Game-Theoretic Incentive Mechanism for Blockchain-Based Federated Learning

Wenzheng Tang, Erwu Liu, Wei Ni, *Fellow, IEEE*, Xinyu Qu, Butian Huang, Kezhi Li, Dusit Niyato, *Fellow, IEEE*, and Abbas Jamalipour, *Fellow, IEEE*

***Abstract*—Blockchain-based federated learning (BFL) has gained attention for its potential to establish decentralized trust. While existing research primarily focuses on personalized frameworks for various applications, essential aspects including incentive mechanisms—critical for ensuring stable system operation—remain under-explored. To bridge this gap, we propose a game-theoretic incentive mechanism designed to foster active participation in BFL tasks. Specifically, we model a BFL system comprising a model owner (MO), i.e., task publisher, multiple miners, and training terminals, framing their interactions through two-tier Stackelberg games. In the first-tier game, the MO designs reward strategies to incentivize training terminals to contribute more data, enhancing model accuracy. The second-tier game introduces a multi-leader multi-follower Stackelberg game, enabling miners to set model packaging prices based on competitors' strategies and anticipated user behavior. By deriving the Stackelberg equilibrium, we identify optimal strategies for all participants, leading to an incentive mechanism balancing individual interests with overall performance. Compared to its benchmarks, our incentive mechanism offers 5.8% and 53.4% higher utilities in the two games compared to its alternatives, accelerating convergence and improving accuracy.**

***Index Terms*—Federated learning, blockchain, Stackelberg game, incentive mechanism, multi-leader multi-follower game.**

I. Introduction

The Internet of Things (IoT) not only generates immense volumes of data but also offers substantial distributed computing and storage resources [1]. Federated Learning (FL), a technology for distributed training, has emerged to enable devices (e.g., mobile phones or vehicles) to collaboratively train a model with their data remaining local [2], [3], thus offering privacy protection for the participating devices. FL typically allows the participation of many users with diverse identity backgrounds and complex behavioral patterns, posing challenges in establishing trust among participants. Dishonest participants and vulnerable central servers may affect the security of the global model [4], [5]. Participants may also be malicious or vulnerable, leading to the leakage or tampering of transmitted information [6]–[8].

Researchers have resorted to blockchain technology to enhance FL, with blockchains commonly functioning as decentralized platforms for offering incentives and verifying data [9]–[13]. The combination of blockchain and FL is called blockchain-based FL (BFL) [14], [15]. In BFL, miners validate the model updates submitted by training terminals, prior to executing global aggregation algorithms. After obtaining the global model, it is uploaded to the main chain, which is accessible to all eligible participants [16]. Although BFL can tackle some of the challenges faced by traditional FL, there are still issues that require attention [17]–[19].

A. Challenges

One critical issue in BFL is the incentive mechanisms [5], [6], [9]. BFL encompasses multiple participants, including model owner (MO), miners, and training terminals, whose interests are intricately intertwined. Designing an incentive mechanism that balances the interests of all participants and encourages their participation is a challenging problem. BFL often operates in a dynamic environment where the behavior and strategies of participants change over time. Fluctuating market conditions and varying contributions of the participants can render static incentives ineffective, leading to underparticipation or overrewarding. The incentive mechanisms must possess flexibility, guaranteeing the sustained and stable operation of the system over the long term.

There has been few studies on the incentive mechanisms for BFL [9], [20], [21], most of which were based on two unrealistic assumptions: (a) the data volume and training costs of all terminals are the same, and (b) the incentive mechanism is static, without considering the potential impact of market conditions on the participants' behavior and strategies. In practice, the participants often flexibly adjust their behaviors based on various factors, e.g., market supply and demand, and competitive situation. There is a need for an incentive mechanism that can dynamically adapt to market changes.

B. Contribution

In this paper, we design an incentive mechanism for BFL to achieve long-term system stability. In BFL systems with

W. Tang, E. Liu, and X. Qu are with the College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China. E. Liu is also with the Department of Ophthalmology, Tongji Hospital, School of Medicine, Tongji University (e-mail: 2232962@tongji.edu.cn, erwu.liu@ieee.org, xinyuqu@tongji.edu.cn).

W. Ni is with the School of Electrical and Data Engineering, University of Technology Sydney, NSW 2007, Australia (e-mail: wei.ni@ieee.org).

B. Huang is with Hangzhou Yunphant Network Technology Co. Ltd., Hangzhou, China (e-mail: hbt@yunphant.com).

K. Li is with the Institute of Health Informatics, University College London, London WC1E 6BT, UK (e-mail: ken.li@ucl.ac.uk).

D. Niyato is with the College of Computing and Data Science, Singapore 639798 (e-mail: dniyato@ntu.edu.sg).

A. Jamalipour is with the School of Electrical and Computer Engineering, University of Sydney, NSW 2006, Australia (e-mail: a.jamalipour@ieee.org)

This work is supported in part by grants from the National Natural Science Foundation of China (No. 42171404) and Shanghai Engineering Research Center for Blockchain Applications And Services (No. 19DZ2255100). Corresponding authors: Erwu Liu, Xinyu Qu.

TABLE I
NOTATION AND DEFINITION

Notation	Definition
ω	The system parameter of MO
a_i, b	The parameters of the linear-quadratic function
c_i	The computation cost for each data sample unit
c_j	The unit packaging cost
d_i	The training data size of terminal i
$\mathcal{I}$	The set of training terminals
I	The number of training terminals
$\mathcal{N}$	The set of miners
N	The number of miners
p_j	The price set by miner j for packaging models
r	The total reward given by MO
x_{ij}	The number of packaged models from terminal i to miner j

heterogeneous terminals, an MO delegates tasks to terminals and miners, who receive rewards based on their contributions. We model the reward allocation problem as two-tier Stackelberg games, and derive a unique equilibrium rigorously under complete information conditions. Among them, the second-tier Stackelberg game model specifically considers the multi-leader multi-follower (MLMF) relationship between terminals and miners, aiming to capture the intricate interactions between these two entities. Hence, our incentive mechanism effectively promotes the long-term stability and efficiency of BFL.

The contributions of this paper are summarized as follows:

- We formulate a Stackelberg game between an MO and terminals to help the MO determine how many rewards to allocate to each terminal for model training, and assist terminals in deciding the appropriate amount of data to allocate to each subtask, thereby optimizing their utility.
- We model the relationships between miners and terminals as an MLMF Stackelberg game. In this game, the miners act as the leaders, and the terminals act as the followers, jointly achieving the profit maximization of the miners and utility maximization of the terminals.
- We provide numerical results to analyze the complex strategic interactions among participants in the BFL system, and validate the effectiveness and reliability of the proposed algorithm. Compared to several baselines, our incentive mechanism offers 5.8% and 53.4% higher utilities in two games. It also accelerates model convergence and improves accuracy.

The rest of this paper is structured as follows. Section II presents an overview of the related literature. Section III describes the system model and formulates the problem as a two-tier Stackelberg game framework. Sections V and VI detail the models and solutions for the two-tier Stackelberg games, respectively. Section VII presents the experimental evaluations. Finally, we conclude the paper in Section VIII. The key notation is provided in Table I.

II. Related Work

A. Research on BFL

Although BFL technology is still in its early stages of development with limited literature research, the BFL architecture is receiving increasing attention and is in a state of continuous development. For instance, the authors in [22] discussed how blockchain technology can effectively address the issues present in FL, and explores the feasibility and potential advantages of integrating blockchain technology into FL. The authors in [20] demonstrated the enormous potential of integrating blockchain technology with FL by proposing the BLADE-FL framework and showcasing its performance advantages. The analysis and optimization methods presented in the article provide valuable references and insights for other researchers exploring the integration of BFL systems.

The types and platforms of blockchain used in some existing BFL research work, as well as a comparison of BFL frameworks, are introduced in [23], [24]. The authors in [25], [26] offered an overview of BFL, highlighted the limitations of FL, and investigated it from perspectives such as architectural features and resource allocation. It also outlined the future potential of BFL in AI. Likewise, The authors in [27] addressed the issues and deficiencies in current FL mechanisms and elaborated on the potential enhancements through the integration of blockchain technology with FL.

These works have primarily focused on BFL frameworks and their application prospects in the field of AI, yet they have not provided a comprehensive examination of the core challenges tackled by BFL.

B. Incentive Mechanism for BFL

Some research on BFL centers on designing incentive mechanisms to regulate client behavior, thereby motivating them to act honestly and efficiently in accordance with established rules. The incentive mechanisms proposed in [18], [28], include two aspects: data rewards and mining rewards. The data rewards of training terminals are received from their corresponding miners, and the amount of the rewards is proportional to the size of their data samples. After the miners complete model aggregation and generate blocks, they can receive mining rewards from the blockchain network. The mining rewards are proportional to the number of aggregated data samples contributed by the training terminals to which the miners are assigned.

In [29], the Federated Reputation Evaluation Blockchain (FREB) framework was proposed for participant selection in FL. It integrates reputation evaluation with blockchain technology to ensure secure and reliable participant selection based on factors such as model contribution, activity, data quality, and stability. The authors in [5] proposed an incentive mechanism for cross-silo FL that addresses the free-rider attack and ensures social efficiency, individual rationality, and budget balance without relying on private information of organizations. The authors in [30] developed an incentive mechanism utilizing smart contracts based on the volume and centroid distance of customer data during local model training. The authors in [9] presented BIT-FL, a blockchain-enabled incentivized and secure FL framework. BIT-FL leverages a loop-based sharded consensus algorithm to accelerate model validation, and integrates a randomized incentive mechanism to attract participants while preserving cost privacy. Moreover, the authors in [31] introduced a decentralized and incentivized

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3567355

FL framework enabled by blockchain technology. The framework utilizes compressed soft-labels and Peer Truth Serum for Federated Distillation (PTSFD) to incentivize honest participation and evaluate contributions.

In [21] and [32], incentive mechanisms were designed under a fully coupled BFL system model in which the FL clients simultaneously serve as a blockchain node, participating in training and mining. The authors in [21] solved for the optimal strategies in both complete and incomplete information scenarios, and implemented them by solving two optimization problems. The authors in [32] proposed a Long-Term Proof-of-Contribution (LPoC) algorithm for BFL, which selects block producers and allocates rewards by considering the long-term contributions of clients in both model training and blockchain consensus processes. Unfortunately, fully coupled BFL system models may adapt to different FL scenarios or blockchain platforms, limiting their practical applicability and scalability.

Most existing research either assumes unrealistically that all terminals have the same data volume and training costs, or treats the incentive mechanism as static, disregarding the impact of market dynamics on participant behavior. To tackle this issue, we design an incentive mechanism for the BFL system, which can fairly distribute rewards and dynamically adapt to market changes.

III. System Model

A. System Overview

Fig. 1 depicts the considered BFL framework, where there are three types of participants: An MO, a set of training terminals, and a set of miners [14].

The MO publishes FL tasks to obtain well-trained final global models from the BFL system. The training terminals participate in the FL tasks in an attempt to receive rewards from the MO. The set of training terminals is denoted as $\mathcal{I} = \{1, \dots, I\}$ with I being the total number of training terminals in the considered BFL system.

Assume that these I terminals each maintain a local dataset, denoted by $\mathcal{D}_i$. Define $d_i \triangleq |\mathcal{D}_i|$, where $|\cdot|$ denotes cardinality of the set $\mathcal{D}_i$. Each terminal downloads the global model, denoted as θ, and proceeds to train it locally. The terminals employ their local data to train their local models and then forward the local model updates to the miners responsible for maintaining the blockchain. The loss function of terminal i with the dataset is

$$F_i(\theta) = \frac{1}{d_i} \sum_{k \in \mathcal{D}_i} f_k(\theta), \tag{1}$$

where $f_k(\theta)$ is the loss function on the k-th data sample of $\mathcal{D}_i$.

The goal of the BFL system is to optimize the global loss function $F(\theta)$ by aggregating the local models, e.g., using FedAvg [33], as given by

$$F(\theta) = \sum_{i \in \mathcal{I}} \frac{d_i F_i(\theta)}{\sum\limits_{i \in \mathcal{I}} d_i}, \tag{2}$$

with the optimal model $\theta^* = \arg\min F(\theta)$ obtained upon the convergence of BFL training.

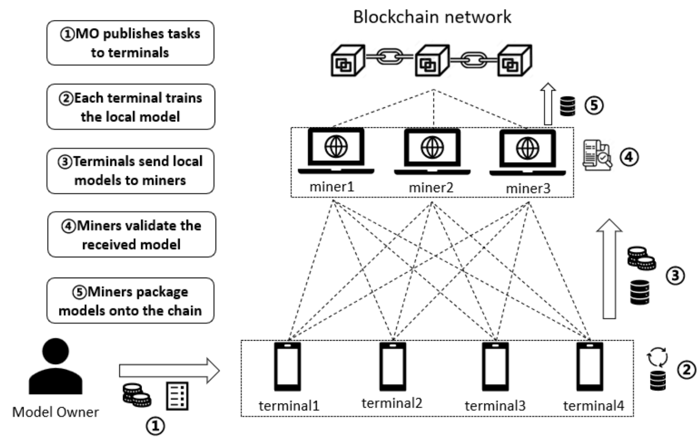

Fig. 1. The architecture of the considered BFL system, which has three participants: an MO, training terminals, and miners. Once receiving the FL task, each terminal trains the local model and then broadcasts the model updates to the miners. Miners validate these models and execute on-chain operations through consensus mechanisms. Once all selected miners finalize their on-chain models, the system aggregates them into a new model.

There are N miners in the blockchain networks. Unlike traditional blockchain miners who primarily validate transactions and maintain the ledger, the miners in the BFL system also validate and encapsulate the local models uploaded by the training terminals into blocks in pursuit of rewards. The miners perform cross-validation and model aggregation, and generate a consistent global model based on a given consensus mechanism. The global model is stored and propagated in the blockchain, allowing the training terminals to download the consistent global model from the blockchain for the next round of training. The set of miners is denoted as $\mathcal{N} = \{1, \dots, N\}$.

It is noted that the considered BFL system and the proposed incentive mechanism are flexible and can be adapted to various consensus mechanisms, such as Proof-of-Stake (PoS), Delegated Proof-of-Stake (DPoS), or Byzantine Fault Tolerance (BFT), as they focus primarily on the interactions between the MO, miners, and training terminals, rather than being tied to a specific consensus protocol.

We outline the workflow of the considered BFL system as follows: a) Upon receiving the FL task, each training terminal trains its local model and then broadcasts the model updates to the miners; b) The miners validate the received local models and perform on-chain operations on the validated local models through consensus mechanisms; c) When all selected miners have completed the model on the chain, the system aggregates these local models to update the global model.

This BFL system is compatible with typical blockchains, including public blockchain (e.g., Ethereum) and private blockchains. In each training cycle of the BFL system, the terminals send their updated models to their pre-assigned miners. These miners only validate models submitted by the terminals that explicitly entrust them, avoiding the situation where multiple miners in the transaction pool repeatedly validate the same model.

In this system, incentive mechanisms are crucial as they motivate the miners and the training terminals to actively

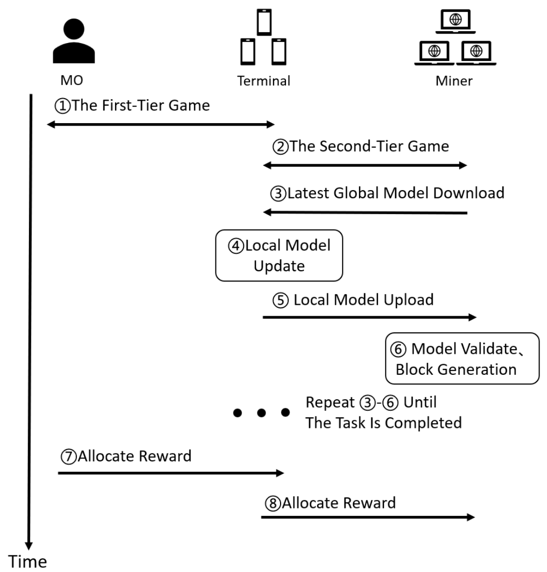

Fig. 2. The timeline of the considered BFL system

participate. By introducing an incentive mechanism, the training terminals can be incentivized to contribute more data and actively participate in training. Moreover, the miners are responsible for validating the legitimacy of the model in the system and packaging it into blocks, which requires a significant amount of computing resources and time. Incentive mechanisms can compensate the miners for their costs and motivate them to continue and actively participate in the validation and packaging of local models.

We design a game-theoretic incentive mechanism for the BFL system, which is structured into two tiers. In the first tier, a game is conducted between the MO and the training terminals, where the MO determines the pricing and the training terminals earn rewards through training. In the second tier, we design an MLMF game model to assist the miners in earning rewards through model validation and packaging, while helping the training terminals select suitable miners to upload their local models to the blockchain. After the two-tier game, each participant starts its training as agreed until the task is completed. The timeline of the considered BFL system is illustrated in Fig. 2. Some participants may deviate from the agreement for various reasons, impacting the system's performance and stability. (For further discussion on potential deviations and their implications, see [34].) The rest of this section defines the utility functions of the MO, training terminals, and miners.

B. Utility Model between MO and Terminals

In this tier of the proposed game-theoretic incentive mechanism, the MO serves as the leader of the game, and all training terminals serve as the followers. The MO encourages the terminals to contribute more data to the model training and hence obtain a more accurate global model.

1) Utility of Training Terminals: The utility of a training terminal consists of revenue and cost. A training terminal receives rewards from the MO by participating in model training. The training terminal also incurs the training cost arising from model training, which is directly proportional to the amount of data used for training. Let U_i^1 denote the utility of training terminal $i \in \mathcal{I} = \{1, \ldots, I\}$. It can be defined as

$$U_i^1(d_i) = \frac{d_i}{\sum\limits_{k\in\mathcal{I}} d_k} r - c_i d_i, \tag{3}$$

where d_i is the data size provided by user i; r is the overall incentive provided by the MO to all participating training terminals; c_i is the computation cost for training a data sample.

2) Utility of MO: Let U_{MO} denote the utility of the MO, which is the gain of global model accuracy subtracted by the total reward paid to the training terminals. The model of utility maximization was first introduced in microeconomics [35] for price policy. Since then, the idea of using concave utility has been considered extensively in the literature [36]–[40].

According to [40], the test accuracy of a training model can be regarded as a concave function with respect to the quantity of training data. The accuracy of the model, as a concave function of the training data volume, reflects the diminishing marginal benefits of model performance improvement as the training data size increases. This is a critical factor in the MO's utility function, as it influences the MO's ability to strike a balance between enhancing accuracy and managing costs. To this end, we specify the utility of the MO as

$$U_{MO}(r) = \omega G(\sum_{k\in\mathcal{I}} d_k) - r, \tag{4}$$

where $\omega > 0$ is a configurable system parameter, and $G(\cdot)$ is a concave function indicating the test accuracy. When ω is large, the MO prioritizes test accuracy and is willing to incur higher costs. When ω is small, the MO prioritizes cost reduction over accuracy gains by increasing the relative weight of cost in (4).

C. Utility Model between Training Terminals and Miners

After completing local training, the training terminals submit their local model updates to the miners to start the second tier of the proposed game-theoretic incentive mechanism for BFL. The miners are responsible for validating and packaging these models before uploading them to the blockchain. This validation and packaging process incurs costs for the miners. To compensate for these costs, the training terminals pay fees for utilizing the miners' resources. These fees are the primary source of income for the miners.

1) Utility of Miners: The miners who win the consensus mechanism gain the privilege of recording these models onto the blockchain and are authorized to charge a fee for each local model they package. However, the miners must also account for operational and maintenance costs, such as computational costs. Consequently, the profit for each miner can be formulated as

$$U_j^n(p_j) = \sum_{i\in\mathcal{I}} p_j x_{ij} - c_j \sum_{i\in\mathcal{I}} x_{ij}, \tag{5}$$

where x_{ij} is the number of local models that training terminal i opts to have packaged by miner j. p_j is the price set by each miner j for the packaging of local models, and c_j indicates the unit packaging cost incurred by miner j.

2) Utility of Training Terminals: The training terminals can reward the miners based on their performance in packaging and uploading local models to the blockchain, thereby incentivizing their participation in completing the tasks of BFL. Typically, the contribution of training terminals is determined by the number of local models successfully packaged into blocks and uploaded onto the blockchain. Consequently, the profit for each training terminal can be formulated as

$$U_i^2(x_{ij}) = \sum_{j\in\mathcal{N}} (a_i x_{ij} - b_i x_{ij}^2) - \sum_{j\in\mathcal{N}} p_j x_{ij}. \tag{6}$$

The first term on the right-hand side (RHS) of (6) represents the internal benefits that training terminal i gains from participating in FL tasks. These benefits can be modeled as a concave linear-quadratic function, $\sum\limits_{j\in\mathcal{N}} (a_i x_{ij} - b_i x_{ij}^2)$. This modeling captures the increase in benefits from participation while accounting for diminishing marginal returns, as described in [41]. The choice of a linear-quadratic function not only facilitates analysis but also serves as a close second-order approximation for a wide range of concave utility functions, effectively reflecting the diminishing marginal returns commonly observed in such settings.

IV. Incentive Mechanism via Two-Tier Stackelberg Game: Overview

In this paper, we design a new incentive mechanism for BFL as a two-tier Stackelberg game [42], [43] to align the interests of the MO, training terminals, and miners. The two-tier Stackelberg game framework is chosen for its ability to model the hierarchical leader-follower dynamics in BFL systems. This framework inherently captures the asymmetric decision-making structure and supports dynamic adjustments.

The proposed incentive mechanism addresses interest distribution at different levels through two interconnected Stackelberg games. Within each tier of the interconnected Stackelberg games, there are two distinct stages: The first stage where the leader sets the price, and the second stage where the followers respond with their decisions.

- In the first-tier game, the MO, as the leader, incentivizes training terminals by setting a unit price for data. This motivates the terminals to use high-quality and abundant data for model training, while ensuring sufficient returns for their efforts (Section V).
- In the second-tier game, the training terminals, using the earnings from the MO, compensate the miners for services, i.e., computational resources and model validation. This encourages the miners' participation (Section VI).

The profits earned by the training terminals in the first-tier game form the financial foundation for their participation in the second-tier game, creating a feedback loop that promotes a virtuous cycle within the BFL system. The synergy between these two-tier games effectively aligns the interests of the MO, training terminals, and miners.

While the training terminals contribute vast amounts of data, it is essential to ensure the authenticity and integrity of the data provided by these terminals. Some existing studies, e.g., [44], [45], have proposed various methods to address this. In BFL system, we can leverage the core characteristics of blockchain, including its distributed ledger and immutability, to effectively resolve issues of participant trust.

V. Game between MO And Terminals

This section focuses on the first-tier game between the MO and the training terminals. As the leader, the MO sets a unit price for the data, representing the fee it is willing to pay per unit. This price directly impacts the terminals' willingness to provide data. The MO anticipates the response of the training terminals to different price points, predicting how much data they are likely to supply based on the offered incentives.

The training terminals, acting as the followers, decide the amount of data to provide based on the cost of data preparation and the benefits offered by the MO. This direct incentive mechanism encourages the training terminals to actively participate in data provision while ensuring that the MO secures the required high-quality data at a reasonable cost.

The Stackelberg game model offers a strategic framework for coordinating the interactions between the MO and training terminals. The MO optimizes the unit price strategy by considering the terminals' responses, while the terminals adjust their decisions to maximize their benefits under the given prices. The model is adaptable to dynamic market conditions [46], [47]. For example, when new training terminals enter the market or task publishers revise their strategies, the optimal solution can be quickly recalculated using backward induction. This adaptability ensures the incentive mechanism remains effective and responsive to changing conditions.

A. Game Formulation

In the first stage, the MO aims to acquire a global model that delivers exceptional performance while minimizing both the time and cost invested in generating rewards. To this end, its main goal is to optimize the utility function U_{MO}. The associated optimization problem can be framed as

$$\mathbf{P1}: \max_{r\geq 0} \quad U_{MO}(r) = \omega G(\sum_{k\in\mathcal{I}} d_k) - r \tag{7a}$$

$$\text{s.t.} \quad r \geq 0, \tag{7b}$$

$$r \leq r^{max}, \tag{7c}$$

where (7b) and (7c) specify the pricing lower and upper bounds of the MO, respectively.

In the second stage, with the reward r in place, the training terminals decide on their best training approaches to maximizing their benefits. Consequently, an optimal training strategy can be derived by addressing the following issue:

$$\mathbf{P2}: \max_{d_i\geq 0} \quad U_i^1(d_i) = \frac{d_i}{\sum\limits_{k\in\mathcal{I}} d_k} r - c_i d_i \tag{8a}$$

$$\text{s.t.} \quad d_i \geq 0. \tag{8b}$$

The second stage can be viewed as a competitive game where each training terminal strives to optimize its own profit. With a given reward r and the training strategies d_{-i} of other terminals, training terminal i decides on the best strategy d_i

to maximize its benefit by weighing the income and expenses associated with model training.

Consequently, a stable state known as a Nash equilibrium arises, at which each player's strategy is optimal given the strategies of the other players, meaning no player has the incentive to deviate from their chosen strategy. The formal definition of the Nash equilibrium for the second-stage game is provided in Definition 1.

Definition 1 (Nash Equilibrium [48])**.** *A Nash equilibrium in a game is a state where players have no motivation to individually alter their strategies in order to achieve higher utility. A set of strategies* $d^* = (d_1^*, d_2^*, \ldots, d_I^*)$ *constitutes a Nash equilibrium, for every terminal* i*, it holds true that*

$$U_i^1(d_i^*, d_{-i}^*, r) \geq U_i^1(d_i, d_{-i}^*, r) \tag{9}$$

where r *is known a-priori.*

The above-mentioned two stages form a two-stage Stackelberg game involving the MO and the terminals. When the followers adopt their optimal reactions (i.e., the Nash equilibrium), the leader can achieve maximum utility. The solution to the game will be presented in the following section.

B. Analysis of MO-Terminal Game

We analyze the Stackelberg game between MO and the terminal and derived their optimal strategy. Specifically, we focus on the existence of a Nash equilibrium for the game.

Theorem 1. *For a specific* r *set by the MO, a Nash equilibrium exists, along with an optimal training strategy for each training terminal* $i \in \mathcal{I} = \{1, \ldots, I\}$ *is given by*

$$d_i^* = \frac{(I-1)r}{\sum\limits_{k\in\mathcal{I}} c_k}\Big(1 - \frac{(I-1)c_i}{\sum\limits_{k\in\mathcal{I}} c_k}\Big). \tag{10}$$

Proof: To demonstrate the presence of a Nash equilibrium, we start with the following lemma from [48].

Lemma 1. *A Nash equilibrium exists in a game when the following conditions are satisfied:*

1) The number of players is finite.

2) The sets of strategies are compact, confined, and convex.

3) The utility functions are continuous and exhibit quasi-concavity within the strategy space.

According to **Lemma 1**, we first verify that the first-order derivative of $U_i^1(d_i)$ with respect to d_i is positive, and the second-order derivative is persistently negative. Specifically, the first-order derivative of $U_i^1(d_i)$ is given by

$$\frac{\partial U_i^1(d_i)}{\partial d_i} = \frac{\sum\limits_{k\in\mathcal{I}^-} d_k}{(\sum\limits_{k\in\mathcal{I}} d_k)^2} r - c_i, \tag{11}$$

and the second-order derivative of $U_i(d_i)$ is given by

$$\frac{\partial^2 U_i^1(d_i)}{\partial d_i^2} = -2\frac{\sum\limits_{k\in\mathcal{I}^-} d_k}{(\sum\limits_{k\in\mathcal{I}} d_k)^3} r < 0, \tag{12}$$

where $i \in \mathcal{I}^- = \{1, \ldots, i-1, i+1, \ldots, I\}$.

According to (12) and the finite set of training terminals, we can confirm the existence of a Nash equilibrium in the second stage based on **Lemma 1**. By setting the first-order derivative to zero, it follows that

$$d_i^* = \sqrt{\frac{r}{c_i}\sum_{k\in\mathcal{I}^-} d_k} - \sum_{k\in\mathcal{I}^-} d_k. \tag{13}$$

According to the translation of (13), we can obtain

$$\sum_{k\in\mathcal{I}} d_k^* = \sqrt{\frac{r}{c_i}\sum_{k\in\mathcal{I}^-} d_k}. \tag{14}$$

By setting $\sum\limits_{k\in\mathcal{I}} d_k^* = \zeta$, it readily follows that

$$d_1^* = \zeta - \tfrac{\zeta^2 c_1}{r}, \ldots, d_I^* = \zeta - \tfrac{\zeta^2 c_I}{r}. \tag{15}$$

Based on (15), we can obtain

$$\zeta = \zeta I - \frac{\zeta^2}{r}\sum_{k\in\mathcal{I}} c_k \Rightarrow \zeta = \frac{(I-1)r}{\sum\limits_{k\in\mathcal{I}} c_k}. \tag{16}$$

By applying (16) into (14), eventually (10) follows. ■

Given the existence of Nash equilibrium, we further prove that for a unique Nash equilibrium, the MO possesses a unique optimal payment strategy, denoted as r^*.

Theorem 2. *The MO's optimal payment strategy* $r^* > 0$ *is unique if a Nash equilibrium exists and is unique among the training terminals for any given* r*.*

Proof: By substituting (10) into (4), we can derive the first-order derivative of $U_{MO}(r)$ as follows:

$$\begin{aligned}\frac{\partial U_{MO}(r)}{\partial r} &= \omega G'(\sum_{k\in\mathcal{I}} d_k)(\frac{\partial(\sum\limits_{k\in\mathcal{I}} d_k^*)}{\partial r}) - 1 \\ &= \omega G'(\sum_{k\in\mathcal{I}} d_k)(\frac{\partial(d_1^*)}{\partial r} + \cdots + \frac{\partial(d_I^*)}{\partial r}) - 1.\end{aligned} \tag{17}$$

Hence, the second-order derivative of $U_{MO}(r)$ is given by

$$\begin{aligned}\frac{\partial^2 U_{MO}(r)}{\partial r^2} &= \omega G''(\sum_{k\in\mathcal{I}} d_k)(\frac{\partial(d_1^*)}{\partial r} + \cdots + \frac{\partial(d_I^*)}{\partial r})^2 \\ &\quad + \omega G'(\sum_{k\in\mathcal{I}} d_k)(\frac{\partial^2(d_1^*)}{\partial r^2} + \cdots + \frac{\partial^2(d_I^*)}{\partial r^2}) \\ &= \omega G''(\sum_{k\in\mathcal{I}} d_k)(\frac{\partial(d_1^*)}{\partial r} + \cdots + \frac{\partial(d_I^*)}{\partial r})^2.\end{aligned} \tag{18}$$

Since $G(\cdot)$ is a concave function, we have $\frac{\partial^2 U_{MO}(r)}{\partial r^2} < 0$. Therefore, the utility of the MO, $U_{MO}(r)$, is a strictly concave function of r. Since $U_{MO}(r) = 0$ for $r = 0$ and $U_{MO}(r) \to -\infty$ as $r \to \infty$, it has a unique maximizer r^*. To this end, there exists a unique optimal payment strategy r^* for the MO to achieve a unique Stackelberg equilibrium. ■

Algorithm 1 outlines the incentive mechanism between the MO and the terminals. The process begins with the MO determining the pricing for terminal participation in training and calculating its utility based on these prices. Using the optimal utility U_{MO}, the MO derives its optimal decision

Algorithm 1 Incentive mechanism b/w MO and terminals.

Input $\omega, G(X), c_i$
Output d_i^*, r^*
The MO calculates $\hat{r}$ by setting (17) to 0;
The MO calculates U_{MO} based on $\hat{r}$ using (4);
if $U_{MO}(\hat{r}) \geq U_{MO}(r)$ **then**
$r^* \leftarrow \hat{r}$;
end if
The MO sends r^* to the terminals;
for $i \in \mathcal{I}$ **do**
Terminal i calculates $\hat{d}_i$ using (10);
if $U_i^1(\hat{d}_i) \geq U_i^1(d_i)$ **then**
$d_i^* \leftarrow \hat{d}_i$;
Terminal i updates its local model using d_i^*;
end if
end for
return d_i^*, r^*

strategy and communicates the pricing to the training terminals. Each terminal then calculates the appropriate data size for training. When a terminal's utility is maximized, it makes optimal decisions and initiates training.

The computational complexity of **Algorithm 1** primarily depends on the number of training terminals, with a complexity of $\mathcal{O}(I)$ [21]. Since the uniqueness of the Stackelberg equilibrium is ensured, the best response strategy described in **Algorithm 1** is guaranteed to converge.

VI. Game between Terminals and Miners

This section introduces the second-tier game within the considered BFL system. To address the complex interactions between many training terminals and miners, we adopt the MLMF game model. The MLMF game model is chosen over traditional game-theoretic approaches because it better captures the intricate and dynamic interactions among multiple miners and training terminals in the considered BFL system. Unlike conventional games, which often assume a single leader or simplified interactions, the MLMF framework allows for a more realistic representation of the competitive and cooperative behaviors among miners, as well as the strategic decisions of terminals in selecting miners for model validation and packaging.

In this model, the miners act as leaders, competing by offering essential computational resources and validation services while setting their service prices to attract terminals. The training terminals act as followers, aiming to submit their local model updates to the miners for validation, packaging, and eventual uploading to the blockchain. Their decisions are guided by their individual requirements and the service prices set by the miners.

A. MLMF Game

In this Stackelberg game, the miners are leaders and the terminals are followers. The game has two stages. In the first stage, each miner formulates its pricing strategy p_j with the objective of optimizing its earnings, while considering the pricing approaches of fellow miners and the resource needs of the terminals. Consequently, the miner's profit maximization problem can be represented as follows:

$$\mathbf{P3}: \quad \max_{\mathbf{x}_{ij}} \quad U_j^n(p_j) = \sum_{i\in\mathcal{I}} p_j x_{ij} - c_j \sum_{i\in I} x_{ij} \tag{19a}$$

$$\text{s.t.} \quad p_j > 0, \tag{19b}$$

$$p_j \leq p_j^{max}, \tag{19c}$$

$$\sum_{i\in I} x_{ij} \leq B_{max}, \tag{19d}$$

where p_j^{max} is the maximum pricing of miner j, and B_{max} is the maximum capacity of miner j. Constraints (19b) and (19c) specify the pricing lower and upper bounds of the miners, respectively. These bounds are determined by factors, such as the miner's operational costs c_j, market competition, and the budget constraints of the training terminals. Given the limited package resources available at the terminals, constraint (19d) outlines the maximum package resource limit for the miners. B_{max} is defined by the computational resources and operational constraints of the system, ensuring that local models are processed and validated without exceeding capacity.

Suppose that every training terminal has access to the resource prices of all miners. A training terminal can determine its packaging strategy by solving the following problem:

$$\mathbf{P4}: \quad \max_{\mathbf{x}_{ij}} \quad U_i^2(x_{ij}) = \sum_{j\in\mathcal{N}} (a_i x_{ij} - b_i x_{ij}^2) - \sum_{j\in\mathcal{N}} p_j x_{ij}, \tag{20a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{N}} x_{ij} \leq K, \tag{20b}$$

$$\sum_{j\in\mathcal{N}} p_j x_{ij} \leq U_i^1, \tag{20c}$$

$$x_{ij} \geq 0, \tag{20d}$$

$$x_{ij} \leq G_{ij}, \tag{20e}$$

where (20b) presents the constraint on the number of rounds that a terminal participates in the training task; i.e., K is the total number of rounds required for the task. (20c) sets the upper limit on the budget for a terminal; i.e., U_i^1 is the profit obtained by the terminal from the MO in the game between the MO and terminals. (20d) ensures that the resource acquired by a training terminal is non-negative.

Constraint (20e) outlines the maximum resource allocation that the training terminal can obtain, contingent on the miners' trustworthiness. The terminals take into account the trust level of the miners when acquiring packaging resources and determine the quantity of resources to purchase to safeguard data security. On the other hand, under a given consensus mechanism (e.g., PoW), variations in the miners' computing powers lead to differences in chain speeds (i.e., the rate at which transactions are processed and blocks are packaged). As a result, each training terminal faces an upper limit when selecting miners for packaging services.

Next, we proceed to analyze the Stackelberg equilibrium (SE) of the MLMF game.

Definition 2 (Stackelberg Equilibrium [42]). *The SE represents the Nash equilibrium between the leaders and the followers in the game. It is presumed that the Nash equilibrium* (p_j^*, x_{ij}^*) *and the accompanying conditions hold true:*

- *For miner* j,

$$U_j^n(p_j^*, p_{-j}^*, x_{ij}^*) \geq U_j^n(p_j, p_{-j}^*, x_{ij}^*), j \in \mathcal{N}. \tag{21}$$

- *For terminal* i *during the second stage,*

$$U_i^2(p_j^*, x_{-ij}^*, x_{ij}^*) \geq U_i^2(p_j^*, x_{-ij}^*, x_{ij}), i \in I, \tag{22}$$

where (21) corresponds to the Nash equilibrium condition for the leaders, and (22) corresponds to the Nash equilibrium condition for the followers.

In what follows, we first find the optimal resource demands of the terminals by analyzing their reactions to the pricing strategies adopted by the miners. Once the terminals' strategies are identified, we use them to solve the Nash equilibrium solutions of the miners' game by substituting these strategies into the miners' optimization problems. The Stackelberg equilibrium ensures that the leaders' strategies are optimal given the followers' reactions, and the followers' strategies are optimal given the leaders' actions, resulting in a stable and efficient outcome for all participants.

B. Analysis of MLMF Game

We start by analyzing the optimal resource demands of the training terminals and establishing the following theorem.

Theorem 3. *Taking into account the resource prices set by the miners, the maximum budget constraint of each terminal, and the level of trust the terminals have in the miners, the solution to Problem* $\mathbf{P4}$ *yields the optimal quantity of resources that terminal* i *should purchase:*

$$x_{ij}^* = \min\Big\{G_{i,j}, \frac{(1-N)p_j + 2Kb_i + \sum\limits_{j'\in\mathcal{N}^-} p_{j'}}{2Nb_i}\Big\}. \tag{23}$$

where $\mathcal{N}^- = \{1, \ldots, j-1, j+1, \ldots, N\}$.

Proof: We can readily verify that the second-order derivative of $U_i^2(x_{ij})$ with respect to x_{ij} is persistently negative. The second-order derivative of $U_i^2(x_{ij})$ is given by

$$\frac{\partial^2 U_i^2(x_{ij})}{\partial x_{ij}^2} = -2b_i < 0, \tag{24}$$

According to (24), we can confirm the existence of a Nash equilibrium in the second stage. The Karush-Kuhn-Tucker (KKT) optimality conditions can be employed to obtain the optimal solution for Problem $\mathbf{P4}$, which can be recast as

$$\mathbf{P5}: \quad \max_{\mathbf{x}_{ij}} \quad \sum_{j\in\mathcal{N}}(a_i x_{ij} - b_i x_{ij}^2) - \sum_{j\in\mathcal{N}} p_j x_{ij} + \lambda_1\big(K - \sum_{j\in\mathcal{N}} x_{ij}\big) + \lambda_2\big(U_i^1 - \sum_{j\in\mathcal{N}} p_j x_{ij}\big), \tag{25a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{N}} x_{ij} \leq K, \tag{25b}$$

$$\sum_{j\in\mathcal{N}} p_j x_{ij} \leq U_i^1, \tag{25c}$$

$$\lambda_1\big(K - \sum_{j\in\mathcal{N}} x_{ij}\big) = 0, \tag{25d}$$

$$\lambda_2\big(U_i^1 - \sum_{j\in\mathcal{N}} p_j x_{ij}\big) = 0, \tag{25e}$$

$$\lambda_1 \geq 0, \tag{25f}$$

$$\lambda_2 \geq 0, \tag{25g}$$

$$K - \sum_{j\in\mathcal{N}} x_{ij} \geq 0, \tag{25h}$$

$$U_i^1 - \sum_{j\in\mathcal{N}} p_j x_{ij} \geq 0, \tag{25i}$$

where λ_1 and λ_2 are the Lagrangian multipliers. Then, the optimal solution is Problem $\mathbf{P5}$ is given by

$$x_{ij}^* = \frac{a_i - p_j - \lambda_1 - \lambda_2 p_j}{2b_i}, \forall i, j. \tag{26}$$

Note that the terminals are self-interested in their own profit; in other words, the terminals demand the profit obtained from the MO in the first tier of the two-tier Stackelberg game to be greater than the rewards offered to the miners in the second tier. As a result, $U_i^1 - \sum_{j\in\mathcal{N}} p_j x_{ij} > 0$, i.e., $\lambda_2 = 0$. Suppose that terminal i participates in all training rounds, i.e. $K - \sum_{j\in\mathcal{N}} x_{ij} = 0$. Hence, $\lambda_2 > 0$. Then, we have

$$K = \sum_{j\in\mathcal{N}} \frac{a_i - p_j - \lambda_1}{2b_i}. \tag{27}$$

By substituting (26) into (27), it follows that

$$\lambda_1 = a_i - \frac{2Kb_i + \sum\limits_{j\in N} p_j}{N}. \tag{28}$$

Further substituting (28) into (26), we obtain

$$x_{ij}^* = \frac{1}{2Nb_i}\left[(1-N)p_j + 2Kb_i + \sum_{j'\in\mathcal{N}^-} p_{j'}\right]. \tag{29}$$

Considering the range of x_{ij}, (29) leads to (23). ■

Operating independently, each miner can predict the actions of its competitors and devise an optimal strategy based on the predicted behavior of the terminals. We convert problem **P3** by first considering the utility of the miners based on the packaging resource demand of the terminals. Specifically, by substituting (29) into (5), it follows that

$$\begin{aligned} U_j^n(p_j, p_{-j}, x_{ij}) = &\sum_{i\in\mathcal{I}} p_j \Big[\tfrac{(1-N)p_j}{2Nb_i} + \tfrac{2Kb_i + \sum\limits_{j\in\mathcal{N}^-} p_j}{2Nb_i}\Big] \\ &- \sum_{i\in\mathcal{I}} c_j \Big[\tfrac{(1-N)p_j}{2Nb_i} + \tfrac{2Kb_i + \sum\limits_{j\in\mathcal{N}^-} p_j}{2Nb_i}\Big]. \end{aligned} \tag{30}$$

For illustration convenience, we define

$$S = \sum_{i\in\mathcal{I}} \frac{1-N}{2Nb_i}; \tag{31a}$$

$$T(p_{-j}) = \sum_{i\in\mathcal{I}} \frac{2Kb_i + \sum\limits_{j\in N^-} pj}{2Nb_i}. \tag{31b}$$

Then, (30) can be rewritten as

$$U_j^n(p_j, p_{-j}, x_{ij}) = Sp_j^2 + T(p{-}j)p_j - Sc_j p_j - T(p_{-j})c_j, \tag{32}$$

where $T(p_{-j})$ reflects the influence of the strategies p_{-j} adopted by other miners on the utility of miner j, i.e., how the pricing choices of other miners affect the utility of miner j.

The second-order derivative of $U_j^n(p_j)$ is given by

$$\frac{\partial^2 U_j^n(p_j)}{\partial x_{ij}^2} = 2S = \sum_{i\in\mathcal{I}} \frac{1-N}{Nb_i}, \tag{33}$$

When the number of miners satisfies $N > 1$, the negativity of $\frac{\partial^2 U_j^n(p_j)}{\partial x_{ij}^2}$ is satisfied, from which the concavity of $U_j^n(p_j)$ follows. Then, our model can converge to the optimum results.

By substituting (29), we rewrite (19d) as

$$\sum_{i\in\mathcal{I}} \Big[\frac{(1-N)p_j}{2Nb_i} + \frac{2Kb_i + \sum\limits_{j\in\mathcal{N}^-} p_j}{2Nb_i}\Big] \le B_{max} \tag{34a}$$

$$\Rightarrow Sp_j + T(p_{-j}) \le B_{max}. \tag{34b}$$

Considering the pricing decisions p_{-j} made by the other miners, the miners' pricing frame is framed as

$$\mathbf{P6}: \max_{\mathbf{p}_j} \quad Sp_j^2 + T(p_j)p_j - Sc_jp_j - T(p_{-j})c_j \tag{35a}$$

$$\text{s.t.} \quad p_j > 0, \tag{35b}$$

$$p_j \le p_{max}, \tag{35c}$$

$$Sp_j + T(p_{-j}) \le B_{max}. \tag{35d}$$

The optimal solution to Problem $\mathbf{P6}$ represents the best pricing approach for miner j, as established in the following theorem.

Theorem 4. *Taking into account unit packaging cost of miner j, the prices p_{-j} of resources packaged by other miners, and the demand for resources from the terminals, the optimal pricing strategy for miner j can be determined by solving Problem $\mathbf{P6}$ under the two cases.*

Case 1: *Miner j has not yet been at its packaging capacity ceiling, and the optimal pricing strategy is provided by*

$$p_j^* = \frac{c_jS - T(p_{-j})}{2S}. \tag{36}$$

Case 2: *Miner j has reached its maximum packaging limit, and the optimal pricing strategy is given by*

$$p_j^* = \frac{B_{max} - T(p_{-j})}{S}. \tag{37}$$

Proof: The KKT optimality conditions can be used to obtain the optimal solution for Problem $\mathbf{P6}$. Alternatively, Problem $\mathbf{P6}$ can be reformulated as follows:

$$\mathbf{P7}: \max_{\mathbf{p}_j} \quad Sp_j^2 + T(p-j)p_j - Sc_jp_j - T(p_{-j})c_j + \lambda_3[B_{max} - Sp_j - T(p_{-j})] \tag{38a}$$

$$\text{s.t.} \quad \lambda_3[B_{max} - Sp_j - T(p_{-j})] \ge 0, \tag{38b}$$

$$\lambda_3 \ge 0, \tag{38c}$$

$$B_{max} - Sp_j - T(p_{-j}) \ge 0, \tag{38d}$$

where λ_3 is the Lagrange multiplier. By addressing Problem $\mathbf{P7}$, we arrive at the following optimal pricing strategy p_j^*:

$$p_j^* = \frac{c_jS - T(p_{-j}) + \lambda_3 S}{2S}. \tag{39}$$

According to the complementary slackness condition, we have two scenarios: either the equality constraint of (34a) is satisfied, or it is not, as discussed below.

Case 1: When miner j has not reached its maximum packaging limit, (34a) does not satisfy the equation,i.e., $\lambda_3 = 0$. Consequently, the ultimate pricing strategy can be derived as follows

$$p_j^* = \frac{c_jS - T(p_{-j})}{2S}. \tag{40}$$

Case 2: When miner j has reached its maximum packaging limit, (34a) satisfies

$$B_{max} - Sp_j - T(p_{-j}) = 0. \tag{41}$$

By substituting (39) into (41), it readily follows that

$$\lambda_3 = \frac{2B_{max} - T(p_{-j}) - c_jS}{S}. \tag{42}$$

We substitute (42) into (39), and obtain p_j^* as

$$p_j^* = \frac{B_{max} - T(p_{-j})}{S}. \tag{43}$$

In summary, when miner j satisfies (34a), it reaches its maximum packaging limit to maximize its profit. ■

Algorithm 2 outlines the incentive mechanism between the training terminals and miners. Each miner begins by setting an initial price p_0, observes the prices set by other miners, calculates a new price, and repeats this process until the change in the pricing strategy is smaller than a predetermined precision ξ. The final price is then communicated to the terminals, which calculate the number of local models to allocate to each miner for packaging. Terminal i makes optimal decisions when its utility is maximized.

The computational complexity of **Algorithm 2** can be analyzed as follows: For a convergence accuracy of ξ, the complexity of the pricing adjustment process among the miners is $\mathcal{O}\big(\frac{1}{\xi}\big)$. With N miners, the total complexity becomes $\mathcal{O}\big(\frac{N}{\xi}\big)$. With I terminals, determining the pricing strategy for each terminal has a complexity of $O(NI)$. Therefore, the total computational complexity of the algorithm is $\mathcal{O}\big(\frac{N}{\xi} + NI\big)$ [21].

VII. Experimental Study

This section presents the numerical results to evaluate the complex strategic dynamics among the BFL participants. We also conduct a detailed comparison between our proposed incentive mechanism and the existing benchmark framework to verify the effectiveness and reliability of our incentive mechanism. Our simulations were conducted using Python 3.8 on a system configured with an AMD Ryzen 7 5800H CPU, GTX 3060 GPU, and 16 GB RAM. We can integrate our incentive mechanism into the smart contracts of simulated blockchain platforms, e.g., BlockEmulator and Ganache [49], to validate optimal pricing and packaging strategies in a blockchain environment through simulated transaction processing, consensus mechanisms, and reward distribution processes.

Algorithm 2 Incentive mechanism b/w terminals and miners

Input $N, K, b_i, a_i, c_j, B_{max}, G_{i,j}$
Output $x^*_{i,j}, p^*_j$
Set $t = 0$ and define the precision as ξ;
Every miner configures an initial price $p_j = p_j(0)$;
Repeat
Miners monitor p_{-j} of other miners;
for $j \in \mathcal{N}$ **do**
if the equality condition of (34a) holds **then**
Miner j calculates $\hat{p}_j$ using (37);
else then
Miner j calculates $\hat{p}_j$ using (36);
end if
end for
$t \leftarrow t + 1$;
until $\|p^t_j - p^{t-1}_j\| \leq \xi$
$p^*_j \leftarrow p^t_j$;
The miners send p^*_j, $\forall j$ to the terminals;
for $i \in \mathcal{I}$ **do**
Terminal i calculates $\hat{x}_{i,j}$ using (26);
if $U^2_i(\hat{x}_{i,j}) \geq U^2_i(x_{i,j})$ **then**
$x^*_{i,j} \leftarrow \hat{x}_{i,j}$;
Terminal i selects miner j to package $x^*_{i,j}$ models;
end if
end for
return $x^*_{i,j}, p^*_j$

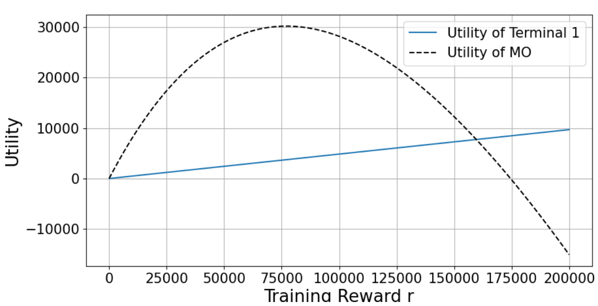

Fig. 3. Utility model of the MO and terminal vs. training reward

A. Numerical Results

Numerical results are presented to illustrate the interplay among the entities within the BFL system as the system parameters vary. Fig. 3 plots the utilities of the MO and the terminals with the increasing reward of the MO r, where the number of terminals is set to $I = 20$. It is observed that the terminals' utility increases almost linearly as r increases. The reason is that as the rewards increase, so does the anticipated profit for each terminal. On the other hand, the MO's utility initially rises and then declines as r increases. This is because when r is small, increasing r may incentivize the terminals and boost their demand. As r continues to rise, the terminals' demands gradually saturate, causing the MO's revenue to decline. In this sense, there is an optimal strategy r^* for the MO to reach its maximum utility.

Fig. 4 plots the impact of the training data size d_i on

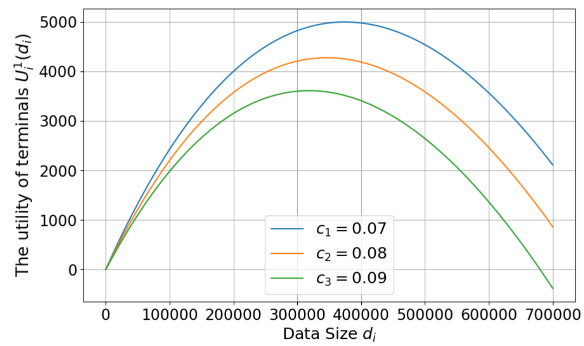

Fig. 4. Terminal utility $U^1_i(d_i)$ under different training costs c_i and different data size d_i

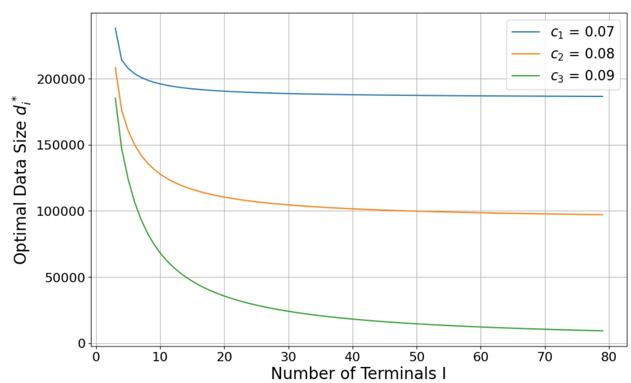

Fig. 5. Terminal's optimal data size d^*_i vs. number of terminals

the utility of the terminal, where U^1_i represents the utility of terminal i under different training costs. As d_i increases, the utility of U^1_i first increases and then decreases. The reason for this is that when d_i exceeds the optimal data size d^*_i, the terminal i must consider the fast-growing training cost. The utility of the terminals U^1_i is also affected by their costs c_i, and high costs can lead to a decrease in utility.

Fig. 5 plots the optimal data sizes d^*_i, $\forall i$ of the terminals under the variation of the number of terminals. As the number I of terminals increases, the optimal data size d^*_i decreases consistently. This is because as I increases, the competition between the terminals becomes more intense, resulting in a reduction in their rewards. Consequently, each terminal needs to conserve its costs to maximize its utility. In addition, a higher unit cost leads to a smaller d^*_i, since a terminal decreases its data size to cut cost when c_i is large.

Fig. 6 shows the stabilization of prices as the number of iterations increases, where each terminal is set to participate in 100 rounds of training, meaning each terminal must purchase 100 packaged resources. A price precision is set to 0.0001. It is observed that the prices of the miners tend to converge, indicating the convergence of the proposed algorithm. This convergence is evident after five iterations, when the prices reach the set precision requirement. The reason for this convergence is the fast convergence rate of the algorithm. It is

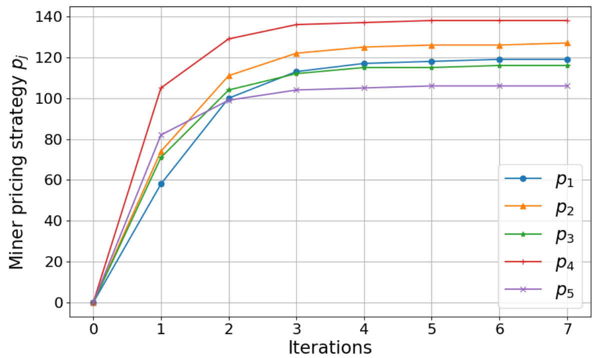

Fig. 6. Convergence behavior of the miner's prices p_j over iterations

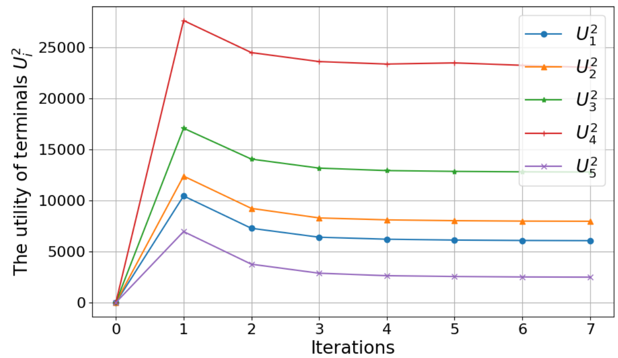

Fig. 8. Convergence behavior of the terminal's utility U_i^2 over iterations

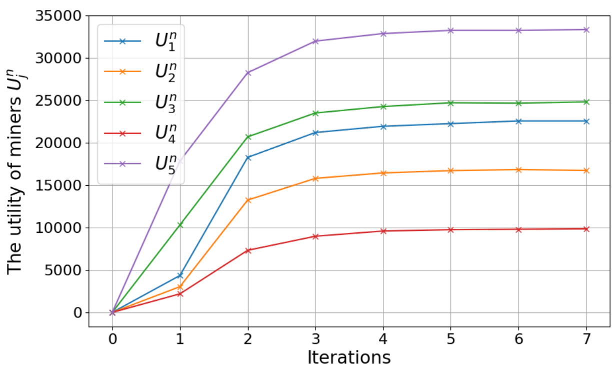

Fig. 7. Convergence behavior of the miner's utility U_j^n over iterations

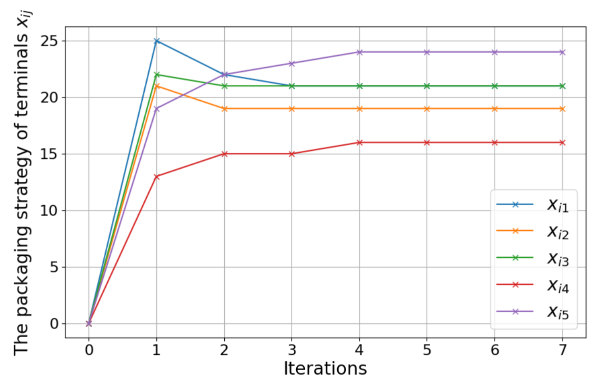

Fig. 9. Convergence of the number of package resources x_{ij} over iterations

noticed that when a miner observes the pricing of the other miners, it adjusts its own pricing strategy accordingly.

Fig. 7 illustrates the variation of the miners' revenue with pricing, where the miners exercise control over their revenue by adjusting the price. As the price stabilizes, the revenue of the miners converges accordingly. Notably, when miners set a high price, terminals reduce the amount of packaged resources they purchase, subsequently leading to a decrease in the miners' revenue. Nonetheless, a miner can increase its revenue by observing and adjusting their pricing strategies based on those of other miners.

For analysis convenience, we choose five terminals that generate income by completing BFL tasks, as depicted in Fig. 8. The income earned by these terminals fluctuates based on the pricing strategy adopted by the miners. It is worth noting that as the number of packaged resources attains a specific threshold, the growth rate of the users' revenue tends to decline. The pricing strategy employed by the miners also has a direct impact on the quantity of packaged resources, as demonstrated in Fig. 9. When the miners set a high resource price, the terminals are inclined to decrease their purchase quantities accordingly. As the miners lower their resource prices, the terminals respond by increasing resources requested, allowing them to generate higher revenues. As the prices set by the miners gradually align, the demand for resources from the terminals also converges.

B. Performance Comparison with Benchmarks

In Figs. 10 and 11, we conduct a comparison study between our proposed incentive mechanism (PIM) and benchmarks, which are classified into rational and irrational categories. The rational benchmarks include *Dynamic-Select*, *Max-Contribute*, and *Fixed-Pool* [50]. These mechanisms differ from the PIM solely in user selection strategies. Dynamic-Select introduces randomness by selecting a subset of terminals to contribute data, promoting diversity and unpredictability. Max-Contribute maximizes data contributions by involving all available terminals, overlooking their individual constraints or rational behavior. Fixed-Pool adopts a structured and consistent approach, selecting a predetermined set of terminals for data contribution. In contrast, the irrational benchmarks – *Blind-Select*, *Overload*, and *Partial-Pool* [50], [51] – disregard the rationality and selfishness of terminals. Blind-Select blindly requests data from randomly chosen terminals. Overload demands full data contributions from all terminals, ignoring their capacity or willingness. Partial-Pool takes a balanced yet arbitrary approach, requesting partial data contributions (half) from a subset (half) of the terminals.

To further validate the effectiveness of our mechanism, we expand the contrastive framework established in [50]. Specifically, we first vary the number of participating terminals (followers) and the amount of data contributed by each, and evaluate the utility outcomes for both terminals and the MO

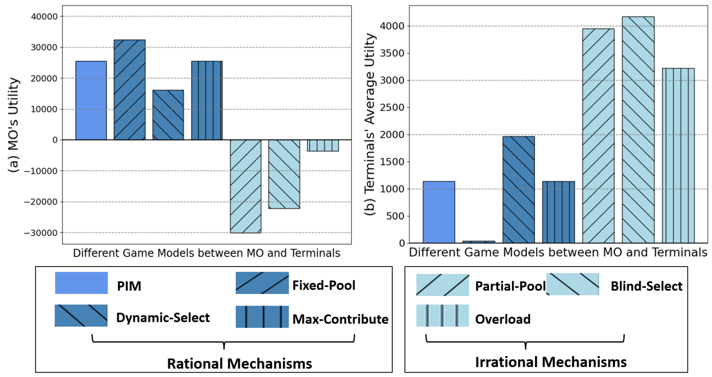

Fig. 10. Comparison of the proposed mechanism and the benchmarks: (a) MO's utility, and (b) terminals' average utility

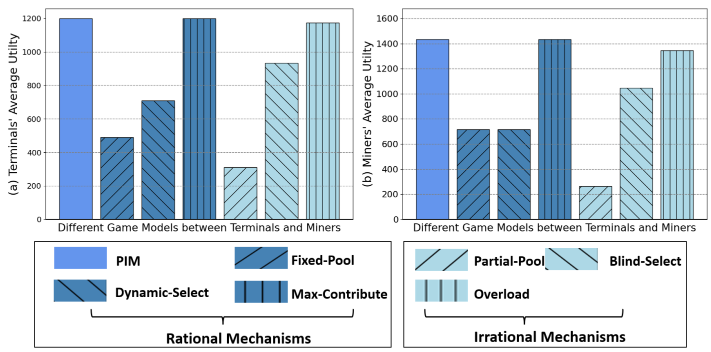

Fig. 11. Comparison of the proposed mechanism and the benchmarks: (a) terminal's utility, and (b) miners' average utility

under different data contributions. This approach provides insights into the robustness of our mechanism against fluctuations in terminal behavior and data volume. An interesting observation from Fig. 10 is that although the individual utilities of the MO and the terminals under the PIM may occasionally fall below those achieved by some rational or irrational benchmarks, the combined utility of the MO and terminals consistently surpasses those of all benchmarks.

Next, we examine the impacts of different packaging quantities, once again using the number of terminals as the primary variable. As illustrated in Fig. 11, by comparing the utilities of the terminals and miners across various packaging strategies, we observe that the aggregate utility of the miners and terminals under the PIM consistently outperforms that of both rational and irrational benchmarks. This superior performance can be attributed to the balanced approach of PIM, which adheres to the principles of individual rationality and optimized terminal selection.

C. FL Convergence

We further test the impact of PIM on the learning performance of the FedAvg algorithm [52]. Our experiments utilize the CIFAR-10 dataset [53] to generate local datasets for the terminals, with a Convolutional Neural Network (CNN) selected as the neural network model. The non-IID degree of the terminals' datasets is adjusted as outlined in [54], where $s\%$ of the data is allocated to each terminal in an IID manner to

TABLE II
COMPARISON OF DIFFERENT INCENTIVE MECHANISMS IN DATA SIZE, DATA CONTRIBUTION, AND ACCURACY.

Incentive Mechanisms	Data Size	Data Contribution	Accuracy
PIM	43415	86.8%	58.1%
URA	37553	75.1%	56.1%
RRA	35127	70.3%	55.9%

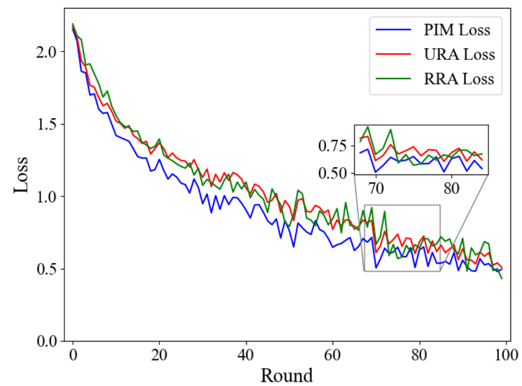

Fig. 12. Training loss of different incentive mechanisms

ensure similarity, while the remaining $(100-s)\%$ is distributed based on label sorting to introduce non-IID characteristics. As a result, different data distributions are experienced across the terminals. A higher non-IID ratio reflects greater dissimilarity in data distributions across terminals from different groups.

We consider two widely recognized incentive schemes:

- *Uniform Reward Allocation (URA) [51], [55]:* The total reward r provided by the MO for FL tasks is evenly distributed among all terminals. Each terminal receives an identical reward, regardless of data contribution.
- *Random Reward Allocation (RRA) [55], [56]:* The total reward r is distributed randomly among the terminals. While the overall payout remains fixed, the reward allocated to each terminal varies randomly.

For a fair comparison, we maintain a constant total reward r paid by the MO across all mechanisms. Additionally, the terminal's utility $U_i^1(d_i)$—calculated as the reward received minus the associated cost—is standardized across mechanisms. This allows for a consistent evaluation of the data contribution made by terminals under different incentive schemes. Variations in data contribution are driven by differences in the underlying incentive structures. We investigate the influence of these incentive mechanisms on FL convergence by selecting 20 terminals for training. The model's training loss and accuracy are evaluated after 100 communication rounds.

Figs. 12 and 13 illustrate the convergence performance of PIM. During the initial 100 training rounds, the PIM exhibits faster loss reduction compared to URA and RRA. By the 100th round, the model trained under PIM achieves higher accuracy than both baselines. Table II evaluates the data contribution of the users under different incentive mechanisms. Our study utilizes the CIFAR-10 dataset, comprising 60,000 images, with 50,000 for training and 10,000 for testing. Compared

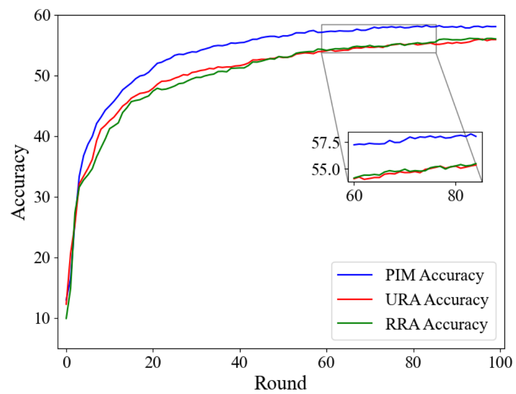

Fig. 13. Model accuracy of different incentive mechanisms

to the other benchmarks, the PIM incentivizes the terminals to contribute more data and, thus, the MO can obtain a model with finer accuracy. This demonstrates the PIM effectively optimizes the participants' utilities at the Stackelberg equilibrium. This finding highlights the robustness of PIM in aligning individual contributions with system-wide objectives.

VIII. Conclusion

This paper has presented a novel game-theoretic incentive mechanism for BFL, addressing the challenges of user participation and system stability. By modeling the interactions within BFL through two subsequent Stackelberg games, we have analyzed how MO, miners, and training terminals can align their strategies to optimize both individual utilities and overall performance. Our mechanism motivates participants to actively engage in BFL, enhancing model accuracy and fostering the stability and efficiency of BFL. Our experiments validated this mechanism, demonstrating its ability to offer 5.8% and 53.4% higher utilities in the two game than those of the alternatives. Future work will integrate reinforcement learning to further refine and adapt the incentive mechanism in dynamic BFL environments. By leveraging reinforcement learning, the system can better handle the non-stationary behaviors of the participants, optimize reward allocation in real time, and enhance adaptability and efficiency in complex scenarios.

References

[1] K. Li, B. P. L. Lau, *et al.*, "Toward ubiquitous semantic metaverse: Challenges, approaches, and opportunities," *IEEE Internet of Things Journal*, vol. 10, no. 24, pp. 21855–21872, 2023.

[2] Y. Zuo, L. Gui, K. Cui, *et al.*, "Mobile blockchain-enabled secure and efficient information management for indoor positioning with federated learning," *IEEE Transactions on Mobile Computing*, vol. 23, no. 12, pp. 12176–12194, 2024.

[3] T. Xiang, Y. Bi, X. Chen, *et al.*, "Federated learning with dynamic epoch adjustment and collaborative training in mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 4092–4106, 2024.

[4] S. Hu, X. Chen, W. Ni, *et al.*, "Distributed machine learning for wireless communication networks: Techniques, architectures, and applications," *IEEE Commun. Surveys Tutorials*, vol. 23, no. 3, pp. 1458–1493, 2021.

[5] M. Tang, F. Peng, and V. W. Wong, "A blockchain-empowered incentive mechanism for cross-silo federated learning," *IEEE Trans. Mobile Comput.*, vol. 23, no. 10, pp. 9240–9253, 2024.

[6] Y. Li, F. Li, S. Yang, *et al.*, "A cooperative analysis to incentivize communication-efficient federated learning," *IEEE Trans. Mobile Comput.*, vol. 23, no. 10, pp. 10175–10190, 2024.

[7] Y. Wang, T. Sun, S. Li, *et al.*, "Adversarial attacks and defenses in machine learning-empowered communication systems and networks: A contemporary survey," *IEEE Commun. Surveys Tutorials*, vol. 25, no. 4, pp. 2245–2298, 2023.

[8] X. Yuan, W. Ni, M. Ding, *et al.*, "Amplitude-varying perturbation for balancing privacy and utility in federated learning," *IEEE Trans. Info. Forensics Security*, vol. 18, pp. 1884–1897, 2023.

[9] C. Ying, F. Xia, D. S. L. Wei, *et al.*, "BIT-FL: Blockchain-enabled incentivized and secure federated learning framework," *IEEE Trans. Mobile Comput.*, vol. 1, no. 1, pp. 1–18, 2024.

[10] Y. Gong, H. Yao, Z. Xiong, *et al.*, "Blockchain-aided digital twin offloading mechanism in space-air-ground networks," *IEEE Trans. Mobile Comput.*, vol. 24, no. 1, pp. 183–197, 2025.

[11] I. Makhdoom, M. Abolhasan, H. Abbas, *et al.*, "Blockchain's adoption in IoT: The challenges, and a way forward," *Journal of Network and Computer Applications*, vol. 125, pp. 251–279, 2019.

[12] G. Yu, X. Wang, K. Yu, *et al.*, "Survey: Sharding in blockchains," *IEEE Access*, vol. 8, pp. 14155–14181, 2020.

[13] Y. Qu, S. Yu, L. Gao, *et al.*, "Blockchained dual-asynchronous federated learning services for digital twin empowered edge-cloud continuum," *IEEE Trans. Services Comput.*, vol. 17, no. 3, pp. 836–849, 2024.

[14] Y. Wang, J. Zhou, G. Feng, *et al.*, "Blockchain assisted federated learning for enabling network edge intelligence," *IEEE Netw.*, vol. 37, no. 1, pp. 96–102, 2023.

[15] H. Chen, R. Zhou, Y.-H. Chan, *et al.*, "LiteChain: A lightweight blockchain for verifiable and scalable federated learning in massive edge networks," *IEEE Trans. Mobile Comput.*, pp. 1–17, 2024.

[16] A. P. Kalapaaking, I. Khalil, M. S. Rahman, *et al.*, "Blockchain-based federated learning with secure aggregation in trusted execution environment for Internet-of-Things," *IEEE Trans. Industrial Informatics*, vol. 19, no. 2, pp. 1703–1714, 2023.

[17] N. Agrawal, D. Mishra, and S. Agrawal, "A comprehensive survey on blockchained federated learning system and challenges," in *Proc. IEEE ICTBIG*, pp. 1–4, 2023.

[18] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, 2020.

[19] C. Huang, E. Liu, R. Wang, *et al.*, "Personalized federated learning via directed acyclic graph based blockchain," *IET Blockchain*, vol. 5, no. 1, pp. 73–82, 2024.

[20] J. Li, Y. Shao, K. Wei, *et al.*, "Blockchain assisted decentralized federated learning (BLADE-FL): Performance analysis and resource allocation," *IEEE Trans. Parallel Distributed Syst.*, vol. 33, no. 10, pp. 2401–2415, 2022.

[21] Z. Wang, Q. Hu, and X. Z. Xiong, "Incentive mechanism design for joint resource allocation in blockchain-based federated learning," *IEEE Trans. Parallel Distributed Syst.*, vol. 34, no. 5, pp. 1536–1547, 2023.

[22] Y. Wang, J. Zhou, G. Feng, *et al.*, "Blockchain assisted federated learning for enabling network edge intelligence," *IEEE Netw.*, vol. 37, no. 1, pp. 96–102, 2023.

[23] K. Toyoda and A. N. Zhang, "Mechanism design for an incentive-aware blockchain-enabled federated learning platform," in *2019 IEEE Int'l Conf. Big Data (Big Data)*, pp. 395–403, 2019.

[24] E. Bandara, X. Liang, S. Shetty, *et al.*, "Skunk — a blockchain and zero trust security enabled federated learning platform for 5G/6G network slicing," in *Proc. SECON*, pp. 109–117, 2022.

[25] Y. Jiang, B. Ma, X. Wang, *et al.*, "Blockchained federated learning for Internet of Things: A comprehensive survey," *ACM Computing Survey*, vol. 56, no. 10, p. 258, 2024.

[26] J. Huang, L. Kong, G. Chen, *et al.*, "Blockchain-based federated learning: A systematic survey," *IEEE Netw.*, vol. 37, no. 6, pp. 150–157, 2023.

[27] L. Bhatia and S. Samet, "Decentralized federated learning: A comprehensive survey and a new blockchain-based data evaluation scheme," in *Int'l Conf. Blockchain Comput. Appl. (BCCA)*, pp. 289–296, 2022.

[28] S. R. Pokhrel and J. Choi, "Federated learning with blockchain for autonomous vehicles: Analysis and design," *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 4734 – 4746, 2020.

[29] J. An, S. Tang, X. Sun, *et al.*, "FREB: Participant selection in federated learning with reputation evaluation and blockchain," *IEEE Trans. Services Comput.*, vol. 17, no. 6, pp. 3685–3698, 2024.

[30] W. Zhang, Q. Lu, Q. Yu, *et al.*, "Blockchain-based federated learning for device failure detection in industrial IoT," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5926–5937, 2021.

[31] L. Witt, U. Zafar, K. Shen, *et al.*, "Decentralized and incentivized federated learning: A blockchain-enabled framework utilising compressed soft-labels and peer consistency," *IEEE Trans. Services Comput.*, vol. 17, no. 4, pp. 1449–1464, 2024.

[32] Y. Zhao, Y. Qu, Y. Xiang, *et al.*, "Long-term proof-of-contribution: An incentivized consensus algorithm for blockchain-enabled federated learning," *IEEE Trans. Services Comput.*, vol. 17, no. 5, pp. 2558–2570, 2024.

[33] P. Sun, E. Liu, W. Ni, *et al.*, "Reconfigurable intelligent surface-assisted wireless federated learning with imperfect aggregation," *IEEE Transactions on Communications*, pp. 1–13, 2024. Early access.

[34] Z. Chen, F. Zhou, Y. Tian, *et al.*, "A blockchain-based dynamic incentive model in mobile edge computing," in *2024 Int'l Conf. Networking and Network Applications (NaNA)*, pp. 54–59, 2024.

[35] H. R. Varian, *Microeconomic Analysis, Third Edition*. Microeconomic analysis, third edition., 1992.

[36] F. P. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, Feb 1997.

[37] E. Liu and K. K. Leung, "Fair resource allocation under Rayleigh and/or Rician fading environments," in *2008 IEEE 19th PRIMC*, pp. 1–5, 2008.

[38] E. Liu, Q. Zhang, and K. K. Leung, "Clique-based utility maximization in wireless mesh networks," *IEEE Trans. Wirel. Commun.*, vol. 10, no. 3, pp. 948–957, 2011.

[39] E. Liu, Q. Zhang, and K. K. Leung, "Relay-assisted transmission with fairness constraint for cellular networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 2, pp. 230–239, 2011.

[40] Y. Zhan, P. Li, K. Wang, S. Guo, and Y. Xia, "Big data analytics by crowdlearning: Architecture and mechanism design," *IEEE Netw.*, vol. 34, no. 3, pp. 143–147, 2020.

[41] O. Candogan, K. Bimpikis, and A. Ozdaglar, "Optimal pricing in networks with externalities," *Operations Research: The Journal of the Operations Research Society of America*, no. 4, p. 60, 2012.

[42] R. B. Myerson, *Game Theory: Analysis of Conflict*. Game Theory: Analysis of Conflict, 1997.

[43] Y. Geng, E. Liu, W. Ni, *et al.*, "Balancing performance and cost for two-hop cooperative communications: Stackelberg game and distributed multi-agent reinforcement learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 10, no. 6, pp. 2193–2208, 2024.

[44] Y. Tian, S. Wang, J. Xiong, *et al.*, "Robust and privacy-preserving decentralized deep federated learning training: Focusing on digital healthcare applications," *IEEE/ACM Trans. Computational Biology Bioinformatics*, vol. 21, no. 4, pp. 890–901, 2024.

[45] J. Lu, H. Liu, Z. Zhang, *et al.*, "Toward fairness-aware time-sensitive asynchronous federated learning for critical energy infrastructure," *IEEE Trans. Industrial Informatics*, vol. 18, no. 5, pp. 3462–3472, 2022.

[46] L. Cheng and T. Yu, "Game-theoretic approaches applied to transactions in the open and ever-growing electricity markets from the perspective of power demand response: An overview," *IEEE Access*, vol. 7, pp. 25727–25762, 2019.

[47] B. Huang and A. Guo, "A dynamic hierarchical game approach for user association and resource allocation in HetNets with wireless backhaul," *IEEE Wirel. Commun. Lett.*, vol. 13, no. 1, pp. 59–63, 2024.

[48] R. B. Myerson, "On the value of game theory in social science," *Rationality and Society*, vol. 4, no. 1, pp. 62–73, 1992.

[49] K. Singla, J. Bose, and S. Katariya, "Machine learning for secure device personalization using blockchain," in *2018 ICACCI*, pp. 67–73, 2018.

[50] X. Wang, Y. Zhao, C. Qiu, *et al.*, "InFEDge: A blockchain-based incentive mechanism in hierarchical federated learning for end-edge-cloud communications," *IEEE J. Select. Areas Commun.*, vol. 40, no. 12, pp. 3325–3342, 2022.

[51] Y. Zhan and J. Zhang, "An incentive mechanism design for efficient edge learning by deep reinforcement learning approach," in *IEEE INFOCOM*, pp. 2489–2498, 2020.

[52] H. B. McMahan, E. Moore, D. Ramage, *et al.*, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, 2017.

[53] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," *Handbook of Systemic Autoimmune Diseases*, vol. 1, no. 4, 2009.

[54] S. P. Karimireddy, S. Kale, M. Mohri, *et al.*, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. the 37th Int'l Conf. Machine Learning*, vol. 476 of *ICML'20*, p. 12, 2020.

[55] X. Tang, Y. Wang, R. Huang, *et al.*, "Stackelberg game based resource allocation algorithm for federated learning in MEC systems," in *Proc. WCCCT*, pp. 7–12, 2023.

[56] S. Wang, B. Luo, and M. Tang, "Tackling system-induced bias in federated learning: A pricing-based incentive mechanism," in *Pro. IEEE ICDCS*, pp. 902–912, 2024.