VARF: An Incentive Mechanism of Cross-Silo Federated Learning in MEC

Ying Li, Xingwei Wang, Rongfei Zeng, Mingzhou Yang, Kexin Li, *Student Member, IEEE*, Min Huang, *Member, IEEE*, and Schahram Dustdar, *Fellow, IEEE*

***Abstract*—Cross-silo federated learning (FL) is a privacy-preserving distributed machine learning where organizations acting as clients cooperatively train a global model without uploading their raw local data. Recently, the cross-silo FL in multiaccess edge computing (MEC) is used in increasing industrial applications. Most existing research on cross-silo FL pays attention to the performance aspect, ignoring the incentive mechanism for high-quality client selection and long participation in model training for efficient and stable FL, which has prevented the widespread adoption of cross-silo FL in MEC. In this article, we propose an incentive mechanism with quality-Aware and reputation-Aware based on the infinitely repeated game for cross-silo FL named VARF. VARF selects high-quality and high-reputation edge nodes (ENs) as candidates for model training in the cross-silo FL by a heuristic algorithm and then motivates the selected ENs to actively contribute their resources. VARF also models the long-term behavior of ENs in cross-silo FL as an infinitely repeated game and derives a stable and long-term cooperative strategy for clients while maximizing the amount of local data for model learning in cross-silo FL. Extensive simulations with real-world data sets demonstrate that the performance of VARF is more beneficial than other benchmarks. Meanwhile, experimental results show that cloud platforms (CPs) and ENs eventually form a long and stable cooperative relationship under the trigger strategy.**

***Index Terms*—Cross-silo federated learning (FL), long-term, multiaccess edge computing (MEC), quality, repeated game, reputation.**

I. Introduction

With the rise of machine learning (ML) and the popularity of Internet of Things (IoT) devices, tremendous data produced by IoT devices makes it difficult to process the data using traditional methods. In the traditional cloud-based approach, data collected by IoT devices is uploaded and processed at cloud servers or data centers in a centralized manner. Nevertheless, this approach is no longer sustainable for the following reasons. First, data owners are progressively sensitive to privacy. Second, the cloud-based approach involves long propagation delays and leads to unacceptable delays for applications that must make real-time decisions. Third, transferring data to the cloud for processing places a burden on the backbone network. Fortunately, multiaccess edge computing (MEC) [1], [2], [3], [4] can enable edge nodes (ENs) to locally collect and process various data under the coordination of remote cloud [5], which alleviates the data privacy issues, latency, and communication inefficiency of the traditional cloud-based approach. However, ML in MEC still needs to provide shared data externally, which leads to the disclosure of data privacy.

In view of increasingly stringent data privacy regulations, federated learning (FL), a privacy-preserving distributed ML, was introduced in MEC [5], [6], [7], [8]. FL is aimed to learn a shared model by performing distributed training locally on participating clients and aggregating the local models into a global one. Compared with traditional ML, the client (i.e., device or organization) in FL just sends a local model update trained on its own local raw data to the central server rather than uploading the raw data of the client. Simply put, the utility of data is well maintained and data privacy can be preserved. FL as an enabling technology of edge intelligence (EI) enables ENs to conduct model training without transmitting their raw data to central servers [9], which counters the privacy issue of clients and alleviates the communication burden.

In recent years, FL as an emerging branch of ML has attracted a lot of attention from academia and industry. Extensive research was conducted on the statistical Challenges, privacy and security issues, model, and system Challenges areas. Specifically, researchers studied distributed optimization [10], [11], [12], privacy [13], [14], adversarial attack and defense [15], [16], communication efficiency [17], [18], [19], computation efficiency [20], [21], and model [22]. Above all focus on the performance improvement of FL, these studies are based on an optimistic assumption that all the clients contribute their resources unconditionally [23], [24], which is not practical in the real scenarios of MEC.

The research on the incentive mechanism is fundamental and significant to FL in MEC. Most importantly, the

Manuscript received 22 November 2022; revised 1 March 2023; accepted 24 March 2023. Date of publication 5 April 2023; date of current version 24 August 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 92267206 and Grant 62032013, and in part by the Science and Technology Planning Project of Shenyang City under Grant 21-108-9-19. *(Corresponding author: Xingwei Wang.)*

Ying Li, Xingwei Wang, and Kexin Li are with the College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China (e-mail: liying1771@163.com; wangxw@mail.neu.edu.cn; NEULiKexin@stumail.neu.edu.cn).

Rongfei Zeng is with the College of Software, Northeastern University, Shenyang 110819, China (e-mail: zengrf@swc.neu.edu.cn).

Mingzhou Yang is with the School of Information Science and Engineering, Shenyang University of Technology, Shenyang 110178, China (e-mail: yangmingzhou11@163.com).

Min Huang is with the College of Information Science and Engineering, Northeastern University, Shenyang 110819, China (e-mail: mhuang@mail.neu.edu.cn).

Schahram Dustdar is with the Distributed Systems Group, TU Wien, 1040 Vienna, Austria (e-mail: dustdar@dsg.tuwien.ac.at).

Digital Object Identifier 10.1109/JIOT.2023.3264611

clients participating in model training consume their computation resources and communication resources [25]. Without well-designed economic compensation, self-interested clients are not enthusiastic to participate in model training of MEC. Furthermore, uploading their model update to the central server may lead to security and privacy issues (e.g., model stealing, and inference attack). Moreover, clients may intentionally or unintentionally perform undesirable behaviors resulting in low-quality local model updates, which can degrade the performance of the global model and cause model convergence problems [26], [27]. Namely, deficient participants might cause FL to malfunction in reality [28]. Besides, cross-silo FL in MEC has a wider range of industrial applications compared with cross-device FL. Consequently, it is necessary and critical to formulate an efficient and stable incentive mechanism for cross-silo FL in MEC.

Existing incentive mechanisms are not directly applicable to cross-silo FL in MEC. First of all, the resource gap [29] between different ENs and unreliable ENs may perform undesirable behaviors both will depress the performance of the global model. Therefore, the incentive mechanism should take model update quality and client reliability into consideration. Moreover, although there have been existing studies and research on designing incentive mechanisms for cross-device FL, the incentive mechanisms for cross-device FL cannot be utilized for cross-silo FL. This is because cross-silo FL involves multiple organizations or entities with their own data, goals, and interests, which makes it more complex and challenging to incentivize participation. In such scenarios, a different set of incentive mechanisms that consider the varying needs and interests of different organizations must be designed and implemented. These mechanisms may involve incentives such as revenue sharing, or access to valuable data or resources. On the other hand, ENs acted as organizations in cross-silo FL may execute multiple cross-silo FL processes repeatedly as a consequence of their time-varying local data set but the challenge that long-term selfish participation behaviors of ENs is unsolved. According to the above challenges, We should design a suitable incentive mechanism for cross-silo FL in MEC.

However, the current cross-silo FL incentive mechanisms face several challenges, including a lack of trust, unequal contributions, and free-riding. Therefore, we aim to design an incentive mechanism to motivate high-quality and high-reputation ENs to actively and long participate in model learning of cross-silo FL and ultimately enhance the performance of FL in MEC. Meanwhile, the ENs who participate in model training will be rewarded according to their contribution degree. To achieve the goal, We first estimate the learning quality and learning reputation of clients leveraging historical learning records, where the freshness of the historical records is considered, and assign weights using an exponential forgetting function. Then, we select the high-quality and high-reputation ENs as candidates for training by a heuristic algorithm. Infinitely repeated games allow for the possibility of building long-term relationships and establishing reputations, which can have a significant impact on the incentive mechanisms. Thus, we next model the long-term behavior of ENs in cross-silo FL as an infinitely repeated game, where the trigger strategy ensures that both the cloud platform (CP) and ENs actively and long participate in the task of cross-silo FL. Finally, we demonstrate the performance of the proposal through extensive simulations with multiple real-world data sets and learning models.

The major contributions of this article are as follows.

1) *Design of Incentive Mechanism for Cross-Silo FL in MEC:* We study the quality-aware and reputation-aware cross-silo FL in MEC, where learning quality and learning reputation are estimated to select high-quality and high-reputation ENs for model training. Moreover, we optimize the long-term profits of ENs by allowing them to selfishly choose degrees of contribution. It is the first work to investigate the incentive mechanism of cross-silo FL in MEC under the consideration of data quality, the reliability of ENs, and the long-term participation of ENs.
2) *Incentive Mechanism (VARF):* We propose an incentive mechanism with quality-aware and reputation-aware for cross-silo FL based on the infinitely repeated game named VARF. VARF can select the high-quality and high-reputation ENs as candidates by a heuristic algorithm, then optimize the long-term profits of ENs by solving profit maximization problems of the efficiency wage model based on the infinitely repeated game for CP and ENs. Finally, we verify the stability of the trigger strategy in the infinitely repeated game and the trigger strategy ensures a long and stable cooperative relationship between CPs and ENs.
3) *Performance Evaluation:* We conduct extensive experiments based on real-world data sets and widely adopted learning models, where the incentive mechanism can recruit high-reputation ENs to contribute high-quality data and motivate ENs to maintain long-term cooperation. The simulation results express that the proposed schema has higher accuracy and lower loss than the other three benchmarks, the detail as shown in Section VI. Moreover, experimental results also show that CPs and ENs eventually form a long and stable cooperative relationship under the trigger strategy.

The structure of this article is as follows. Section II presents some related work. In Section III, we introduce the system overview, followed by the model and formulation in Section IV. We express the incentive mechanism VARF based on the infinitely repeated game in Section V. The experimental studies are presented in Section VI. In Section VII, we conclude this article.

II. Related Work

In this section, we introduce related work regarding FL, cross-silo FL, and incentive mechanism.

A. Federated Learning

Recently, the performance optimization of promising FL has gained the attention of industry and academia. Wang et al. [8] notionally considered the convergence bound of distributed gradient descent and made an optimal tradeoff between local updates and global parameter aggregation to minimize the loss function for a given resource budget. Wang et al. [30] proposed communication-mitigated FL (CMFL) which can markedly reduce the communication burden while ensuring learning convergence. Luo et al. [31] proposed a low-cost sampling-based algorithm to learn the unknown parameters associated with convergence. Wang et al. [32] achieved an experience-driven control framework that can counterpoise the bias conducted by non-IID data and accelerate convergence. These researches are based on the assumption of voluntary participation of clients. However, without well-designed incentive mechanisms, clients are unwilling to voluntarily participate in model training.

B. Cross-Silo FL

A few researchers have analyzed cross-silo FL. Tang and Wong [33] proposed an incentive mechanism for the features of public goods in cross-silo FL. Zhang et al. analyzed the selfish participation and long-term behaviors of heterogeneous clients in cross-silo FL. Marfoq et al. [34] defined the topology design problem for cross-silo to maximize the number of communication. Majeed et al. [35] built a horizontal cross-silo FL model for traffic classification using flow-based time-related features. Heikkilä et al. [36] learned complex models and guaranteed rigorous privacy in cross-silo FL under the combination of additively homomorphic secure summation protocols with differential privacy. Although there is some research about cross-silo FL, none of them considers the quality, reputation, and long-term participation and cooperation of clients in cross-silo.

C. Incentive Mechanism

There are more and more papers concentrating on the incentive mechanism design for FL in recent years. Deng et al. [25] proposed a quality-aware FL system, which significantly improves the quality of distributed learning through precise user incentives and model aggregation. Kang et al. [24] presented an incentive mechanism that considers reputation and contract theory to motivate reliable clients to participate in the task of FL. Tang and Wong [33] designed an incentive mechanism for cross-silo FL targeting the features of public goods. Ma et al. [37] designed an auction mechanism that integrates the concept of padding and an efficient pricing strategy to guarantee desired properties in restricted MEC environments. Zeng et al. [38] proposed three multidimensional game-theoretic models to study the economic behaviors of participants and validated their applicability. However, none of the above papers analyzes the long-term participation of high-quality and high-reputation clients for cross-silo FL in MEC.

D. Infinitely Repeated Game

Infinitely repeated games allow for the development of long-term relationships and the emergence of cooperative behavior, which can lead to outcomes that are more efficient and beneficial for all players involved. Thus, incentive mechanisms

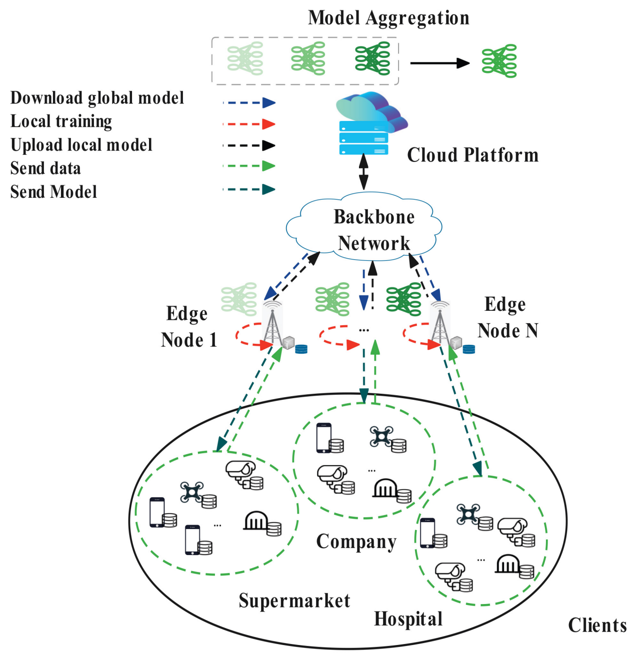

Fig. 1. Overview of cloud-edge-based cross-silo FL system.

in infinitely repeated games can be very effective at encouraging cooperation and deterring defection. Chi et al. [39] proposed a multistrategy repeated game-based incentive mechanism to guide participants to provide long-term participation and high-quality data, but the quality of data and reputation of participants are not evaluated before model training. In addition, the stability of the trigger strategy is not demonstrated. Li et al. [40] proposed an incentive model based on the infinitely repeated game and designed a trigger strategy with credible threats. Zhao et al. [41] proposed a three-party repeated game model based on game theory, which indicates that the proposed model is feasible and effective in maximizing the benefits of game participants. However, the issue of quality and reputation is ignored in both papers above.

In summary, research on the long-term participation of high-quality and high-reputation clients in cross-silo is largely unexplored. Moreover, incentive mechanisms previously used in other scenarios cannot be directly applied to FL in MEC [5]. That's because the widening resource gap between different ENs might degrade the performance of the global model for FL. Consequently, designing an incentive scheme that can encourage active and long-term participation of high-quality and high-reputation nodes is essential and crucial.

III. System Overview

In this section, we first present a cloud-edge-based cross-silo FL system in MEC. Next, we introduce the cost incurred, utilities and profits during the process of participating in the task in detail. Then, we formulate the long-term profit maximization problems for both CP and ENs.

A. System Description

As depicted in Fig. 1, we consider a scene with a CP and N ENs. Let $\mathcal{N} = \{1, 2, \ldots, n, \ldots, N\}$ denotes the set of ENs.

TABLE I
KEY SYMBOLS USED IN THIS ARTICLE

Symbol	Description
N	The number of ENs
$\mathcal{N}$	The set of ENs
$\mathcal{N}'$	The set of selected ENs
K	The number of iterations in one round
B	The total batch size
j	The number of the selected ENs
ξ	The fade parameter about freshness
ζ	The effective capacitance parameter for EN n
G	Transmission bandwidth
θ_n	The valuation of EN n for the model accuracy
$\mathcal{S}_n$	The local dataset of EN n
S_n	The size of $\mathcal{S}_n$
$\mathcal{X}_n$	The chosen subset for EN n
x_n	The size of $\mathcal{X}_n$
f_n^t	The processing capacity used by EN n
c_n^t	The number of CPU cycles needed by EN n for a data unit
h_n^t	Channel gain between the cloud platform and EN n
ρ_n^t	Transmission power of the EN n
N_0^t	Background noise
d_n^t	The data quality of EN n in round t
q_n^t	The learning quality of EN n in round t
$R_{c\to n}^t$	The reputation of the CP for the EN n in round t

EN n, $n \in \mathcal{N}$, has a local data set $\mathcal{S}_n$ collected from the edge devices (EDs) it covers which may change over time, where the number of the local data set is $S_n = |\mathcal{S}_n|$. Therefore, each EN may iteratively perform cross-silo FL processes owing to its time-varying data set. EN n can choose a subset $\mathcal{X}_n \subseteq \mathcal{S}_n$ of its local data set for local model training. Let x_n denote the size of the chosen subset, i.e., $x_n = |\mathcal{X}_n|$. Then, all local model updates on the CP are aggregated to generate the global model. The detailed key symbols used in this article are listed in Table I. Cross-silo FL aims to get the optimal weights of the global model ω^* to minimize the global loss function $L(\omega)$ [42]

$$L(\omega) = \sum_{n\in\mathcal{N}} \frac{x_n}{\sum_{n'\in\mathcal{N}} x_{n'}} L_n(\omega; \mathcal{X}_n) \quad (1)$$

where $L_n(\omega; \mathcal{X}_n)$ is the loss for subset $\mathcal{X}_n$ given ω

$$\omega^* = \arg\min_{\omega} L(\omega). \quad (2)$$

In round t, the CP distributes the global parameter w^t to the ENs selected. Here, we suppose that $\mathcal{N}'$ is the set of the ENs selected and the number of the ENs selected is j. Then, the ENs selected train their own local model with the local data based on the global parameter w^t as follows:

$$\omega_n^{t+1} = \omega^t - \eta \nabla L_n\left(w_n^t\right) \quad (3)$$

where η is the step size.

Next, the selected ENs transfer their local model ω_n^{t+1} to CP, and the updated global model is calculated as

$$\omega^{t+1} = \frac{\sum_{n=1}^{j} x_n \omega_n^{t+1}}{\sum_{n=1}^{j} x_n}. \quad (4)$$

Several rounds are required to achieve convergence for cross-silo FL, in which one round consists of K iterations. The training process is terminated when the accuracy of the global model meets the requirements or the training time exceeds a predefined threshold.

B. Profit of CP and ENs in One Cross-Silo FL Process

In this section, we present the cost, utility, and profit of the efficiency wage model for the CP and ENs in one cross-silo FL process in detail.

1) Cost of ENs in One Cross-Silo FL: Undoubtedly, the cloud-edge-based cross-silo FL process in MEC mentioned above brings the cost to ENs. The cost of ENs is given in detail as follows.

Model Accuracy Loss: The purpose of model training by ENs is to obtain a global model with good accuracy. The global model with high accuracy means the global model with little loss of accuracy [43]. $L(w^K) - L(w^*)$ is devoted to calculating the accuracy loss of the global model, where $L(w^K)$ and $L(w^*)$ are the global losses with parameters w^K and w^*, respectively, and K is the number of iterations in one cross-silo FL process. As stated by [43], [44], and [45], the expected global model accuracy loss is bounded by $O([1/\sqrt{BK}] + [1/K])$, where B is the total batch size, namely, $B = \sum_{n\in\mathcal{N}'} \mathcal{X}_n^t$. As noted above, $O([1/\sqrt{BK}] + [1/K])$ decreases when the total amount of all chosen local data $\sum_{n\in\mathcal{N}'} \mathcal{X}_n^t$ for model training and the number of iterations K increase. Thus, we denote the model accuracy loss of EN n, $n \in \mathcal{N}'$, in round t as $L(\mathcal{X}_n^t, \mathcal{X}_{-n}^t)$, which depends on the amount of data contributed by all ENs selected for model training when the number of iterations K is fixed and can be computed by

$$L(\mathcal{X}_n^t, \mathcal{X}_{-n}^t) = \frac{1}{\sqrt{\left(x_n^t + \sum_{n'\in\mathcal{N}', n'\neq n} x_{n'}^t\right)K}} + \frac{1}{K} \quad (5)$$

where $x_{-n}^t = \{x_1^t, x_2^t, \ldots, x_{n-1}^t, x_{n+1}^t, \ldots, x_j^t\}$.

Computation Cost: In the cloud-edge-based cross-silo FL, the processing capacity for EN n is denoted as f_n^t (i.e., CPU-cycle frequency). Denote the number of CPU cycles required by EN n to execute a data unit by c_n^t. According to [24], [33], [45], [46], and [47] the computation cost of EN n for one iteration in round t can be calculated by

$$E_{n,t}^{\text{cmp}} = \zeta c_n^t x_n^t f_n^{t\,2} \quad (6)$$

where ζ is the effective capacitance parameter of the computing chipset for EN n. This article assumes that ζ, c_n^t, and f_n^t are the same for all ENs. In this case, we denote the computation cost of the EN $n \in \mathcal{N}$ as $E_{n,t}^{\text{cmp}} = Hx_n^t$, where $H = \zeta c_n^t f_n^{t\,2}$.

Communication Cost: After K local training iterations, ENs selected transfer local model updates w_n^t to CP for model aggregation, then receive the global model w^t for the next round of model training. The data transmission between ENs and CP can be through either the wired network or the wireless networks [45]. G is transmission bandwidth, h_n^t is channel gain between EN n and the CP, ρ_n^t is transmission power of the EN n and N_0^t is the background noise. The communication consumption of the EN n in a global training is denoted as follows:

$$E_n^{\text{com}} = \frac{\sigma \rho_n^t}{G \ln\left(1 + \left[\frac{\rho_n^t h_n^t}{N_0^t}\right]\right)}. \quad (7)$$

Here, we assume that all ENs have the same communication resource, and the parameter of a local model update is σ which is an identical constant value for all ENs. In this context, all ENs consume the same communication cost. In this article, the communication cost of EN n is denoted as E_n^{com}.

Total Cost: As stated above, we describe the total cost of EN n as

$$C\left(\mathcal{X}_n^t\right) = \theta_n L\left(\mathcal{X}_n^t, \mathcal{X}_{-n}^t\right) + \text{KE}_{n,t}^{\text{cmp}} + E_n^{\text{com}} \tag{8}$$

where θ_n denotes the valuation of EN n toward the model accuracy, which expresses the importance of model accuracy to EN n [33], [45]. That is, θ_n is the unit benefit of EN n by applying the global model. For instance, θ_n can be the unit benefit loss of a supermarket when using the customer experience-driven service, or the customer churn for per unit loss of model accuracy of an insurance company when using a customer churn analysis model.

2) Profit of the CP and ENs in One Cross-Silo FL: In this section, we analyze the profit of the CP and ENs in one cross-silo FL.

EN: ENs will incur costs when they contribute their resources in cross-silo FL, and they need to be paid to incentive them to complete the training. The profit of edge ENs is the difference between reward and cost

$$U_n\left(\mathcal{X}_n^t\right) = r\left(\mathcal{X}_n^t\right) - C\left(\mathcal{X}_n^t\right) \tag{9}$$

where $U_n(\mathcal{X}_n^t)$ represents the profit that EN n participates in the training in round t, $r(\mathcal{X}_n^t)$ means the reward to EN n that CP distribute according their data contribution $\mathcal{X}_n^t$ in round t, $C(\mathcal{X}_n^t)$ is the incurred cost that EN n contributes resources in round t.

CP: The data contributed by the ENs can bring utility to the CP, and the CP needs to pay ENs according to their data contribution. The profit of CP is the difference between the utility and the reward

$$U_c\left(\mathcal{X}_n^t\right) = y\left(\mathcal{X}_n^t\right) - r\left(\mathcal{X}_n^t\right) \tag{10}$$

where $U_c(\mathcal{X}^t)$ represents the profit of CP in the round t and $y(\mathcal{X}_n^t)$ is the utility brought to CP by the data contributed $\mathcal{X}_n^t$ in round t.

C. Problem Definition

Since CPs and ENs are highly autonomous service providers, motivating them to forego short-term interests in consideration of long-term profits and to provide high-quality and reliable data continuously and stably is crucial to the research of incentive mechanisms for cross-silo FL in MEC. The CP and ENs are both rational individuals: the CP obtains more via incentivizing ENs to contribute high-quality and reliable resources for a long time while ENs benefit more by continuously and stably providing high-quality and high-reputation data. Consequently, there are two problems to be solved in this study. First, the CP has to determine the cross-silo FL which performs in round t by which ENs at which price to maximize its long-term profit. Second, ENs have to make a tradeoff between the reward obtained by contributing resources and the incurred cost. In other words, the EN n needs to decide if it should participate in the cross-silo FL or not, and how to contribute resources (i.e., the number of resources and the quality of the resources) for maximizing its own long-term profit. In this section, we present the long-term profit maximization problem for the CP and ENs. It can be formulated as follows:

$$\begin{aligned} \text{Aim}:\ & \max\left(\sum_{t\in T} U_n\left(\mathcal{X}_n^t\right), \sum_{t\in T} U_c\left(\mathcal{X}_n^t\right)\right) \\ \text{s.t.}\quad & n \in \mathcal{N}, T = 1, 2, 3, \ldots, t, \ldots \end{aligned} \tag{11}$$

As shown in (11), the aim of this article is to formulate an incentive mechanism to maximize the long-term profits of both CP and ENs.

IV. Model and Formulation

To long obtain high-quality and high-reputation resources, in this section, we first estimate the learning quality and the learning reputation of the EN n in each round according to the historical learning quality and reputation record. Then, we select the high-quality and high-reputation ENs as candidates for model training in the cross-silo FL by a heuristic algorithm. Finally, we analyze the interactions among the CP and ENs in one single time slot.

A. Learning Quality Estimation

The quantity and quality of the contributed local data both can influence the aggregated global model accuracy of FL significantly. At present, there is a lack of appropriate metrics to measure the quality of the local model update and global aggregation model. To ensure the quality of the collected data, we can form a schema for estimating the quality of the contributed data.

1) Learning Quality Quantification: Contribution evaluation is the primary task in formulating the incentive mechanism of cross-silo federal learning. The quantification of learning quality can be used to measure the contribution of clients for a trained global model. One feasible approach is to compute the cross entropy between LL^i (evaluation of the local data set on the global FL model) and LS^i (evaluation of the global data set on the local model) [48]. Nonetheless, the method consumes significant overhead when transmitting the data. Another approach is to calculate the difference between $\text{loss}_j(t_s)$ (the average test loss of task l_j^t's global model) and $\text{loss}_{i,j}(t_e)$ (the average training loss of node i's local model) as the data quality of node i in round t [25]. However, the local data of ENs actually is incremental in cross-silo FL. Thus, the difference between $\text{loss}_j(t_s)$ generated at the start of round t and $\text{loss}_{i,j}(t_e)$ formed at the end of round t cannot fully reflect the contribution of clients. Based on the loss reduction measurement method [25], we formulate a new loss reduction measurement method.

As shown in Fig. 2, the round t begins at I_{t-1}^t and terminates at I_t^{t+1}. ENs willing to participate in training need to contribute their data at I_n^t within $[I_{t-1}^t, I_t^{t+1}]$. Local model updates that are not submitted during the time slot will be rejected in round t. The selected local model updates are aggregated to generate the final global model at time I_t^{t+1}. Then, the next round starts.

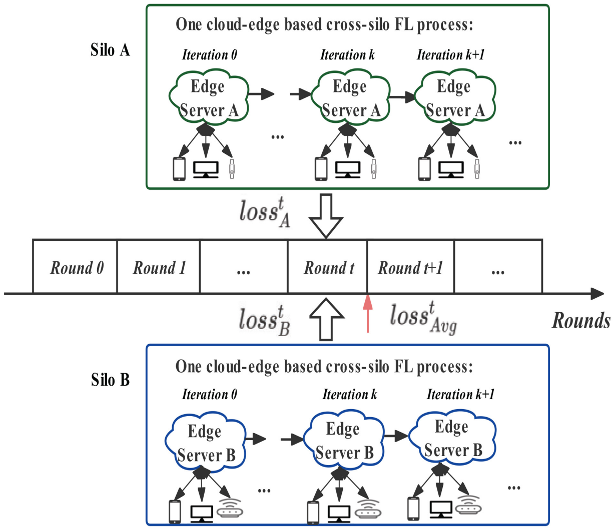

Fig. 2. Computational process for data quality in one Cross-Silo FL.

Support the average training loss of EN n is loss_n^t in round t and the average training loss of all ENs in round t is $\text{loss}_{\text{avg}}^t$. We describe the data quality of EN n in round t as follows:

$$d_n^t = \text{loss}_{\text{avg}}^t - \text{loss}_n^t. \tag{12}$$

The quantity of the training data is of vital importance, and the learning quality of EN n is defined as the product of the size of data contributed and the data quality in round t

$$q_n^t = \begin{cases} d_n^t x_n^t, & \text{if} \quad I_{t-1}^t < I_n^t \le I_t^{t+1} \\ 0, & \text{Otherwise.} \end{cases} \tag{13}$$

2) Overall Learning Quality Estimation of EN With Interaction Freshness: Data contributed by EN n Change over time in each round. And malicious nodes may contribute high-quality data only in one round to gain a chance to participate in model training and do evil during the training process. Unlike cross-device FL, the clients have an identity and can carry state from round to round in cross-silo FL [49]. Thus, we can regard the interaction among the ENs during the process of model training in cross-silos FL as long-term cooperation. According to the historical learning quality record of EN n, we can estimate the overall learning quality of EN n. The learning quality of ENs changes with time and recent learning quality has a greater weight than past learning quality. To reflect the time effect on learning quality, we adopt a freshness fading function to assign the weights: $\varkappa(t) = \xi^{T-t}$, where $\xi \in (0, 1)$ is a given fade parameter about quality freshness, T is the current round and t is the training round within $[1, T]$ [24]. Therefore, the overall learning quality of EN n up to round t is computed by

$$\hat{q}_n^t = \frac{\sum_{t=1}^{T} \varkappa(t) q_n^t}{\sum_{t=1}^{T} \varkappa(t)}. \tag{14}$$

B. Learning Reputation Estimation

1) Learning Reputation Quantification: One unreliable EN may execute intentionally or unintentionally unwelcome behaviors to mislead the training. The quantification of learning reputation is crucial to select high-quality ENs and reliable model training. The subjective logic is a widely adopted probabilistic reasoning framework [24], [50], [51], [52], [53], which is utilized to evaluate the trustworthiness or reliability of different clients. The subjective logic uses "opinion" to indicate that subjective beliefs are expressed by positive, negative, and uncertain statements [50]. In this article, we leverage a subjective logic model to generate the learning reputation of the EN n according to the learning quality of EN n

$$\begin{aligned} b_{c\to n}^t &= p_{c\to n}^t \frac{\varsigma \alpha_n^t}{\varsigma \alpha_n^t + \varphi \beta_n^t} \\ d_{c\to n}^t &= p_{c\to n}^t \frac{\varphi \beta_n^t}{\varsigma \alpha_n^t + \varphi \beta_n^t} \\ u_{c\to n}^t &= 1 - p_{c\to n}^t. \end{aligned} \tag{15}$$

In the cloud-edge-based FL system, the reputation opinion of CP c for EN n is displayed by a tuple vector $\gamma_{c\to n}^t = \{b_{c\to n}, d_{c\to n}, u_{c\to n}\}$ in round t. $b_{c\to n}, d_{c\to n}$, and $u_{c\to n}$ are belief, disbelief, and uncertainty, respectively [54]. $b_{c\to n} + d_{c\to n} + u_{c\to n} = 1$, and $b_{c\to n}, d_{c\to n}, u_{c\to n} \in [0, 1]$. α_n^t and β_n^t are the numbers of positive interactions and negative interactions in the round t separately. The CP determines whether the interaction between itself and ENs n in round t is positive or negative according to the learning quality of the EN n. Namely, the CP regards the training process as a positive interaction event between the CP and the EN n in round t if the learning quality of the EN n is greater than 0, and vice versa. $p_{c\to n}^t$ represents the probability of success for the packet transmission, which affects the uncertainty of the opinion [24]. Thus, the reputation of the CP for the EN n in round t is expressed as

$$R_{c\to n}^t = b_{c\to n}^t + a u_{c\to n}^t. \tag{16}$$

Positive interactions can enhance the reputation of ENs while negative interactions would reduce the reputation of ENs. To depress negative interaction events, the negative interactions are put on a higher weight than the positive interactions during the process of reputation calculation. ς and φ are denoted as the weights of positive and negative interactions, respectively. Here, $\varsigma \ll \varphi$ and $\varsigma + \varphi = 1$.

2) Overall Learning Reputation Estimation of EN With Interaction Freshness: In cross-silo FL, the trustworthiness of the EN changes with rounds, and the EN is not always reliable for model training. Similar to the overall learning Quality estimation of EN with interaction freshness, we adopt the freshness fading function $\varkappa(t)$ to assign the weights. Hence, the overall learning reputation of the EN n up to round t can be computed by

$$\begin{aligned} \hat{b}_{c\to n}^t &= \frac{\sum_{t=1}^{T} \varkappa(t) b_{c\to n}^t}{\sum_{t=1}^{T} \varkappa(t)} \\ \hat{d}_{c\to n}^t &= \frac{\sum_{t=1}^{T} \varkappa(t) d_{c\to n}^t}{\sum_{t=1}^{T} \varkappa(t)} \\ \hat{u}_{c\to n}^t &= \frac{\sum_{t=1}^{T} \varkappa(t) u_{c\to n}^t}{\sum_{t=1}^{T} \varkappa(t)} \end{aligned} \tag{17}$$

and $\hat{R}_{c\to n} = ([\sum_{t=1}^{T} \varkappa(t) R_{c\to n}^t]/[\sum_{t=1}^{T} \varkappa(t)])$.

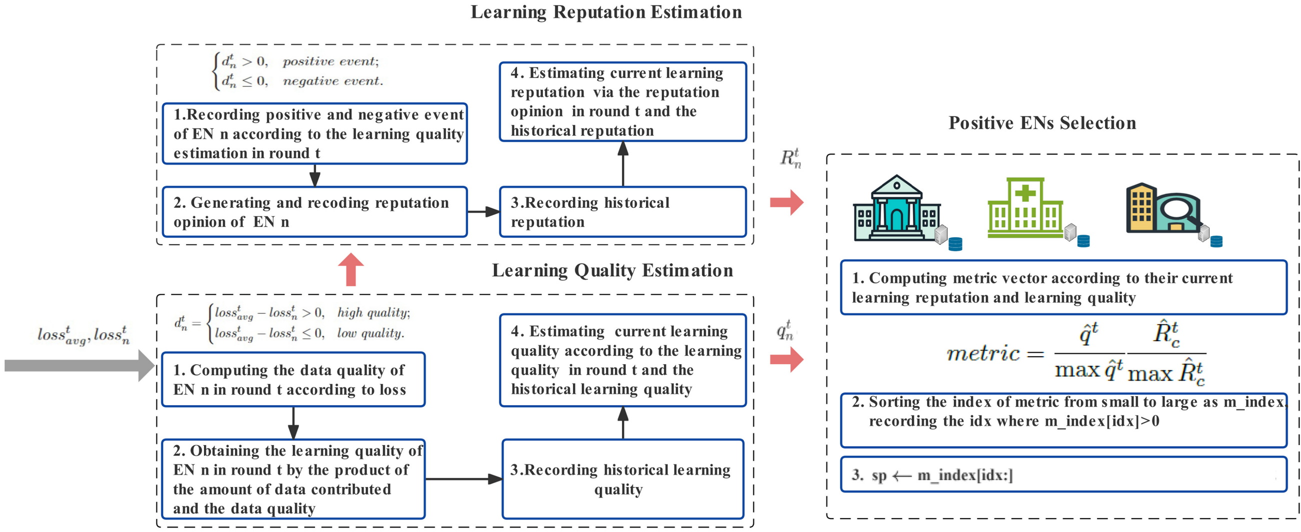

Fig. 3. Process of selecting positive ENs.

Algorithm 1 ENs Selected Algorithm by the Contribution Degree

Input: $\hat{q}^t$: the set of the learning quality of ENs in round t; $\hat{R}_c^t$: the set of the learning reputation of ENs in round t; $\hat{b}^t$: the set of the self-employment income of ENs in round t.

Output: sp: the set of ENs where $metric > 0$; mp_n: the normalize where $metric > 0$.

Compute the maximum value of all the learning quality and all the learning reputation in round t $\max \hat{q}^t$ and $\max \hat{R}_c^t$;

$metric \leftarrow \frac{\hat{q}^t}{\max \hat{q}^t}\frac{\hat{R}_c^t}{\max \hat{R}_c^t}$;

Sort index values of metric from small to large as m_index;

for $m \in m\_index$ **do**

  Record the index position of m idx;

  if metric[m] > 0 **then**

    break

  end if

end for

$sp \leftarrow m\_index[idx:]$;

$mp \leftarrow metric[sp]$;

Normalize mp to get mp_n;

return sp, mp_n.

C. Process of Selecting ENs

The ENs may intentionally or unintentionally update low-quality parameters to deteriorate the global model quality of the cross-silo FL. For better performance of the FL, we apply quality and reputation as the metrics to assess the contribution degree of an EN in one cross-silo FL, as illustrated in Fig. 3. To ensure that the ENs selected are high-quality and high-reputation, we develop a measurement schema that considers both the quality of the contributed resources and the reputation of the EN and the detailed process as shown in Algorithm 1. We first achieve a measurement schema according to the data contribution in the current round and the historical learning quality record and learning reputation record by a heuristic algorithm, and obtain the contribution degree *metric* of ENs, as shown in the following:

$$\text{metric} = \frac{\hat{q}^t}{\max \hat{q}^t}\frac{\hat{R}_c^t}{\max \hat{R}_c^t} \tag{18}$$

where $\hat{q}^t = \{\hat{q}_1^t, \hat{q}_2^t, \ldots, \hat{q}_N^t\}$ is the set of the learning quality of ENs in round t, $\max \hat{q}^t$ is the maximum value of all the learning qualities in round t; where $\hat{R}_c^t = \{\hat{R}_{c\to 1}^t, \hat{R}_{c\to 2}^t, \ldots, \hat{R}_{c\to N}^t\}$ denotes the set of the learning reputation of ENs in round t, $\max \hat{R}_c^t$ is the maximum value all the learning reputation in round t. If the metric is greater than 0, it indicates that the EN is reliable or the data contributed is helpful for improving the performance of model training, and vice versa. As shown in Fig. 3, We can finally select the ENs that have a positive utility on the model performance, denoted by sp. Namely, sp represents the set of ENs where metric > 0. To this end, mp measures the significance of selected model updates and mp_n denotes the normalized result of mp.

D. Behaviors Among the CP and ENs in Consecutive Time Slots

Due to the time-varying local data sets of ENs, they may repeatedly execute cross-silo FL processes. Consequently, the cloud-edge-based FL system operates as a time slot with a time span divided into T consecutive slots with equal span. And each time slot with equal duration means one cross-silo FL process. We describe the behaviors among ENs in consecutive time slots as demonstrated in Fig. 2. Strictly speaking, we first present the interactions of ENs in one slot and then outline the interactions in the infinite time slots.

We first introduce the interactions among the CP and ENs in one slot. As illustrated in Fig. 2, one slot represents one cross-silo FL process which performs one global aggregation and k local iterations. In each time slot, the CP first gives rewards to ENs, then the EN n decides if it should participate in the cross-silo FL or not, and how to choose the degree

TABLE II
PROFIT MATRIX OF BOTH CP AND ENS

Cloud Platform \ EN	P	Q
Y	$\mathcal{U}_c(Y,P), \mathcal{U}_n(Y,P)$	$\mathcal{U}_c(Y,Q), \mathcal{U}_n(Y,Q)$
N	$\mathcal{U}_c(N,P), \mathcal{U}_n(N,P)$	$\mathcal{U}_c(N,Q), \mathcal{U}_n(N,Q)$

of data contribution to maximize their own profit formulated in (9) while the degrees of other ENs' data contribution are determined. When ENs are short-sighted and only focus on their profits in the current slot, we build their interactions as a selfish stage game in one cross-silo FL (SSFL), which is presented in Section IV-E in detail.

Then, we introduce the interactions among the CP and ENs in the infinite time span which is split into T consecutive slots with equal duration. In [33], [45], [55], and [56] each time slot can be a week for hospitals or a month for an insurance company. In the infinite time span, the data sets of ENs change over time. In this article, $\mathcal{X}_n^{t-1}$ and $\mathcal{X}_n^t$ are the local data sets of EN n in slot $t-1$ and slot t, respectively, where $\mathcal{X}_n^{t-1} \neq \mathcal{X}_n^t$ and $x_n^{t-1} < x_n^t$ $\forall n \in \mathcal{N}$. To update the global models with data sets changed with time, ENs repeat the cross-silo FL process. In time slot t, ENs first download the global model ω^t and then execute the next cross-silo FL according to their current local data sets to train the global model ω^{t+1}. In the infinite time span, the CP provides appropriate rewards for ENs to motivate ENs to join the model training from a long-term perspective to maximize its profit. Next, ENs choose the degree of data contribution for the task in cross-silo FL to maximize their own long-term profit. We model the long-term interactions of the CP and ENs as an efficiency wage model of an infinitely repeated game based on SSFL, which is described in Section V-A in detail.

E. Stage Game Analysis

As we all know, both the CP and ENs are rational individuals. Simply put, the CP should not only consider reducing cost but also motivate ENs to work hard with appropriately high rewards depending on their contribution so as to improve model accuracy; after that, the EN decides whether to accept the model training task and whether to work hard according to the reward given by the CP. Between them is a game process. Next, we analyze the behaviors among the CP and ENs in one cross-silo FL process.

1) Selfish Stage Game in One Cross-Silo FL: In one cross-silo FL process, both CP and ENs selfishly choose their strategies to maximize their own profits. We model the selfish behaviors of the CP and ENs in the one-time slot as follows.

Game 1 (SSFL):

1) *Players:* CP and the set of ENs $\mathcal{N}$.
2) *Strategies:* CP has two strategies, namely, the strategy Y that provides the ENs the optimal reward r^* and the strategy N that does not provide the optimal reward r^*, and the strategy set is $\mathcal{S}_c = (Y, N)$.Each EN has two strategies, specifically, P means that the EN n works hard on the model training task, and Q means that the EN n slacks off in the model training task, and the strategy set of ENs is $\mathcal{S}_n = (P, Q)$.
3) *Objectives:* CP aims to maximize its profit $\mathcal{U}_c(\mathcal{S}_c, \mathcal{S}_n)$, and the EN n aims to maximize its profit $\mathcal{U}_n(\mathcal{S}_c, \mathcal{S}_n)$.

SSFL can be defined by $G = (\mathcal{P}, \mathcal{S}_m, \mathcal{U}_m)$, $m \in \mathcal{P}$, where $\mathcal{P} = \{c, n\}$, $c = CP, n \in \mathcal{N}$, is a set of players in Game 1, $\mathcal{S}_m$ is the set of player m's strategies, and $\mathcal{U}_m$ is the profit of player m. In SSFL, the strategy of the ENs is whether to accept the model training task and whether to work hard for the task, while the strategy of the CP is whether to provide the optimal reward.

In one cross-silo FL, the profit matrix of both the CP and EN are shown in Table II. The calculation formulas of CP and EN are as follows:

$$\begin{cases} \mathcal{U}_c(Y,P) = y\big(\mathcal{X}_n^t\big) - r\big(\mathcal{X}_n^t\big) \\ \mathcal{U}_c(Y,Q) = p * y\big(\mathcal{X}_n^t\big) - r\big(\mathcal{X}_n^t\big) \\ \mathcal{U}_c(N,P) = 0 \\ \mathcal{U}_c(N,Q) = 0 \end{cases} \tag{19}$$
$$\text{s.t.} \quad r\big(\mathcal{X}_n^t\big) \geq 0, y\big(\mathcal{X}_n^t\big) \geq 0, p > 0, n \in \mathcal{N}'$$

where $\mathcal{U}_c(\mathcal{S}_c, \mathcal{S}_n)$ represents the profit of the CP in the round t, $y(\mathcal{X}_n^t)$ is the utilities that the trained model generated by ENs bring in round t, and $r(\mathcal{X}_n^t)$ is the reward that the CP pays to the EN n according to their data contributed $\mathcal{X}_n^t$ in round t

$$\begin{cases} \mathcal{U}_n(Y,P) = r\big(\mathcal{X}_n^t\big) - C\big(\mathcal{X}_n^t\big) \\ \mathcal{U}_n(Y,Q) = r\big(\mathcal{X}_n^t\big) \\ \mathcal{U}_n(N,P) = r^0\big(\mathcal{X}_n^t\big) - C\big(\mathcal{X}_n^t\big) \\ \mathcal{U}_n(N,Q) = r^0\big(\mathcal{X}_n^t\big) \end{cases} \tag{20}$$
$$\text{s.t.} \quad r^0\big(\mathcal{X}_n^t\big) \geq 0, r\big(\mathcal{X}_n^t\big) \geq 0, C\big(\mathcal{X}_n^t\big) \geq 0, n \in \mathcal{N}$$

where $\mathcal{U}_n(\mathcal{S}_c, \mathcal{S}_n)$ represents the profit for EN n which proceeds in round t, $C(\mathcal{X}_n^t)$ is the incurred cost of EN n in round t, which can be obtained by (8).

2) Nash Equilibrium Solution of SSFL: Following we establish the Nash equilibrium (NE) of SSFL and then draw the unique NE:

Definition 1 (Nash Equilibrium): When $\mathcal{S}_c$ and $\mathcal{S}_n$ are fixed, if $\mathcal{S}_c^* \in \{Y, N\}$ satisfies $\mathcal{U}(\mathcal{S}_c^*) \geq \mathcal{U}(\mathcal{S}_c)$ and $\mathcal{S}_n^* \in \{P, Q\}$ satisfies $\mathcal{U}(\mathcal{S}_n^*) \geq \mathcal{U}(\mathcal{S}_n)$, $(\mathcal{S}_c^*, \mathcal{S}_n^*)$ is the NE in SSFL.

Backward induction can be utilized to address the NE in Game 1. As shown in Fig. 4, the EN n chooses Q strategy to obtain more profit when the CP does not provide the optimal reward. When the CP provides the optimal reward, the EN n

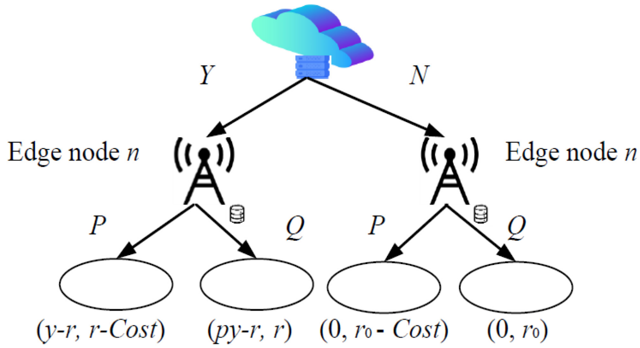

Fig. 4. Profit of both CP and ENs in one cross-silo FL.

decides whether to work hard or not. That is, the EN n chooses their degree of data contribution (P, Q). Under the condition that the CP must provide the reward r, the EN n will choose the Q strategy. At the same time, the profit $p * y - r$ that the EN n brings to the CP is most likely negative, so the reward r to the EN paid by the CP is low $(r < r_0)$. The CP will choose the N strategy to benefit more while the EN n chooses the self-employment strategy Q. Consequently, the NE strategy of the Game 1 is (N, Q). Namely, without proper incentives, the game outcome in one cross-silo FL process is (N, Q), which is called the MEC dilemma in this article.

V. Incentive Mechanism Design

In this section, we formulate an incentive mechanism to achieve the long-time aims of the CP and ENs in MEC. As described above, both the CP and ENs are rational individuals, which results in the MEC dilemma. Since neither the CP nor ENs knows at which stage the cross-silo FL terminates, the game processes are identical with repeated games without the final stage [57]. To solve the MEC dilemma, we introduce the efficiency wage model of the infinitely repeated game to realize a cooperative incentive model for the CP and ENs over a long time. In this article, the dynamic repeated game is applied to model the interactions between the CP and ENs about contribution and rewards. We first analyze the behaviors among the CP and ENs where the infinite time is split into many time slots. Then, we introduce the basic process of the proposed incentive mechanism. Finally, we evolve the dynamic evolution process of CP and ENs reaching equilibrium by establishing the replicator dynamic equation.

A. Infinitely Repeated Game Analysis

In this section, we explore the repeated game, i.e., the long-term behaviors among CP and ENs in the infinite time span with SSFL as illustrated in Fig. 5. As described above, the game processes are identical to an infinitely dynamic repeated game that will not end. K local iterations and one global aggregation are executed in one time slot. In the infinite time span, the CP first chooses whether to give the optimal reward r^* to maximize its long-term total profit, then the EN chooses whether to accept the model training task and chooses its degree of data contribution if accepts the task according to the reward provided by CP to maximize its long-term total

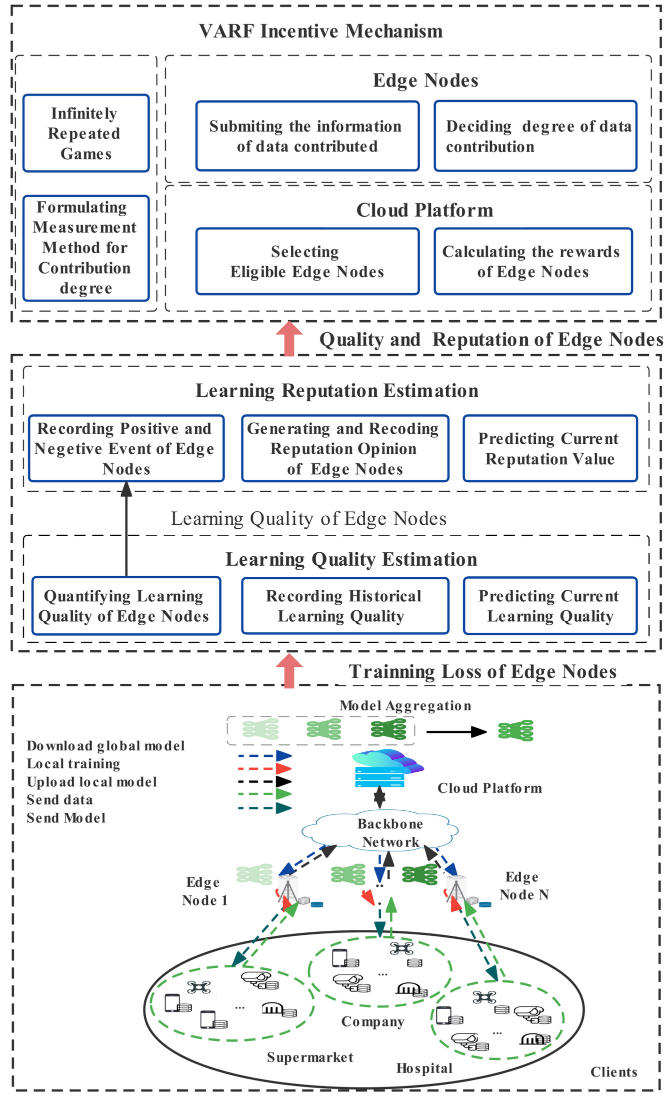

Fig. 5. Framework of the VARF incentive mechanism.

profit. It moves on to the next round after one cross-silo FL is completed. The learning quality and learning reputation are estimated according to historical records and transferred to the CP. Then, the CP and ENs reselect their own strategies based on each other's behavior. In short, the repeated game is a staged game in which the same stage game is repeatedly executed over and over.

Definition 2 (Infinitely Repeated Game): Given a staged game G, the process of infinitely repeated G game is called repeated game $G(\infty, \delta)$, where $\delta(\delta \in [0, 1))$ is the discount factor of players [58] and it describes the importance of future gains in the current stage. Moreover, for any t, before playing the t-stage game, all players can see the histories of the previous $(t-1)$ stage game

$$G(\infty, \delta) = (\mathcal{P}, \mathcal{S}_m, \mathcal{U}_m, T, \delta), m \in \mathcal{P}. \quad (21)$$

In the repeated game $G(\infty, \delta)$, $\mathcal{S}_m^t$ $(\mathcal{S}_m^t \in \mathcal{S}_m)$ denotes the strategy selected in stage t by the player m, and $\mathcal{S}^t = (\mathcal{S}_c^t, \mathcal{S}_1^t, \mathcal{S}_2^t, \mathcal{S}_3^t, \ldots, \mathcal{S}_n^t)$ is described as the strategy combination. The benefit of each player in $G(\infty, \delta)$ is equal to the present value of the benefit of each stage. Supposed that $\mathcal{U}_i(\mathcal{S}^t)$ is the benefit of the player i in stage t, the present value of the total profit is calculated by

$$\pi_i = \pi_1 + \pi_2\delta + \pi_3\delta^2 + \cdots = \sum_{t=1}^{\infty} \delta^{t-1}\mathcal{U}_i\left(\mathcal{S}^t\right). \tag{22}$$

If the constant $\bar{\pi}$, as the long-term profit of each stage of an infinitely repeated game, can yield the same discounted value as the infinite profit sequence $\pi_1, \pi_2, \ldots$ of a certain player in the infinitely repeated game, then $\bar{\pi}$ is called the average return of $\pi_1, \pi_2, \ldots$ The present value can be computed by (23)

$$\pi_i = \bar{\pi} + \bar{\pi}\delta + \bar{\pi}\delta^2 + \cdots = \frac{\bar{\pi}}{1-\delta}. \tag{23}$$

According to (22) and (23), the average return is calculated as follows:

$$\bar{\pi} = (1-\delta)\sum_{t=1}^{\infty} \delta^{t-1}\mathcal{U}_i\left(\mathcal{S}^t\right). \tag{24}$$

Thus, the present value of the CP is

$$\mathcal{U}_c\left(\mathcal{S}_c^{\infty}, \mathcal{S}_n^{\infty}\right) = \frac{\bar{\pi}_c}{1-\delta} \tag{25}$$

where $\bar{\pi}_c$ is the average return of the CP.

Then, for the EN n, its present value is

$$\mathcal{U}_n\left(\mathcal{S}_c^{\infty}, \mathcal{S}_n^{\infty}\right) = \frac{\bar{\pi}_n}{1-\delta} \tag{26}$$

where $\bar{\pi}_n$ is the average return of the EN n.

The repeated game is modeled by the behaviors between CP and ENs over an infinite time span as below.

Game 2 (Repeated Game in Infinite cross-silo FL, RGFL).

1) *Players:* CP and the set of ENs $\mathcal{N}$.
2) *Strategies:* CP has two strategies in stage t, namely, the strategy Y that gives ENs the optimal reward r^*, the strategy N that does not give the optimal reward r^*, and the strategy set is $\mathcal{S}_c = (Y, N)$.Each EN has three strategies in stage t, specifically, P means that the EN n accepts the task and works hard, and Q means that the EN n accepts the task but slacks off, and the strategy set of ENs is $\mathcal{S}_n = (P, Q)$.
3) *Histories:* The strategy profile history of CP $\mathcal{S}_c^t$ till the stage t and the strategy profile history of each EN $\mathcal{S}_n^t$ till the stage t.
4) *Objectives:* CP targets to maximize its long-term profit $\mathcal{U}_c(\mathcal{S}_c^{\infty}, \mathcal{S}_n^{\infty})$, and EN n targets to maximize its long-term profit $\mathcal{U}_n(\mathcal{S}_c^{\infty}, \mathcal{S}_n^{\infty})$.

Theorem 1 (Folk Theorem of Infinite Repeated Game): $G(\infty, \delta)$ is an infinitely repeated game with G as a stage game, and its benefit of NE is denoted as $(a_1, a_2, \ldots, a_n)$, and its feasible benefit is denoted as $(b_1, b_2, \ldots, b_n)$. If there is $b_i > a_i$ for any player i and δ is close enough to 1, then there must be an subgame perfect Nash equilibrium (SPNE) path for infinite repeated games $G(\infty, \delta)$, which can realize the average return of all players in the infinitely repeated games.

According to Theorem 1, any realizable and individually rational strategies of the CP and ENs can be an SPNE. As the success of cross-silo FL is determined by the long-term active and credible contribution of ENs, we typify the SPNE that can maximize the local data contributed by high-quality and high-reputation ENs to increase the accuracy of the global model.

Algorithm 2 VARF Incentive Mechanism Algorithm for CP

Input: the cross-silo FL task ϕ, the set of ENs $\mathcal{N}$, the set of the learning quality estimation of all ENs $\hat{q}$, the set of the learning reputation estimation of all ENs $\hat{R}_c$

Output: the optimal reward $r^*(\mathcal{X}_n^t)$, the degree of contributing data m_n^t;

for $t \leftarrow 1$ to ∞ **do**
  Publish information of the task ϕ;
  for $n \in \mathcal{N}$ **do**
    Compute m_n^t depending on $\hat{q}_n^t$ and $\hat{R}_{c\to n}$ by Eq.(18);
    Check the history information of the EN n;
    if $m_n^t > 0$ and $(m_n^0, m_n^1, \ldots, m_n^{t-1}) > 0$ **then**
      Hire the EN n;
      Compute the optimal reward $r^*(\mathcal{X}_n^t)$ by Eq.(29);
    else
      Refuse to hire the EN n;
    end if
  end for
  Record and return $r^*(\mathcal{X}_n^t)$ and m_n^t;
end for

B. Basic Process of the Proposed Incentive Mechanism (VARF)

In this article, the cross-silo FL processes with the incentive mechanism can be built as an infinitely repeated game as follows. First, the CP chooses whether to give the optimal reward r^* to maximize its long-term total profit. Then, the EN n chooses whether to accept the model training task and chooses its degree of data contribution if accepts the task according to the reward provided by the CP to maximize its long-term total profit.

The CP considers whether to hire the EN for the cross-silo FL according to the learning quality quantification and the learning reputation estimation based on the current and historical data. If the CP determines to hire the ENs, it will compute and prepay the matched payment to the ENs according to (29). Then, ENs who are hired contribute their local model updates to the CP.

The trigger strategies of the CP and ENs in the tth stage are as follows.

CP: If the metric is greater than 0 in the previous $t-1$ stages and it is greater than 0 in the t stage, the EN will be hired and prepaid the corresponding optimal reward r^* calculated by (29) in stage t; otherwise, it will not be hired. The detailed process is described in Algorithm 2.

EN: The optimal reward r^* not only compensates for the self-employment income r^0 and the negative utility Cost but also there are additional benefits. If the CP does not provide the optimal reward r^*, ENs will refuse the task. ENs will

Algorithm 3 VARF Incentive Mechanism Algorithm for ENs

Input: the subset $\mathcal{X}_n^t$ of the local dataset for model training, the reward $r(\mathcal{X}_n^t)$ paid by CP in round t; the self-employment income $r^0(\mathcal{X}_n^t)$ in round t;

Output: the strategy of EN n;

for $t \leftarrow 1$ to ∞ **do**
  if $r_n > r_n^0$ **then**
    Compute $\hat{q}_n^t$ by Eq.(14) and $\hat{R}_{c\to n}$ by Eq.(17);
    Submit $\hat{q}_n^t$ and $\hat{R}_{c\to n}$ to the CP;
    if the EN is hired **then**
      Compute $C(\mathcal{X}_n^t)$ by Eq.(8);
      Check history information of the EN n;
      if $\sum_1^{t-1} r(\mathcal{X}_n^t) > \sum_1^{t-1}(r^0(\mathcal{X}_n^t) + C(\mathcal{X}_n^t))$ **then**
        Choose to work hard;
        Submit model update to the CP;
      end if
    else
      Wait for next task;
    end if
  else
    Choose to slack off;
  end if
end for

accept the task and work hard when the optimal reward r^* is paid in the previous $t-1$ stages and in the t stage; Otherwise, ENs would accept the task but slack off. The detailed process is expressed in Algorithm 3.

The trigger strategy makes both CP and ENs threatened and the discount factor δ controls a credible threat. The threat of the CP is that once the EN slacks off, it will not employ the EN in the next stage. The threat of ENs is that they will refuse the task if the reward is lower than r^0 and they will not work hard if the reward r is less than r^0 plus negative utility Cost. Both threats make no CP and ENs are reluctant to violate the trigger strategy lonely, which meets the principle of NE.

Definition 3 (Subgame Perfect Nash Equilibrium): In a dynamic game, a strategy profile $S^* = (S_c^*, S_1^*, S_2^*, \ldots, S_n^*)$ composed of the strategies of CP and ENs is an SPNE if it represents a NE of every subgame of the original game.

To guarantee the trigger strategy formulated is credible, the combination of the trigger strategies of the CP and ENs is required to be an SPNE. Therefore, finding the SPNE of the RGFL is crucial in the infinitely repeated game.

Definition 4 (One-Shot Deviation Principle [59]): In an infinitely repeated game, a strategy combination is an SPNE when and only when each player passes the one-shot deviation test at each stage.

In short, if no player can gain more profits by deviating from the strategy of the original game, the strategies chosen by all players form an SPNE. Namely, no player can benefit by deviating from its original strategy at a certain stage and then returning to the original strategy.

Next, we prove the trigger strategy of VARF is an SPNE. And one-shot deviation principle is used to characterize players' behaviors at the SPNE. Therefore, the long-term discounted total profit π_h^e of the EN n works hard should be higher than the long-term discounted total profit π_s^e of the EN n slacks off when performing the task in the cross-silo FL. The long-term discounted total profit π_h^r of optimal reward by the CP should be higher than the long-term discounted total profit π_s^r that does not give optimal reward to motivate the EN to work hard to perform tasks with high rewards. Therefore, the following is an analysis of the effectiveness of the incentive mechanism.

EN: For ENs, we assume the long-term profits when ENs do not violate the trigger strategy is π_h^e as

$$\begin{aligned}\pi_h^e &= \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \delta \times \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big)\\ &\quad + \delta^2 \times \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \cdots + \delta^{t-1}\\ &\quad \times \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \cdots\\ &= \sum_{T=1}^{\infty} \delta^{T-1}\big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big)\\ &= \frac{(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n))}{1-\delta}. \end{aligned} \tag{27}$$

Assuming that the ENs violate the trigger strategy in stage t, p is the probability of the EN improving the accuracy of model training and obtaining a high reward r_* when the EN slacks off in the cross-silo FL. In this case, the total profit is π_s^e as

$$\begin{aligned}\pi_s^e &= \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \delta \times \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \cdots\\ &\quad + \delta^{t-2} \times \big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \delta^{t-1} \times r^*(\mathcal{X}_n)\\ &\quad + \delta^{t} \times \big(p \times r^*(\mathcal{X}_n) + (1-p) \times r_0(\mathcal{X}_n)\big)\\ &\quad + \delta^{t+1} \times \big(p \times r^*(\mathcal{X}_n) + (1-p) \times r_0(\mathcal{X}_n)\big) + \cdots\\ &= \sum_{T=1}^{t-1} \delta^{T-1}\big(r^*(\mathcal{X}_n) - \mathrm{Cost}(\mathcal{X}_n)\big) + \delta^{t-1} \times r^*(\mathcal{X}_n)\\ &\quad + \sum_{T=t+1}^{\infty} \delta^{T-1}\big(p \times r^*(\mathcal{X}_n) + (1-p) \times r_0(\mathcal{X}_n)\big)\\ &= \frac{(1-\delta)r^*(\mathcal{X}_n) + (1-p)\delta r_0(\mathcal{X}_n)}{(1-p\delta)(1-\delta)}. \end{aligned} \tag{28}$$

If the total profit of the EN satisfies $\pi_h^e > \pi_s^e$, the EN n will not violate its trigger strategy. Therefore, ENs choose not to violate the trigger strategy if the optimal reward r^* satisfies the following conditions:

$$r_n^* > r_0(\mathcal{X}_n) + \mathrm{Cost}(\mathcal{X}_n) + \frac{1-\delta}{\delta(1-p)} \times \mathrm{Cost}(\mathcal{X}_n). \tag{29}$$

CP: If the CP always provides the optimal reward r^* satisfying (29) and accordingly ENs choose to work hard, the total profit of the CP in the infinite repeated game is π_h^c as shown in (30). The total profit π_s^c of the CP is 0 if the CP chooses not to hire ENs

$$\begin{aligned}\pi_h^c &= \big(y(\mathcal{X}_n) - r^*(\mathcal{X}_n)\big) + \delta \times \big(y(\mathcal{X}_n) - r^*(\mathcal{X}_n)\big) + \delta^2\\ &\quad \times \big(y(\mathcal{X}_n) - r^*(\mathcal{X}_n)\big) + \cdots + \delta^{t-1}\\ &\quad \times \big(y(\mathcal{X}_n) - r^*(\mathcal{X}_n)\big) + \cdots\\ &= \frac{1}{1-\delta} \times \big(y(\mathcal{X}_n) - r^*(\mathcal{X}_n)\big). \end{aligned} \tag{30}$$

Therefore, the CP will follow its trigger strategy in the infinitely repeated game if it satisfies the condition as follows:

$$\frac{(y(\mathcal{X}_n) - r^*(\mathcal{X}_n))}{1-\delta} > 0. \tag{31}$$

Simply put, CP will always provide the optimal reward r^* if $(y(\mathcal{X}_n) - r^*(\mathcal{X}_n)) > 0$.

Thus, the strategy profile of CP and ENs is the SPNE of RGFL under the constraint (29) and (31). According to (29) and (31), we can obtain the result as follows:

$$\begin{aligned} y(\mathcal{X}_n) > r^*(\mathcal{X}_n) > r_0(\mathcal{X}_n) + \text{Cost}(\mathcal{X}_n) \\ + \frac{1-\delta}{\delta(1-p)} \times \text{Cost}(\mathcal{X}_n). \end{aligned} \tag{32}$$

Therefore, the rewards of ENs can be calculated by (32), and m_n^t represents the measurement of the learning quality and learning reputation calculated by (18)

$$\begin{aligned} r^*(\mathcal{X}_n) = r_0(\mathcal{X}_n) + \text{Cost}(\mathcal{X}_n) + \frac{1-\delta}{\delta(1-p)} \\ \times \text{Cost}(\mathcal{X}_n) \times \left(1 + m_n^t\right). \end{aligned} \tag{33}$$

According to (27) and (28), the discount factor δ in this article is given by

$$\begin{aligned} \delta > \delta_n^* &= \frac{(r^*(\mathcal{X}_n) - \text{Cost}(\mathcal{X}_n)) - r^*(\mathcal{X}_n)}{(p \times r^*(\mathcal{X}_n) + (1-p) \times r_0(\mathcal{X}_n)) - r^*(\mathcal{X}_n)} \\ &= \frac{\text{Cost}(\mathcal{X}_n)}{\text{Cost}(\mathcal{X}_n) + (1-p)(y(\mathcal{X}_n) - r_0(\mathcal{X}_n) - \text{Cost}(\mathcal{X}_n))} \end{aligned} \tag{34}$$

where δ_n^* is the threshold discounted factor of EN n to perform the trigger strategy $S*$ in Game 2. If the CP and ENs are patient enough, that is, the discounted factor $\delta_n \geq \delta_n^*$, the CP and ENs will play the trigger strategy at the SPNE.

C. Evolution Analysis

The long-term profit of the repeated game in infinite cross-silo FL is static equilibrium. Namely, the repeated game cannot express the dynamic evolution process of players reaching equilibrium. To show the process of players from SSFL to RGFL under the incentive and threat of trigger strategy, we explore the relations between the two from the perspective of both CP and ENs and prove the stability of the trigger strategy in RGFL via establishing the replicator dynamic equation between CP and ENs. Suppose that x and $1-x$ are the proportions of the CP that choose Y strategy and N strategy, respectively, while z and $1-z$ are the proportions of the ENs that choose Y strategy and Y strategy separately.

1) Strategy Stability Analysis of the Cloud Platform: According to the profit matrix shown in Table II and the profit result calculated by the discount calculation method above, we know that $\mathcal{U}_c(Y,P) = ([y - r^*]/[1-\delta])$, $\mathcal{U}_n(Y,P) = ([r^* - \text{Cost}]/[1-\delta])$, $\mathcal{U}_c(Y,Q) = ([p * y - r^*]/[1-\delta])$, $\mathcal{U}_n(Y,Q) = ([r^*]/[1-\delta])$, $\mathcal{U}_c(N,P) = 0$, $\mathcal{U}_n(N,P) = ([r^0 - \text{Cost}]/[1-\delta])$, $\mathcal{U}_c(N,Q) = 0$, and $\mathcal{U}_n(N,Q) = ([r^0]/[1-\delta])$.

The expected profit when the CP gives the optimal reward to the EN is

$$\begin{aligned} e_1 &= z\mathcal{U}_c(Y,P) + (1-z)\mathcal{U}_c(Y,Q) \\ &= z\frac{y-r^*}{1-\delta} + (1-z)\frac{p*y-r^*}{1-\delta}. \end{aligned} \tag{35}$$

The expected profit when the CP does not give the optimal reward to the EN is

$$e_2 = z\mathcal{U}_c(N,P) + (1-z)\mathcal{U}_c(N,Q) = 0. \tag{36}$$

Therefore, the average revenue of the CP population is as follows:

$$\bar{e_c} = x\left[z\frac{y-r^*}{1-\delta} + (1-z)\frac{p*y-r^*}{1-\delta}\right]. \tag{37}$$

The dynamic equation of the replicators of CP is

$$\frac{dx}{dt} = x(e_1 - \bar{e_c}) = x(1-x)\left[z\frac{y-r^*}{1-\delta} + (1-z)\frac{p*y-r^*}{1-\delta}\right]. \tag{38}$$

2) Strategy Stability Analysis of Edge Nodes: The expected profits of ENs actively perform tasks are

$$\begin{aligned} e_3 &= x\mathcal{U}_n(Y,P) + (1-x)\mathcal{U}_n(Y,Q) \\ &= x\frac{r^* - \text{Cost}}{1-\delta} + (1-x)\frac{r^*}{1-\delta} \\ &= \frac{r^* - \text{xCost}}{1-\delta}. \end{aligned} \tag{39}$$

The expected profit of ENs passively perform tasks is

$$\begin{aligned} e_4 &= x\mathcal{U}_n(N,P) + (1-x)\mathcal{U}_n(N,Q) \\ &= x\frac{r^0 - \text{Cost}}{1-\delta} + (1-x)\frac{r^0}{1-\delta} \\ &= \frac{r^0 - \text{xCost}}{1-\delta}. \end{aligned} \tag{40}$$

Therefore, the average revenue of the ENs population is as follows:

$$\bar{e_n} = z\left(\frac{r^* - \text{xCost}}{1-\delta}\right) + (1-z)\left(\frac{r^0 - \text{xCost}}{1-\delta}\right). \tag{41}$$

The dynamic equation of the replicators of ENs is

$$\begin{aligned} \frac{dz}{dt} &= z(e_3 - \bar{e_n}) \\ &= z\left[\frac{r^* - \text{xCost}}{1-\delta} - z\left(\frac{r^* - \text{xCost}}{1-\delta}\right) \right. \\ &\left. \quad - (1-z)\left(\frac{r^0 - \text{xCost}}{1-\delta}\right)\right]. \end{aligned} \tag{42}$$

3) Trigger Strategy TS Stability Analysis: On the basis of evolutionary game theory, a dynamic replication system is built for CP and ENs based on (38) and (42). The corresponding Jacobian matrix can be expressed as shown in (43), at the bottom of the next page.

According to (43), we can observe that the trace of the matrix $trJ < 0$ and the Jacobian determinant $\det J > 0$ when $x = 1$ and $z = 1$. That is, ($x = 1$ and $z = 1$) is an equilibrium point, locally stable. At the same time, it is the evolutionarily stable strategy of the dynamic replication system. This

TABLE III
COMPARISON BETWEEN EXISTING RESEARCH WITH PROPOSED SCHEME

Research	Quality	Reputation	Long-term Participation	Stability Analysis
Reliable FL based on Blockchain [24]	-	✓	-	-
Quality-Aware FL [25]	✓	-	-	-
MSRG Incentive Mechanism [39]	✓	-	✓	-
Novel Incentive Mechanism [40]	-	-	✓	✓
Proposed Scheme	✓	✓	✓	✓

reveals that if the CP chooses the strategy that does not give the optimal reward, it will be punished by ENs; likewise, if the EN chooses the strategy of passively and inefficiently performing the task, then it will be punished by the CP. As a result of being punished, the expected profit of both parties in the game is less than the profit when the cooperation is selected. In the infinite time horizon, all players will eventually choose the trigger strategy through Continuous learning and adjustment. Finally, both the CP and ENs will build a stable cooperative relationship. To summarize, the evolutionarily stable strategy of the dynamic replication system can resist the invasion of other noncooperative strategies.

Table III summarizes the comparison between existing research and the proposed scheme for quality, reputation, long-term participation, and stability analysis, see the related references for detail.

VI. EXPERIMENTAL EVALUATION

In this section, we evaluate the performance of VARF for cross-silo FL in the cloud-edge system by simulation. We evaluate the proposed incentive mechanism by comparing it with several benchmarks [all trained mechanism (ATM), Bid price first mechanism (BFM), and Positive effects mechanism (PEM)] under different scenarios: 1) Clean data sets (CDs) and 2) Noisy label data sets (NLDs); c) Error label data sets (ELDs).

A. Experimental Settings

The simulation environment in this article is a server with RTX 8000 GPU (48G). To compare with previous methods, we perform simulations based on the data set used in [5], [24], [25], [33], and [45], and the data set has been widely used by many existing FL works. We simulate the algorithms by setting $k = 5$, $\varsigma = 0.03$, and $\varphi = 0.97$ after testing. The other detailed parameter settings are shown in Table IV, in which N_{EN} means the number of ENs having a noisy training data set. And N_{level} represents the noise level of the noisy ENs in the range [0, 1], a higher value of N_{level} indicates a higher noise level of the ENs. This is an important consideration when designing MEC systems because the noise level can affect the accuracy and reliability of any ML. Therefore, it is important to develop methods to consider the specific application or use case and the tolerable level of noise in the data. Here, we set N_{level} with $\{0, 0.3, 0.5, 1\}$ for different scenarios to demonstrate the tolerance of our scheme to noise, respectively.

TABLE IV
PARAMETERS SETTINGS

Settings	$\mathcal{N}$	N_{EN}	N_{level}	G
1	20	6	0	10
2	20	10	0	10
3	50	15	0	10
4	50	25	0	10
5	100	30	[0, 1]	10
6	100	50	[0, 1]	10
7	100	30	[0, 1]	20
8	100	50	[0, 1]	20
9	100	30	[0, 1]	30
10	100	50	[0, 1]	30

1) Non-IID Setting: Simulating the non-IID setting is important for evaluating the robustness and effectiveness of FL algorithms in real-world scenarios. Here, we simulate the non-IID setting by data partitioning.

1) *Determine the Size of Data Set and the Number of Shards:* Suppose the size of the data set is $\mathbb{N}$ and the number of shards is M, then the size of shards is $\mathbb{N}/M$.
2) *Partition the Data:* Allocate subsets according to the number of ENs N and the ratio rt ([6:5:4:3:2]). The ratio

$$J = \begin{bmatrix} (1-2x)\left[z\frac{y-r^*}{1-\delta} + (1-z)\frac{p*y-r^*}{1-\delta}\right] & x(1-x)\left[\frac{y-r^*}{1-\delta} + \frac{p*y-r^*}{1-\delta}\right] \\ 0 & \left\{(1-2z)\frac{r^*-x*\text{Cost}}{1-\delta} + \frac{r^0-x*\text{Cost}}{1-\delta}\right. \end{bmatrix} \tag{43}$$

of all subsets l and the number of shards in subsets can be computed by

$$\begin{aligned} a &= \lfloor N/\text{len}(rt) \rfloor \\ b &= N\%\text{len}(rt) \\ l &= rt * a + \left[rt[i] \text{ for } i \text{ in range}(b)\right] \\ c &= M * \frac{\left[l[j] \text{ for } j \text{ in range}(N)\right]}{\text{sum}(l)}. \end{aligned} \tag{44}$$

3) *Distribute Subsets to ENs:* Finally, we distribute each subset to a different EN to perform training.

2) Data Set Setting: All incentive mechanisms are trained with the same data sets and each learning model is trained with N nodes under three different scenarios.

1) *Clean Data Sets:* All ENs have unchanged training data sets to train the model normally.
2) *Noisy Label Data Sets:* Among N ENs, the training data sets of $n_1\%$ of ENs are clean, but in the training data sets of other nodes, $n_2\%$ of data samples are incorrectly labeled, i.e., labels are generated by $\text{label}(d) = (\text{label}(d) + 1)\ \%10$.
3) *Error Label Data Sets:* Among N ENs, the training data sets of $n_3\%$ of ENs are clean, but the training data sets of other $n_4\%$ of ENs are incorrectly labeled.

B. Benchmarks

To compare the performance of VARF with other schemes, we design reasonable benchmarks as follows.

1) *All Trained Mechanism:* All the ENs are selected for model training disregarding the data quality and the truthfulness of ENs.
2) *Bid Price First Mechanism:* Under the condition that the truthfulness of ENs is not guaranteed, ENs with the lowest bid are preferentially selected.
3) *Positive Effects Mechanism:* ENs with positive effects are selected for model training, regardless of bid, data quality, truthfulness, and reputation of ENs.

C. Impact of Different Incentive Mechanisms

In this article, we perform four different learning tasks. Specifically, the MLP and CNN models are trained with MNIST and CIFAR, respectively. To conduct a fair comparison, VARF and other all benchmarks adopt the Federated Averaging algorithm for model aggregation to complete four learning tasks. Besides, all incentive mechanisms run four learning tasks to compare the performance of each learning model with 100 ENs and four learning tasks in each iteration. There are four different scenarios among the 100 ENs: 1) 100 ENs are clean; 2) the training data sets of 70 ENs are clean but the other 30 ENs have noisy training data sets with a 30% noisy level; 3) the training data sets of 70 ENs are clean but the other 30 ENs have noisy training data sets with a 50% noisy level; and 4) the data sets of 70 ENs are clean but the data sets of the other 30 ENs are error. Each incentive mechanism runs 30 iterations, then the average accuracy and loss results of the learning model in the cross-silo FL are plotted in Figs. 6 and 7, respectively.

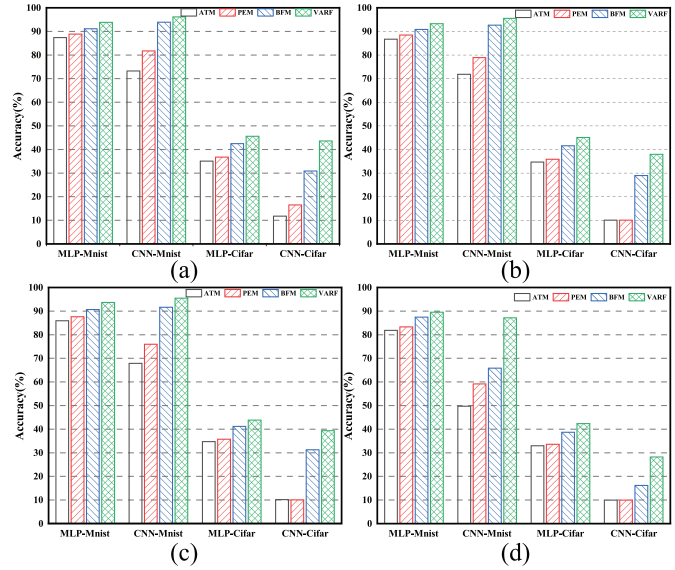

Fig. 6. Average model accuracy of MLP-Mnist (MM), CNN-Mnist (CM), MLP-Cifar (MC), CNN-Cifar (CC) with different incentive mechanisms under different scenarios: (a) CD; (b) 30% NLD; (c) 50% NLD; and (d) ELD.

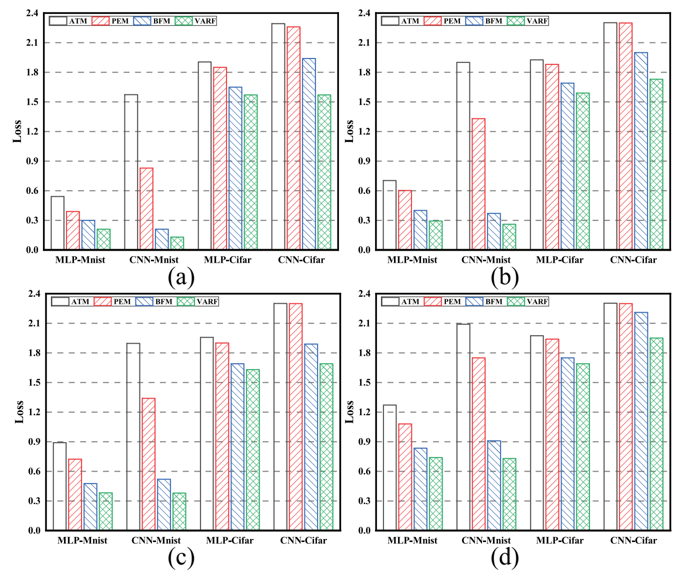

Fig. 7. Average loss of MLP-Mnist (MM), CNN-Mnist (CM), MLP-Cifar (MC), CNN-Cifar (CC) with different incentive mechanisms under different scenarios: (a) CD; (b) 30% NLD; (c) 50% NLD; and (d) ELD.

We can discover that VARF achieves the highest accuracy score compared with other all benchmarks under any scenario. As shown in Fig. 6(a), when evaluating the CNN trained with MNIST under CDs, ATM achieves a 73.27% accuracy score, PEM achieves a score of 81.75% and BFM achieves a score of 93.88%, while VARF can achieve a 96.13% accuracy score, which can improve the accuracy score by 31.2%, 17.6%, and 2.4%, respectively. Similar observations can be made for other learning models in other scenarios. Moreover, we observe that the accuracy of the model will be lower when the quality of the data set gets worse. With the same learning model, the accuracy score of VARF is 96.13%, 95.53%, 95.49%, and 87.15% under scenarios 1)–4), respectively.

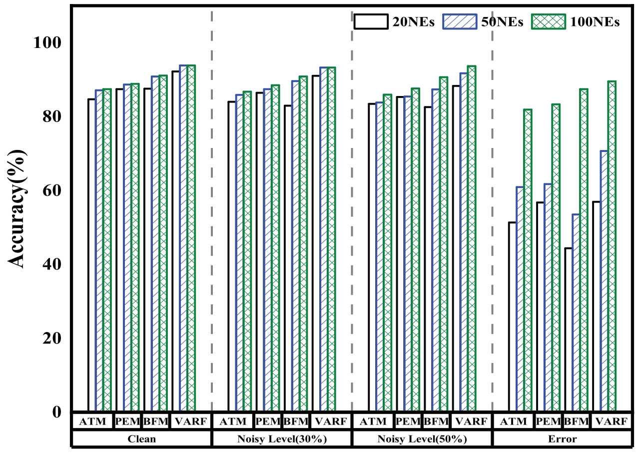

Fig. 8. Average model accuracy of the learning model with different numbers of clients N when 30% labels are variable under different scenarios: (a) CD; (b) 30% NLD; (c) 50% NLD; and (d) ELD.

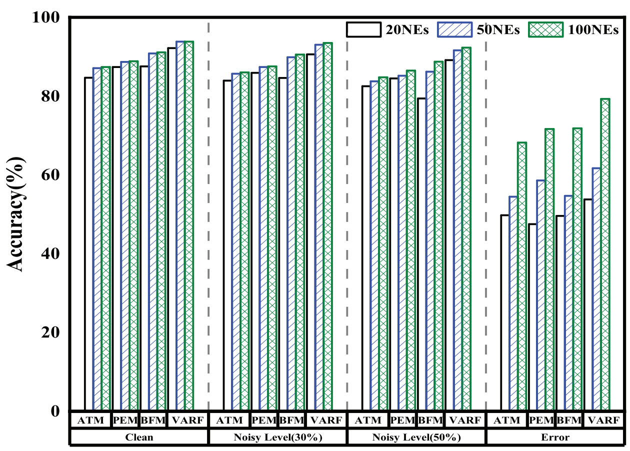

Fig. 9. Average model accuracy of the learning model with different numbers of clients N when 50% labels are variable under different scenarios: (a) CD; (b) 30% NLD; (c) 50% NLD; and (d) ELD.

Fig. 7 shows that VARF reaches the lowest loss compared with other all benchmarks under any scenarios. As shown in Fig. 7(a), when evaluating the CNN trained with MNIST under CDs, the loss of ATM is 1.573, the loss of PEM is 0.83, the loss of BFM is 0.21 and the loss of VARF is 0.13. Besides, it's obvious that the loss of the model will be higher when the quality of the data set gets worse. With the same learning model, the loss value of VARF is 0.13, 0.26, 0.38, and 0.73 under scenarios 1)–4) separately.

D. Impact of the Number of ENs N

We show how the number of ENs N affects the average model accuracy of all the learning models after running 30 iterations under different scenarios in Figs. 8 and 9.

Figs. 8 and 9 show that the score of average model accuracy after running 30 iterations increases with N under different scenarios: 1) CDs; 2) 30% NLDs; 3) 50% NLDs; and 4) ELDs. There are three major statements in this section. First of all, the score of average model accuracy increases with N in any scenario. Second, VARF outperforms all benchmarks in all scenarios. Moreover, the performance of the model increases

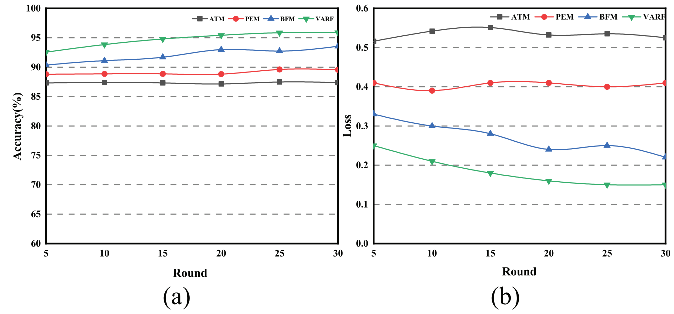

Fig. 10. Average model accuracy and loss of learning models with the different numbers of iterations G. (a) Accuracy & iteration. (b) Loss & iteration.

significantly with N when the data quality is low for all learning models. For instance, the accuracy score of VARF is 92.19%, 93.83%, and 93.85% in CDs when the number of ENs N is 20, 50, and 100, respectively. And the accuracy score of ATM, PEM, BFM, and VARF is 87.45%, 88.87%, 91.11%, and 93.85% in CDs when the number of ENs N is 100, respectively. In addition, the model accuracy performed by VARF decreases from 93.85% in the CDs to 89.51% in the Error data sets while the model accuracy of ATM decreases from 87.4% in the CDs to 68.19% in the Error data sets.

E. Impact of the Number of Rounds G

We show how the number of rounds G affects the average model accuracy and average model loss of all the learning models after running 30 iterations under different scenarios: 1) 5 rounds; 2) 10 rounds; 3) 15 rounds; 4) 20 rounds; 5) 25 rounds; and 6) 30 rounds. Then, we plot the average model accuracy results of the learning model in Fig. 10(a) and the average model loss results of the learning model in Fig. 10(b).

According to Fig. 10(a), we can observe that the model accuracy increases with G for all learning models and VARF can achieve optimal performance. However, the model accuracy of ATM and PEM is only slightly improved compared with VARF and BFM. Besides, Fig. 10(b) shows that the model loss decreases significantly with G for VARF and BFM while the model losses of ATM and PEM are basically unchanged. As shown in Fig. 10, VARF performing 30 rounds improves the accuracy score by 3.6% and reduces the model loss by 66.7% compared to running 5 rounds.

F. Impact of Noisy Level Y

This section shows how the noisy level Y affects the average model accuracy and average model loss of all the learning models after running 30 iterations with different noisy level Y under different scenarios: 1) the training data sets of 50 ENs are clean but the other 50 ENs have noisy training data sets and 2) the training data sets of 70 ENs are clean but the other 30 ENs have noisy training data sets.

As illustrated in Fig. 11, you can see that the model accuracy decreases with noisy level Y for all learning models and VARF performs optimally with any noisy level Y under different scenarios. Comparing Fig. 11(a) and (b), we find that the score of accuracy can deteriorate when upgrading the number

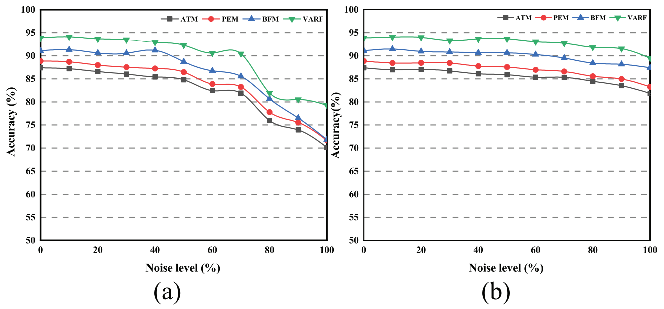

Fig. 11. Average model accuracy of the learning model with different noisy levels under different scenarios: (a) 50% labels are variable and (b) 30% labels are variable.

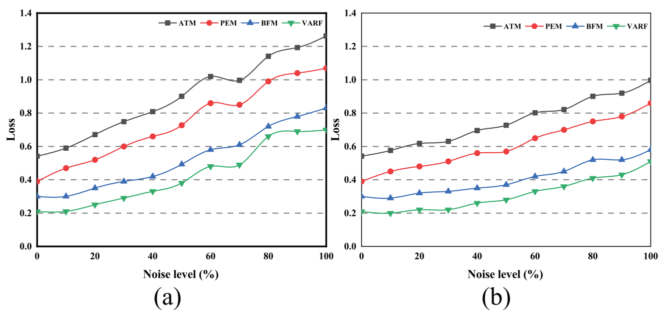

Fig. 12. Average loss of learning model with a different noise level under different scenarios: (a) 50% labels are variable and (b) 50% labels are variable.

of noisy training data sets. Moreover, we can observe that the model accuracy will decline dramatically when the noise level exceeds 50%. Taking the VARF learning model as an example, the model accuracy is 79.28% when the number of noisy training data sets is 50 and the noisy level is 100% whereas the model accuracy is 89.51% when the number of noisy training data sets is 30 and the noisy level is 100%.

In accordance with Fig. 12, we discover that the model loss improves with noisy level Y for all learning models and the model loss of VARF remains the smallest among all learning models. Besides, Fig. 12 shows that the model loss will improve when upgrading the number of noisy training data sets for all learning models. For instance, the model accuracy is 0.7 when the number of noisy training data sets is 50 and the noisy level is 100% while the model accuracy is 0.51 when the number of noisy training data sets is 30 and the noisy level is 100%.

G. Evolutionary Stability

We show how the strategy of both CPs and ENs evolve with iterations under the action of the trigger strategy. 10 CPs and 100ENs are selected as players in the evolutionary game. The initial proportions of CPs and ENs are set as 0,4, 0.5, 0.6, and 0.7, respectively. Fig. 13(a) and (b) show the strategy selection of CPs and ENs under the action of trigger strategy.

As illustrated in Fig. 13(a) and (b), we can find that at the beginning of the game, there exist deviations from the evolutionary stability strategy between CPs and ENs. After long-term learning, all players including CPs and ENs eventually form a stable cooperative relationship under the trigger

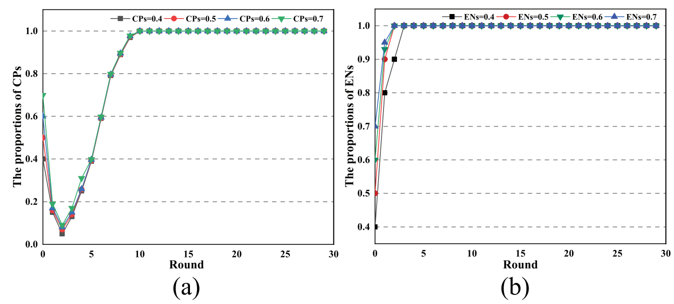

Fig. 13. Verification of strategy selection of CPs and ENs under the action of trigger strategy. (a) Evolution analysis of CPs. (b) Evolution analysis of ENs.

strategy. The final stable strategy is that the ENs choose strategy P to actively participate in the task while the CPs choose strategy Y to provide the optimal reward r^*.

VII. Conclusion

In this work, we propose an incentive mechanism for cross-silo FL in MEC, VARF, which can enhance the performance of learning tasks in cross-silo FL via motivating active and long-term participation of high-quality and high-reputation organizations. we estimate the learning quality and learning reputation of organizations according to the historical learning record. We model interactions of organizations in the long-term cross-silo FL as an infinitely repeated game. We obtain the optimal SPNE that can select high-quality and high-reputation organizations for participating in long-term model training while maximizing the amount of local data for model learning in cross-silo FL, which is a trigger strategy complied with a punishment strategy. Extensive simulations under diverse distributed learning tasks have been executed, and simulation results show that the infinitely repeated game can motivate active and long-term cooperation among organizations with high quality and high reputation and eventually CPs and ENs form a long and stable cooperative relationship under the trigger strategy. Meanwhile, VARF improves the performance of learning tasks and degrades the loss of learning tasks in cross-silo FL.

For future work, we take into consideration the combination of VARF and blockchain to ensure the credibility of historical learning quality and learning reputation and control access to the final global model.

References

[1] J. Zhang and K. B. Letaief, "Mobile edge intelligence and computing for the Internet of Vehicles," *Proc. IEEE*, vol. 108, no. 2, pp. 246–261, Feb. 2020.

[2] F. Lyn et al., "LeaD: Large-scale edge cache deployment based on spatio-temporal WiFi traffic statistics," *IEEE Trans. Mobile Comput.*, vol. 20, no. 8, pp. 2607–2623, Aug. 2021.

[3] M. Shen, J. Duan, L. Zhu, J. Zhang, X. Du, and M. Guizani, "Blockchain-based incentives for secure and collaborative data sharing in multiple clouds," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 1229–1241, Jun. 2020.

[4] M. Shen, H. Lu, F. Wang, H. Liu, and L. Zhu, "Secure and efficient blockchain-assisted authentication for edge-integrated Internet-of-Vehicles," *IEEE Trans. Veh. Technol.*, vol. 71, no. 11, pp. 12250–12263, Nov. 2022.

[5] R. Zeng, S. Zhang, J. Wang, and X. Chu, "FMore: An incentive scheme of multi-dimensional auction for federated learning in MEC," in *Proc. IEEE 40th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2020, pp. 278–288.

[6] D. Chen et al., "Matching-theory-based low-latency scheme for multitask federated learning in MEC networks," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11415–11426, Jul. 2021.

[7] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–7.

[8] S. Wang et al., "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[9] Q. He et al., "Routing optimization with deep reinforcement learning in knowledge defined networking," *IEEE Trans. Mobile Comput.*, early access, Jan. 9, 2023, doi: 10.1109/TMC.2023.3235446.

[10] S. Reddi et al., "Adaptive federated optimization," 2020, *arXiv:2003.00295*.

[11] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 7611–7623.

[12] Z. Li, D. Kovalev, X. Qian, and P. Richtárik, "Acceleration for compressed gradient descent in distributed and federated optimization," 2020, *arXiv:2002.11364*.

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

[14] M. Shen, X. Tang, L. Zhu, X. Du, and M. Guizani, "Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7702–7712, Oct. 2019.

[15] C. Xie, M. Chen, P.-Y. Chen, and B. Li, "CRFL: Certifiably robust federated learning against backdoor attacks," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 11372–11382.

[16] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the GAN: Information leakage from collaborative deep learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2017, pp. 603–618.

[17] X. Ma, H. Sun, and R. Q. Hu, "Scheduling policy and power allocation for federated learning in NOMA based MEC," in *Proc. IEEE Global Commun. Conf. GLOBECOM*, 2020, pp. 1–7.

[18] L. Zhu, H. Lin, Y. Lu, Y. Lin, and S. Han, "Delayed gradient averaging: Tolerate the communication latency for federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 29995–30007.

[19] Z. Tang, S. Shi, B. Li, and X. Chu, "GossipFL: A decentralized federated learning framework with sparsified and adaptive communication," *IEEE Trans. Parallel Distrib. Syst.*, vol. 34, no. 3, pp. 909–922, Mar. 2023.

[20] C. Thapa, M. A. P. Chamikara, S. Camtepe, and L. Sun, "SplitFed: When federated learning meets split learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, 2022, pp. 8485–8493.

[21] N. Skatchkovsky, H. Jang, and O. Simeone, "Federated neuromorphic learning of spiking neural networks for low-power edge intelligence," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2020, pp. 8524–8528.

[22] Q. Li, Z. Wen, and B. He, "Practical federated gradient boosting decision trees," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 4642–4649.

[23] J. Kang et al., "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.

[24] 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 Things J.*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019.

[25] Y. Deng et al., "FAIR: Quality-aware federated learning with precise user incentive and model aggregation," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2021, pp. 1–10.

[26] P. Blanchard, E. M. E. Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 119–129.

[27] E. M. E. Mhamdi, R. Guerraoui, and S. Rouault, "The hidden vulnerability of distributed learning in Byzantium," in *Proc. ICML*, 2018, pp. 3518–3527.

[28] R. Zeng, C. Zeng, X. Wang, B. Li, and X. Chu, "A comprehensive survey of incentive mechanism for federated learning," 2021, *arXiv:2106.15406*.

[29] Z. Tang, Y. Zhang, S. Shi, X. He, B. Han, and X. Chu, "Virtual homogeneity learning: Defending against data heterogeneity in federated learning," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 21111–21132.

[30] L. Wang, W. Wang, and B. Li, "CMFL: Mitigating communication overhead for federated learning," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2019, pp. 954–964.

[31] B. Luo, X. Li, S. Wang, J. Huang, and L. Tassiulas, "Cost-effective federated learning design," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2021, pp. 1–10.

[32] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing federated learning on non-IID data with reinforcement learning," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2020, pp. 1698–1707.

[33] M. Tang and V. W. S. Wong, "An incentive mechanism for cross-silo federated learning: A public goods perspective," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2021, pp. 1–10.

[34] O. Marfoq, C. Xu, G. Neglia, and R. Vidal, "Throughput-optimal topology design for cross-silo federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 19478–19487.

[35] U. Majeed, L. U. Khan, and C. S. Hong, "Cross-silo horizontal federated learning for flow-based time-related-features oriented traffic classification," in *Proc. 21st Asia–Pac. Netw. Oper. Manag. Symp. (APNOMS)*, 2020, pp. 389–392.

[36] M. A. Heikkilä, A. Koskela, K. Shimizu, S. Kaski, and A. Honkela, "Differentially private cross-silo federated learning," 2020, *arXiv:2007.05553*.

[37] L. Ma, X. Wang, X. Wang, L. Wang, Y. Shi, and M. Huang, "TCDA: Truthful combinatorial double auctions for mobile edge computing in Industrial Internet of Things," *IEEE Trans. Mobile Comput.*, vol. 21, no. 11, pp. 4125–4138, Nov. 2022.

[38] R. Zeng, C. Zeng, X. Wang, B. Li, and X. Chu, "Incentive mechanisms in federated learning and game-theoretical approach," *IEEE Netw.*, vol. 36, no. 6, pp. 229–235, Nov./Dec. 2022.

[39] C. Chi, Y. Wang, Y. Li, and X. Tong, "Multistrategy repeated game-based mobile crowdsourcing incentive mechanism for mobile edge computing in Internet of Things," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–18, Jan. 2021.

[40] J. Li et al., "A novel incentive mechanism based on repeated game in fog computing," in *Proc. IEEE 3rd Int. Conf. Adv. Comput. Technol. Inf. Sci. Commun. (CTISC)*, 2021, pp. 112–119.

[41] M. Zhao, L. Chen, J. Xiong, and Y. Tian, "A three-party repeated game model for data privacy in mobile edge crowdsensing of IoT," in *Proc. 13th EAI Int. Conf. Mobile Multimedia Commun. (Mobimedia)*, Aug. 2020, pp. 1–5.

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

[43] N. Ding, Z. Fang, and J. Huang, "Incentive mechanism design for federated learning with multi-dimensional private information," in *Proc. IEEE 18th Int. Symp. Model. Optim. Mobile Ad Hoc Wireless Netw. (WiOPT)*, 2020, pp. 1–8.

[44] M. Li, T. Zhang, Y. Chen, and A. J. Smola, "Efficient mini-batch training for stochastic optimization," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2014, pp. 661–670.

[45] N. Zhang, Q. Ma, and X. Chen, "Enabling long-term cooperation in cross-silo federated learning: A repeated game perspective," *IEEE Trans. Mobile Comput.*, early access, Feb. 4, 2022, doi: 10.1109/TMC.2022.3148263.

[46] T. D. Burd and R. W. Brodersen, "Processor design for portable systems," *J. VLSI Signal Process. Syst. Signal Image Video Technol.*, vol. 13, no. 2, pp. 203–221, 1996.

[47] N. H. Tran, W. Bao, A. Y. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2019, pp. 1387–1395.

[48] Y. Chen, X. Yang, X. Qin, H. Yu, B. Chen, and Z. Shen, "Focus: Dealing with label quality disparity in federated learning," 2020, *arXiv:2001.11359*.

[49] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, nos. 1–2, pp. 1–210, 2021.

[50] Y. Liu, K. Li, Y. Jin, Y. Zhang, and W. Qu, "A novel reputation computation model based on subjective logic for mobile ad hoc networks," *Future Gener. Comput. Syst.*, vol. 27, no. 5, pp. 547–554, 2011.

[51] Y. Hu, Y. Wang, Y. Li, and X. Tong, "An incentive mechanism in mobile crowdsourcing based on multi-attribute reverse auctions," *Sensors*, vol. 18, no. 10, p. 3453, 2018.

[52] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled Internet of Vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2906–2920, Mar. 2019.

[53] X. Huang, R. Yu, J. Kang, Z. Xia, and Y. Zhang, "Software defined networking for energy harvesting Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1389–1399, Jun. 2018.

[54] J. Kang, Z. Xiong, D. Niyato, P. Wang, D. Ye, and D. I. Kim, "Incentivizing consensus propagation in proof-of-stake based consortium blockchain networks," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 157–160, Feb. 2018.

[55] P. J. Miller and D. M. Chin, "Using monthly data to improve quarterly model forecasts," *Fed. Reserve Bank Minneapolis Quart. Rev.*, vol. 20, pp. 16–28, Jul. 1996.

[56] N. J. Van Leersum et al., "The dutch surgical colorectal audit," *Eur. J. Surg. Oncol.*, vol. 39, no. 10, pp. 1063–1070, 2013.

[57] D. Abreu, "On the theory of infinitely repeated games with discounting," *Econometrica J. Econometric Soc.*, vol. 56, no. 2, pp. 383–396, 1988.

[58] S. Shakkottai and R. Srikant, "Economics of network pricing with multiple ISPs," *IEEE/ACM Trans. Netw.*, vol. 14, no. 6, pp. 1233–1245, Dec. 2006.

[59] E. Hendon, H. J. Jacobsen, and B. Sloth, "The one-shot-deviation principle for sequential rationality," *Games Econ. Behav.*, vol. 12, no. 2, pp. 274–282, 1996.

Ying Li received the B.S. degree in Internet of Things from Anyang Institute of Technology, Anyang, China, in 2017, and the M.S. degree in computer technology from Northeastern University, Shenyang, China, in 2020, where she is currently pursuing the Ph.D. degree in computer science and technology.

Her research interests include distributed machine learning, blockchain, and knowledge-defined networking.

Xingwei Wang received the B.S., M.S., and Ph.D. degrees in computer science from Northeastern University, Shenyang, China, in 1989, 1992, and 1998, respectively.

He is currently a Professor with the College of Computer Science and Engineering, Northeastern University. He has published more than 100 journal articles, books and book chapters, and refereed conference papers. His research interests include cloud computing and future Internet.

Prof. Wang has received several best paper awards.

Rongfei Zeng received the Ph.D. degree (Hons.) in computer science and technology from Tsinghua University, Beijing, China, in 2012.

He is the Associate Professor with Software College, Northeastern University. He has published several papers in top journals and conferences, such as the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, *Computer Networks* (Elsevier), and IEEE ICDCS. His research interests include network security and privacy, machine learning and its security, and industrial networks and IoT.

Mingzhou Yang received the B.Sc. degree in computer science and technology from Shenyang University of Technology, Shenyang, China, in 2015, and the M.Sc. degree in computer software and theory and the Ph.D. degree in computer application technology from Northeastern University, Shenyang, in 2017 and 2022, respectively.

She is currently a Lecturer with Shenyang University of Technology. Her research interests include social network analysis, computational intelligence, and machine learning.

Kexin Li (Student Member, IEEE) received the B.S. degree in software engineering from Harbin University of Science and Technology, Harbin, China, in 2015, and the M.S. degree in computer technology from Northeastern University, Shenyang, China, in 2018, where she is currently pursuing the Ph.D. degree in computer application technology.

Her research interests include software-defined networking, edge computing, and machine learning.

Min Huang (Member, IEEE) received the B.S. degree in automatic instrument, the M.S. degree in systems engineering, and the Ph.D. degree in control theory from Northeastern University, Shenyang, China, in 1990, 1993, and 1999, respectively.

She is currently a Professor with the College of Information Science and Engineering, Northeastern University. She has published more than 100 journal articles, books, and refereed conference papers. Her research interests include modeling and optimization for logistics and supply chain system.

Schahram Dustdar (Fellow, IEEE) received the M.S. and Ph.D. degrees in business informatics from Johannes Kepler Universität Linz, Linz, Austria, in 1989 and 1992, respectively, and the Postdoctoral degree in information systems from The London School of Economics and Political Science, London, U.K., in 1994.

He is a Full Professor of Computer Science (Informatics) with a focus on Internet Technologies heading the Distributed Systems Group, TU Wien, Vienna, Austria.

Prof. Dustdar is the recipient of the ACM Distinguished Scientist Award in 2009 and the IBM Faculty Award in 2012. He has been a member of the IEEE Conference Activities Committee since 2016, the Section Committee of Informatics of the Academia Europaea since 2015, and the Academia Europaea: The Academy of Europe, Informatics Section since 2013. He has been the Chairman of the Informatics Section of the Academia Europaea since 9 December 2016. He is an Associate Editor of the IEEE TRANSACTIONS ON SERVICES COMPUTING, ACM TRANSACTIONS ON THE WEB, and ACM TRANSACTIONS ON INTERNET TECHNOLOGY, and on the editorial board of IEEE INTERNET COMPUTING. He is the Editor-in-Chief of *Computing* (an SCI-ranked journal of Springer).