Hindawi
International Journal of Intelligent Systems
Volume 2024, Article ID 6402026, 15 pages
https://doi.org/10.1155/2024/6402026

Research Article

Hierarchical Incentive Mechanism for Federated Learning: A Single Contract to Dual Contract Approach for Smart Industries

Tao Wan,[1] **Tiantian Jiang**,[1] **Weichuan Liao**,[2] **and Nan Jiang**[1]

[1]*The School of Information Engineer, East China Jiaotong University, Nanchang 330013, China*
[2]*The School of Science, East China Jiaotong University, Nanchang 330013, China*

Correspondence should be addressed to Tao Wan; wantao217@163.com

Received 17 July 2023; Revised 6 October 2023; Accepted 28 December 2023; Published 4 January 2024

Academic Editor: Yu-an Tan

Federated learning (FL) has shown promise in smart industries as a means of training machine-learning models while preserving privacy. However, it contradicts FL's low communication latency requirement to rely on the cloud to transmit information with data owners in model training tasks. Furthermore, data owners may not be willing to contribute their resources for free. To address this, we propose a single contract to dual contract approach to incentivize both model owners and workers to participate in FL-based machine learning tasks. The single-contract incentivizes model owners to contribute their model parameters, and the dual contract incentivizes workers to use their latest data to participate in the training task. The latest data draw out the trade-off between data quantity and data update frequency. Performance evaluation shows that our dual contract satisfies different preferences for data quantity and update frequency, and validates that the proposed incentive mechanism is incentive compatible and flexible.

1. Introduction

In recent years, the rise of the Industrial Internet of Things (IIoT) [1] and relevant intelligent technologies such as deep learning (DL) [2] algorithms have ushered in innovative changes and development opportunities for smart industries. Consequently, the deployment of the IIoT is becoming increasingly popular for various applications, e.g., smart grid [3], logistics [4], and healthcare [5].

IIoT integrates various technologies to digitally transform manufacturing and service operations. In 2020, Liu et al. [6] proposed a novel tracker based on response region reliability with edge computing, which achieved accurate and fast tracking with high reliability in IIoT applications. Then, considering security concerns while bringing computing power to the edge of industrial systems, Houda et al. [7] designed a novel MEC-based framework to secure IIoT applications using FL. Feng et al. [8] proposed a trustworthy self-healing scheme based on blockchain and digital twin and implemented trustworthy self-healing in the edge AI-enabled IIoT environment. Recently, IIoT has become a core component of smart applications, which can capture various valuable events and objects. Rahman et al. [9] proposed an AI-enabled IIoT to automate event management in smart cities. Farahani and Monsefi [10] combined multiparty technologies, privacy-enhancing techniques, and AI to further facilitate the industrial data economy and innovation process.

DL algorithms require substantial amounts of data to outperform traditional methods in model training. Furthermore, restricted by regulations such as the General Data Protection Regulation (GDPR) [11], they are also reluctant to share data. Federated learning (FL) [12] as an emerging technology came into being in IIoT, where different model owners collaborate by sharing their gradients instead of raw data, thereby preserving privacy. FL is a technique that can train statistical models using data distributed over remote devices or siloed data centers while keeping data localized and preserving privacy. By leveraging the diversity and heterogeneity of data from different parties, FL can train a model that has more generalization ability and adaptability, thus improving the model's performance and reliability. Additionally, the hierarchical computing architecture [13] has been proposed to enhance data transmission efficiency between the cloud and the terminal. In the hierarchical computing architecture, edge devices transmit data to the cloud through the nearest edge node, thereby reducing the number of communications for global aggregation and updates between terminal devices and remote cloud servers, energy consumption during communication, and latency.

Traditionally, FL research usually involves the model owner issuing model training tasks to workers and motivating them to participate through a single contract, i.e., the user contribution in the contract is related to only one independent variable. Workers only collect information when requested, and their contributions are only relevant to the data quantity. However, in many real-time tasks, service latency may be unbearable, requiring frequent data caching and updates to maintain data freshness. This trade-off between data quantity and data update frequency is underexplored. For example, disaster prediction in the mining industry relies on up-to-date information to prevent safety hazards [14].

In addition, when a model owner manages the trade-off between data quantity and data update frequency, it can create an incentive mismatch with the worker. For example, the model owner may prefer a large data quantity to ensure optimal model performance, but the collection expense such as energy consumption incurred in data collection and data storage in caching may be prohibitive for the worker. So, a uniform reward allocation may not result in optimal utility for both parties, as the model owner may lack information about the worker's unit cost of data collection.

Therefore, we propose an FL-based hierarchical incentive mechanism that utilizes a single contract to dual contract task-aware model. Our mechanism utilizes a single contract to incentivize model owners to collect gradients and a dual contract to motivate workers to perform both data collection and updates. We design the dual contract to flexibly adjust the different preferences of requesters regarding data quantity and update frequency. When data quantity is more important, the contract can be designed to motivate workers to collect more data. When data update frequency is more important, the contract can be designed to encourage more frequent updates. Through the self-revealing mechanism of contract theory, workers and model owners can maximize their utility by choosing the contract that best suits their needs. The contributions of this paper can be summarized as follows:

(i) We propose a new approach for federated learning that addresses the challenges of data quality in the context of edge computing. Specifically, we use a single contract to dual contract structure that enables a three-way collaboration between model owners, workers, and requesters. This approach assesses data quality through two different aspects: data quantity and data update frequency.

(ii) We utilize the self-revealing and weighting characteristics of the dual contract-theoretic incentive mechanism design to incentive worker. Specifically, the model owners can adjust the proportion of data quantity and data update frequency in their dual contracts according to their preferences of single-contract requesters. The self-revealing mechanism of the contract allows workers to maximize their benefits when choosing a contract that suits them. So the approach can justly compensate workers for the costs associated with data collection and updating.

(iii) We show that model owners can calibrate to suit the varying preferences for data quantity and data update frequency in the dual contract according to their willingness to participate in the single contract. This flexibility in calibration enables model owners to strike a balance between maximizing the quality of the data set and minimizing their own resource usage and workload, ultimately promoting a more effective and efficient federated learning process.

The paper is structured as follows. Section 2 discusses the related works, Section 3 presents the system model and problem formulation, Section 4 formulates the contract design, Section 5 discusses the performance evaluation, and Section 6 concludes the paper.

2. Related Work

The percentage of studies involving Industrial Internet of Things (IIoT) applications has significantly increased in recent years, including in the areas of smart grids [15] and supply chains [16]. The former investigates the use of IIoT to improve task resilience in the event of accidents, while the latter explores the potential of IIoT to enhance task efficiency. In particular, the study presented in [17] proposes a cloud-based solution for detecting patients in their homes.

The above studies usually assume that machine learning models are trained in an ideal environment, which is not always the case. Thus, it is necessary to investigate decentralized learning schemes, such as federated learning (FL), for industrial applications. FL was designed to enable efficient machine learning among multiple computing nodes while ensuring data owner privacy and security. In the context of FL, a privacy-preserving Byzantine-robust federated learning (PBFL) scheme based on blockchain was proposed in [18], which achieves convergence and provides privacy protection on different datasets. Additionally, a compressed and privacy-preserving FL scheme in deep neural network (DNN) architecture was proposed in [19] to address the curse of dimensionality in FL.

FL has developed successful applications in various fields, including autonomous driving car [20] and smart home [21]. In particular, the study in [22] enabled the FL-based differential privacy algorithm to enhance the privacy level of the feature in smart healthcare. Influenced by transmission efficiency, the implementation of the data transmission architecture has changed from the initial central server to the edge server, which provides edge computing [23] services locally to meet the demands of real time, security, and privacy preservation. As such, the study in [24] proposes a hierarchical edge-cloud framework to reduce the system resource requirements and data transmission time, to satisfy efficient storage and rapid response.

However, existing works have primarily focused on user privacy preservation or efficient problem-solving, rather than incentive mechanism design. With the popularization of FL tasks, designing an incentive mechanism to encourage data owners to participate in FL has become increasingly important. He et al. [25] propose a game theory-based incentive mechanism for collaborative security of FL in an energy blockchain environment, which can discourage nodes from taking malicious behaviors in iterative training of FL. Chen et al. [26] devise a multifactor reward function based on reputation, model accuracy, and reward rate, which ensures that data owners with a high reputation and high model accuracy will receive more rewards. The study in [27] selected users with bids and contributions through a reverse auction mechanism. Reputation was integrated into incentives in studies such as [28]. The study in [29] used deep reinforcement learning to design a learning-based incentive mechanism.

Currently, the contract theory approach of FL has been well explored in the literature. Xu et al. [30] present a contract-based dynamically federated learning optimized personal deep learning scheme, which enabled edge devices to reach a consensus on the optimal weights of personalized models. Also, the study in [31] designed a multidimensional contract approach. The study in [32] jointly considered data quality and computation effort. Privacy concerns have led to the emergence of contracts that incorporate privacy cost considerations, such as in the study presented in [33]. The study in [34] designed a two-period incentive mechanism that allowed for extension to multi-periods. Li et al. [35] develop a novel incentive-based federated learning framework, where a contract-based reputation mechanism and a Stackelberg-based interclient incentive mechanism are incorporated. Chen et al. [36] propose a contract-based edge-assisted federated learning model-sharing incentive mechanism, which maximize the EFL model consumers' profit and ensure the quality of training services. Feng et al. [37] incorporate local differential privacy into contract theory-based private data trading to support personalized privacy preferences. However, most of these incentive mechanisms consider worker contribution based on the data quantity without focusing on data updates in FL.

According to the work of [38], the age of information (AoI) has been considered in FL tasks, and a hierarchical incentive mechanism framework has been proposed in [39] to improve the efficiency between cloud and terminal. In this paper, we propose an FL-based single contract to dual contract hierarchical incentive mechanism that introduces a dual contract to solve the worker data multifaceted issue contribution problem. We model data quantity along with data update frequency to design incentives that take into account worker effort and appropriate profit. By adding the hierarchical framework, we aim to improve the efficiency of the federated learning task.

Table 1 provides a comparison of the prominent features of this paper's incentive mechanism with other studies discussed in this paper.

Table 1: The comparison of the prominent features of this paper's incentive mechanism with other studies.

	Technique	Contribution of users	Layered
[25, 26]	Stackelberg	Data quantity	×
[27, 28]	Auction	Data quantity	×
[29]	Reinforce	Data quantity	×
[31]	Contract	Data quantity	×
[33]	Contract	Task expenditure + privacy risk	×
[34]	Contract	Data quantity	×
[36]	Contract	Data quantity	✓
[38]	Contract	Data freshness	×
[39]	Contract	Data quantity	✓
Our	Contract	Data quantity + data freshness	✓

3. System Model and Problem Formulation

3.1. System Model. Our system model consists of requester, model owner, and worker. The requester publishes its model training task requirements to the model owner through the corresponding platform and associates with them by the contract. The model owner completes the tasks by collecting relevant gradients from the worker who has signed contracts with them, and workers train the model on data in response to tasks. The specific details are given in Figure 1.

Our federated learning model is based on a three-tiered architecture of terminal-to-edge-to-cloud. When uploading parameters at the terminal, they are not directly transmitted to the cloud server. Instead, the parameters are first subjected to edge aggregation on the edge server located at the network edge and close to the terminal device. Among these, the edge nodes in the blockchain will select the most reputable ones as the consensus committee to check the gradient parameters uploaded by the edge devices, and a leader is responsible for collecting qualified gradient parameters. Then, the edge-aggregated parameters are uploaded to the cloud server for global model aggregation and updating. When uploading global parameters to the cloud, they are not directly transmitted to the terminal. Instead, the global parameters are first subjected to edge on the edge server located at the network edge and close to the terminal device. Subsequently, the global parameters are transmitted from the edge server to the terminal. Unlike the traditional terminal-to-cloud architecture, our architecture performs edge aggregation at the edge layer, reducing unnecessary updates and communications, thereby reducing the number of communications for global aggregation and updates between terminals and remote cloud servers, energy consumption during communication and latency. This improves the computational and communication efficiency of the federated learning model.

We assume that the requester initiates a task that involves a set $\mathscr{F} = \{1, \ldots, f, \ldots, F\}$ of F model owners. As tasks are initiated, each model owner initiates a task that involves a set $\mathscr{G} = \{1, \ldots, g, \ldots, G\}$ of G workers in a synchronous task that spans a fixed duration T, with multiple instances of model training requests. In this scenario, the tasks issued by the model owner follow the Poisson process [38]. Unlike in conventional FL studies where workers collect data after request arrival, we now consider the

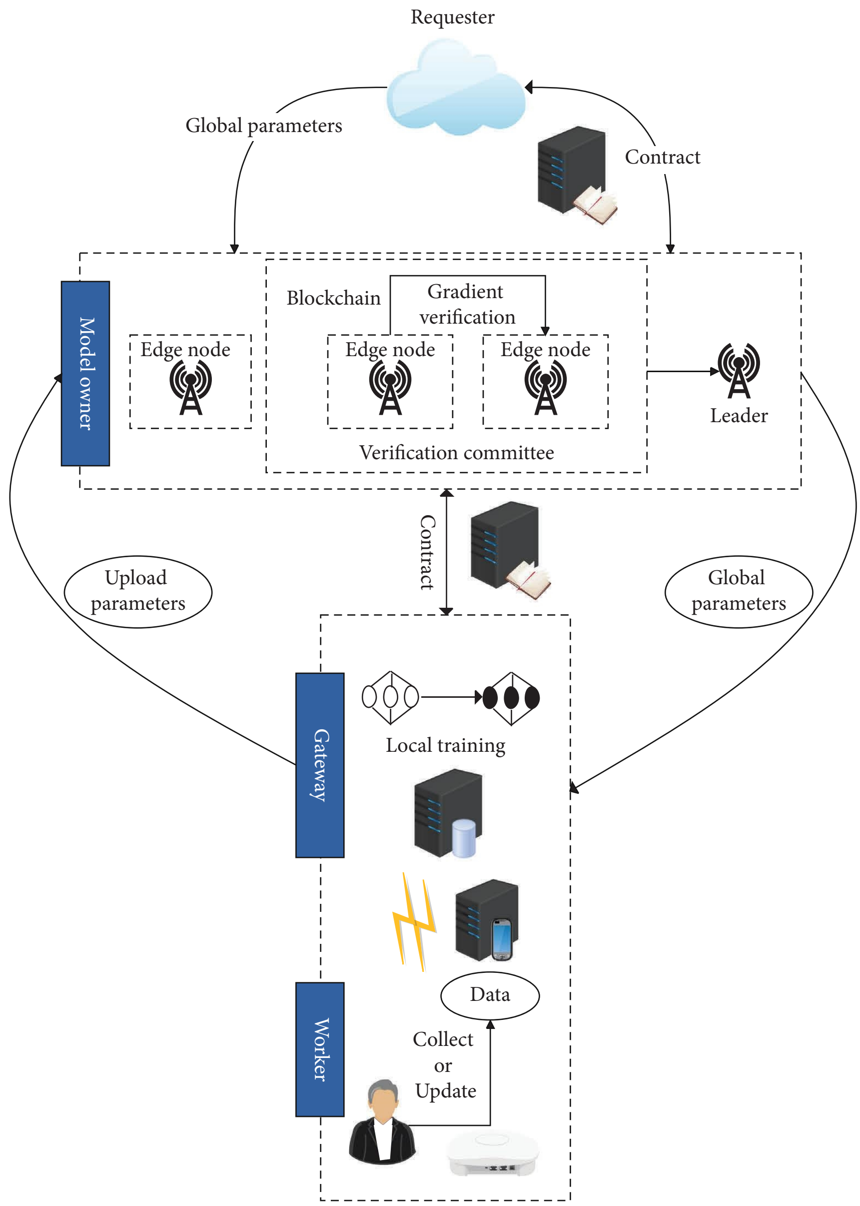

Figure 1: System model.

possibility of data caching due to service delays. Workers periodically update the cached data and proceed to train the model on the gateway. In a privacy-preserving FL method adopting differential privacy encryption, only the model parameters are sent to their corresponding model owners for aggregation. The model owners generate a consensus committee and leader based on their reputation with the requester and proceed to verify the legal gradient and perform on-chain operations. Finally, the leader submits the aggregation gradient to the requester.

3.2. Problem Formulation. In the FL network, the single contract of the incentive mechanism is comprised of $Z_1 = \{\varphi_m\colon 1 \le m \le M\}$, which represents M types of willingness to participate for model owners. Each model owner type φ_m can be characterized by a probability mass function $P(\varphi_m)$, where the types are indexed in nondecreasing order such that $0 < \varphi_1 \le \cdots \le \varphi_m \le \cdots \le \varphi_M$. Similarly, the quality of gradient contribution is represented by $Z_2 = \{V_m\colon 1 \le m \le M\}$, where the quality of gradient types is indexed in nondecreasing order such that $0 < V_1 \le \cdots \le V_m \le \cdots \le V_M$. The model owner type reflects each model owner's level of willingness to participate and determines the quality of the collected gradient, whereby model owners who are more willing to participate collect higher-quality data. We define the utility function of the requester for a model owner with type m as follows:

$$w_m(\varphi_m) = \sigma_1 \varphi_m \log(1 + V_m) - R_m, \tag{1}$$

where R_m represents the reward paid by the requester to the model owner and σ_1 is the conversion parameter from gradient performance to profits. The function represents the diminishing returns of gradient quality on model accuracy and is concave.

The dual contract of the incentive mechanism has $L_1 = \{c_{i_m}: 1 \leq m \leq M\}$ of M unit data collection cost types belonging to model owner i. The worker type c_{i_m} can be characterized by a probability mass function $p(c_{i_m})$, where the worker types are indexed in a nondecreasing order $0 < c_{i_1} \leq \cdots \leq c_{i_m} \leq \cdots \leq c_{i_M}$. The data quantity contribution is denoted as $L_2 = \{q_{i_m}: 1 \leq m \leq M\}$, where the data quantity types are indexed in a decreasing order $q_{i_1} \geq \cdots \geq q_{i_m} \geq \cdots \geq q_{i_M} > 0$. Immediately after, there has $L_3 = \left\{Q^m_{c_{i_j}}: 1 \leq m \leq M\right\}$ of M data update cost type belonging to the unit data collection cost type c_{i_j}, where the data update cost types are indexed in a nondecreasing order $0 < Q^1_{c_{i_j}} \leq \cdots \leq Q^m_{c_{i_j}} \leq \cdots \leq Q^M_{c_{i_j}}$. Similarly, the data update frequency is denoted as $L_4 = \left\{\overline{\eta^m_{c_{i_j}}}: 1 \leq m \leq M\right\}$, where the data update frequency types are indexed in a decreasing order $\overline{\eta^1_{c_{i_j}}} \geq \cdots \geq \overline{\eta^m_{c_{i_j}}} \geq \cdots \geq \overline{\eta^M_{c_{i_j}}} > 0$. The type of unit data collection cost reflects each worker's level of data collection willingness to participate and determines the quantity of data collected, i.e., workers with lower unit data collection costs are more willing to collect data. The cost of data update is also not the same, leading to different data update willingness, i.e., workers with lower data update cost are more willing to update data.

The type m cost of data update type belonging to the unit data collection cost type c_{i_j} denoted by $Q^m_{c_{i_j}}$ can be expressed as follows:

$$Q^m_{c_{i_j}} = \left(c_{i_m} + \partial^T + \partial^S\right) q_{i_m}, \tag{2}$$

where ∂^T refers to the energy consumed for unit data transmission from the IIoT network to the gateway and ∂^S refers to the energy consumed for unit data caching [40].

The type m worker utility of data quantity denoted by ρ_m can be expressed as follows:

$$\rho_m\left(c_{i_m}\right) = \sigma_2 \log\left(1 + \varrho q_m\right) - r_m, \tag{3}$$

where q_m and r_m represent the amounts of data used to train the model and reward from the i model owner of worker m, respectively. Moreover, σ_2 and ϱ represent the conversion parameter from data quantity performance to profits. The diminishing returns of data quantity are represented by a concave function.

The type m worker utility of data update frequency denoted by χ_m can be expressed as follows:

$$\chi_m\left(Q_m\right) = \sigma_3\left(s_a \log\left(1 + \alpha\overline{\eta_m}\right) + s_b\left(-\frac{\overline{\eta_m}^2}{\beta} + \vartheta\right)\right) - k_m, \tag{4}$$

where s_a and s_b refer to the weighted preference for the data freshness and service delays, respectively. In addition, $s_a + s_b = 1$ and $s_a, s_b \in [0, 1]$. α represents calibratable system model parameters that determine the data freshness on model accuracy. β and ϑ are calibratable system model parameters that determine the service delays on model accuracy. σ_3 represents the conversion parameter from update frequency performance to profits. The data freshness is fit by a monotonically increasing concave function showing incremental return decreases with the data update frequency. The service delay is fit by a monotonically decreasing concave function showing the decline rate of return decreases with the data update frequency.

In our hierarchical incentive mechanism design, we take into account the contract formulated by the requester to obtain an FL-based model from the model owner. In order to gather gradients for the relevant data quality, the model owner has the flexibility to adjust z_a and z_b to accommodate varying preferences for data quantity and update frequency, as well as s_a and s_b to cater to varying preferences for data freshness and service delays. These adjustments are crucial to incentivize workers to participate in the task.

4. Contract-Theoretic Incentive Mechanism Design

In this section, the requester incentivizes the model owner to collect models with a single contract that takes into account data quality, while the model owner incentivizes workers to train the model with a dual contract that considers both data quantity and freshness. We begin by studying the single contract of a representative model owner and then studying the dual contract of a representative worker. Then, we discuss the conditions for contract feasibility and relax the constraints to derive the optimal contract.

4.1. Single Contract. For ease of notation, we study one of the model owners as a representative for now. The model owner of type m utility maximize problem is denoted by U_m as follows:

$$\max_{(R_m, V_m)} U_m = \varphi_m R_m - C V_m, \tag{5}$$

where V_m refers to the quality of the gradient collected, R_m refers to the contract rewards from the requester, and C represents the cost incurred per unit quality of the gradient collected. In the formulation of contract theory, each model owner has a bundle $\{R_m, V_m\}$ that maximizes its utility U_m.

From (1), the requester utility maximization function denoted by γ_1 can be expressed as follows:

$$\begin{aligned}\gamma_1 &= \sum_{m=1}^{M} w_m\left(\varphi_m\right)\\ &= \sum_{m=1}^{M} \mathrm{NP}\left(\varphi_m\right)\left(\sigma_1 \varphi_m \log\left(1 + V_m\right) - R_m\right),\end{aligned} \tag{6}$$

where $P(\varphi_m)$ refers to the proportion of model owner type m, $\sum_{m=1}^{M} P(\varphi_m) = 1$, N refers to the number of model owners, and R_m refers to the reward to each model owner of type m for its gradient collection efforts. For ease of reference, we refer the readers to Table 2 for commonly used notations.

Table 2: Table of commonly used notations.

Notations	Description
φ	Model owner's level of willingness
V	Model owner gradient quality contribution
C	Unit cost of gradient collection
q	Worker data quantity contribution
$\overline{\eta}$	Worker data update frequency contribution
c	Unit cost of data collection
Q	The cost of data update
R	Model owner contract reward
r	Worker data collect contract reward
k	Worker data update contract reward
$\sigma_1, \sigma_2, \sigma_3$	Model owner conversion parameter from data quality and worker conversion parameter from data quantity and data update frequency

To ensure the feasibility of the contract, if and only if satisfy the following constraints.

Definition 1 (individual rationality (IR)). Each model owner participates in the task when they can get the positive utility, i.e.,

$$U_m = \varphi_m R_m - CV_m \geq 0. \tag{7}$$

Definition 2 (incentive compatibility (IC)). The utility of each model owner can be maximized if and only if they choose the contract design for its type, i.e.,

$$\varphi_m R_m - CV_m \geq \varphi_z R_z - CV_z, \quad m \neq z. \tag{8}$$

To guarantee a feasible contract, we have to deal with M IR constraint and $M(M-1)$ IC constraint to reduce and relax conditions.

Lemma 3. *For any feasible contract, we have $R_m \geq R_z$ if and only if $\varphi_m \geq \varphi_z$, $m \neq z$, $\forall m, z \in \{1, \ldots, M\}$.*

Proof. Using the IC constraint in Definition 2, we first prove if $\varphi_m \geq \varphi_z$, it follows $R_m \geq R_z$.

As such, we have

$$\begin{aligned} &\varphi_m R_m - \mathrm{CV}_m \geq \varphi_m R_z - \mathrm{CV}_z \text{ and} \\ &\varphi_z R_z - \mathrm{CV}_z \geq \varphi_z R_m - \mathrm{CV}_m. \end{aligned} \tag{9}$$

Then, we add these two inequalities

$$\varphi_m R_m + \varphi_z R_z \geq \varphi_m R_z + \varphi_z R_m. \tag{10}$$

By swapping left and right, we can obtain

$$\begin{aligned} \varphi_m R_m - \varphi_z R_m &\geq \varphi_m R_z - \varphi_z R_z, \\ R_m(\varphi_m - \varphi_z) &\geq R_z(\varphi_m - \varphi_z). \end{aligned} \tag{11}$$

When $\varphi_m - \varphi_z \geq 0$, it follows $R_m \geq R_z$. Likewise, from the IC constraints, we have

$$\varphi_m(R_m - R_z) \geq \varphi_z(R_m - R_z). \tag{12}$$

When $R_m \geq R_z \geq 0$, it follows $\varphi_m \geq \varphi_z$. Lemma 3 is proven. □

Lemma 3 implies that model owners with a higher willingness to participate φ to collect the higher quality of gradient V, and the more rewards R will be obtained. As such, the contract bundles are designed such that higher gradient quality contributed translate to higher rewards. As such, a feasible contract establishment must have the necessary condition for the following monotonicity conditions.

Theorem 4 (monotonicity). *A feasible contract must satisfy the following conditions:*

$$\begin{cases} 0 \leq R_1 \leq \cdots \leq R_m \leq \cdots \leq R_M, \\ 0 \leq V_1 \leq \cdots \leq V_m \leq \cdots \leq V_M. \end{cases} \tag{13}$$

Next, we further relax the IR and IC constraints. Intuitively, the maximum utility model owner incurs the highest quality of data, i.e., the type M model owner.

Lemma 5 (reduce single-contract IR constraints). *If the IR constraints of model owner type 1 are satisfied, the other IR constraints will also remain the same.*

Proof. According to the IC constraints and conditions $\varphi_1 \leq \cdots \leq \varphi_m \leq \cdots \leq \varphi_M$, we have

$$\varphi_i R_i - \mathrm{CV}_i \geq \varphi_i R_1 - \mathrm{CV}_1 \geq \varphi_1 R_1 - \mathrm{CV}_1. \tag{14}$$

As such, if the IR constraint of model owner type 1 is satisfied, the type m, $m \in \{1, \ldots, M\}$, IR constraints are automatically satisfied. □

Lemma 6 (reduce single-contract IC constraints). *The constraints can be reduced to local down incentive constraints (LDIC).*

Proof. Consider three model owner types, where $\varphi_{m-1} \leq \varphi_m \leq \varphi_{m+1}$. The two LDICs, i.e., constraints between type m and type $m-1$ model owners, are as follows:

$$\varphi_{m+1}R_{m+1} - \mathrm{CV}_{m+1} \geq \varphi_{m+1}R_m - \mathrm{CV}_m \text{ and} \quad \varphi_m R_m - \mathrm{CV}_m \geq \varphi_m R_{m-1} - \mathrm{CV}_{m-1}. \tag{15}$$

It can be obtained from Lemma 3 that when $R_m \geq R_z$, it follows $\varphi_m \geq \varphi_z$, and we can rewrite LDICs as follows:

$$\begin{aligned}&\varphi_{m+1}\left(R_m - R_{m-1}\right) \geq \\ &\quad \varphi_m\left(R_m - R_{m-1}\right) \geq C\left(V_m - V_{m-1}\right), \text{ and} \\ &\varphi_{m+1}R_{m+1} - \mathrm{CV}_{m+1} \geq \\ &\quad \varphi_{m+1}R_m - \mathrm{CV}_m \geq \varphi_{m+1}R_{m-1} - \mathrm{CV}_{m-1}.\end{aligned} \tag{16}$$

As such, we have

$$\varphi_{m+1}R_{m+1} - \mathrm{CV}_{m+1} \geq \varphi_{m+1}R_{m-1} - \mathrm{CV}_{m-1}. \tag{17}$$

As such, if the IC constraint applies to model owners of type m, it will also apply to model owners of type $m-1$. This process can be extended down from type $m-1$ to model owners of type 1, i.e., all LDICs remain the same, so we can rewrite that as follows:

$$\begin{aligned}\varphi_{m+1}R_{m+1} - \mathrm{CV}_{m+1} &\geq \varphi_{m+1}R_{m-1} - \mathrm{CV}_{m-1} \\ &\geq \cdots \\ &\geq \varphi_{m+1}R_1 - \mathrm{CV}_1.\end{aligned} \tag{18}$$

We can find that if the local upward incentive constraint (LUIC) holds, all UICs are also satisfied. From the monotonicity condition in Theorem 4, LDIC also implies a local upward incentive constraint (LUIC) as follows:

$$\varphi_{m-1}R_m - \mathrm{CV}_m \leq \varphi_{m-1}R_{m-1} - \mathrm{CV}_{m-1}. \tag{19}$$

As such, the IC constraints can be reduced to LDIC constraints, and it also guarantees that all UIC and DIC constraints hold.

With the constraints relaxed, we can derive a tractable set of sufficient conditions for the feasible contract. □

Theorem 7. *A feasible contract must meet the following sufficient conditions:*

$$\begin{aligned}&\varphi_1 R_1 - \mathrm{CV}_1 \geq 0, \\ &\varphi_m R_m - \mathrm{CV}_m \geq \varphi_{m-1}R_{m-1} - \mathrm{CV}_{m-1}.\end{aligned} \tag{20}$$

From the optimization contract established, we take V as the only influencing factor to study gradient collection. As such, the optimal rewarding scheme can be summarized in the following theorem.

Theorem 8. *For a known set of data quantity V satisfying $0 \leq V_1 \leq \cdots \leq V_m \leq \cdots \leq V_M$, the optimal reward is given by*

$$R_m^* = \begin{cases} \dfrac{1}{\varphi_m}\mathrm{CV}_m, & \text{if } m = 1, \\ R_{m-1} - \dfrac{1}{\varphi_m}\mathrm{CV}_{m-1} + \dfrac{1}{\varphi_m}\mathrm{CV}_m, & \text{otherwise.}\end{cases} \tag{21}$$

Proof. We use contradiction to validate this theorem. There exists a R^Γ that yields greater profit for the model owner, meaning that Theorem 7 is incorrect, i.e., $\gamma_1(R^\Gamma) \geq \gamma_1(R^*)$. This implies there exists at least a $t \in \{1, 2, \ldots, M\}$ that satisfies the inequality $R_t^\Gamma \leq R_t^*$.

According to Lemma 6, we have

$$R_t^\Gamma \geq R_{t-1}^\Gamma - \frac{1}{\varphi_t}\mathrm{CV}_{t-1} + \frac{1}{\varphi_t}\mathrm{CV}_t. \tag{22}$$

From Theorem 7, we also can get

$$R_t^\Gamma = R_{t-1}^\Gamma - \frac{1}{\varphi_t}\mathrm{CV}_{t-1} + \frac{1}{\varphi_t}\mathrm{CV}_t. \tag{23}$$

From (22) and (23), we can deduce that $R_1^\Gamma \leq R_1^* = 1/\varphi_1 \mathrm{CV}_1$. This violates the IR constraint. Therefore, there does not exist the rewards R^Γ in the feasible contract that yields greater profit for the model owner. □

Substitute (10) into (6), the variable of each data quantity V_m can be derived by separately optimizing each V_m^* as follows:

$$\begin{aligned}V_m^* = \arg\max_{V_m}\ & P(\varphi_m)\left(\sigma_1\varphi_m \log\left(V_m + 1\right)\right) \\ & - \sum_{m=1}^{M}\left(P(\varphi_m)\frac{1}{\varphi_m}\mathrm{CV}_m + \Theta_m \sum_{d=m+1}^{M} P(\varphi_t)\right),\end{aligned} \tag{24}$$

where $\Theta_m = 1/\varphi_m \mathrm{CV}_m - 1/\varphi_{m+1}\mathrm{CV}_m$ and $\Theta_M = 0$. The derived solutions are feasible when they satisfy the monotonicity constraint. Otherwise, we adopt the "Bunching and Ironing" algorithm [39] to adjust the solutions iteratively (see Algorithm 1).

4.2. Dual Contract. For ease of notation, we study one of the workers as a representative for now. The worker of type m data collect and data update frequency utility maximize the problem denoted by u_m and λ_m as follows:

$$\max_{(r_m, q_m)} u_m = r_m - c_m q_m, \tag{25}$$

$$\max_{(k_m, \overline{\eta_m})} \lambda_m = k_m - Q_m \overline{\eta_m}, \tag{26}$$

where r_m represents the contract reward from data quantity and k_m represents the contract reward from data update frequency. In the formulation of contract theory, each worker has a bundles $\{r_m, q_m\}$ and $\{k_m, \overline{\eta_m}\}$ that, respectively, maximizes its utility u_m and λ_m.

From (3) and (4), the model owner utility is expressed as follows.

For the data quantity performance profit γ_2,

$$\begin{aligned}\gamma_2 &= \sum_{m=1}^{M} \rho_m(c_m) \\ &= \sum_{m=1}^{M} np(c_m)\left(\sigma_2 \log\left(1 + q_m\right) - r_m\right).\end{aligned} \tag{27}$$

For the data update frequency performance profit γ_3,

(1) Initialization: Let $V_m^* = \arg \max_{V_m} G_m(V_m), \forall m \in \{1, \ldots, M\}$
(2) **while** The set of $V_m = \{V_m^*\}$ violates the monotonicity constraint, **do**
(3) Find an infeasible subsequence $\{V_i^*, V_{i+1}^*, \ldots, V_j^*\}$, where $V_i^* \leq \cdots \leq V_j^*$ and $i < j$;
(4) Set $V_l^* = \arg \max_V \sum_{t=i}^{j} G_t(V), \forall l \in \{i, \ldots, j\}$;
(5) **end while**
(6) **Return** The feasible set $V^* = \{V_m^*\}, m \in \{1, \ldots, M\}$

Algorithm 1: "Bunching and Ironing" adjusted algorithm.

$$\gamma_3 = \sum_{m=1}^{M} \chi_m(Q_m) = \sum_{m=1}^{M} \mathrm{np}(Q_m)\left(\sigma_3\left(s_a \log(1 + \alpha\overline{\eta_m}) + s_b\left(-\frac{\overline{\eta_m}^2}{\beta} + \vartheta\right)\right) - k_m\right), \tag{28}$$

where n represents the number of the worker participating in the task and $p(\omega_m)$ denotes the proportion of m type worker. In the formula of contract theory, q_m represents the data quantity provided by workers, $\overline{\eta_m}$ represents the data update frequency provided by workers, and the quality of data is equal to the weighted preference of data quantity and data update frequency.

Overall, the utility of the worker is denoted by L_m that can be expressed as follows:

$$L_m = z_a u_m + z_b \lambda_m, \tag{29}$$

where z_a and z_b represent the weighted preference for data quantity and data update frequency, respectively. In addition, $z_a + z_b = 1$ and $z_a, z_b \in [0, 1]$.

Also, for the data quantity and data update frequency contract, each contract must satisfy the following constraints.

Definition 9 (individual rationality (IR)). Each worker participates in the task if and only if its utility is not less than zero, i.e.,

$$u_m = r_m - c_m q_m \geq 0, \tag{30}$$

$$\lambda_m = k_m - Q_m\overline{\eta_m} \geq 0. \tag{31}$$

Definition 10 (incentive compatibility (IC)). Each worker of type m only chooses contracts designed for its type, not any other contract to maximize utility, i.e.,

$$r_m - c_m q_m \geq r_z - c_z q_z, \quad m \neq z, \tag{32}$$

and

$$k_m - Q_m\overline{\eta_m} \geq k_z - Q_z\overline{\eta_z}, \quad m \neq z. \tag{33}$$

However, this means that we have to deal with 2 M IR constraints and 2 $M(M-1)$ IC constraints, both of which are nonconvex. As such, we continue to reduce and relax the conditions that guarantee the contract is feasible.

Lemma 11. *For any feasible contract, we have if $c_m \leq c_z$, $Q_m \leq Q_z$, it follows $q_m \geq q_z$, $\overline{\eta_m} \geq \overline{\eta_z}$, $m \neq z$, $\forall m, z \in \{1, \ldots, M\}$.*

Proof. Using the IC in Definition 10, we have

$$\begin{aligned} r_m - c_m q_m &\geq r_z - c_m q_z \text{ and} \\ r_z - c_z q_z &\geq r_m - c_z q_m. \end{aligned} \tag{34}$$

Then, we add these two inequalities

$$c_z q_m + c_m q_z \geq c_m q_m + c_z q_z. \tag{35}$$

Tidying it up, we can get

$$(c_m - c_z)(q_m - q_z) \geq 0. \tag{36}$$

When $c_m \leq c_z$, it follows $q_m \geq q_z$. Likewise, we can prove if $Q_m \leq Q_z$, it follows $\overline{\eta_m} \geq \overline{\eta_z}$ in this way. □

Lemma 11 implies that workers with the lower unit data collection cost c collect the higher data quantities q and the more reward r. Similarly, workers with the lower cost of data update Q to update the higher data and more rewards k will be obtained. As such, the contract bundles are designed such that higher data quantities and data update frequency contribute translate to higher rewards. Also, a feasible contract has the necessary conditions for the following monotonicity conditions.

Theorem 12 (monotonicity). *The feasible dual contract must satisfy the following conditions:*

$$\begin{cases} q_1 \geq \cdots \geq q_m \geq \cdots \geq q_M, \\ r_1 \geq \cdots \geq r_m \geq \cdots \geq r_M, \end{cases} \tag{37}$$

and

$$\begin{cases} \overline{\eta_1} \geq \cdots \geq \overline{\eta_m} \geq \cdots \geq \overline{\eta_M}, \\ k_1 \geq \cdots \geq k_m \geq \cdots \geq k_M. \end{cases} \tag{38}$$

Next, we further relax the IR and IC constraints. Intuitively, the maximum utility worker is the worker that incurs the highest quality of data, i.e., the type M worker.

Lemma 13 (reduce dual contract IR constraints). *If the IR constraints of worker type 1 are satisfied, the other IR constraints will also remain the same.*

Proof. In the data collection stage, according to the IC constraints in Definition 10 and conditions $c_1 \geq \cdots \geq c_m \geq \cdots \geq c_M$, we have

$$r_i - c_i q_i \geq r_1 - c_i q_1 \geq r_1 - c_1 q_1. \tag{39}$$

In the data update stage, the worker data update cost Q is satisfied $Q_1 \geq \cdots \geq Q_m \geq \cdots \geq Q_M$, and we have

$$k_i - Q_i \overline{\eta_i} \geq k_1 - Q_i \overline{\eta_1} \geq k_1 - Q_1 \overline{\eta_1}. \tag{40}$$

As such, if the IR constraint of worker type 1 is satisfied, the other IR constraints are automatically satisfied. □

Lemma 14 (reduce dual contract IC constraints). *The constraints can be reduced to local down incentive constraints (LDIC).*

Proof. In the data collection stage, consider three worker types, where $c_{m-1} \geq c_m \geq c_{m+1}$. The two LDICs, i.e., constraints between owners of type m and type $m-1$, are as follows:

$$\begin{aligned} r_{m+1} - c_{m+1} q_{m+1} &\geq r_m - c_{m+1} q_m, \text{ and} \\ r_m - c_m q_m &\geq r_{m-1} - c_m q_{m-1}. \end{aligned} \tag{41}$$

It can be obtained from Lemma 11 that when $c_m \leq c_z$, it follows $q_m \geq q_z$. As such, we can rewrite LDICs as follows:

$$\begin{aligned} &c_{m-1}\left(q_{m-1} - q_m\right) \geq \\ &\quad c_m\left(q_{m-1} - q_m\right) \geq r_{m-1} - r_m. \end{aligned} \tag{42}$$

Tidy it up, we have

$$\begin{aligned} &r_{m+1} - c_{m+1} q_{m+1} \geq \\ &\quad r_m - c_{m+1} q_m \geq r_{m-1} - c_{m+1} q_{m-1}. \end{aligned} \tag{43}$$

As such, we have

$$r_{m+1} - c_{m+1} q_{m+1} \geq r_{m-1} - c_{m+1} q_{m-1}. \tag{44}$$

If the IC constraint in Definition 10 applies to workers of type m, it will also apply to workers of type $m-1$. This process can be extended down from type $m-1$ to workers of type 1, i.e., all LDICs remain the same, and we can rewrite that as follows:

$$\begin{aligned} r_{m+1} - c_{m+1} q_{m+1} &\geq r_{m-1} - c_{m+1} q_{m-1} \\ &\geq \cdots \\ &\geq r_1 - c_{m+1} q_1. \end{aligned} \tag{45}$$

We can find that if the local upward incentive constraint (LUIC) holds, then all UICs are also satisfied. From the condition in Theorem 12, if $r_m \geq r_{m-1}$, LDIC also implies a local upward incentive constraint (LUIC) as follows:

$$r_m - c_{m-1} q_m \leq r_{m-1} - c_{m-1} q_{m-1}. \tag{46}$$

Likewise, we can prove the data update contract in this way. As such, we have shown that the IC constraints in Definition 10 can be reduced to LDIC constraints since it also guarantees that all UIC and DIC constraints hold. □

With the constraints relaxed, we can derive a tractable set of sufficient conditions for the feasible contract.

Theorem 15. *A feasible contract must meet the following sufficient conditions:*

$$\begin{gathered} r_M - c_M q_M \geq 0, \\ k_M - Q_M \eta_M \geq 0, \\ r_{m-1} - c_{m-1} q_{m-1} \geq r_m - c_m q_m, \\ k_{m-1} - Q_{m-1} \eta_{m-1} \geq k_m - Q_m \overline{\eta_m}, \\ \sum_{m=1}^{M} \left(z_a r_m + z_b k_m\right) \leq R_m^*, \\ \sum_{m=1}^{M} \left(z_a q_m + z_b \mu q_m \overline{\eta_m}\right) = V_m^*. \end{gathered} \tag{47}$$

Theorem 15 implies that the weighted quality of data quantity and data update frequency is equal to the data quality required in the single contract, where we add a quality conversion parameter μ from data update frequency to data quantity. The reward obtained by the weighted data quantity and data update frequency is smaller than the reward obtained in the single contract. Thereafter, we take c and $\overline{\eta}$ as the only influencing factor to study data collection and data update, respectively. As such, the optimal rewarding scheme can be summarized in the following theorem.

Theorem 16. *For a known set of data quantity q satisfying $q_1 \geq \cdots \geq q_m \geq \cdots \geq q_M$ in a feasible contract, the optimal reward is given by*

$$r_m^* = \begin{cases} c_m q_m, & \text{if } m = M, \\ r_{m-1} - c_{m-1} q_{m-1} + c_{m-1} q_{m-1}, & \text{otherwise.} \end{cases} \tag{48}$$

Similar, for a $\overline{\eta}$ satisfying $\overline{\eta_1} \geq \cdots \geq \overline{\eta_m} \geq \cdots \geq \overline{\eta_M}$ in a feasible contract, the optimal reward is given by

$$k_m^* = \begin{cases} Q_m \overline{\eta_m}, & \text{if } m = M, \\ k_{m-1} - Q_{m-1} \overline{\eta_{m-1}} + Q_{m-1} \overline{\eta_{m-1}}, & \text{otherwise.} \end{cases} \tag{49}$$

Proof. We adopt the proof by contradiction to validate this theorem. We first assume that there exists a r^Γ that yields greater profit for the worker, meaning that Theorem 15 is incorrect, i.e., $\delta_1(r^\Gamma) \geq \delta_1(r^*)$. This implies there exists at least a $x \in \{1, 2, \ldots, M\}$ that satisfies the inequality $r_x^\Gamma \leq r_x^*$.

According to Lemma 13 and Theorem 15, we have

$$r_x^\Gamma \geq r_{x-1}^\Gamma - c_x q_{x-1} + c_x q_x, \tag{50}$$

$$r_x^\Gamma = r_{x-1}^\Gamma - c_x q_{x-1} + c_x q_x. \tag{51}$$

From (50) and (51), we can deduce that $r_1^{\Gamma} \leq r_1^* = c_1 q_1$. This violates the IR constraint. Therefore, there does not exist the rewards c^{Γ} in the feasible contract that yields greater profit for the model owner. Likewise, we can prove the (26) in this way. □

Substitute (25) into (16), the variable of each data quantity q_m can be derived by separately optimizing each q_m^* as follows:

$$V_m^* = \arg\max_{V_m} P(\phi_m)(\sigma_1 \phi_m \log(V_m + 1)) - \sum_{m=1}^{M}\left(P(\phi_m)\frac{1}{\phi_m}CV_m + \Theta_m \sum_{d=m+1}^{M} P(\phi_t)\right). \tag{52}$$

Similarly, substitute (49) into (28), the variable of each data update frequency $\overline{\eta_m}$ can be derived by separately optimizing each $\overline{\eta_m^*}$ as follows:

$$\overline{\eta_m^*} = \arg\max_{\overline{\eta_m}} p(Q_m)\left(\sigma_3\left(s_a \log(1 + \alpha\overline{\eta_m}) + s_b\left(-\frac{\overline{\eta_m}^2}{\beta} + \vartheta\right)\right)\right) + Q_{m-1}\overline{\eta_m}\sum_{t=1}^{m-1} p(Q_t) - Q_m\overline{\eta_m}\sum_{t=1}^{m} p(Q_t). \tag{53}$$

The derived solutions are feasible if and only if they satisfy the monotonicity constraint. Otherwise, we can replace V with q or $\overline{\eta}$ using the Bunching and Ironing algorithm.

5. Performance Evaluation

In this section, we evaluate the feasibility of the single contract and the dual contract and the optimality of our designed dual contract-based incentive mechanism. Then, we evaluate the data quantity and data update frequency changes under the different preferences. Also, we evaluate the utility of workers, the utility of model owners, and social welfare changes under the different worker types.

The key simulation parameters are provided in Table 3, and we assume that there are 100 workers to research our incentive mechanism, with varying collection costs q_m modeled by the normal distribution. We adopt the k-means clustering method to derive $M = 5$ clusters of workers and $M = 8$ clusters of model owners. We keep $s_a = s_b = 0.5$, i.e., both data freshness and service delays are of equal importance to the model owner.

5.1. Contract Optimization. To study the contract feasibility, we set $\sigma_1 = 100$, $\sigma_2 = 20$, and $\sigma_3 = 70$.

For the single contract of the incentive mechanism, Figure 2 shows that both the quality of the gradient contributed and rewards for the model owners increase as the model owner's willingness to participate increases. The monotonicity constraint in Theorem 4 is satisfied. Also, Figure 3 shows that all model owner utilities are positive and can maximize utility when they choose the contract design for its type. The IR constraint in Definition 1 and the IC constraint in Definition 2 are satisfied.

The dual contract of the incentive mechanism contains two superimposed contracts. For the first stage contract, Figure 4 shows that both the data quantity contributed and rewards for the workers decrease as the cost incurred per unit quantity of the data collected increases. The monotonicity constraint in Theorem 12 is satisfied. Also, Figure 5 shows that all worker utilities are positive and can maximize utility when they choose the contract design for its type. The IR constraint in Definition 9 and the IC constraint in Definition 10 are satisfied. The workers collect more data in the first stage, i.e., the higher update costs in the second stage. As such, Figure 6 shows that both the data update frequency contributed and rewards for the workers increase as the cost of the data update decreases. The monotonicity in Theorem 12 is satisfied. Also, Figure 7 shows that all worker utilities are positive and can maximize utility when they choose the contract design for its type. The IR constraints in Definition 9 and IC constraints in Definition 10 are satisfied.

5.2. Performance Comparison. To facilitate the analysis, we consider one task as representative of the continuous task, while the other tasks are similar. Therefore, our proposed incentive mechanism can be applied to any number of tasks in a continuous task with a fixed duration T. Moreover, we take the example of the requester who prefers data quantity with a set of values for $\sigma_2 = 5.9z_a^2 + 0.85z_a + 20.88$ and $\sigma_3 = 6z_a^2 - 11z_a + 75$. Figure 8 illustrates the variations in the model owner's utility as the data quantity preference z_a changes. As expected, the model owner's utility increases as the data quantity weighing z_a becomes higher, indicating a greater inclination toward the requester's preference. Furthermore, when $z_a > 0.5$, the value of z_a is equivalent to the model owner's willingness to participate φ in the single contract. A higher z_a implies a greater willingness to participate. This finding validates that the model owner can adjust the weights z_a and z_b to accommodate the requester's preference and own willingness to participate.

We present a comparison between the proposed incentive scheme and two other contract-based schemes: the contract-based social maximization scheme (CS) and the contract-based complete information scheme (CC) presented in [33]. The CC scenario assumes that the model owner has full knowledge of the cost types of each worker,

Table 3: Table of key simulation parameters.

Simulation parameters	Value
Gradient collect parameters: ϕ, C, and μ	$N(0.8, 0.01)$, 1.5, and 0.5
Data collect parameters: c and ϱ	$N(0.7, 0.01)$ and 1
Data update parameters: α, β, ϑ, ∂^T, and ∂^S	1, 100, 1, 0.015, and 0.04

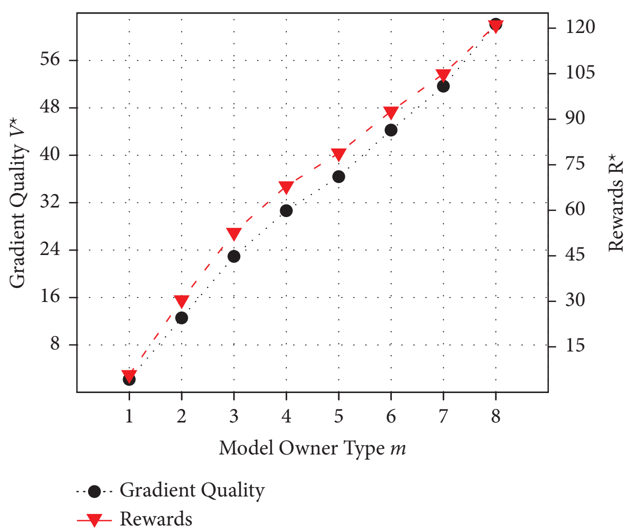

Figure 2: Gradient quality vs. model owner types.

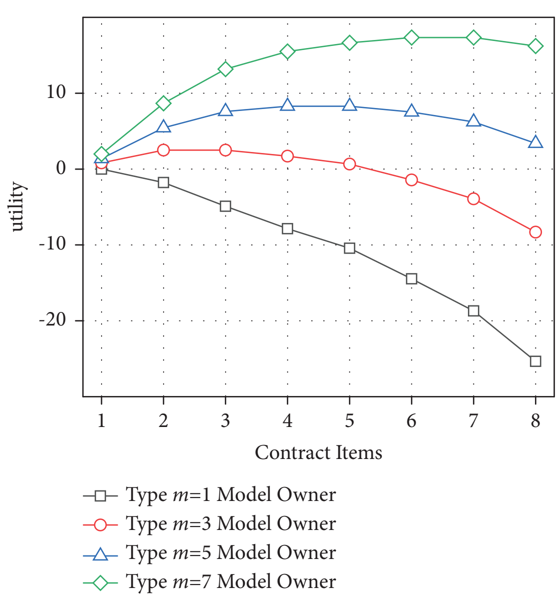

Figure 3: Model owner utility vs. contract items.

while the CS scenario aims to maximize social welfare. We observe from Figure 8 that the utility of model owners cannot always be maximized under the single contract in CC. On the other hand, the single contract in CS yields similar results to the dual contract in terms of model owner utility, which is consistently lower than that of the proposed scheme. We attribute this difference in

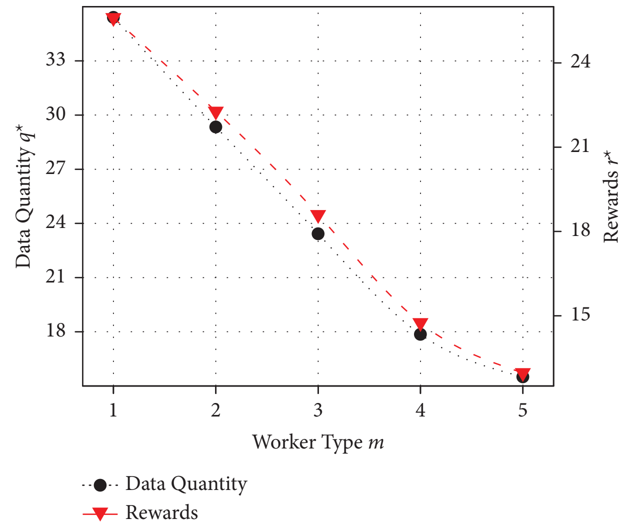

Figure 4: Data quantity vs. worker types.

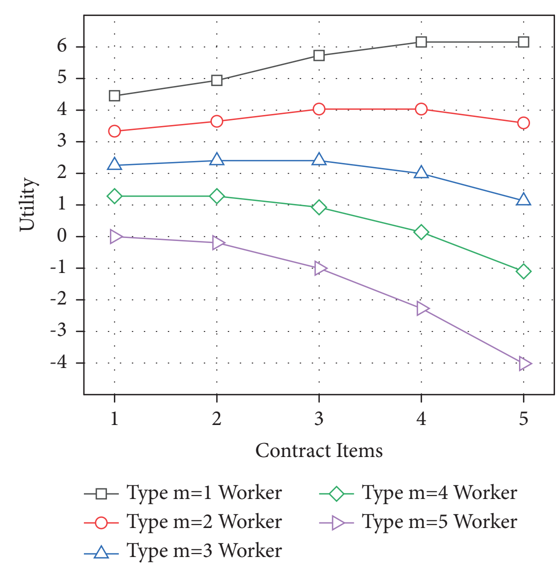

Figure 5: Stage 1 worker utility vs. contract items.

performance to the second stage of the proposed contract, where the average data update cost is smaller compared to that of the CS and CC scenarios. This leads to a better model owner utility under the proposed scheme compared to the single contract in CS and CC. Furthermore, the proposed contract also outperforms the single contract in CA, as the effect of the second stage contract improves its overall performance.

Overall, our results highlight the importance of considering a multistage contract in incentivizing workers for collaborative data updates.

5.3. Managing the AoI-Service Latency Trade-Off. In practice, a model owner may have different preferences for varying tasks. We vary the weights z_a and z_b within the range [0.1, 0.9] to study the changes in data quantity and data update frequency when the model owner preferences vary.

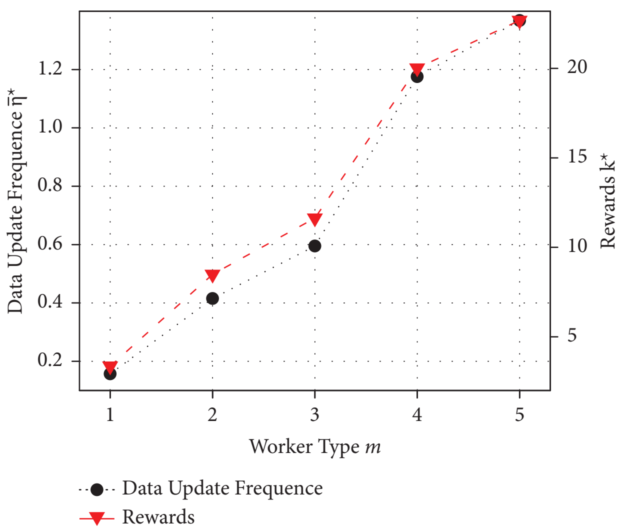

Figure 6: Data update frequency vs. worker types.

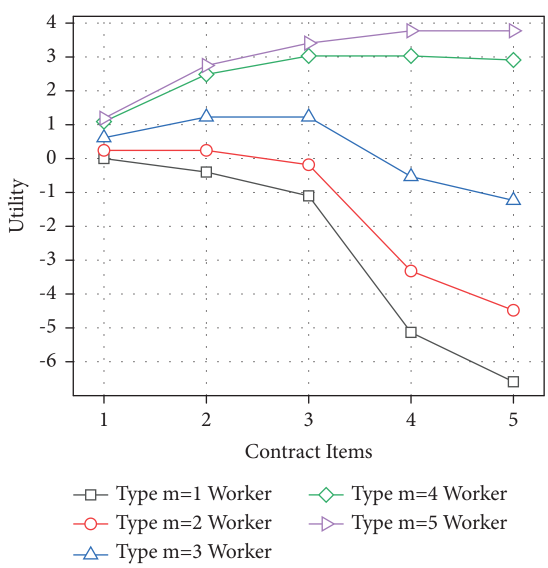

Figure 7: Stage 2 worker utility vs. contract items.

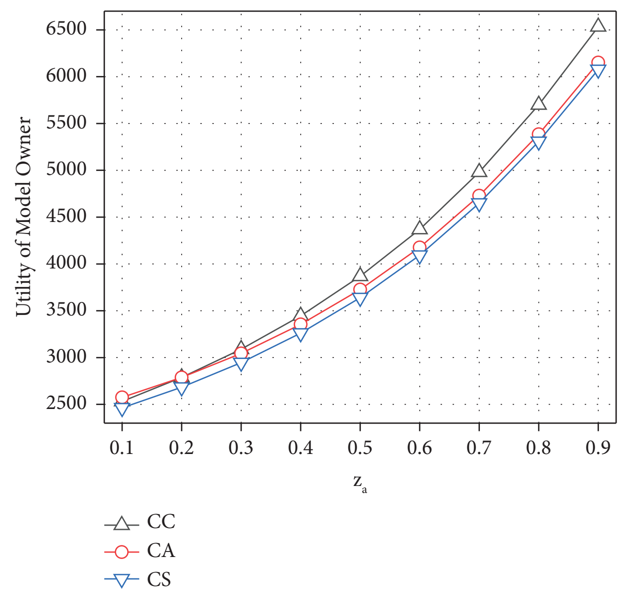

Figure 8: Utility of model owner.

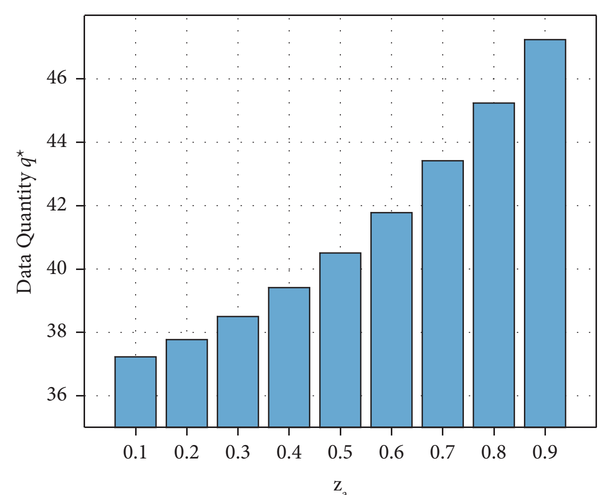

Figure 9: Data quantity for different preferences.

Figure 9 depicts the changes in the number of data quantities as the preference towards z_a varies. As expected, data quantities and the growth rate increase as z_a increases. Figure 10 depicts the changes in data update frequency as the preference towards z_a varies. As expected, the number of data update frequencies decreases, and the rate of decline increases as z_a increases.

5.4. Impact of Worker Types. Figures 11–13 depict the system performance concerning z_a under a different number of worker types. When the number of z_a increases, both the model owner and the workers obtain higher utilities. Because the collected quality of data improves, model owners can obtain more utility for training the model and gaining more rewards. Thus, social welfare is also improved. Also, we found when the number of worker types increases, the utility of the model owner decreases but the utilities of workers increase. The reason is that when the number of worker types increases, it becomes more difficult for the model owner to mine the information of the worker type and design the corresponding contract. Therefore, the workers can extract more rewards from the model owner.

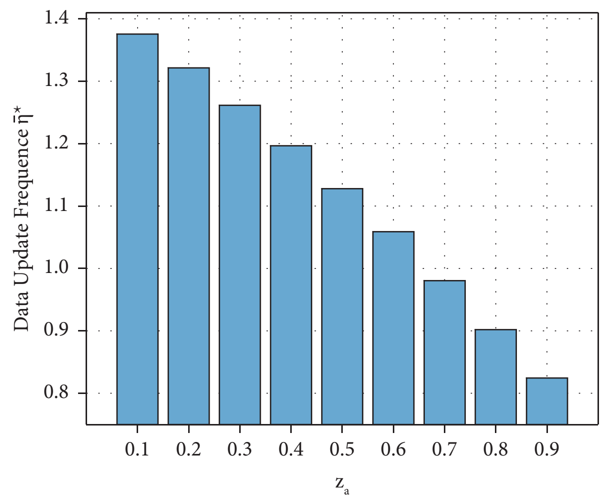

Figure 10: Data update frequency for different preferences.

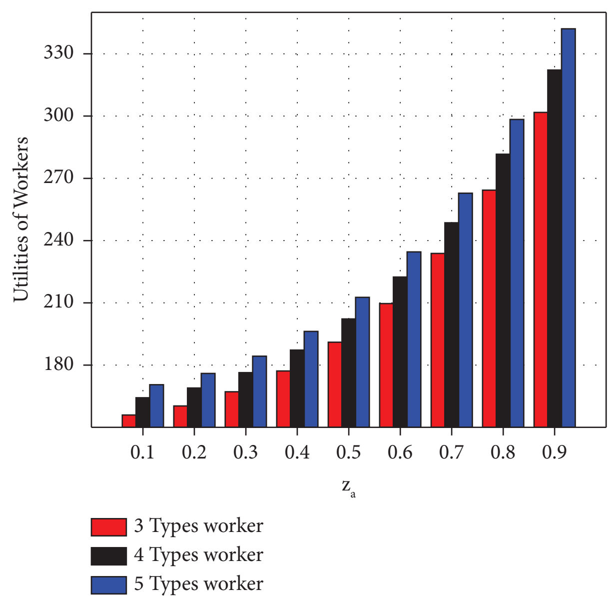

Figure 11: Utility of workers under different worker types.

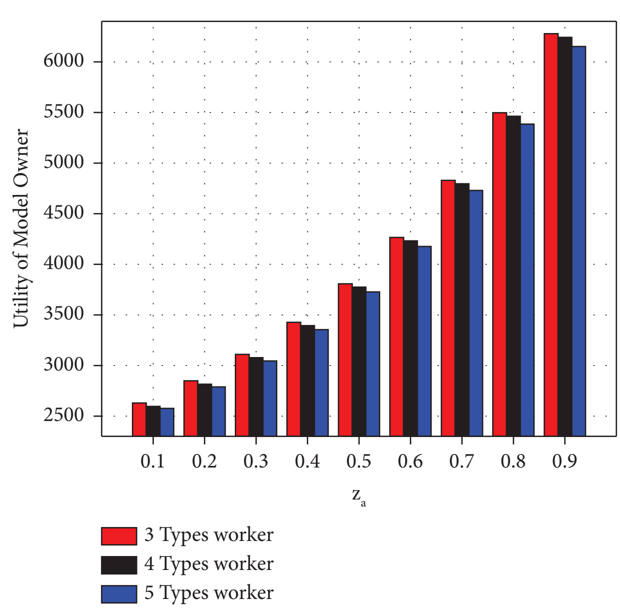

Figure 12: Utility of model owners under different worker types.

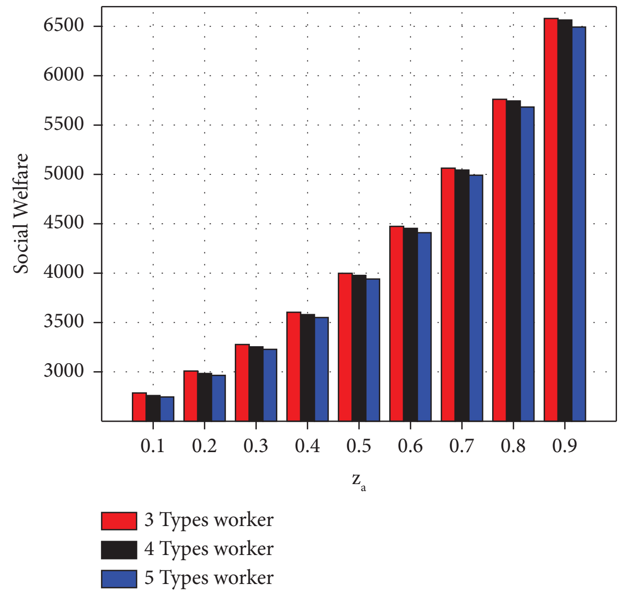

Figure 13: Social welfare under different worker types.

6. Conclusion

This paper presents a hierarchical incentive mechanism for an FL-based system with caching where the models trained by the workers are all based on their latest data and investigate the trade-off between data quantity and data update frequency. Specifically, we design a single contract to a dual contract based on the model owner's willingness to participate and the gradient quality that the worker provides. Our proposed mechanism uses contract theory to incentivize high-quality gradient updates from different types of workers.

As a future research direction, we can explore superior incentive mechanisms to improve FL efficiency. Furthermore, by considering worker data quality in more aspects and practical conditions limitations, this may involve incorporating other metrics, such as data completeness or relevance or data correlation between different people into the incentive mechanism design.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Nature Science Foundation of China (No. 61962022 and No. 62062034).

References

[1] K. Yu, L. Tan, M. Aloqaily, H. Yang, and Y. Jararweh, "Blockchain-enhanced data sharing with traceable and direct revocation in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7669–7678, 2021.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[3] Z. Zhou, C. Zhang, C. Xu, F. Xiong, Y. Zhang, and T. Umer, "Energy-efficient industrial internet of uavs for power line inspection in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2705–2714, 2018.

[4] Z. Guo, Y. Zhang, X. Zhao, and X. Song, "Cps-based self-adaptive collaborative control for smart production-logistics systems," *IEEE Transactions on Cybernetics*, vol. 51, no. 1, pp. 188–198, 2021.

[5] F. Al-Turjman and S. Alturjman, "Context-sensitive access in industrial internet of things (IIoT) healthcare applications," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2736–2744, 2018.

[6] S. Liu, C. Guo, F. Al-Turjman, K. Muhammad, and V. H. C. de Albuquerque, "Reliability of response region: a novel mechanism in visual tracking by edge computing for IIoT environments," *Mechanical Systems and Signal Processing*, vol. 138, Article ID 106537, 2020.

[7] Z. A. E. Houda, B. Brik, A. Ksentini, L. Khoukhi, and M. Guizani, "When federated learning meets game theory: a cooperative framework to secure IIoT applications on edge computing," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 7988–7997, 2022.

[8] X. Feng, J. Wu, Y. Wu, J. Li, and W. Yang, "Blockchain and digital twin empowered trustworthy self-healing for edge-AI enabled industrial internet of things," *Information Sciences*, vol. 642, Article ID 119169, 2023.

[9] M. A. Rahman, M. S. Hossain, A. J. Showail, N. A. Alrajeh, and A. Ghoneim, "AI-enabled IIoT for live smart city event monitoring," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 2872–2880, 2023.

[10] B. Farahani and A. K. Monsefi, "Smart and collaborative industrial IoT: a federated learning and data space approach," *Digital Communications and Networks*, vol. 9, no. 2, pp. 436–447, 2023.

[11] P. Voigt and A. Von dem Bussche, *The Eu General Data protection Regulation (Gdpr). A Practical Guide*, Springer International Publishing, Cham, 2017.

[12] P. Kairouz, H. B. McMahan, B. Avent et al., "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[13] B. Yang, X. Cao, X. Li, Q. Zhang, and L. Qian, "Mobile-edge-computing-based hierarchical machine learning tasks distribution for IIoT," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2169–2180, 2020.

[14] L. D. Xu, W. He, and S. Li, "Internet of things in industries: a survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

[15] R. Chaudhary, G. S. Aujla, S. Garg, N. Kumar, and J. J. Rodrigues, "Sdn-enabled multi-attribute-based secure communication for smart grid in IIoT environment," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2629–2640, 2018.

[16] R. Y. Zhong, C. Xu, C. Chen, and G. Q. Huang, "Big data analytics for physical internet-based intelligent manufacturing shop floors," *International Journal of Production Research*, vol. 55, no. 9, pp. 2610–2621, 2017.

[17] T. Qu, S. Lei, Z. Wang, D. Nie, X. Chen, and G. Q. Huang, "IoT-based real-time production logistics synchronization system under smart cloud manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 84, no. 1-4, pp. 147–164, 2016.

[18] Y. Miao, Z. Liu, H. Li, K. K. R. Choo, and R. H. Deng, "Privacy-preserving byzantine-robust federated learning via blockchain systems," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2848–2861, 2022.

[19] Y. Miao, R. Xie, X. Li, X. Liu, Z. Ma, and R. H. Deng, "Compressed federated learning based on adaptive local differential privacy," in *Proceedings of the 38th Annual Computer Security Applications Conference*, Austin, TX, USA, December 2022.

[20] I. Yaqoob, L. U. Khan, S. M. A. Kazmi, M. Imran, N. Guizani, and C. S. Hong, "Autonomous driving cars in smart cities: recent advances, requirements, and challenges," *IEEE Network*, vol. 34, no. 1, pp. 174–181, 2020.

[21] U. M. Aïvodji, S. Gambs, and A. Martin, "Iotfla: a secured and privacy-preserving smart home architecture implementing federated learning," in *Proceedings of the 2019 IEEE Security and Privacy Workshops (SPW)*, IEEE, San Francisco, CA, USA, May 2019.

[22] J. Li, Y. Meng, L. Ma et al., "A federated learning based privacy-preserving smart healthcare system," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 2021–2031, 2022.

[23] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. Shen, "Energy efficient dynamic offloading in mobile edge computing for internet of things," *IEEE Transactions on Cloud Computing*, vol. 9, no. 3, pp. 1050–1060, 2021.

[24] Y. Yu, S. Liu, P. L. Yeoh, B. Vucetic, and Y. Li, "Layerchain: a hierarchical edge-cloud blockchain for large-scale low-delay industrial internet of things applications," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 5077–5086, 2021.

[25] Y. He, M. Luo, B. Wu et al., "A game theory-based incentive mechanism for collaborative security of federated learning in energy blockchain environment," *IEEE Internet of Things Journal*, vol. 10, no. 24, pp. 21294–21308, 2023.

[26] Y. Chen, H. Zhou, T. Li, J. Li, and H. Zhou, "Multi-factor incentive mechanism for federated learning in IoT: a stackelberg game approach," *IEEE Internet of Things Journal*, vol. 10, no. 24, pp. 21595–21606, 2023.

[27] T. H. Thi Le, N. H. Tran, Y. K. Tun et al., "An incentive mechanism for federated learning in wireless cellular networks: an auction approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 8, pp. 4874–4887, 2021.

[28] J. Zhang, Y. Wu, and R. Pan, "Incentive mechanism for horizontal federated learning based on reputation and reverse auction," in *Proceedings of the Web Conference 2021*, Ljubljana Slovenia, April 2021.

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

[30] C. Xu, J. Ge, Y. Li et al., "SCEI: a smart-contract driven edge intelligence framework for IoT systems," *IEEE Transactions on Mobile Computing*.

[31] W. Y. B. Lim, J. Huang, Z. Xiong et al., "Towards federated learning in uav-enabled internet of vehicles: a multi-dimensional contract-matching approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 5140–5154, 2021.

[32] M. Tian, Y. Chen, Y. Liu, Z. Xiong, C. Leung, and C. Miao, "A contract theory based incentive mechanism for federated learning," 2021, https://arxiv.org/abs/2108.05568.

[33] M. Wu, D. Ye, J. Ding, Y. Guo, R. Yu, and M. Pan, "Incentivizing differentially private federated learning: a multi-dimensional contract approach," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10639–10651, 2021.

[34] W. Y. B. Lim, S. Garg, Z. Xiong et al., "Dynamic contract design for federated learning in smart healthcare applications," *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 16853–16862, 2021.

[35] B. Li, Y. Shi, Q. Kong, Q. Du, and R. Lu, "Incentive-based federated learning for digital twin driven industrial mobile crowdsensing," *IEEE Internet of Things Journal*, vol. 10, no. 20, pp. 17851–17864, 2023.

[36] J. Chen, S. Guo, T. Shen, Y. Feng, J. Gao, and X. Qiu, "IncEFL: a sharing incentive mechanism for edge-assisted federated learning in industrial IoT," *Digital Communications and Networks*, keai publishing, Beijing, China, 2023.

[37] Z. Feng, S. Yu, and Y. Zhu, "Towards personalized privacy preference aware data trading: a contract theory based approach," *Computer Networks*, vol. 224, Article ID 109637, 2023.

[38] W. Y. B. Lim, Z. Xiong, J. Kang et al., "When information freshness meets service latency in federated learning: a task-aware incentive scheme for smart industries," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 1, pp. 457–466, 2022.

[39] W. Y. B. Lim, Z. Xiong, C. Miao et al., "Hierarchical incentive mechanism design for federated machine learning in mobile networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9575–9588, 2020.

[40] X. Xia, F. Chen, Q. He, J. Grundy, M. Abdelrazek, and H. Jin, "Online collaborative data caching in edge computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 2, pp. 281–294, 2021.