcal{N}} (A({p_i}^2 \frac{\sigma_i^2}{D_t^i} + 6Lp_i d_i + 2p_i(H-1)^2 \frac{\sigma_i^2}{D_t^i}) + c_l + T c^i_p D_i).
\end{aligned}
$$

It can be shown that the above upper bound is a convex function of D_i. The optimal mini-batch size ${D_i}^*$ can be obtained by calculating the partial derivative of the bound with respect to D_i and letting the derivative equals to 0.

REFERENCES

[1] B. McMahan and D. Ramage, "Federated learning: Collaborative machine learning without centralized training data," *Google Research Blog*, vol. 3, 2017.

[2] N. H. Tran, W. Bao, A. Zomaya, N. M. NH, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *International Conference on Computer Communications (INFOCOM)*. IEEE, 2019.

[3] H. H. Yang, Z. Liu, T. Q. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 317–333, 2019.

[4] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, 2020.

[5] J. Xu and H. Wang, "Client selection and bandwidth allocation in wireless federated learning networks: A long-term perspective," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1188–1200, 2020.

[6] W. Shi, S. Zhou, Z. Niu, M. Jiang, and L. Geng, "Joint device scheduling and resource allocation for latency constrained wireless federated learning," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 453–467, 2020.

[7] J. Ren, Y. He, D. Wen, G. Yu, K. Huang, and D. Guo, "Scheduling for cellular federated edge learning with importance and channel awareness," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7690–7703, 2020.

[8] S. Wang, M. Lee, S. Hosseinalipour, R. Morabito, M. Chiang, and C. G. Brinton, "Device sampling for heterogeneous federated learning: Theory, algorithms, and implementation," in *International Conference on Computer Communications (INFOCOM)*. IEEE, 2021.

[9] J. Zhang, N. Li, and M. Dedeoglu, "Federated learning over wireless networks: A band-limited coordinated descent approach," in *International Conference on Computer Communications (INFOCOM)*. IEEE, 2021.

[10] R. H. L. Sim, Y. Zhang, M. C. Chan, and B. K. H. Low, "Collaborative machine learning with incentive-aware model rewards," in *International Conference on Machine Learning*. PMLR, 2020, pp. 8927–8936.

[11] H. Yu, Z. Liu, Y. Liu, T. Chen, M. Cong, X. Weng, D. Niyato, and Q. Yang, "A sustainable incentive scheme for federated learning," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 58–69, 2020.

[12] P. Sun, H. Che, Z. Wang, Y. Wang, T. Wang, L. Wu, and H. Shao, "Pain-fl: Personalized privacy-preserving incentive for federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3805–3820, 2021.

[13] M. Zhang, E. Wei, and R. Berry, "Faithful edge federated learning: Scalability and privacy," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3790–3804, 2021.

[14] S. R. Pandey, N. H. Tran, M. Bennis, Y. K. Tun, A. Manzoor, and C. S. Hong, "A crowdsourcing framework for on-device federated learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3241–3256, 2020.

[15] Y. Jiao, P. Wang, D. Niyato, B. Lin, and D. I. Kim, "Toward an automated auction framework for wireless federated learning services market," *IEEE Transactions on Mobile Computing*, vol. 20, no. 10, pp. 3034–3048, 2020.

[16] K. Donahue and J. Kleinberg, "Model-sharing games: Analyzing federated learning under voluntary participation," in *AAAI Conference on Artificial Intelligence*, 2021.

[17] ——, "Optimality and stability in federated learning: A game-theoretic approach," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[18] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 700–10 714, 2019.

[19] N. Ding, Z. Fang, and J. Huang, "Optimal contract design for efficient federated learning with multi-dimensional private information," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 186–200, 2020.

[20] M. Zhang, E. Wei, and R. Berry, "Faithful edge federated learning: Scalability and privacy," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3790–3804, 2021.

[21] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand, "Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2012.

[22] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2012.

[23] I. Koutsopoulos, "Optimal incentive-driven design of participatory sensing systems," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2013.

[24] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "Trac: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2014.

[25] A. Tarable, A. Nordio, E. Leonardi, and M. A. Marsan, "The importance of being earnest in crowdsourcing systems," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2015.

[26] N. B. Shah and D. Zhou, "Double or nothing: Multiplicative incentive mechanisms for crowdsourcing," in *Conference on Neural Information Processing Systems (NIPS)*, 2015.

[27] Y. Luo, N. B. Shah, J. Huang, and J. Walrand, "Parametric prediction from parametric agents," in *The 10th Workshop on the Economics of Networks, Systems and Computation (NetEcon)*, 2015.

[28] W. Wang, L. Ying, and J. Zhang, "The value of privacy: Strategic data subjects, incentive mechanisms and fundamental limits," in *ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, 2016.

[29] L. Pu, X. Chen, J. Xu, and X. Fu, "Crowdlet: Optimal worker recruitment for self-organized mobile crowdsourcing," in *IEEE Conference on Computer Communications (INFOCOM)*, 2016.

[30] H. Zhang, B. Liu, H. Susanto, G. Xue, and T. Sun, "Incentive mechanism for proximity-based mobile crowd service systems," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2016.

[31] P. Bolton and M. Dewatripont, *Contract theory*. MIT press, 2005.

[32] A. Dasgupta and A. Ghosh, "Crowdsourced judgement elicitation with endogenous proficiency," in *International World Wide Web Conference (WWW)*, 2013.

[33] Y. Cai, C. Daskalakis, and C. H. Papadimitriou, "Optimum statistical estimation with strategic data sources," in *Conference on Learning Theory (COLT)*, 2015.

[34] Y. Liu and Y. Chen, "Learning to incentivize: Eliciting effort via output agreement," *International Joint Conference on Artificial Intelligence (IJCAI)*, 2016.

[35] ——, "Sequential peer prediction: Learning to elicit effort using posted prices." in *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[36] D. Prelec, "A Bayesian truth serum for subjective data," *Science*, vol. 306, no. 5695, pp. 462–466, 2004.

[37] H. Jin, L. Su, and K. Nahrstedt, "Theseus: Incentivizing truth discovery in mobile crowd sensing systems," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2017.

[38] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao, "Optimal distributed online prediction using mini-batches," *The Journal of Machine Learning Research*, vol. 13, pp. 165–202, 2012.

[39] J. Song, M.-h. Oh, and H.-S. Kim, "Personalized federated learning with server-side information," *arXiv preprint arXiv:2205.11044*, 2022.

[40] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," in *International Conference on Learning Representations (ICLR)*, 2020.

[41] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010. [Online]. Available: http://yann.lecun.com/exdb/mnist/