An Incentive Mechanism for Cross-Silo Federated Learning: A Public Goods Perspective

Ming Tang and Vincent W.S. Wong
Department of Electrical and Computer Engineering
The University of British Columbia, Vancouver, Canada
email: {mingt, vincentw}@ece.ubc.ca

***Abstract*—In cross-silo federated learning (FL), organizations cooperatively train a global model with their local data. The organizations, however, may be heterogeneous in terms of their valuation on the precision of the trained global model and their training cost. Meanwhile, the computational and communication resources of the organizations are non-excludable public goods. That is, even if an organization does not perform any local training, other organizations cannot prevent that organization from using the outcome of their resources (i.e., the trained global model). To address the organization heterogeneity and the public goods feature, in this paper, we formulate a social welfare maximization problem and propose an incentive mechanism for cross-silo FL. With the proposed mechanism, organizations can achieve not only social welfare maximization but also individual rationality and budget balance. Moreover, we propose a distributed algorithm that enables organizations to maximize the social welfare without knowing the valuation and cost of each other. Our simulations with MNIST dataset show that the proposed algorithm converges faster than a benchmark method. Furthermore, when organizations have higher valuation on precision, the proposed mechanism and algorithm are more beneficial in the sense that the organizations can achieve higher social welfare through participating in cross-silo FL.**

***Index Terms*—Federated learning, incentive mechanism, game theory, public goods, resource allocation.**

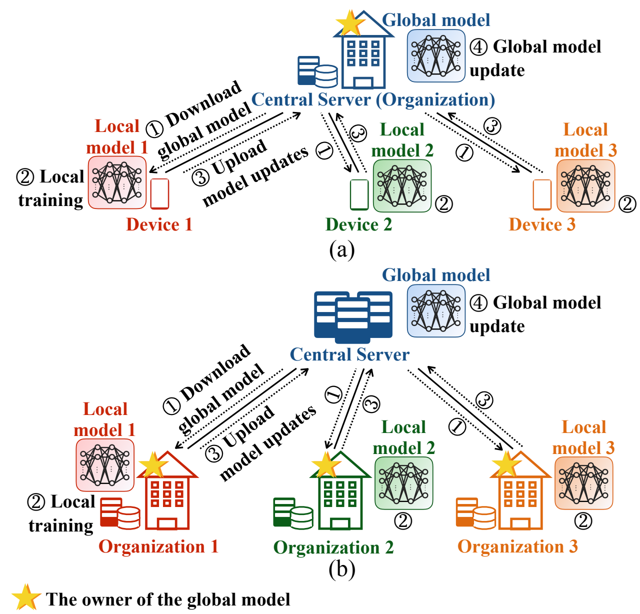


Fig. 1. An illustration of (a) cross-device FL and (b) cross-silo FL.

I. Introduction

A. Background and Motivation

Federated learning (FL) [1] is a decentralized machine learning approach. In FL, multiple clients cooperatively train a global model with their local data under the coordination of a central server. During the training phase, each client periodically downloads the global model from the central server, updates its local model by training the downloaded global model with its local data, and uploads the model updates to the central server for global model updating. Since each client does not need to transfer its local data to the central server, data privacy can be preserved. FL can be classified into two types [1]: cross-device FL and cross-silo FL. In cross-device FL, as shown in Fig. 1 (a), an organization (e.g., company, institution) acts as the central server. This organization is the owner of the global model. That is, it initiates the FL and owns the trained global model. The devices are the clients and perform local training. On the other hand, in cross-silo FL, as shown in Fig. 1 (b), a third party entity acts as the central server and is responsible for the coordination of training. A set of organizations act as the clients to perform local training. They are also the owners of the global model and can make use of the trained global model.

In this work, we focus on cross-silo FL. There have been various industrial applications of cross-silo FL. Owkin [2] cooperates with medical institutions for biomedical data analysis. WeBank and Swiss Re [3] collaborate for data analysis in finance and insurance. Other industrial examples include the MELLODDY [4] for drug discovery and FeatureCloud [5] for medical data mining. Meanwhile, algorithms have been developed to enable cross-silo FL. McMahan *et al.* [6] proposed the federated average (FedAvg) algorithm based on stochastic gradient descent. Built upon the FedAvg algorithm, some recent works have proposed algorithms to improve the convergence rate [7]–[9], accuracy [10], [11], and security [12]. Some other works considered resource optimization problem for FL, e.g., [13], [14]. Kairouz *et al.* in [1] and Lim *et al.* in [15] provided comprehensive surveys in the area of FL.

In cross-silo FL, each organization can choose the processing capacity that it allocates for local training. The processing capacities from different organizations will affect the *precision* of the trained global model, i.e., the degree that the trained global model fits the local data of the organizations. The selection of the processing capacity will also affect the computational cost of the organizations. Moreover, the communication cost of each organization depends on how often it needs to exchange model updates with the central server. On the other hand, the organizations may be heterogeneous in terms of their valuation on the precision (i.e., the degree that an organization is concerned about the precision of the trained global model) as well as the computational and communication costs. Hence, the selection of the processing capacity has to be optimized to ensure efficient cooperation, i.e., maximize the social welfare. However, the organizations are independent entities and may even be business competitors. Each organization may be self-interest and may not aim at optimizing the social welfare. As a result, it is necessary to design an incentive mechanism to motivate efficient cooperation in cross-silo FL.

There are various challenging issues to address for the design of an incentive mechanism for cross-silo FL. First, the operation of the incentive mechanism should not rely on the private information (e.g., valuation on precision, costs) of the organizations. Second, the mechanism has to address the fact that the computational and communication resources in cross-silo FL provided by the organizations are *public goods* [16]. Specifically, public goods are both *non-excludable* (i.e., individuals cannot be excluded from using them) and *non-rivalrous* (i.e., there is no competition in terms of using them). An example of public goods is radio broadcast, where everyone can receive the radio signal within a certain area without competition. For cross-silo FL, the computational and communication resources are non-excludable in the sense that even if an organization does not perform any local training, other organizations cannot exclude that organization from using the outcome of their resources, i.e., the trained global model. The resources are non-rivalrous since there is no competition in terms of using the outcome of the resources. In contrast to public goods, *private goods* are excludable (i.e., individuals can be excluded from using the goods) and rivalrous (i.e., there is competition in using the goods). For example, when a person owns a laptop, other people can be prevented from using it (i.e., excludable). If the person is using the laptop, others cannot use it (i.e., rivalrous). Many incentive mechanisms in the literature are for private goods, such as the auction mechanisms (e.g., [17]), contract based mechanisms (e.g., [18]), and pricing based mechanisms (e.g., [19]). Those mechanisms are not applicable in cross-silo FL due to the non-excludable and non-rivalrous features of the resources.

Despite the fact that cross-silo FL has various industrial applications, there is a lack of comprehensive understanding on the incentive mechanism for cross-silo FL. Although Lim *et al.* in [20] proposed a coalition game based incentive mechanism that can be applicable to cross-silo FL, it does not consider how much processing capacity that the organizations should use for local training. On the other hand, there are existing works on the incentive mechanism design for cross-device FL [21]–[23]. For example, Pandey *et al.* [21] proposed a pricing scheme to motivate the devices to achieve certain training precision. Kang *et al.* in [22] considered a contract based algorithm to incentivize devices with high quality data to participate in FL. Zhang *et al.* in [23] proposed a deep reinforcement learning (DRL) based algorithm to determine the optimal pricing strategy. However, those incentive mechanisms for cross-device FL cannot be applied to cross-silo FL. This is because in those mechanisms for cross-device FL, the central server (as the owner of the global model) has to provide payment to the devices for FL training. In cross-silo FL, however, the central server does not own the global model, so it does not need to pay the organizations. In addition, the public goods feature in cross-silo FL should be addressed.

In other application scenarios, there are existing works on public goods. For example, Kakhbod *et al.* [24] proposed an efficient incentive mechanism for multicasting in wireless networks. Zhang *et al.* [25] designed an incentive mechanism for wireless powered networks. Those mechanisms, however, cannot be directly applied in cross-silo FL. First, those works considered the scenario with one producer and multiple consumers. In cross-silo FL, however, each organization is both a producer (who contributes resources) and a consumer (who utilizes the trained global model). Second, the social welfare maximization problem in cross-silo FL is nonconvex, which poses additional challenges for incentive mechanism design.

B. Solution and Contributions

To motivate efficient cooperation, in this paper, we propose an incentive mechanism for cross-silo FL that addresses the challenges resulting from the private information of the organizations and the public goods feature. The proposed mechanism helps the organizations to answer the following questions: (a) How much processing capacity should each organization allocate for local training? (b) How much should each organization be compensated (by other organizations) for its local training? Moreover, based on the proposed mechanism, we propose a distributed algorithm that enables the organizations to maximize the social welfare in a decentralized manner. The main contributions of this paper are as follows:

- Incentive Mechanism Design Problem for Cross-Silo FL: We first formulate a social welfare maximization problem for cross-silo FL, which is a nonconvex problem. Then, we formulate an incentive mechanism design problem, taking into account the heterogeneity of the organizations and the public goods feature.
- Incentive Mechanism: We propose an incentive mechanism for cross-silo FL. The proposed mechanism addresses the public goods feature, and its operation does not rely on the private information of the organizations. Given the proposed mechanism, the strategic interaction among the organizations is modeled as a non-cooperative game with perfect information (assuming that the organizations know the private information of each other). We prove that the Nash equilibrium (NE) of the game, i.e., the

strategy profile such that no organization has an incentive to deviate, can lead to several properties. These include *social efficiency* (i.e., the social welfare is maximized), *individual rationality* (i.e., each organization is no worse off by participating in cross-silo FL), and *budget balance* (i.e., no third party investment is required).

- Distributed Algorithm: Based on our proposed incentive mechanism, we further propose a distributed algorithm that enables the organizations to achieve the NE of the game *without* knowing the private information of each other. That is, the algorithm solves the social welfare maximization problem in a decentralized manner and maintains the individual rationality and budget balance.
- Performance Evaluation: We conduct simulations with MNIST dataset [26]. The simulation results show that the proposed algorithm converges faster than the conventional Lagrangian method [27, Section 8.1]. Moreover, with the proposed mechanism and algorithm, organizations can increase the social welfare through participating in cross-silo FL. This enhancement is more significant when the organizations have higher valuation on precision.

This paper is organized as follows. We present the system model in Section II and the incentive mechanism in Section III. The distributed algorithm is given in Section IV. We conduct simulations in Section V and conclude in Section VI. We use $\mathbb{R}$, $\mathbb{R}_+$, and $\mathbb{Z}_+$ to denote the sets of real numbers, nonnegative real numbers, and nonnegative integers, respectively.

II. System Model

We consider a scenario with N organizations. Let $\mathcal{N} = \{0, \ldots, N-1\}$ denote the set of organizations. Each organization has its own local data. Let $\mathcal{S}_n$ denote the collected dataset of organization $n \in \mathcal{N}$. Let S_n denote the number of data units in set $\mathcal{S}_n$, i.e., $S_n = |\mathcal{S}_n|$. The organizations train a global model using cross-silo FL to learn the collected data. Let $\boldsymbol{\omega}$ denote the weights of the global model. The organizations aim to find the optimal weights of the global model $\boldsymbol{\omega}^*$ that minimize the expected loss $L(\boldsymbol{\omega})$ over the datasets [6], [9]:

$$\boldsymbol{\omega}^* = \arg\min_{\boldsymbol{\omega}} \left\{ L(\boldsymbol{\omega}) \triangleq \sum_{n\in\mathcal{N}} \frac{S_n}{\sum_{n'\in\mathcal{N}} S_{n'}} l(\boldsymbol{\omega}; \mathcal{S}_n) \right\}, \quad (1)$$

where $l(\boldsymbol{\omega}; \mathcal{S}_n)$ is the loss over dataset $\mathcal{S}_n$ given $\boldsymbol{\omega}$. On the other hand, the organizations may have different valuation on the precision of the trained global model and have different computational and communication costs. Thus, some organizations may need to pay other organizations to compensate the cost of the latter ones. Otherwise, some organizations may not cooperate.

In the following, we first present the FL model and the payoff of the organizations. We then formulate the social welfare maximization and incentive mechanism design problems.

A. FL Model

The organizations cooperate to train a global model with their collected datasets using the widely applied FedAvg algorithm [6].[1] As shown in Fig. 1 (b), a central server helps the organizations with the training. The central server maintains the global model. Each organization has a local model, which has the same neural network structure as the global model.

At the beginning of the algorithm, the central server first randomly initializes the weights of the global model $\boldsymbol{\omega}^0$. The algorithm iterates for multiple training rounds. Let $\boldsymbol{\omega}^r$ and $\boldsymbol{\omega}_n^r$ denote the weights of the global model and the local model of organization $n \in \mathcal{N}$ in training round r, respectively. In training round r, each organization n downloads the previous weights of the global model $\boldsymbol{\omega}^{r-1}$ from the central server. It then performs K local updates over the downloaded global model with its dataset $\mathcal{S}_n$, where each local update corresponds to a mini-batch stochastic gradient descent [6]. The updated model is the local model of organization n, where $\boldsymbol{\omega}_n^r$ is the weights of the local model. After that, organization n uploads $\boldsymbol{\omega}_n^r$ to the central server. The central server updates the global model by taking an average over the received weights from all organizations, i.e., $\boldsymbol{\omega}^r = \sum_{n\in\mathcal{N}} S_n \boldsymbol{\omega}_n^r / (\sum_{n'\in\mathcal{N}} S_{n'})$. Note that the central server knows the values of S_n for $n \in \mathcal{N}$ [6].

Let f_n (in CPU cycles per second) denote the processing capacity used by organization $n \in \mathcal{N}$ for its local training. Let $\boldsymbol{f} = (f_n, n \in \mathcal{N})$ denote the processing capacity vector. Let D_n denote the number of CPU cycles required by organization n to process one data unit. Let T_n^{UL} and T_n^{DL} (in seconds) denote the time that organization n is required for uploading and downloading the model updates in each training round, respectively. The duration of each training round is as follows:

$$\tau(\boldsymbol{f}) = \max_{n\in\mathcal{N}} \left\{ \frac{S_n D_n K}{f_n} + T_n^{\mathrm{UL}} + T_n^{\mathrm{DL}} \right\}. \quad (2)$$

The $\max$ operator is due to the fact that the central server updates the global model after the weights of the local models from all organizations have been received in a training round.

Similar to other existing works [21]–[23], we consider a scenario where the datasets are unchanged across time. We consider that there is a fixed total training time $T \in \mathbb{R}_+$. This corresponds to the scenario where the organizations have a deadline for the FL process. For example, the FL process over the clinical data of infectious disease (e.g., COVID-19) collected by multiple pharmaceutical companies on or before a day may need to be completed within a deadline, as the results may need to be used for the subsequent research on the vaccine development. The number of training rounds is equal to the total training time divided by the duration of each round:

$$r(\boldsymbol{f}) = T/\tau(\boldsymbol{f}). \quad (3)$$

Similar to [21], [22], we do not round up $r(\boldsymbol{f})$ for simplicity. This is reasonable as $r(\boldsymbol{f})$ is usually large in practice, so the difference between $r(\boldsymbol{f})$ and its rounded value is negligible.

Let m_n (in dollars) denote the monetary transfer to organization $n \in \mathcal{N}$. If $m_n > 0$, then organization n receives m_n from other organizations. If $m_n < 0$, then organization n

[1]This work can be extended to other synchronous FL algorithms (e.g., [7], [8]) by modifying the precision formulation in (4).

pays $|m_n|$ to some other organizations. The monetary transfer is between the organizations, i.e., some organizations pay other organizations.[2] Let vector $\boldsymbol{m} = (m_n, n \in \mathcal{N})$.

B. Payoff of the Organizations

We now define the utility and cost functions of the organizations as well as the payoff function.

1) Utility: As in the existing work [21], we define the utility of each organization as a function of the precision of the trained global model.[3] The precision of the trained global model is defined as the difference between the expected loss of the trained global model (after $r(\boldsymbol{f})$ training rounds) and the minimum expected loss [7], [9], [21], i.e., $L(\boldsymbol{\omega}^{r(\boldsymbol{f})}) - L(\boldsymbol{\omega}^*)$. Note that a smaller precision implies a smaller loss of the trained global model and hence a better fit of the model to the datasets. Let $\epsilon(r(\boldsymbol{f}))$ denote the precision of the trained global model with $r(\boldsymbol{f})$ training rounds.[4] Under a strongly convex loss function $L(\boldsymbol{\omega})$, $\epsilon(r(\boldsymbol{f}))$ can be modeled as follows [9]:

$$\epsilon(r(\boldsymbol{f})) = \epsilon_0/(\epsilon_1 + Kr(\boldsymbol{f})), \tag{4}$$

where ϵ_0 and ϵ_1 are positive coefficients. These coefficients can be derived based on the loss function, the neural network structure, and the distribution of the datasets [9]. In (4), as $r(\boldsymbol{f})$ increases, the precision is non-increasing, and the marginal decrease of the precision reduces. Our proposed mechanism and algorithm are applicable to any function of $\epsilon(r(\boldsymbol{f}))$ that is non-increasing and convex in $r(\boldsymbol{f})$ as well as having a bounded $\epsilon(0)$ (i.e., the initial global model has a bounded precision).

The utility of organization $n \in \mathcal{N}$ is its valuation on the difference between the precision of the global model without training (i.e., $\epsilon(0)$) and that of the trained global model:

$$U_n(r(\boldsymbol{f})) = u_n\left(\epsilon(0) - \epsilon(r(\boldsymbol{f}))\right),\ n \in \mathcal{N}, \tag{5}$$

where u_n (in dollars per unit of loss) is the unit revenue that organization n can earn from its market by using the trained global model. For example, u_n can be the unit revenue that a pharmaceutical company can earn from its customers by selling vaccine, where the clinical data analysis for developing vaccine was performed by cross-silo FL. This revenue is different from the monetary transfer between organizations (i.e., $\boldsymbol{m}$) for motivating cooperation. Note that u_n of organization n may be unknown to other organizations and the central server.

2) Cost: The cost of an organization is defined as follows:

$$\begin{aligned} C_n(f_n, r(\boldsymbol{f})) = {} & (C_n^{\text{UL}} + C_n^{\text{DL}})\, r(\boldsymbol{f}) + C_n^{\text{invt}} f_n \\ & + C_n^{\text{comp}}(f_n)^2 S_n D_n K r(\boldsymbol{f}),\ n \in \mathcal{N}. \end{aligned} \tag{6}$$

The parameters C_n^{UL} and C_n^{DL} are the operating costs for uploading and downloading the model updates in each training round, respectively. The product $C_n^{\text{invt}} f_n$ corresponds to the investment cost (e.g., leasing servers) per processing capacity, where the linearity is due to the linear server leasing rate [28]. The term $C_n^{\text{comp}}(f_n)^2$ is the operating cost of organization n for performing one CPU cycle, where the quadratic form is supported by the widely adopted quadratic energy consumption model for processors (e.g., [13], [22]). Note that the cost function $C_n(f_n, r(\boldsymbol{f}))$ of any organization n may be unknown to other organizations and the central server.

3) Payoff: The payoff of organization $n \in \mathcal{N}$ is defined as the difference between its utility and cost and the additional monetary transfer. For organization $n \in \mathcal{N}$, the payoff is

$$V_n(f_n, r(\boldsymbol{f}), m_n) = U_n(r(\boldsymbol{f})) - C_n(f_n, r(\boldsymbol{f})) + m_n. \tag{7}$$

C. Problem Formulation

We now present the social welfare maximization problem and formulate the incentive mechanism design problem.

1) Social Welfare Maximization: From the social perspective, the organizations should choose the processing capacity vector $\boldsymbol{f}$ that maximizes the social welfare of the system:

$$\underset{\boldsymbol{f}}{\text{maximize}} \quad \textstyle\sum_{n \in \mathcal{N}} \left(U_n(r(\boldsymbol{f})) - C_n(f_n, r(\boldsymbol{f}))\right) \tag{8a}$$

$$\text{subject to} \quad f_n \geq 0,\ n \in \mathcal{N}. \tag{8b}$$

Problem (8) is nonconvex, since $C_n(f_n, r(\boldsymbol{f}))$ is noncovex in $\boldsymbol{f}$. Each organization, however, is interested in maximizing its payoff instead of the social welfare.

2) Incentive Mechanism Design Problem: To motivate efficient cooperation among the organizations, we aim to design an incentive mechanism. This mechanism should be able to address the public goods feature of the resources of the organizations. We consider an incentive mechanism as follows:

- Each organization $n \in \mathcal{N}$ submits a message profile (γ_n, π_n) to the central server. Message γ_n indicates the number of training rounds that organization n expects to have. Message π_n indicates the unit monetary transfer per training round that organization n expects to pay or receive. Let $\boldsymbol{\gamma} = (\gamma_n, n \in \mathcal{N})$ and $\boldsymbol{\pi} = (\pi_n, n \in \mathcal{N})$.
- The central server computes and announces the processing capacity vector $\boldsymbol{f}(\boldsymbol{\gamma}) = (f_n(\boldsymbol{\gamma}), n \in \mathcal{N})$ and the monetary transfer vector $\boldsymbol{m}(\boldsymbol{\gamma}, \boldsymbol{\pi}) = (m_n(\boldsymbol{\gamma}, \boldsymbol{\pi}), n \in \mathcal{N})$.

We aim to design $\boldsymbol{f}(\boldsymbol{\gamma})$ and $\boldsymbol{m}(\boldsymbol{\gamma}, \boldsymbol{\pi})$ in the incentive mechanism. Note that the actual monetary transfer to organization n (i.e., $m_n(\boldsymbol{\gamma}, \boldsymbol{\pi})$) depends on the choice of $\boldsymbol{m}(\boldsymbol{\gamma}, \boldsymbol{\pi})$. Thus, it is not necessarily equal to the monetary transfer specified by organization n in message profile (i.e., the number of training rounds γ_n multiplied by the unit monetary transfer π_n).

Given the incentive mechanism, the organizations have to decide the message profiles to submit. The strategic interaction among the organizations can be modeled as a non-cooperative game. Let $(\boldsymbol{\gamma}^{\text{NE}}, \boldsymbol{\pi}^{\text{NE}})$ denote the NE of the game. That is, given the message profiles of other organizations, organization $n \in \mathcal{N}$ cannot increase its payoff by deviating

[2]The central server in cross-silo FL can help the organizations to collect the payment from the organizations with negative m_n and distribute the collected payment to those with positive m_n. We will later require $\sum_{n \in \mathcal{N}} m_n = 0$.

[3]We follow [9] to use the term 'precision'. The term 'accuracy' is used in [7], [21] with the same meaning and definition.

[4]Due to the complicated neural network structure, it is challenging to obtain the precision of the trained global model. As in [7], [21], $\epsilon(r(\boldsymbol{f}))$ corresponds to the upper-bound of the precision of the trained global model. That is, $L(\boldsymbol{\omega}^{r(\boldsymbol{f})}) - L(\boldsymbol{\omega}^*) \leq \epsilon(r(\boldsymbol{f}))$. Recall that $L(\boldsymbol{\omega}^*)$ is the minimum expected loss, thus the inequality $L(\boldsymbol{\omega}^{r(\boldsymbol{f})}) \geq L(\boldsymbol{\omega}^*)$ always holds. We follow [7], [21] to refer to $\epsilon(r(\boldsymbol{f}))$ as the precision of the trained global model.

from $(\gamma_n^{\rm NE}, \pi_n^{\rm NE})$. We aim to design $\boldsymbol{f}(\boldsymbol{\gamma})$ and $\boldsymbol{m}(\boldsymbol{\gamma}, \boldsymbol{\pi})$ in the incentive mechanism such that the NE of the game satisfies the following properties P1, P2, and P3:

P1 Social efficiency: The processing capacity vector under NE, i.e., $\boldsymbol{f}(\boldsymbol{\gamma}^{\rm NE})$, is the optimal solution of problem (8).

P2 Individual rationality: Each organization is no worse off through participating in cross-silo FL, i.e., $V_n(f_n(\boldsymbol{\gamma}^{\rm NE}), r(\boldsymbol{f}(\boldsymbol{\gamma}^{\rm NE})), m_n(\boldsymbol{\gamma}^{\rm NE}, \boldsymbol{\pi}^{\rm NE})) \geq 0$ for $n \in \mathcal{N}$.

P3 Budget balance: The summation of the monetary transfer of all organizations is zero, i.e., $\sum_{n\in\mathcal{N}} m_n(\boldsymbol{\gamma}^{\rm NE}, \boldsymbol{\pi}^{\rm NE}) = 0$. In other words, the monetary transfer can operate among the organizations without any third party investment.

III. Incentive Mechanism Design

In this section, we design an incentive mechanism for cross-silo FL to achieve properties P1−P3. To address the public goods feature of the resources, our proposed incentive mechanism is inspired by the existing mechanisms for public goods in other application scenarios, e.g., [24], [25]. Different from those existing mechanisms, our proposed mechanism can further address the following new challenges in cross-silo FL. First, the social welfare maximization problem (8) is nonconvex. Second, each organization is both a producer who contributes resources and a consumer who utilizes the trained global model.

In the following, we first propose the incentive mechanism. We then analyze the strategies of the organizations and the properties of the incentive mechanism.

A. Incentive Mechanism

In the incentive mechanism, we consider the central server in cross-silo FL helps the organizations to collect the message profiles as well as compute the processing capacity and monetary transfer of the organizations. Note that it may be infeasible to let the central server directly solve problem (8) and the incentive mechanism design problem due to the private information of the organizations. Let $\bar{r}$ denote the maximum possible number of training rounds, i.e., $\bar{r} \triangleq \max_{f_n\geq 0, n\in\mathcal{N}} r(\boldsymbol{f})$. By substituting $r(\boldsymbol{f}) = T/\tau(\boldsymbol{f})$ and $\tau(\boldsymbol{f})$ defined in (2), we can obtain $\bar{r} = \min_{n\in\mathcal{N}}\{T/(T_n^{\rm UL} + T_n^{\rm DL})\}$. The incentive mechanism for cross-silo FL is as follows.

Mechanism 1 (Incentive Mechanism for Cross-Silo FL)**.** *First, each organization $n \in \mathcal{N}$ submits a message profile (γ_n, π_n) to the central server, where*

- $\gamma_n \in [0, \bar{r}]$ *indicates the number of training rounds that organization n expects to have.*
- $\pi_n \in \mathbb{R}$ *indicates the unit monetary transfer per training round that organization n expects to receive or send.*

Second, the central server computes and announces the number of training rounds that each organization needs to perform, denoted by $\widetilde{r}(\boldsymbol{\gamma})$, processing capacity $\boldsymbol{f}(\boldsymbol{\gamma}) = (f_n(\boldsymbol{\gamma}), n \in \mathcal{N})$, and monetary transfer $\boldsymbol{m}(\boldsymbol{\gamma}, \boldsymbol{\pi}) = (m_n(\boldsymbol{\gamma}, \boldsymbol{\pi}), n \in \mathcal{N})$:

- $\widetilde{r}(\boldsymbol{\gamma})$ *is computed as follows:*
$$\widetilde{r}(\boldsymbol{\gamma}) = \textstyle\sum_{n\in\mathcal{N}} \gamma_n / N. \tag{9}$$
- $f_n(\boldsymbol{\gamma})$ *is the processing capacity that organization n should use in its local training:*
$$f_n(\boldsymbol{\gamma}) = f_n^\star(\widetilde{r}(\boldsymbol{\gamma})) \triangleq \frac{S_n D_n K}{\frac{T}{\widetilde{r}(\boldsymbol{\gamma})} - T_n^{\rm UL} - T_n^{\rm DL}},\ n \in \mathcal{N}, \tag{10}$$
where $f_n^\star(\widetilde{r}(\boldsymbol{\gamma}))$ is the processing capacity that organization n should use to achieve $\widetilde{r}(\boldsymbol{\gamma})$ training rounds.
- $m_n(\boldsymbol{\gamma}, \boldsymbol{\pi})$ *is the monetary transfer to organization n:*
$$m_n(\boldsymbol{\gamma}, \boldsymbol{\pi}) = \widetilde{r}(\boldsymbol{\gamma}) \left(\pi_{\mu(n+1)} - \pi_{\mu(n+2)}\right), \tag{11}$$
where $\mu(n+1)$ is equal to $n+1$ modulo N.[5] *In (11), the monetary transfer to organization n is equal to the number of training rounds multiplied by the difference between the unit monetary transfer submitted by the organizations with indices $\mu(n+1)$ and $\mu(n+2)$.*

Intuitively, in Mechanism 1, to address the public goods feature (i.e., non-excludable, non-rivalrous), each organization $n \in \mathcal{N}$ is treated equally regardless of its π_n. In Mechanism 1, the number of training rounds that each organization has to perform (i.e., $\widetilde{r}(\boldsymbol{\gamma})$) is equal to the average value of the number of training rounds that the organizations expect to have. The processing capacity of each organization has to lead to $\widetilde{r}(\boldsymbol{\gamma})$ training rounds. The definition of monetary transfer in (11) has three features that make the proposed incentive mechanism achieve properties P1−P3 mathematically. First, the payoff of organization n does not rely on the choice of π_n. Second, the payoff of organization n is linear in the number of training rounds. Third, it always holds that the summation of the monetary transfer to all organizations, i.e., the summation of $m_n(\boldsymbol{\gamma}, \boldsymbol{\pi})$ over all $n \in \mathcal{N}$, is equal to zero. Moreover, we will discuss in Section IV that when the organizations do not know the private information of each other, equation (11) ensures any organizations $\mu(n+1)$ and $\mu(n+2)$ for $n \in \mathcal{N}$ to gradually change the value of $\pi_{\mu(n+1)} - \pi_{\mu(n+2)}$ in order to motivate organization n to choose the value of γ_n that leads to social efficiency.

B. Analysis of the Strategies of the Organizations

We first define the game of the organizations. Then, we derive the NE and the properties of Mechanism 1.

1) Game of the Organizations: Given Mechanism 1, each organization can optimize its message profile to maximize its payoff. Such strategic interaction can be modeled as a non-cooperative game with perfect information (i.e., the organizations know the private information of each other). We will relax the perfect information requirement in Section IV. The game of the organizations is defined as follows.

Game 1 (Message Profile Submission)**.**

- *Player: all organizations $n \in \mathcal{N}$.*
- *Strategy: message profile (γ_n, π_n) with $\gamma_n \in [0, \bar{r}]$ and $\pi_n \in \mathbb{R}$ for each organization $n \in \mathcal{N}$.*

[5]For example, consider $N = 10$, i.e., the set of organizations is $\mathcal{N} = \{0, 1, \ldots, 9\}$. If $n \in \{0, 1, \ldots 7\}$, then $\mu(n+1) = n+1$ and $\mu(n+2) = n+2$. If $n = 8$, then $\mu(n+1) = 9$ and $\mu(n+2) = 0$. If $n = 9$, then $\mu(n+1) = 0$ and $\mu(n+2) = 1$.

- *Payoff function:* $V_n(f_n(\boldsymbol{\gamma}), \widetilde{r}(\boldsymbol{\gamma}), m_n(\boldsymbol{\gamma}, \boldsymbol{\pi}))$ *for* $n \in \mathcal{N}$. *Note that* $V_n(f_n(\boldsymbol{\gamma}), r(\boldsymbol{f}(\boldsymbol{\gamma})), m_n(\boldsymbol{\gamma}, \boldsymbol{\pi}))$ *is equal to* $V_n(f_n(\boldsymbol{\gamma}), \widetilde{r}(\boldsymbol{\gamma}), m_n(\boldsymbol{\gamma}, \boldsymbol{\pi}))$ *according to* (10).

Let $(\boldsymbol{\gamma}_{-n}, \boldsymbol{\pi}_{-n})$ denote the message profiles submitted by all organizations excluding organization $n \in \mathcal{N}$, i.e., $\boldsymbol{\gamma}_{-n} = (\gamma_{n'}, n' \in \mathcal{N} \setminus \{n\})$ and $\boldsymbol{\pi}_{-n} = (\pi_{n'}, n' \in \mathcal{N} \setminus \{n\})$. For simplicity, we will use $\boldsymbol{f}(\gamma_n, \boldsymbol{\gamma}_{-n})$ and $\boldsymbol{f}(\boldsymbol{\gamma})$, $\widetilde{r}(\gamma_n, \boldsymbol{\gamma}_{-n})$ and $\widetilde{r}(\boldsymbol{\gamma})$, as well as $m_n(\gamma_n, \pi_n, \boldsymbol{\gamma}_{-n}, \boldsymbol{\pi}_{-n})$ and $m_n(\boldsymbol{\gamma}, \boldsymbol{\pi})$ interchangeably. The NE of Game 1 is defined as follows.

Definition 1 (Nash equilibrium)**.** *An NE of Game 1 is a message profile* $(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$ *that satisfies*

$$\begin{aligned} &V_n(f_n(\boldsymbol{\gamma}^{\mathrm{NE}}), \widetilde{r}(\boldsymbol{\gamma}^{\mathrm{NE}}), m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})) \\ &\geq V_n(f_n(\gamma_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}), \widetilde{r}(\gamma_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}), m_n(\gamma_n, \pi_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}, \boldsymbol{\pi}_{-n}^{\mathrm{NE}})), \\ &\qquad \gamma_n \in [0, \bar{r}], \pi_n \in \mathbb{R}, n \in \mathcal{N}. \end{aligned} \tag{12}$$

According to (11), we have $m_n(\gamma_n, \pi_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}, \boldsymbol{\pi}_{-n}^{\mathrm{NE}}) = m_n(\gamma_n, \pi_n^{\mathrm{NE}}, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}, \boldsymbol{\pi}_{-n}^{\mathrm{NE}})$ for $\pi_n \in \mathbb{R}$ under any $N \geq 3$. Hence, the following inequality is equivalent to inequality (12):

$$\begin{aligned} &V_n(f_n(\boldsymbol{\gamma}^{\mathrm{NE}}), \widetilde{r}(\boldsymbol{\gamma}^{\mathrm{NE}}), m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})) \\ &\geq V_n(f_n(\gamma_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}), \widetilde{r}(\gamma_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}), m_n(\gamma_n, \pi_n^{\mathrm{NE}}, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}, \boldsymbol{\pi}_{-n}^{\mathrm{NE}})), \\ &\qquad \gamma_n \in [0, \bar{r}], n \in \mathcal{N}. \end{aligned} \tag{13}$$

From (12) to (13), we replace $m_n(\gamma_n, \pi_n, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}, \boldsymbol{\pi}_{-n}^{\mathrm{NE}})$ with $m_n(\gamma_n, \pi_n^{\mathrm{NE}}, \boldsymbol{\gamma}_{-n}^{\mathrm{NE}}, \boldsymbol{\pi}_{-n}^{\mathrm{NE}})$. In the rest of this paper, we focus on the scenario with $N \geq 3$. If $N = 1$, then cross-silo FL cannot be operated. If $N = 2$, then we can introduce an additional virtual organization with zero utility and cost.

2) Nash Equilibrium and Properties: According to Definition 1 and inequality (13), any NE should satisfy the following.

Lemma 1 (Nash Equilibrium)**.** *A message profile* $(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$ *is an NE of Game 1 if and only if*

$$\begin{aligned} \gamma_n^{\mathrm{NE}} = N \arg \max_{r \in [0, \bar{r}]} V_n(f_n^\star(r), r, m_n(r\mathbf{1}, \boldsymbol{\pi}^{\mathrm{NE}})) \\ - \textstyle\sum_{n' \in \mathcal{N} \setminus \{n\}} \gamma_{n'}^{\mathrm{NE}}, \ n \in \mathcal{N}, \end{aligned} \tag{14}$$

where $\mathbf{1}$ *is an all-one vector with length* N. *The function* $m_n(r\mathbf{1}, \boldsymbol{\pi}^{\mathrm{NE}}) = r(\pi_{\mu(n+1)}^{\mathrm{NE}} - \pi_{\mu(n+2)}^{\mathrm{NE}})$ *defines the monetary transfer to organization* n *under* r *training rounds given* $\boldsymbol{\pi}^{\mathrm{NE}}$.

The proof is given in Appendix A. Intuitively, Lemma 1 implies that $(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$ is an NE if and only if under $\boldsymbol{\pi}^{\mathrm{NE}}$, $\boldsymbol{\gamma}^{\mathrm{NE}}$ leads to the number of training rounds $\widetilde{r}(\boldsymbol{\gamma}^{\mathrm{NE}})$ that maximizes the payoff of each organization, i.e., $\sum_{n' \in \mathcal{N}} \gamma_{n'}^{\mathrm{NE}}/N = \widetilde{r}(\boldsymbol{\gamma}^{\mathrm{NE}}) = \arg\max_{r \in [0, \bar{r}]} V_n(f_n^\star(r), r, m_n(r\mathbf{1}, \boldsymbol{\pi}^{\mathrm{NE}}))$, $n \in \mathcal{N}$.

According to Lemma 1 as well as equations (9), (10), and (11), Mechanism 1 satisfies properties P1, P2, and P3.

Theorem 1 (Efficiency)**.** *Under any NE of Game 1, i.e.,* $(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$, $\boldsymbol{f}(\boldsymbol{\gamma}^{\mathrm{NE}})$ *optimizes problem* (8).

The proof of Theorem 1 is given in Appendix B.

Proposition 1 (Individual Rationality)**.** *Under any NE of Game 1, i.e.* $(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$, *each organization has nonnegative payoff, i.e.,* $V_n(f_n(\boldsymbol{\gamma}^{\mathrm{NE}}), \widetilde{r}(\boldsymbol{\gamma}^{\mathrm{NE}}), m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})) \geq 0$, $n \in \mathcal{N}$.

This holds because $V_n(f_n(\boldsymbol{\gamma}^{\mathrm{NE}}), \widetilde{r}(\boldsymbol{\gamma}^{\mathrm{NE}}), m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})) \geq V_n(f_n^\star(0), 0, m_n(\mathbf{0}, \boldsymbol{\pi}^{\mathrm{NE}})) = 0$ for $n \in \mathcal{N}$ due to Lemma 1, where $\mathbf{0}$ is a zero vector with length N.

Proposition 2 (Budget Balance)**.** *Under any NE of Game 1, i.e.,* $(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$, *the summation of the monetary transfer of all organizations is equal to zero, i.e.,* $\sum_{n \in \mathcal{N}} m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}}) = 0$.

Proposition 2 is proven by substituting $m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$ defined in (11) into $\sum_{n \in \mathcal{N}} m_n(\boldsymbol{\gamma}^{\mathrm{NE}}, \boldsymbol{\pi}^{\mathrm{NE}})$.

IV. Distributed Algorithm Design

In this section, we propose a distributed algorithm that enables the organizations to achieve the NE of Game 1. The proposed algorithm can address the challenges regarding the nonconvexity of problem (8) and the unknown private information of the organizations. The main idea is to first reformulate the social welfare maximization problem. Since the saddle point of the Lagrangian of the reformulated problem is the NE of Game 1, we can then design a distributed algorithm that converges to the saddle point of the Lagrangian and hence the NE of Game 1. In the following, we first reformulate the social welfare maximization problem. Then, we propose the distributed algorithm.

A. Problem Reformulation

We introduce an auxiliary variable $\boldsymbol{r} = (r_n, n \in \mathcal{N})$, where r_n is the number of training rounds that organization n performs. Since r_n should be identical for all $n \in \mathcal{N}$, we have the following social welfare maximization problem:

$$\underset{\boldsymbol{r}}{\text{maximize}} \quad \textstyle\sum_{n \in \mathcal{N}} \left(U_n\left(r_n\right) - C_n\left(f_n^\star(r_n), r_n\right)\right) \tag{15a}$$

$$\text{subject to} \quad r_{\mu(n-2)} = r_{\mu(n-1)}, \ n \in \mathcal{N}, \tag{15b}$$

$$r_n \in [0, \bar{r}], \ n \in \mathcal{N}. \tag{15c}$$

Constraint (15b) can be expanded as $r_{N-2} = r_{N-1}$ for $n = 0$, $r_{N-1} = r_0$ for $n = 1$, and $r_{n-2} = r_{n-1}$ for $2 \leq n \leq N-1$. It implies that r_n has to be identical for all $n \in \mathcal{N}$. We write it in the form of $r_{\mu(n-2)} = r_{\mu(n-1)}$, $n \in \mathcal{N}$ for the simplicity of algorithm design. Let $\boldsymbol{r}^*$ be the optimal solution to problem (15), and let $r^* = r_0^* = \cdots = r_{N-1}^*$. Then, the processing capacity $\boldsymbol{f}^\star(r^*) = (f_n^\star(r^*), n \in \mathcal{N})$ optimizes problem (8).

We define the Lagrangian of problem (15), i.e., $\mathcal{L} : [0, \bar{r}]^N \times \mathbb{R}^N \to \mathbb{R}$, as follows:

$$\begin{aligned} \mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda}) = &\textstyle\sum_{n \in \mathcal{N}} \left(U_n\left(r_n\right) - C_n\left(f_n^\star(r_n), r_n\right)\right) \\ &- \textstyle\sum_{n \in \mathcal{N}} \lambda_n \left(r_{\mu(n-2)} - r_{\mu(n-1)}\right), \end{aligned} \tag{16}$$

where $\boldsymbol{\lambda} = (\lambda_n, n \in \mathcal{N})$ is the vector of the Lagrange multipliers. Note that $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda})$ can be decoupled into multiple functions. That is, $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda}) = \sum_{n \in \mathcal{N}} \mathcal{L}_n(r_n, \boldsymbol{\lambda})$, and

$$\begin{aligned} \mathcal{L}_n(r_n, \boldsymbol{\lambda}) = &U_n\left(r_n\right) - C_n\left(f_n^\star(r_n), r_n\right) \\ &- \left(\lambda_{\mu(n+2)} - \lambda_{\mu(n+1)}\right) r_n, \ n \in \mathcal{N}. \end{aligned} \tag{17}$$

For $n \in \mathcal{N}$, $\lambda_{\mu(n+2)}$ and $\lambda_{\mu(n+1)}$ are the Lagrange multipliers correspond to constraints $r_n = r_{\mu(n+1)}$ and $r_{\mu(n-1)} = r_n$, respectively. Based on (7), (11), and (17), we have $\mathcal{L}_n(r_n, \boldsymbol{\lambda}) =$

$V_n\left(f_n^\star\left(r_n\right), r_n, m_n(r_n\mathbf{1}, \boldsymbol{\lambda})\right)$ for all $n \in \mathcal{N}$. This is the payoff of organization n under r_n training rounds, given $\boldsymbol{\lambda}$.

For problem (15), strong duality holds according to Slater's condition. Hence, the saddle point of Lagrangian $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda})$, denoted by $(\boldsymbol{r}^*, \boldsymbol{\lambda}^*)$, exists. That is, $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda}^*) \leq \mathcal{L}(\boldsymbol{r}^*, \boldsymbol{\lambda}^*) \leq \mathcal{L}(\boldsymbol{r}^*, \boldsymbol{\lambda})$ for any $\boldsymbol{r} \in [0, \bar{r}]^N$, $\boldsymbol{\lambda} \in \mathbb{R}^N$. Thus, we can prove that if $(\boldsymbol{r}^*, \boldsymbol{\lambda}^*)$ is a saddle point of $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda})$, then it is an NE of Game 1, with the proof given in Appendix C.

Lemma 2 (Saddle Point and NE)**.** *For any saddle point of $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda})$, denoted by $(\boldsymbol{r}^*, \boldsymbol{\lambda}^*)$, the message profile $(\boldsymbol{\gamma}^{\rm NE} = \boldsymbol{r}^*, \boldsymbol{\pi}^{\rm NE} = \boldsymbol{\lambda}^*)$ is an NE of Game 1.*

B. Distributed Algorithm

Our proposed algorithm is inspired by the distributed accelerated augmented Lagrangian method [29], which is a distributed algorithm for achieving the saddle point of the Lagrangian of a constrained problem. We have modified the algorithm in [29] to adapt to the cross-silo FL scenario. Specifically, we replace the notations $\boldsymbol{r}$ and $\boldsymbol{\lambda}$ in $\mathcal{L}(\boldsymbol{r}, \boldsymbol{\lambda})$ with notations $\boldsymbol{\gamma}$ and $\boldsymbol{\pi}$, respectively. In the proposed algorithm, the organizations aim to find the saddle point of $\mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\pi}) = \sum_{n \in \mathcal{N}} \mathcal{L}_n(\gamma_n, \boldsymbol{\pi}) = \sum_{n \in \mathcal{N}} V_n\left(f_n^\star\left(\gamma_n\right), \gamma_n, m_n(\gamma_n \mathbf{1}, \boldsymbol{\pi})\right)$. The obtained saddle point is the NE of Game 1.

The proposed algorithm is given in Algorithm 1, where the organizations update message profiles for multiple iterations until the algorithm converges. Let $t \in \mathbb{Z}_+$ denote the iteration index. Each organization n first randomly initializes its message profile $(\gamma_n(0), \pi_n(0))$. While the convergence indicator Convg_Indicator $= 0$, each organization n submits message profile $(\gamma_n(t), \pi_n(t))$ in iteration t. Then, the central server sends the submitted message profiles to the organizations. Steps 7-9 correspond to how each organization n updates its message profile. In Step 7, organization n first computes $\hat{\gamma}_n(t)$ by deriving the value of $\gamma_n \in [0, \bar{r}]$ that maximizes

$$V_n^\rho(\gamma_n, \boldsymbol{\gamma}_{-n}, \boldsymbol{\pi}) = V_n(f_n^\star(\gamma_n), \gamma_n, m_n(\gamma_n \mathbf{1}, \boldsymbol{\pi})) - \rho \sum_{n \in \mathcal{N}} \left(\gamma_{\mu(n-2)} - \gamma_{\mu(n-1)}\right)^2, \quad (18)$$

where ρ is a penalty coefficient. The second term can be regarded as a term for penalizing the different number of training rounds submitted by the organizations (i.e., $\gamma_{\mu(n-2)} \neq \gamma_{\mu(n-1)}$ for $n \in \mathcal{N}$). In Steps 8 and 9, organization n computes the updated message profile $(\gamma_n(t+1), \pi_n(t+1))$ by considering a step size $\eta \in (0,1)$. A larger η implies a more aggressive update. An intuition behind Step 9 is as follows. Suppose the number of training rounds submitted by organization $\mu(n-2)$ is much larger than that submitted by organization $\mu(n-1)$ (i.e., the difference $\gamma_{\mu(n-2)}(t) - \gamma_{\mu(n-1)}(t)$ is large). Then, $\pi_n(t+1)$ is large based on Step 9. According to (11), the large value of $\pi_n(t+1)$ will lead to a small monetary transfer to organization $\mu(n-2)$ and a large monetary transfer to organization $\mu(n-1)$. Hence, in the next iteration, organizations $\mu(n-2)$ and $\mu(n-1)$ will reduce and increase the number of training rounds that they submit, respectively. The algorithm terminates when the absolute difference between $\gamma_n(t+1)$ and $\gamma_n(t)$ for all $n \in \mathcal{N}$ is smaller than a predefined threshold ϕ.

Algorithm 1: Distributed Algorithm for Cross-Silo FL

1 Organization $n \in \mathcal{N}$ randomly initializes $\gamma_n(0)$, $\pi_n(0)$;
2 $t \leftarrow 0$, Convg_Indicator $\leftarrow 0$;
3 **while** *Convg_Indicator* $= 0$ **do**
4   Organization $n \in \mathcal{N}$ submits $(\gamma_n(t), \pi_n(t))$;
5   Central server sends organizations $\boldsymbol{\gamma}(t)$ and $\boldsymbol{\pi}(t)$;
6   **for** *organization $n \in \mathcal{N}$ in parallel* **do**
7         $\hat{\gamma}_n(t) \leftarrow \arg\max_{\gamma_n \in [0,\bar{r}]} V_n^\rho(\gamma_n, \boldsymbol{\gamma}_{-n}(t), \boldsymbol{\pi}(t))$;
8         $\gamma_n(t+1) \leftarrow \gamma_n(t) + \eta\left(\hat{\gamma}_n(t) - \gamma_n(t)\right)$;
9         $\pi_n(t+1) \leftarrow \pi_n(t) + \rho\eta(\gamma_{\mu(n-2)}(t) - \gamma_{\mu(n-1)}(t))$;
10   **end**
11     $t \leftarrow t+1$;
12   **if** $|\gamma_n(t+1) - \gamma_n(t)| \leq \phi,\ n \in \mathcal{N}$ **then**
13   Convg_Indicator $\leftarrow 1$;
14   **end**
15 **end**

TABLE I
PARAMETER SETTINGS

Param.	Value	Param.	Value
N	10	Model size	0.16 Mbits
K	5	DL speed	78.26 Mbps [30]
T	60 seconds	UL speed	42.06 Mbps [30]
D_n	0.01 gigacycles	Invest. cost	$0.22 per GHz per hour [31]
S_n	600 samples	DL energy	3 joules per Mbit [32]
ϵ_0	9.82	UL energy	3 joules per Mbit [32]
ϵ_1	4.26	Elec. rate	$0.174 per kWh [33]

Algorithm 1 converges to the NE of Game 1, which can be proven based on [29, Theorem 2] and Lemma 2.

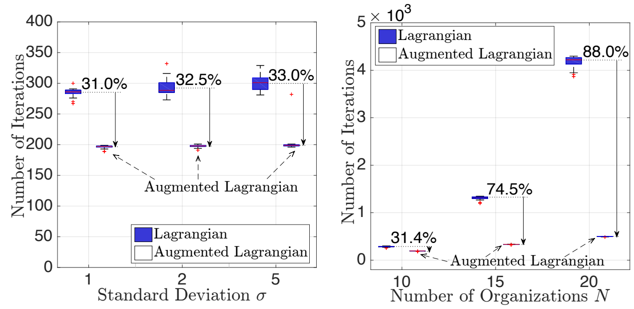
Proposition 3 (Convergence)**.** *For any $\rho \in \mathbb{R}_+$ that ensures $V_n^\rho(\gamma_n, \boldsymbol{\gamma}_{-n}, \boldsymbol{\pi})$ to be strictly concave for $\boldsymbol{\gamma} \in [0, \bar{r}]^N$, $\boldsymbol{\pi} \in \mathbb{R}^N$, $n \in \mathcal{N}$, Algorithm 1 converges to the NE of Game 1.*

V. PERFORMANCE EVALUATION

We conduct simulations based on dataset MNIST [26] using FedAvg [6]. This dataset contains the figures of handwritten digits. It has been used by many existing works on FL, e.g., [7], [9]. Table I shows the parameter settings. In Table I, 'param.', 'invest.', and 'elec.' are the short-forms for 'parameter', 'investment', and 'electricity', respectively. We set $N = 10$, as the number of organizations in cross-silo FL is always small [1]. We consider a small $T = 60$ seconds, as the MNIST dataset is easy to train in the sense that hundreds of training rounds can achieve a high digit recognition correctness rate. Parameters S_n, D_n, model size (i.e., the size of $\boldsymbol{\omega}$), ϵ_0, and ϵ_1 are obtained from the MNIST dataset [26]. We set $\eta = 0.3$, $\phi = 0.0001$, and $\rho = 0.00005$.

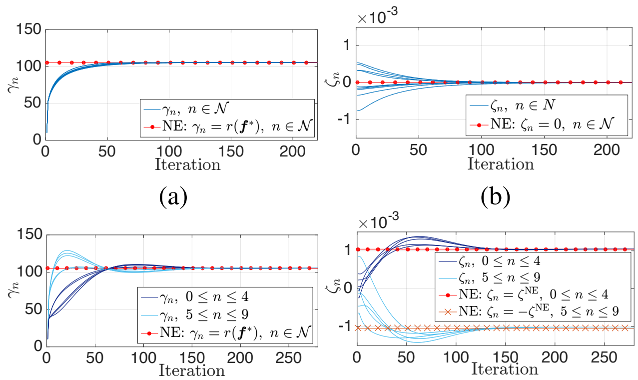
A. Convergence Rate

Fig. 2 shows the convergence of our proposed Algorithm 1, denoted by 'augmented Lagrangian', and another Lagrangian



(a) (b)

Fig. 2. Number of iterations under different (a) standard deviation σ (with $N = 10$) and (b) number of organizations N (with $\sigma = 1$).



(c) (d)

Fig. 3. Convergence of the message profiles: (a) γ_n in homogeneous scenario; (b) ζ_n in homogeneous scenario; (c) γ_n in heterogeneous scenario; (d) ζ_n in heterogeneous scenario. Note that $\zeta_n \triangleq \pi_{\mu(n+1)} - \pi_{\mu(n+2)}$.

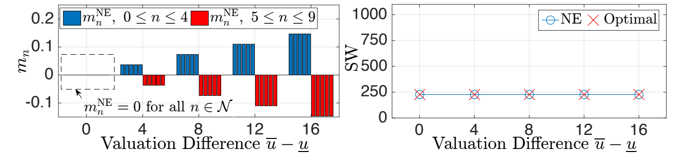
based algorithm [27, Section 8.1],[6] denoted by 'Lagrangian'. Note that both algorithms can converge to the NE of Game 1. In the simulation, u_n for $n \in \mathcal{N}$ is randomly generated using the truncated normal distribution with a mean of 10 and a standard deviation of σ.[7] The results are shown using boxplot [34]. The percentage in the figure gives the average percentage reduction that the proposed algorithm induces when compared with 'Lagrangian'. In Fig. 2, the proposed algorithm significantly reduces the number of iterations, especially when the number of organizations is large.

B. Algorithm Convergence under Two Instances

Fig. 3 shows the convergence of the message profiles submitted by the organizations across iterations. In Fig. 3 (a) and (b), we consider a homogeneous scenario where $u_n = 10$ for all $n \in \mathcal{N}$. As the number of iterations increases, γ_n gradually converges to the optimal number of training rounds $r(\boldsymbol{f}^*)$ for all $n \in \mathcal{N}$, where $\boldsymbol{f}^*$ is the optimal solution to problem (8).

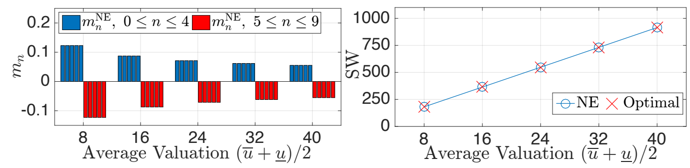
[6]The Lagrangian based algorithm [27, Section 8.1] is an algorithm for finding the saddle point of any Lagrangian. In the simulation, it has been modified to solve the problem in the cross-silo FL scenario.

[7]We set the mean as 10, as the NE under $u_n = 10$ for $n \in \mathcal{N}$ leads to the number of training rounds that can achieve a digit recognition correctness rate of more than 90%. Our observations also hold for other values of mean.



(a) (b)

Fig. 4. Effect of valuation difference $\overline{u} - \underline{u}$ (with $(\overline{u} + \underline{u})/2 = 10$) on: (a) monetary transfer m_n; (b) social welfare (SW).



(a) (b)

Fig. 5. Effect of average valuation $(\overline{u} + \underline{u})/2$ (with $\overline{u} - \underline{u} = 12$) on: (a) monetary transfer m_n; (b) social welfare (SW).

This implies that after the algorithm has converged, the social welfare is maximized. In addition, the unit monetary transfer $\zeta_n \triangleq \pi_{\mu(n+1)} - \pi_{\mu(n+2)}$ converges to zero for all $n \in \mathcal{N}$. Intuitively, since the organizations are homogeneous, they do not need to pay each other to motivate the cooperation.

In Fig. 3 (c) and (d), we consider a heterogeneous scenario where $u_n = 4$ for $0 \leq n \leq 4$ and $u_n = 16$ for $5 \leq n \leq 9$. Similarly, γ_n gradually converges to $r(\boldsymbol{f}^*)$. For the unit monetary transfer, ζ_n for $0 \leq n \leq 4$ converges to a positive value $\zeta^{\rm NE}$, and ζ_n for $5 \leq n \leq 9$ converges to the negative value of $\zeta^{\rm NE}$, i.e., $-\zeta^{\rm NE}$. Note that $\zeta^{\rm NE}$ and $-\zeta^{\rm NE}$ are the unit monetary transfer for an organization $0 \leq n \leq 4$ and $5 \leq n \leq 9$ under NE, respectively. Intuitively, the organizations in set $\{0, 1, 2, 3, 4\}$ have lower valuation, so the other organizations have to pay them to motivate their participation.

C. Nash Equilibrium under Different Scenarios

Figs. 4 and 5 show the monetary transfer of the organizations and the social welfare under the NE of Game 1. Let $m_n^{\rm NE}$ denote the monetary transfer to organization $n \in \mathcal{N}$ under NE. In these simulations, we set $u_n = \underline{u}$ for $0 \leq n \leq 4$ and $u_n = \overline{u}$ for $5 \leq n \leq 9$. In Fig. 4 (a), as the valuation difference $\overline{u} - \underline{u}$ increases, the payment from the organizations with higher valuation (i.e., $5 \leq n \leq 9$) to those with lower valuation (i.e., $0 \leq n \leq 4$) increases. Intuitively, if the organizations are more heterogeneous in terms of their valuation (i.e., $\overline{u} - \underline{u}$ is larger), those with higher valuation should pay more to motivate those with lower valuation to cooperate. In Fig. 4 (b), as we have proven, the social welfare under NE is always equal to the optimal social welfare. The social welfare under NE does not change with $\overline{u} - \underline{u}$, as the optimal processing capacity does not change with $\overline{u} - \underline{u}$ given any $(\overline{u} + \underline{u})/2$.

In Fig. 5 (a), as the average valuation $(\overline{u} + \underline{u})/2$ increases, the payment from the organizations with higher valuation (i.e., $5 \leq n \leq 9$) to those with lower valuation (i.e., $0 \leq n \leq 4$) reduces. This is because the relative valuation difference (i.e.,

$(\overline{u}-\underline{u})/((\overline{u}+\underline{u})/2)$) decreases. In Fig. 5 (b), as $(\overline{u}+\underline{u})/2$ increases, the social welfare under NE increases. This implies that when the organizations have higher valuation, they can achieve higher social welfare by participating in cross-silo FL, so the proposed mechanism and algorithm are more beneficial.

VI. Conclusion

In this work, we proposed an incentive mechanism for cross-silo FL that addresses the public goods feature. The proposed mechanism can achieve social efficiency, individual rationality, and budget balance. We further proposed a distributed algorithm that enables the organizations to achieve social efficiency without knowing the private information of each other. The simulation results with MNIST dataset show that the proposed algorithm can achieve faster convergence than the conventional Lagrangian method. Meanwhile, the proposed mechanism and algorithm enable the organizations to achieve higher social welfare through participating in cross-silo FL, especially when the organizations have high valuation on precision. For future work, one direction is to further consider the non-independent and identically distributed (non-i.i.d.) data of the organizations. Another research direction is to consider the scenario where the organizations can choose the number of training rounds that they participate in according to their valuation on precision and their computational and communication costs.

Appendix

A. Proof for Lemma 1

Suppose $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})$ satisfies (14) but is not an NE. Then, there exists γ_n' for $n\in\mathcal{N}$ such that for $\boldsymbol{\gamma}'=(\gamma_n',\boldsymbol{\gamma}_{-n}^{\rm NE})$,

$$\begin{aligned}&V_n(f_n^\star(\widetilde{r}(\boldsymbol{\gamma}')),\widetilde{r}(\boldsymbol{\gamma}'),m_n(\widetilde{r}(\boldsymbol{\gamma}')\mathbf{1},\boldsymbol{\pi}^{\rm NE}))\\&\quad>V_n(f_n^\star(\widetilde{r}(\boldsymbol{\gamma}^{\rm NE})),\widetilde{r}(\boldsymbol{\gamma}^{\rm NE}),m_n(\widetilde{r}(\boldsymbol{\gamma}^{\rm NE})\mathbf{1},\boldsymbol{\pi}^{\rm NE})),\end{aligned}\tag{19}$$

where inequality (19) contradicts inequality (14).

Suppose $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})$ is an NE of Game 1, while (14) does not hold for some $n\in\mathcal{N}$. Then, there exists γ_n' for an organization $n\in\mathcal{N}$ such that for $\boldsymbol{\gamma}'=(\gamma_n',\boldsymbol{\gamma}_{-n}^{\rm NE})$,

$$\begin{aligned}&V_n(f_n(\boldsymbol{\gamma}'),\widetilde{r}(\boldsymbol{\gamma}'),m_n(\boldsymbol{\gamma}',\boldsymbol{\pi}^{\rm NE}))\\&\quad>V_n(f_n(\boldsymbol{\gamma}^{\rm NE}),\widetilde{r}(\boldsymbol{\gamma}^{\rm NE}),m_n(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})),\end{aligned}\tag{20}$$

which violates the assumption that $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})$ is an NE.

B. Proof for Theorem 1

To prove Theorem 1, we formulate an optimization problem with respect to the number of training rounds:

$$\underset{r}{\text{maximize}}\quad \textstyle\sum_{n\in\mathcal{N}}\left(U_n(r)-C_n(f_n^\star(r),r)\right)\tag{21a}$$

$$\text{subject to}\quad r\in[0,\bar{r}].\tag{21b}$$

If r^* is an optimal solution to problem (21), then we can prove that the processing capacity vector $\boldsymbol{f}^\star(r^*)=(f_n^\star(r^*),n\in\mathcal{N})$ is an optimal solution to problem (8).

We now prove that for any NE of Game 1 $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})$, $\boldsymbol{f}(\boldsymbol{\gamma}^{\rm NE})$ is an optimal solution to problem (8). This is proven by showing that $r^{\rm NE}\triangleq\widetilde{r}(\boldsymbol{\gamma}^{\rm NE})$ is the optimal solution to problem (21). In the following, we first analyze problem (21) and then prove that $r^{\rm NE}$ is an optimal solution to (21).

As problem (21) is a convex problem with strictly concave objective function and continuously differentiable linear constraints, its Karush-Kuhn-Tucker (KKT) conditions are sufficient for optimality. Hence, r^* is an optimal solution to (21) if there exist Lagrange multipliers α^* and β^* such that the following KKT conditions are satisfied:[8]

$$\sum_{n\in\mathcal{N}}\left(\frac{\partial U_n(r^*)}{\partial r^*}-\frac{\partial C_n(f_n^\star(r^*),r^*)}{\partial r^*}\right)+\alpha^*-\beta^*=0,\tag{22a}$$

$$r^*\geq 0,\ r^*\leq\bar{r},\ \alpha^*\geq 0,\ \beta^*\geq 0,\tag{22b}$$

$$\alpha^*r^*=0,\ \beta^*(r^*-\bar{r})=0.\tag{22c}$$

In terms of the NE $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})$ and the resulting $r^{\rm NE}$, according to Lemma 1, there exist $\boldsymbol{\alpha}^{\rm NE}=(\alpha_n^{\rm NE},n\in\mathcal{N})$ and $\boldsymbol{\beta}^{\rm NE}=(\beta_n^{\rm NE},n\in\mathcal{N})$ such that the following holds:

$$r^{\rm NE}=\textstyle\sum_{n\in\mathcal{N}}\gamma_n^{\rm NE}/N,\tag{23a}$$

$$\begin{aligned}&\frac{\partial U_n(r^{\rm NE})}{\partial r^{\rm NE}}-\frac{\partial C_n(f_n^\star(r^{\rm NE}),r^{\rm NE})}{\partial r^{\rm NE}}\\&+(\pi_{\mu(n+1)}^{\rm NE}-\pi_{\mu(n+2)}^{\rm NE})+\alpha_n^{\rm NE}-\beta_n^{\rm NE}=0,\ n\in\mathcal{N},\end{aligned}\tag{23b}$$

$$r^{\rm NE}\geq 0,\ r^{\rm NE}\leq\bar{r},\alpha_n^{\rm NE}\geq 0,\ \beta_n^{\rm NE}\geq 0,\ n\in\mathcal{N},\tag{23c}$$

$$\alpha_n^{\rm NE}r^{\rm NE}=0,\ \beta_n^{\rm NE}(r^{\rm NE}-\bar{r})=0,\ n\in\mathcal{N}.\tag{23d}$$

Conditions (23b)−(23d) correspond to the KKT conditions of $\max_{r\in[0,\bar{r}]}V_n(f_n^\star(r),r,m_n(r\mathbf{1},\boldsymbol{\pi}^{\rm NE}))$ for each $n\in\mathcal{N}$. To prove that $r^{\rm NE}$ is an optimal solution to problem (21), let $\alpha^*=\sum_{n\in\mathcal{N}}(\pi_{\mu(n+1)}^{\rm NE}-\pi_{\mu(n+2)}^{\rm NE})+\sum_{n\in\mathcal{N}}\alpha_n^{\rm NE}=\sum_{n\in\mathcal{N}}\alpha_n^{\rm NE}$, and let $\beta^*=\sum_{n\in\mathcal{N}}\beta_n^{\rm NE}$. Then, we have $(r^*=r^{\rm NE},\alpha^*,\beta^*)$ satisfies the KKT conditions in (22).

C. Proof for Lemma 2

The saddle point of $\mathcal{L}(\boldsymbol{r},\boldsymbol{\lambda})$, denoted by $(\boldsymbol{r}^*,\boldsymbol{\lambda}^*)$, satisfies the KKT conditions of problem (15) as follows:

$$\begin{aligned}&\frac{\partial U_n(r_n^*)}{\partial r_n^*}-\frac{\partial C_n(f_n^\star(r_n^*),r_n^*)}{\partial r_n^*}-\left(\lambda_{\mu(n+2)}^*-\lambda_{\mu(n+1)}^*\right)\\&\qquad+\widetilde{\alpha}_n^*-\widetilde{\beta}_n^*=0,\ n\in\mathcal{N},\end{aligned}\tag{24a}$$

$$r_{\mu(n-2)}^*=r_{\mu(n-1)}^*,\ r_n^*\geq 0,\ r_n^*\leq\bar{r},\ n\in\mathcal{N},\tag{24b}$$

$$\widetilde{\alpha}_n^*\geq 0,\ \widetilde{\beta}_n^*\geq 0,\ \widetilde{\alpha}_n^*r_n^*=0,\ \widetilde{\beta}_n^*(r_n^*-\bar{r})=0,\ n\in\mathcal{N},\tag{24c}$$

where $\widetilde{\alpha}_n^*$ and $\widetilde{\beta}_n^*$ are the Lagrange multipliers corresponding to constraints $r_n\geq 0$ and $r_n\leq\bar{r}$ for $n\in\mathcal{N}$, respectively. Let $\gamma_n^{\rm NE}=r_n^*$, $\pi_n^{\rm NE}=\lambda_n^*$, $\alpha_n^{\rm NE}=\widetilde{\alpha}_n^*$, and $\beta_n^{\rm NE}=\widetilde{\beta}_n^*$ for $n\in\mathcal{N}$. Then, $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE},\boldsymbol{\alpha}^{\rm NE},\boldsymbol{\beta}^{\rm NE})$ leads to (23), i.e., $(\boldsymbol{\gamma}^{\rm NE},\boldsymbol{\pi}^{\rm NE})$ is an NE of Game 1.

[8]We use α and β to refer to the Lagrange multipliers in this appendix to distinguish them from the Lagrange multipliers in Section IV.

REFERENCES

[1] P. Kairouz *et al.*, "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning*, vol. 14, no. 1, Mar. 2021 (Early Access).

[2] Owkin, "Owkin solutions," https://owkin.com/solutions/#data-connect, accessed Jan. 11, 2021.

[3] Life Insurance, "Swiss Re and Chinese firm WeBank partner to explore use of AI in reinsurance," https://www.lifeinsuranceinternational.com/news/swiss-re-webank/, accessed Jan. 11, 2021.

[4] MELLODDY, "Machine learning ledger orchestration for drug discovery," https://www.melloddy.eu, accessed Jan. 11, 2021.

[5] FeatureCloud, "Our vision," https://featurecloud.eu/about/our-vision/, accessed Jan. 11, 2021.

[6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int'l Conf. on Artificial Intelligence and Statistics (AISTATS)*, Fort Lauderdale, FL, Apr. 2017.

[7] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. Int'l Conf. on Machine Learning (ICML)*, Jul. 2020.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Conf. on Machine Learning and Systems (MLSys)*, Austin, TX, Mar. 2020.

[9] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," in *Proc. Int'l Conf. on Learning Representations (ICLR)*, Apr. 2020.

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

[11] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing federated learning on non-IID data with reinforcement learning," in *Proc. IEEE INFOCOM*, Jul. 2020.

[12] X. Zhang, F. Li, Z. Zhang, Q. Li, C. Wang, and J. Wu, "Enabling execution assurance of federated learning at untrusted participants," in *Proc. IEEE INFOCOM*, Jul. 2020.

[13] 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 *Proc. IEEE INFOCOM*, Paris, France, Apr. 2019.

[14] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.

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

[16] R. Cornes and T. Sandler, *The Theory of Externalities, Public Goods, and Club Goods*. Cambridge University Press, 1996.

[17] M. Tang, H. Pang, S. Wang, L. Gao, J. Huang, and L. Sun, "Multi-dimensional auction mechanisms for crowdsourced mobile video streaming," *IEEE/ACM Trans. Netw.*, vol. 26, no. 5, pp. 2062–2075, Oct. 2018.

[18] N. Zhao, Y.-C. Liang, and Y. Pei, "Dynamic contract incentive mechanism for cooperative wireless networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10 970–10 982, Nov. 2018.

[19] Q. Shao, M. H. Cheung, and J. Huang, "Multimedia crowdsourcing with bounded rationality: A cognitive hierarchy perspective," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1478–1488, Jul. 2019.

[20] W. Y. B. Lim, Z. Xiong, C. Miao, D. Niyato, Q. Yang, C. Leung, and H. V. Poor, "Hierarchical incentive mechanism design for federated machine learning in mobile networks," *IEEE Internet of Things J.*, vol. 7, no. 10, pp. 9575–9588, Oct. 2020.

[21] 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 Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3241–3256, May 2020.

[22] 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 J.*, vol. 6, no. 6, pp. 10 700–10 714, Dec. 2019.

[23] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, "A learning-based incentive mechanism for federated learning," *IEEE Internet of Things J.*, vol. 7, no. 7, pp. 6360–6368, Jul. 2020.

[24] A. Kakhbod and D. Teneketzis, "An efficient game form for multi-rate multicast service provisioning," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 11, pp. 2093–2104, Dec. 2012.

[25] M. Zhang, J. Huang, and R. Zhang, "Wireless power transfer with information asymmetry: A public goods perspective," *IEEE Trans. Mobile Comput.*, vol. 20, no. 1, pp. 276–291, Jan. 2021.

[26] Y. LeCun, C. Cortes, and C. J.C. Burges, "The MNIST database," http://yann.lecun.com/exdb/mnist/, accessed Jan. 11, 2021.

[27] M. Benzi, G. H. Golub, and J. Liesen, "Numerical solution of saddle point problems," *Acta Numerica*, vol. 14, pp. 1–137, 2005.

[28] Microsoft Azure, "Linux virtual machines pricing," https://azure.microsoft.com/en-ca/pricing/details/virtual-machines/linux/, accessed Jan. 11, 2021.

[29] N. Chatzipanagiotis, D. Dentcheva, and M. M. Zavlanos, "An augmented Lagrangian method for distributed optimization," *Mathematical Programming*, vol. 152, pp. 405–434, Sep. 2015.

[30] SPEEDTEST, "Speedtest global index," https://www.speedtest.net/global-index, accessed Jan. 11, 2021.

[31] Microsoft Azure, "Cloud services pricing," https://azure.microsoft.com/en-us/pricing/details/cloud-services, accessed Jan. 11, 2021.

[32] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: A measurement study and implications for network applications," in *Proc. ACM SIGCOMM Conf. on Internet Measurement*, Chicago, IL, Nov. 2009.

[33] E. Hub, "Electricity prices in Canada 2020," https://www.energyhub.org/electricity-prices/, accessed Jan. 11, 2021.

[34] MathWorks, "boxplot," https://www.mathworks.com/help/stats/boxplot.html, accessed Jan. 11, 2021.