Incentive Mechanism Design of Federated Learning for Recommendation Systems in MEC

Jiwei Huang, *Senior Member, IEEE*, Bowen Ma, Ming Wang, Xiaokang Zhou, *Member, IEEE*, Lina Yao, *Senior Member, IEEE*, Shoujin Wang, Lianyong Qi, *Senior Member, IEEE*, and Ying Chen, *Senior Member, IEEE*

***Abstract*—With the rapid development of consumer electronics and communication technology, a large amount of data is generated from end users at the edge of the networks. Modern recommendation systems take full advantage of such data for training their various artificial intelligence (AI) models. However, traditional centralized model training has to transmit all the data to the cloud-based servers, which suffers from privacy leakage and resource shortage. Therefore, mobile edge computing (MEC) combined with federated learning (FL) is considered as a promising paradigm to address these issues. The smart devices can provide data and computing resources for the FL and transmit the local model parameters to the base station (BS) equipped with edge servers to aggregate into a global model. Nevertheless, due to the limited physical resources and the risk of privacy leakage, the users (the owners of the devices) would not like to participate in FL voluntarily. To address this issue, we take advantage of game theory to propose an incentive mechanism based on the two-stage Stackelberg game to inspire users to contribute computing resources for FL. We define two utility functions for the users and the BS, and formulate the utility maximization problem. Through theoretical analysis, we obtain the Nash equilibrium strategy of the users and the Stackelberg equilibrium of the utility maximization problem. Furthermore, we propose a game-based incentive mechanism algorithm (GIMA) to achieve the Stackelberg equilibrium. Finally, simulation results are provided to verify the performance of our GIMA algorithm. The experimental results show that our GIMA algorithm converges quickly, and can achieve higher utility value compared to other incentive methods.**

***Index Terms*—Federated learning, recommendation system, mobile edge computing, incentive mechanism, game theory.**

Manuscript received 30 April 2023; revised 20 October 2023; accepted 2 December 2023. Date of publication 13 December 2023; date of current version 26 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 61972414; in part by the Beijing Natural Science Foundation under Grant L232050; in part by the Project of Cultivation for Young Top-Motch Talents of Beijing Municipal Institutions under Grant BPHR202203225; and in part by the Young Elite Scientists Sponsorship Program by BAST under Grant BYESS2023031. *(Corresponding author: Xiaokang Zhou.)*

Jiwei Huang and Bowen Ma are with the Beijing Key Laboratory of Petroleum Data Mining, China University of Petroleum, Beijing 102249, China (e-mail: huangjw@cup.edu.cn; 2021211249@student.cup.edu.cn).

Ming Wang is with the China Unicom Intelligence Network Innovation Center, China United Network Communications Group Company Ltd., Beijing 100089, China (e-mail: wangm316@chinaunicom.cn).

Xiaokang Zhou is with the Faculty of Data Science, Shiga University, Hikone 522-8522, Japan, and also with the RIKEN Center for Advanced Intelligence Project, RIKEN, Tokyo 103-0027, Japan (e-mail: zhou@biwako.shiga-u.ac.jp).

Lina Yao is with CSIRO's Data 61, Eveleigh, NSW 2015, Australia, and also with the School of Computer Science and Engineering, University of New South Wales, Sydney, NSW 2052, Australia (e-mail: lina.yao@unsw.edu.au).

Shoujin Wang is with the Data Science Institute, University of Technology Sydney, Sydney, NSW 2007, Australia (e-mail: shoujin.wang@uts.edu.au).

Lianyong Qi is with the College of Computer Science and Technology, China University of Petroleum (East China), Qingdao 266580, China (e-mail: lianyongqi@gmail.com).

Ying Chen is with the Computer School, Beijing Information Science and Technology University, Beijing 100101, China (e-mail: chenying@bistu.edu.cn).

Digital Object Identifier 10.1109/TCE.2023.3342187

I. Introduction

NOWADAYS, there has been an explosive growth in the number of smart devices. According to recent statistics, there are nearly 3 billions smart-phones and 7 billion connected Internet of Things (IoT) devices all around the world [1], [2], [3], [4]. These devices generate a huge amount of data, which enables a lot of machine learning (ML)-based applications, especially recommendation systems, to be possible. The availability of more personal information to servers enables the generation of more accurate and relevant recommendations for specific users in the recommendation systems. Currently, recommendation systems play a central role in several fields, such as e-commerce, video services, online music services, and advertising, benefiting both users and service providers [5], [6].

Conventionally, the data are usually transmitted to the cloud centers for processing. Although the cloud-based servers have abundant storage and computing resources, the cloud-centric method suffers from the following issues. First, the data collected from devices may contain private information such as personal messages, locations or photos. The privacy leakage will cause great loss to data owner. Second, with the large amount of data being generated by smart devices, transferring the raw data to the cloud can incur significant communication costs. Furthermore, the cloud is usually located far away from the smart devices. Therefore, it is not feasible to send all the raw data to the cloud.

Mobile Edge Computing (MEC) emerges as a transformative paradigm, wielding the power to provide intensive storage and computing resources in close proximity to end-user devices, thereby catalyzing a notable shift from traditional cloud-centric systems. In this innovative ecosystem, base stations (BSs) are fortified with servers brimming with storage and computational capacities, paving the way for a series of benefits, particularly in addressing some of the intrinsic challenges encountered by traditional cloud-centric recommendation systems. The propinquity of MEC to terminal devices substantially trims down the communication overhead and effectively minimizes the delay, ensuring a nimble and more responsive data processing and communication pipeline [7], [8], [9]. Furthermore, the implementation of MEC enables the offloading of computational tasks from resource-constrained devices to the edge, thereby not only conserving the vital resources of the terminal devices but also accelerating the execution of computationally intensive tasks. This proximity enables fast communications that support real-time applications and services, improving the user experience by delivering fast, powerful computing right where the network begins.

In an era where data privacy is important, Federated Learning (FL) is proposed as a decentralized Machine Learning (ML) technology that stands out for its inherent capability to safeguard users' data from potential misuse, while also facilitating collaborative model training across a network of devices [10]. Within the FL framework, each user, being the proprietor of smart devices, conducts training of their local model using their respective data samples, thereby ensuring that the raw, potentially sensitive data do not have to traverse to a centralized server. Beyond this intrinsic data privacy feature, FL extends additional merits including reducing the necessity for massive data transmission across the network, thus conserving bandwidth and minimizing latency, which are essential for maintaining a seamless user experience, especially in real-time applications. Subsequently, a parameter server aggregates these locally-trained model parameters from all users, generating a global FL model, which can be employed for various predictive and recommendation tasks.

While FL inherently enhances privacy by enabling local model training on the device and reducing the need to share raw data, it is susceptible to several privacy issues such as model inversion attacks, membership inference attacks, and eavesdropping during model update transmissions. Various privacy-enhancing technologies, such as differential privacy and homomorphic encryption, offer potential pathways to safeguard user privacy by obfuscating or encrypting transmitted data. The primary intent of our research is to develop incentive mechanisms to motivate user participation and computational resource contribution in FL.

The MEC paradigm and FL techniques show great advantage in protecting data privacy and training a recommendation model. By taking advantage of MEC and FL, many AI and recommendation system services can be better utilized and provided [11], [12], [13], [14]. In 2016, Google utilized federated learning to collect keyboard input data from multiple users' devices, and trained machine learning models locally using these data to improve the prediction and recommendation capabilities of Gboard [15]. However, in FL for recommendation systems, the performance of the ML model depends on many factors like the data quantity, data freshness, communication bandwidth, and the computation power provided by users [16]. It is also an important issue that not all the users voluntarily participate in the FL training. For instance, users suffer from limited local data do not want to join the FL. Therefore, there is an urgent need for an incentive mechanism to motivate users to participate in FL for recommendation systems.

In this paper, in order to incentive the users to contribute more computing resources to improve the performance of the FL model, we take advantage of game theory to model the FL training between the BS and users as a two-stage Stackelberg game in which the BS is the leader and the users are the followers. The BS equipped with server publishes an FL task with payment. The users in the coverage of the BS participate in the FL by contributing computing resources and local data samples. The targets of the BS and the users are to maximize their respective utilities. We theoretically analyze the existence of the Stackelberg equilibrium for the utilities maximization problem, and propose a game-based incentive mechanism algorithm (GIMA) to solve the utilities maximization problems of the users and BS.

The main contributions in this paper are as follows:

- We consider the FL training scenario in the MEC paradigm for recommendation systems. The users communicate with the BS over wireless networks. We design two utility functions for the BS and users. We formulate the utility maximization problems for the BS and users.
- We take advantage of game theory and theoretically analyze the Nash equilibrium of the users and the Stackelberg equilibrium of the problems. We further propose the game-based incentive mechanism algorithm (GIMA) to reach the Stackelberg equilibrium.
- We conduct extensive simulations to verify the performance of our GIMA algorithm. The experimental results show that our GIMA algorithm converges quickly, and can obtain a higher utility value in comparison with other incentive methods.

The rest of this paper is organized as follows. Section II reviews the related work. Section III presents the system model and problem formulation. Section IV gives the game model, proves the existence of the Stackelberg equilibrium and proposes the GIMA algorithm. Section V evaluates the performance of our GIMA algorithm, and Section VI concludes the paper and discusses the future direction.

II. Related Work

A. Federated Learning and Recommendation Systems

Thanks to its feature of privacy-preserving, FL has attracted a lot of attention in recent years [17]. Many studies have been devoted to optimizing the performance of FL. In [18], Dinh et al. proposed an FL algorithm named FEDL which could process heterogeneous data. The FEDL algorithm was only suitable for the problem with strongly convex and smooth loss functions. They further proposed a resource allocation optimization problem in wireless environment and aimed to minimize the FL convergence time and energy consumption. The work of [19] analyzed distributed gradient descent problem. In order to minimize the loss function under a given resource budget, the authors proposed a control algorithm to find the best trade-off between local updates and global parameter aggregation. In [20], Nishio and Yonetani studied the node selection problem and proposed a new FL protocol which can efficiently select clients based on their resource conditions. In [21], Bashir et al. emphasized the potential of FL in enhancing privacy and scalability in Metaverse healthcare, innovatively intertwining it with IoT, AI, and blockchain technologies. In [22], Pandya et al. comprehensively explored FL in smart cities, emphasizing its critical role in privacy preservation and scrutinizing its integration across various applications, while also examining impending challenges and future prospects.

In [23], Qi et al. proposed an MEC-based FL framework that allows distributed model training under privacy preserving conditions. In addition, the proposed FedCS protocol can manage clients for collaborative training, which alleviates the inefficiency of FL. Google employed federated learning to gather keyboard input data from multiple users' devices and trained machine learning models locally using the collected data to enhance the prediction and suggestion capabilities of Gboard. The utilization of federated learning technology ensured user data privacy by enabling the data to remain on their respective devices [15]. The development of language models also facilitates the development of recommendation systems. Moreover, combining recommendation systems with FL can provide better AI services [24]. Chen et al. studied the problem of training FL algorithms over a realistic wireless network [25]. They derived a closed-form expression for the expected convergence rate of the FL algorithm, and optimized the user selection and uplink RB allocation to minimize the FL loss function. Yang et al. provided an overview of FL applications for 6G wireless networks and discussed essential requirements, potential applications, associated problems and challenges in FL [26].

B. Incentive Mechanism in Federated Learning

In FL, it is important to encourage the users to participate in the FL, since the performance of the FL model depends on the sufficient data sample and other resource (e.g., computing resources). Therefore, there are many works studying the incentive mechanism in FL. In [27], the authors considered a blockchain-based MCS system, BSIF, that effectively utilized idle resources of widely used portable devices to perform sensing tasks, thus addressing the weaknesses of traditional, centralized MCS systems. In [28], the authors explored the central role of UAVs as edge intelligence enhancers within the MECN, highlighting in particular their contribution in providing communication, computation, and control services. In [29], the paper adopted a mixed game-based AoI optimization scheme, utilizing edge-based wireless technologies and AI-enabled diagnostic bots, to enhance timely infector detection during the COVID-19 pandemic.

In [30], Zeng et al. proposed a lightweight and incentive compatible incentive mechanism named FMore. They aimed to inspire more high-quality edge nodes with low cost to participate in FL and eventually improve the performance of FL. In [31], the authors proposed a social welfare maximization problem for cross-silo FL. A distributed algorithm based on incentive mechanism was designed to maximize the social welfare. In [32], Zhan et al. studied the impact of data

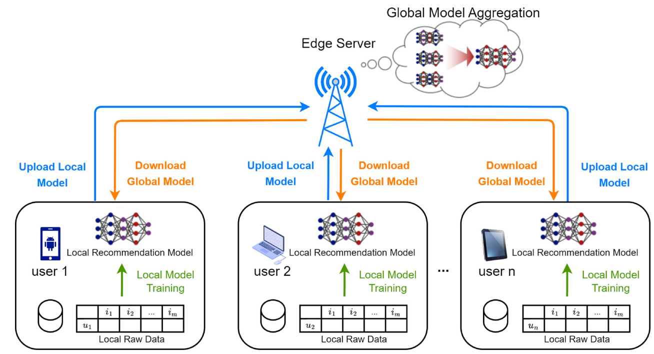


Fig. 1. The system model of FL for recommendation systems over wireless communication networks in MEC.

quantity on performance. A game-based incentive mechanism was proposed to encourage edge nodes to contribute their data samples. The authors aimed to maximize the utilities of the users. In [33], the authors modeled a reverse auction problem to encourage high-quality learning users to participate in FL, and they proposed a novel system named FAIR to optimize the global learning model. In [34], in order to maximize the social welfare of the FL services market, Jiao et al. proposed two auction mechanisms to incentive data owner to participate in FL. They also took the communication traffic congestion into consideration in the FL. Le et al. considered the scenario of wireless communication in [35]. The authors proposed a randomized auction framework for FL and aimed to minimize the energy consumption of each user.

Most previous works on FL for recommendation systems have overlooked the willingness of users to contribute their resources. Especially in resource-constrained MEC environments, the performance of the MEC systems might be significantly degraded if a portion of users are unlikely to provide their computing resources for training the AI model in FL framework. Our incentive mechanism of FL for recommendation systems in MEC, to be described next, is designed to fill this gap.

III. System Model and Problem Formulation

A. Federated Learning Model for Recommendation Systems

We consider a typical FL scenario for a recommendation system over a cellular network environment, shown in Fig. 1. It consists of one BS with edge server and N users denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$. Each user trains his own local recommendation model using local raw data collected from users. Afterwards, only the parameters of the local models are uploaded to the edge server for aggregation to obtain the global model. Finally, the aggregated global model is sent back to the user devices for inference and/or the next iteration of training. In this paper, similar to related works [32], [36], [37], we suppose that the data sample in FL is sufficient.

Formally, each user has a local dataset $\mathcal{D}_n = \{I_n^k, O^k_{n,1\le k\le D_n}\}$ including D_n data samples, where I_n^k denotes the input and O_n^k denotes the corresponding output. Let $\boldsymbol{w}$ denote the parameters of the FL model, and the loss function of user n with one sample is $h_n(\boldsymbol{w}, I_n^k, O_n^k)$, and the total loss function of user n is

$$H_n(\boldsymbol{w}) = \frac{1}{D_n}\sum_{k=1}^{D_n} h_n\left(\boldsymbol{w}, I_n^k, O_n^k\right). \tag{1}$$

Then, in order to obtain an FL model, the target of the FL training problem is to minimize the global loss function

$$\min_{\boldsymbol{w}} H(\boldsymbol{w}) = \frac{1}{D}\sum_{n=1}^{N}\sum_{k=1}^{D_n} h_n\left(\boldsymbol{w}, I_n^k, O_n^k\right), \tag{2}$$

where $D = \sum_{n=1}^{N} D_n$ is the total samples of all users.

In order to solve the problem (2), we adopt a conventional FL method from [38]. It is an iterative method that requires a number of global iterations to achieve a global accuracy level.

Specifically, in each iteration t, each user downloads the global model parameters $\boldsymbol{w}$ from the BS, and trains the local model with local dataset to solve the local FL problem, which can be expressed as:

$$\begin{aligned}&\min_{\boldsymbol{g}_n} \Phi_n\left(\boldsymbol{w}^t, \boldsymbol{g}_n\right)\\ &\quad = H_n\left(\boldsymbol{w}^t + \boldsymbol{g}_n\right) - \left(\nabla H_n(\boldsymbol{w}) - \gamma \nabla H\left(\boldsymbol{w}^t\right)\right)^T \boldsymbol{g}_n,\end{aligned} \tag{3}$$

where γ is a constant value, $\boldsymbol{g}_n$ denotes the solution of the problem (3), and $\boldsymbol{w}^t + \boldsymbol{g}_n$ is the local FL model parameters of user n at iteration t. Similar to [39], each user n adopts the gradient method to solve problem (3) with local accuracy θ that characterizes the quality of the local model, and produces the output $\boldsymbol{g}_n$ that satisfies

$$\Phi_n\left(\boldsymbol{w}^t, \boldsymbol{g}_n\right) - \Phi_n\left(\boldsymbol{w}^t, \boldsymbol{g}_n^*\right) < \theta\left(\Phi_n\left(\boldsymbol{w}^t, 0\right) - \Phi_n\left(\boldsymbol{w}^t, \boldsymbol{g}_n^*\right)\right). \tag{4}$$

Multiple local iterations are required to achieve the local accuracy. Then each user n transmits the local parameter $\boldsymbol{g}_n$ to the BS. Next, the BS aggregates the local parameters from the users and computes the global parameters, which can be expressed as:

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^t + \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{w}_n^t. \tag{5}$$

Finally, the BS broadcasts the global parameters to all users, which is used for the next round of iteration. This process is repeated until the global accuracy ε of (2) is achieved.

In order to obtain the number of local iterations and the number of global iterations, we consider that function $H_n(\boldsymbol{w})$ is $\pi - Lipschitza$ and $\nu - strongly$ convex, i.e.,

$$\nu \boldsymbol{I} \preceq H_n(\boldsymbol{w}) \preceq \pi \boldsymbol{I}, \quad \forall n \in \mathcal{N}. \tag{6}$$

Similar to [39], we can replace the assumption of γ-strong convexity with the γ-bounded nonconvexity condition for the nonconvex loss function. It has been proven that the regularized versions of $L_n(\boldsymbol{w}_n)$ and $L(\boldsymbol{w})$ satisfy γ-strong convexity and $(\pi + 2\gamma)$-Lipschitz properties. Therefore, the convergence analysis of convex loss function can be directly applied. Then, the general lower bound on the number of global iterations depends on the local accuracy ν and the global accuracy ε, which can be expressed as

$$I^g(\varepsilon, \nu) = \frac{C \log(1/\varepsilon)}{1 - \nu}. \tag{7}$$

TABLE I
NOTATIONS AND DEFINITIONS

Notations	Definitions
$\mathcal{N}$	set of users
N	number of users
h_n	loss function of user n with one sample
$H_n(\boldsymbol{w})$	total loss function of user n
$\boldsymbol{w}$	parameter of the global model
I^g	the number of global iterations
I_n^l	the number of local iterations
r_n	transmission rate of user n to the BS
T_n^{tr}	time of transmitting the local model of device n to BS
ξ	the data size of the local model
p_n	transmission power of user n
c_n	the density (CPU cycles for executing one data sample) of data samples
x_n	the number of data samples of user n
f_n	CPU frequency of user n
E_n^{tr}	the transmission energy consumption in each global iteration user n
E^{com}	the computation energy consumption in each global iteration user n
κ	the effective capacitance parameter
pe_n	the process efficiency for user n
η	payment value of the BS
λ	satisfaction level of the BS
ν_n	local accuracy of the user n
ε	global accuracy of the BS

Parameter C is a constant depending on the loss function [40]. For each user n, it is necessary that consuming additional resources to attain an accurate value of ν_n. The lower bound on the number of local iterations needed to achieve local accuracy ν_n, which is derived in [39], can be expressed as:

$$I_n^l = \vartheta_n \log\left(\frac{1}{\nu_n}\right), \tag{8}$$

where $\vartheta_n > 0$ is a parameter that depends on the parameters of $F_n(\boldsymbol{w})$ [39]. Similar to [41], we normalize $\vartheta_n = 1$. The upper bound of the I^g can be obtained based on the worst case of the users' local accuracy, which can be expressed as:

$$I^g = \frac{C \log(1/\varepsilon)}{1 - \max \nu_n}, \quad \forall n \in \mathcal{N}. \tag{9}$$

B. Communication and Computation Models of Federated Learning for Recommendation Systems

In each global iteration, the users transmit the local model parameters to the BS. We consider that the BS communicates with the users by FDMA-based scheme [42]. Therefore, there is no interference between different users. Based on the Shannon Theorem, the transmission rate r_n from user n to BS is expressed as

$$r_n = B_n \log_2\left(1 + \frac{p_n g_n}{B_n \sigma}\right), \tag{10}$$

where B_n denotes the bandwidth allocated to user n, p_n denotes transmission power, g_n is the uplink channel gain, and the white Gaussian noise is expressed by σ.

The transmission delay of the local model update from user n to the BS is

$$T_n^{tr} = \frac{\xi}{r_n}, \tag{11}$$

where ξ is the data size of the local model and it is the same for all users. Moreover, the transmission energy consumption in each global iteration is

$$E_n^{tr} = T_n^{tr} p_n = \frac{\xi p_n}{r_n}. \tag{12}$$

During each global iteration, each user downloads the global model parameters and trains the local model to achieve the level of local accuracy. The process of local model update needs local computing resources and energy. Therefore, for user n, the execution time of one round in local iteration is

$$\tau_n = \frac{c_n x_n}{f_n}, \tag{13}$$

where c_n is the density (CPU cycles for executing one data sample) of data samples, x_n is the number of data samples of user n, f_n is the computation resource that user n contribute to the FL model training. Specifically, the energy consumption in one local iteration is

$$E^{com} = \tau_n \kappa f_n^{\alpha} = c_n x_n \kappa f_n^{\alpha-1}, \tag{14}$$

where κ is the effective capacitance parameter for user n, and $\alpha \geq 2$ is the exponent parameter [43]. We define $\alpha = 2$ in this paper. Thus, we have

$$E^{com} = c_n x_n \kappa f_n. \tag{15}$$

Hence, the total energy consumption is composed of the energy consumed over I_g rounds of global iterations. Each of these global iterations consists of two main components: the energy consumed during I_n^l rounds of local iterative computations and the energy required for a single model transmission. Hence, the total energy consumption for a user n in the process of FL model training is

$$\begin{aligned} E_n^{tol} &= \left(I_n^l E_n^{com} + E_n^{tr}\right) I^g \\ &= \left(\log\left(\frac{1}{\nu_n}\right) E_n^{com} + E_n^{tr}\right) \frac{C \log(1/\varepsilon)}{1 - \max_{n \in \mathcal{N}} \nu_n}. \end{aligned} \tag{16}$$

C. Problem Formulation

In this part, we consider an incentive mechanism based on Stackelberg game to encourage the users to contribute computing resources. Therefore, in order to maximize the profits of the users and BS, we formulate two utility functions to represent the profits of the users and BS, respectively.

1) Utility of Users: From the users' perspective, the rewards are competitively obtained from the BS's payment, and the size of these rewards depends not only on their own participation level, but also on the participation level of other users. Their training costs include both computation and transmission energy consumption. The users may be heterogeneous in terms of their capacity to process data samples. Therefore, we define the process efficiency for user n as

$$pe_n = \frac{f_n}{c_n}, \tag{17}$$

which denotes the number of CPU cycles for processing one data sample. The BS publishes an FL training task with payment η. According to the η and other users' strategies, each user decides how much computing resources it contributes. Therefore, the utility of the user n is

$$u_n\left(f_n, \boldsymbol{f}_{-n}\right) = \frac{\frac{f_n}{c_n}}{\sum_{n=1}^{N} \frac{f_n}{c_n}} \eta - \rho E_n^{tol}, \tag{18}$$

where $\boldsymbol{f}_{-n} = \{f_1, f_2, \ldots, f_{n-1}, f_{n+1}, \ldots, f_N\}$ is the strategies of all users excluding user n, ρ is the conversion factors (e.g., joules to dollars), and ρE_n^{tol} denotes the cost. The user n earns payment according to its contribution.

2) Utility of BS: For the BS, the reward is derived from the computational resources contributed by the users. We use a logarithmic function to represent the diminishing marginal returns of the BS with respect to the computational resource contributed by the users, illustrating that as users contribute more computational resource, the additional utility or benefit received by the BS begins to decrease. The cost to the BS, on the other hand, is primarily the payment made to the users for their computational contributions and participation in FL. The BS publishes an FL training task with payment η, and the utility of the BS is

$$u_{BS}(\eta) = \lambda \log\left(1 + \sum_{n=1}^{N} \frac{f_n}{c_n}\right) - \eta, \tag{19}$$

where λ denotes the satisfaction level to the BS's FL task, and $\log(1 + \sum_{n=1}^{N} f_n/c_n)$ represents the BS's diminishing return on the processing capacity of users. This kind of utility function has been widely adopted to represent the utility in the other works [44], [45].

3) Utility Maximization: In our proposed two-stage Stackelberg game framework, the interactions between the parties involved in FL are highly complex. The BS needs to decide on an appropriate payment to attract users to participate to obtain a high-quality model (Eq. (20)). Users need to decide on their level of participation to maximize their utilities (Eq. (21)). Thus, the optimization problems can be expressed as:

$$BS\colon \max_{\eta} \left(u(\eta) = \lambda \log\left(1 + \sum_{n=1}^{N} \frac{f_n}{c_n}\right) - \eta\right). \tag{20}$$

$$User\colon \max_{f_n} u_n\left(f_n, \boldsymbol{f}_{-n}\right). \tag{21}$$

IV. Game-Based Incentive Mechanism for FL

In this section, we formulate the utility maximization problem to a two-stage Stackelberg game. In the second stage, we prove the existence of Nash equilibrium among users, and then, the Nash equilibrium strategies of the users can be determined based on the given payment. With the computing resources contributed by users, in the first stage, we analyse the existence of Stackelberg equilibrium between the users and BS, and we further determine the optimal payment value of the BS.

A. User Frequency Determination

In the intricately designed two-stage Stackelberg game, the methodology begins with user interactions in the second stage and aims to establish a Nash equilibrium among them. Users seek to maximize their utility, which is a trade-off between rewards and costs in FL. To ensure that no single user can unilaterally deviate to achieve higher utility without affecting others, we validate the existence of Nash equilibrium through rigorous mathematical proofs and logical consistency. Based on Eq. (18), each user competes with others to obtain more profits by contributing more computing resources. It is noted that the strategies among the users are coupled. In order to solve the problem (21), we propose a game-based approach to calculate the optimal strategies for the users in second stage.

Definition 1: $\boldsymbol{f}^* = (f_1^*, f_2^*, \ldots, f_N^*)$ is denoted as a Nash equilibrium of the game. In this case, there is no way for any player to further increase the utility by one-side changing its strategy under the state of Nash equilibrium $\boldsymbol{f}^*$, i.e.,

$$u_n\left(f_n, \boldsymbol{f}_{-n}^*\right) \leq u_n\left(f_n^*, \boldsymbol{f}_{-n}^*\right), \quad \forall n \in \mathcal{N}, \tag{22}$$

To study the Nash equilibrium of the users in FL with the given η, we derive the first order derivative of $u_n(f_n, \boldsymbol{f}_{-n})$ with respect to f_n, i.e.,

$$\frac{\partial u_n\left(f_n, \boldsymbol{f}_{-n}\right)}{\partial f_n} = \frac{\eta}{c_n \sum_{m=1}^N \frac{f_m}{c_m}} - \frac{f_n \eta}{\left(c_n \sum_{m=1}^N \frac{f_m}{c_m}\right)^2} - \rho c_n x_n \kappa \log\left(\frac{1}{\nu_n}\right) \frac{C \log(1/\varepsilon)}{1 - \max \nu_n}. \tag{23}$$

Then, we further derive the second-order derivative of $u_n(f_n, \boldsymbol{f}_{-n})$ with respect to f_n, i.e.,

$$\frac{\partial^2 u_n\left(f_n, \boldsymbol{f}_{-n}\right)}{\partial {f_n}^2} = -\frac{2\eta \sum_{m \neq n} \frac{f_m}{c_m}}{\left(\sum_{m=1}^N \frac{f_m}{c_m}\right)^3} < 0. \tag{24}$$

Lemma 1: The utility functions of the users are continues and convex.

Proof: According to Eq. (24), the second-order derivative of user n is negative in the domain of strategy f_n. Therefore, the utility function of each user is continues and convex. ■

Lemma 2: The strategy sets are convex.

Proof: We set $x_1, x_2 \in [0, f_n^{max}]$, and $\mu \in [0, 1]$. According to the definition of convex set, $\mu \mathbf{x}_1 + (1-\mu)\mathbf{x}_2 \in [0, f_n^{max}]$ always holds. Therefore, The strategy sets are convex. ■

Lemma 3: The strategy sets are compact.

Proof: $f_n \leq f_n^{max}$, where f_n^{max} is the user n's upper bound of the computing resources. Therefore, The strategy sets are compact. ■

Based on Lemma 1-3, we further propose that there exist a Nash equilibrium among the users. In this case, we let the first order derivative of user n equal to zero, i.e.,

$$\frac{\partial u_n\left(f_n, \boldsymbol{f}_{-n}\right)}{\partial f_n} = 0, \tag{25}$$

then, we can obtain the optimal strategy of the user n, which can be expressed as

$$f_n = \sqrt{\frac{\eta c_n \sum_{m \neq n} \frac{f_m}{c_m}}{b_n}} - \sum_{m \neq n} \frac{f_m}{c_m} c_n, \tag{26}$$

where $b_n = \rho c_n x_n \kappa \log(\frac{1}{\nu_n}) \frac{C \log(1/\varepsilon)}{1 - \max \nu_n}$ is a constant. If the right hand of Eq. (26) is negative, the user n does not contribute computing resources in the FL model training. If f_n is more than the upper bound f_n^{max}, then, the strategy of user n should be set to f_n^{max}. Therefore, we obtain

$$f_n = \begin{cases} 0, \quad \eta \leq \sum_{m \neq n} \frac{f_m}{c_m} b_n c_n; \\ \sqrt{\frac{\eta c_n \sum_{m \neq n} \frac{f_m}{c_m}}{b_n}} - \sum_{m \neq n} \frac{f_m}{c_m} c_n \quad 0 < f_n < f_n^{max}; \\ f_n^{max}, \quad \text{otherwise}. \end{cases} \tag{27}$$

Based on definition 1, when the game converge to the Nash equilibrium point, we can obtain the strategies of all users.

Theorem 1: The strategy set $(f_1^*, f_2^*, \ldots, f_M^*)$ is a Nash equilibrium, where

$$f_n^* = \frac{(M-1)\eta c_n}{\sum_{i=1}^M b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^M b_i c_i}\right). \tag{28}$$

Proof: According to Eq. (23), we have

$$\frac{\partial u_n\left(f_n, \boldsymbol{f}_{-n}\right)}{\partial f_n} = 0, \quad 0 < f_n < f_n^{max}, \tag{29}$$

then,

$$\eta \sum_{i=1}^M \frac{f_i^*}{c_i} - \frac{f_n^*}{c_n}\eta = b_n c_n \left(\sum_{i=1}^M \frac{f_i^*}{c_i}\right)^2. \tag{30}$$

For all users $n \in \mathcal{M} \subseteq \mathcal{N}$, we get

$$\begin{cases} \eta \sum_{i=1}^M \frac{f_i^*}{c_i} - \frac{f_1^*}{c_1}\eta = b_1 c_1 \left(\sum_{i=1}^M \frac{f_i^*}{c_i}\right)^2 \\ \eta \sum_{i=1}^M \frac{f_i^*}{c_i} - \frac{f_2^*}{c_2}\eta = b_2 c_2 \left(\sum_{i=1}^M \frac{f_i^*}{c_i}\right)^2 \\ \quad \vdots \\ \eta \sum_{i=1}^M \frac{f_i^*}{c_i} - \frac{f_M^*}{c_M}\eta = b_M c_M \left(\sum_{i=1}^M \frac{f_i^*}{c_i}\right)^2 \end{cases} \tag{31}$$

We sum up the Eq. (31) at the Nash equilibrium point as

$$M\eta \sum_{i=1}^M \frac{f_i^*}{c_i} - \sum_{i=1}^M \frac{f_i^*}{c_i}\eta = \sum_{i=1}^M b_m c_m \left(\sum_{i=1}^M \frac{f_i^*}{c_i}\right)^2. \tag{32}$$

Therefore, we obtain

$$\sum_{i=1}^M \frac{f_i^*}{c_i} = \frac{(M-1)\eta}{\sum_{m=1}^M b_m c_m}. \tag{33}$$

Then, we substitute Eq. (33) into Eq. (25) and obtain $f_n^* = \frac{(M-1)\eta c_n}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right)$. ■

Theorem 2: For each given η, the optimal solution to problem (21) is given by Theorem 1. Here, $\mathcal{S}$ is the set of optimal users to the FL, and the set can be represented as

$$\mathcal{S} = \left\{ v \in \mathcal{M} | b_n^v c_n^v < \frac{\sum_{i=1}^{v} b_i c_i}{v-1} \right\} \tag{34}$$

Proof: For each given η, the Nash equilibrium strategy of the users is positive. Therefore, let $\boldsymbol{f}^*$ be the optimal strategy to (21) and define $\mathcal{S} := \{v \in \mathcal{M} | f_v^* > 0\}$. We reorder the users such that $b_n^1 c_n^1 \leq b_n^2 c_n^2 \leq \ldots \leq b_n^M c_n^M$. If $b_n^{m_1} = c_n^{m_2}$, order them in the increasing order of v. Based on Eq. (28), we have

$$\frac{(M-1)\eta c_n}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right) > 0, \tag{35}$$

then, we obtain

$$b_n c_n < \frac{\sum_{i=1}^{M} b_i c_i}{M-1}. \tag{36}$$

Due to the ordering of the users, we know $m \in \mathcal{S} \Rightarrow m-1 \in \mathcal{S}$, and thus, $\mathcal{S}$ is of the form $\{1, 2, \ldots, V\}$ for some $V \leq M$. Let $\mathcal{S}' = \{1, \ldots, T\}$ be another such set where we assume that $T > M$. Then, $\{V+1, \ldots, T\} \notin \mathcal{S}$. Then, we have

$$\begin{aligned} b_n^{V+1} c_n^{V+1} &\geq \frac{\sum_{i=1}^{V} b_i c_i}{V-1} \\ \Rightarrow b_n^{V+1} c_n^{V+1}(V-1) &\geq \sum_{i=1}^{V} b_i c_i \\ \Rightarrow b_n^{V+1} c_n^{V+1}(V+1-1) &\geq \sum_{i=1}^{V+1} b_i c_i \\ \Rightarrow b_n^{V+1} c_n^{V+1} &\geq \frac{\sum_{i=1}^{V+1} b_i c_i}{V}. \end{aligned} \tag{37}$$

Since $b_n^1 c_n^1 \geq b_n^2 c_n^2$, we can repeat this process and derive

$$b_n^M c_n^M \geq \frac{\sum_{i=1}^{M} b_i c_i}{M-1}, \tag{38}$$

which is a contradiction to the definition of $\mathcal{S}'$. Therefore, in the case of $b_n^1 c_n^1 \leq b_n^2 c_n^2 \leq \ldots \leq b_n^M c_n^M$, $\mathcal{S}$ is of the form $\mathcal{S} = \left\{ v \in \mathcal{M} | b_n^v c_n^v < \frac{\sum_{i=1}^{v} b_i c_i}{v-1} \right\}$. ■

Theorem 3: The Nash equilibrium strategy set $(f_1^*, f_2^*, \ldots, f_M^*)$ is the optimal strategy set of the second stage.

Proof: We substitute the Eq. (33) into Eq. (23), and let the first order derivative equal zero, then, we obtain

$$\begin{aligned} \sum_{i \neq n} \frac{f_i^*}{c_i} &= \frac{b_n c_n}{\eta}\left(\sum_{i=1}^{M} \frac{f_i^*}{c_i}\right)^2 \\ &= \frac{b_n c_n}{\eta}\left(\frac{(M-1)\eta}{\sum_{m=1}^{M} b_m c_m}\right)^2. \end{aligned} \tag{39}$$

Then, substituting the Eq. (39) into Eq. (26), we have

$$\begin{aligned} f_n^* &= \sqrt{\frac{\eta c_n \sum_{m \neq n} \frac{f_m^*}{c_m}}{b_n}} - \sum_{m \neq n} \frac{f_m^*}{c_m} c_n \\ &= \frac{(M-1)\eta c_n}{\sum_{i \in \mathcal{M}} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i \in \mathcal{M}} b_i c_i}\right), \end{aligned} \tag{40}$$

which means Eq. (28) equal the Eq. (26) in the case of Nash equilibrium. Therefore, The Nash equilibrium strategy set is the optimal strategy set of the second stage. ■

Theorem 4: In each iteration of the game, half of the users are eliminated.

Proof: Based on Eq. (27), the user n gives up this opportunity to participate the FL training in the condition of $f_n < 0$ with the given η. In this case, when the number of remaining users M is relatively large, and the parameters b_n and c_n are obtained randomly from a range for the reason of heterogeneity of users. According to Eq. (28), there exist a probability of 50% to eliminate the user n when its parameters meet the following condition:

$$b_n c_n (M-1) \geq \sum_{i=1}^{M} b_i c_i. \tag{41}$$

Therefore,

$$1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i} \leq 0. \tag{42}$$

Then, the user n is eliminated because of $f_n \leq 0$. ■

B. BS Payment Determination

In the first stage, the BS, as the leader, determines the optimal payment to the users. It aims to maximize its utility, while taking into account user participation and the effectiveness of global model updates to design incentive mechanisms that ensure user contributions. Establishing the Stackelberg equilibrium involves confirming that the BS, while anticipating user strategies, sets an optimal payment strategy, and subsequently users respond optimally to that payment, ensuring a stable, sequentially rational outcome. There exist a Nash equilibrium among users with any given payment value η. Therefore, the BS can determine a best payment value for the first stage of Stackelberg game in order to maximize the utility.

Theorem 5: Based on Theorem 1, with the best response of the users, the best strategy of the BS is

$$\eta^* = 1 - \frac{1}{\sum_{n=1}^{M} \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right)}. \tag{43}$$

Proof: According to Eq. (19), we have

$$\begin{aligned} \frac{\partial u_{BS}(\eta)}{\partial \eta} &= \frac{\lambda \sum_{i=1}^{M} \frac{1}{c_i} \frac{\partial f_i}{\partial \eta}}{1 + \sum_{i=1}^{N} \frac{f_i}{c_i}} - 1 \\ &= \frac{\lambda \sum_{n=1}^{M} \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right)}{1 + \sum_{i=1}^{N} \frac{f_i}{c_i}} - 1. \end{aligned} \tag{44}$$

Then, we further derive the second-order derivative of $u_{BS}(\eta)$ with respect to η as

$$\frac{\partial^2 u_{BS}(\eta)}{\partial \eta^2} = -\lambda \frac{1}{\left(1+\sum_{i=1}^{N} \frac{f_i}{c_i}\right)^2}\left(\sum_{i=1}^{M} \frac{1}{c_i}\frac{\partial f_i}{\partial \eta}\right)^2 + \lambda \frac{1}{1+\sum_{i=1}^{N} \frac{f_i}{c_i}} \sum_{i=1}^{N} \frac{1}{c_i}\frac{\partial^2 f_i(\eta)}{\partial \eta^2}. \tag{45}$$

Based on Eq. (28), we have $\frac{\partial^2 f_i(\eta)}{\partial \eta^2} = 0$, and the third part of the Eq. (45) is zero. According to the above analysis, we derive that the second order derivative of $u_{BS}(\eta)$ is negative, i.e., $\frac{\partial^2 u_{BS}(\eta)}{\partial \eta^2} < 0$.

The domain of η is $[0, \infty)$, and we define $G(\eta) = \frac{\partial^2 u_{BS}(\eta)}{\partial \eta^2}$, then, we have

$$\begin{cases} G(0) = 0 \\ G(\infty) = -\infty \end{cases} \tag{46}$$

Since the second order derivative of $u_{BS}(\eta)$ with respect to η is negative, the utility function of BS is convex. Therefore, there exist a unique payment value of BS at Stackelberg equilibrium point.

Furthermore, let the first order derivative of $u_{BS}(\eta)$ equal to zero, i.e.,

$$\frac{\partial u_{BS}(\eta)}{\partial \eta} = 0, \tag{47}$$

then, we have

$$\sum_{i=1}^{M} \frac{1}{c_i}\frac{\partial f_i}{\partial \eta} = 1 + \sum_{i=1}^{M} \frac{f_i}{c_i}. \tag{48}$$

We substitute Eq. (33) into Eq. (48) as

$$\sum_{n=1}^{M} \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right) = 1 + \sum_{n=1}^{M} \frac{(M-1)\eta}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right). \tag{49}$$

In the end, we obtain the best payment value of the BS is $1 - 1/\sum_{n=1}^{M} \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\left(1 - b_n c_n \frac{(M-1)}{\sum_{i=1}^{M} b_i c_i}\right)$. ■

C. Game-Based Incentive Mechanism Algorithm Design

We propose the Game-based Incentive Mechanism Algorithm (GIMA) as shown in Algorithm 1. t stands for the integrative epoch ID. GIMA is primarily constituted of several important stages to ensure a coherent and effective incentive mechanism in FL. Initially, it includes an information initialization and broadcasting phase, wherein relevant parameters and information are established and broadcast through the network to set the stage for subsequent computations. Following this setup, the algorithm advances to a crucial step where both the BS and users independently compute their optimal solutions, pursuing their respective objectives of utility maximization and cost minimization. It is noted that the users and BS operate under a sequential decision-making process, ensuring that the respective strategies are mutually informed and adaptive to one another's decisions. Finally, GIMA converges on a final analytic solution through an iterative process that continuously updates and optimizes the responses of the BS and users. This solution inherently balances the BS and users' utility maximization and computational cost considerations.

Algorithm 1: Ordering-Based Incentive Mechanism Algorithm (GIMA)

```
1  Initialize the iterative epoch t = 0 and the number of
   the users |S(0)| = N.
2  for n ∈ S(t) do
3      each device n sends b_n and c_n to the BS.
4  repeat
5      The BS determines η*(t) based on Eq. (43).
6      if η*(t) < 0 then
7          End the whole game.
8      else
9          Select the optimal users set S based on Eq. (34).
10         Compute u_n(t) and f_n^*(t) based on Eq. (18) and
           Eq. (28).
11         if f_n^*(t) > f_n^max then
12             f_n^*(t) = f_n^max
13         .
14     S(t + 1) = S(t)
15     t = t + 1
16 until The set M is no change for I continuous epochs;
```

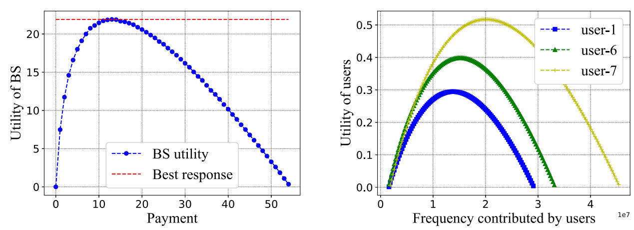
At the beginning of the GIMA, we initialize all the users participating the FL model training (Line 1). Then, all the users send the local parameters to the BS, i.e., the density of data samples c_n and the system parameter b_n (Line 2-3). The BS calculates the optimal payment strategy based on Eq. (46) (Line 5). In the case of $\eta^* < 0$, the BS gives up the opportunity to publish the FL model, otherwise, the BS sends η^* to all users (Line 6-8). The time complexity of the above operation is $\mathcal{O}(1)$.

Then, based on η^*, the BS needs to determine the optimal users set based on Theorem 2. The BS reorders the users with $b_n c_n$ in the increasing order and selects the optimal users set according to Eq. (34). The sorting operation can actually be performed outside the loop, where the computational complexity is $\mathcal{O}(N \log N)$ (Line 9). Once the BS determines the optimal users set, the optimal computing resources can be determined according to Eq. (28). If f_n^* is greater than f_n^{max}, the user will contribute all the available computing resources, where the computational complexity is $\mathcal{O}(1)$ (Line 10-12).

Finally, if the set $\mathcal{S}$ is no change for I continuous epochs, which means no users or BS can further increase utility. Then, the BS broadcasts the optimal strategy of the computing resources to all users (Line 14-16). In the worst case, the game iterates N times until there are no users remaining. Therefore, the computational complexity of GIMA is $\mathcal{O}(N \log N + N)$.

TABLE II
EXPERIMENT SETUP

Parameters	Value
c_n (density of data samples)	$U \sim [1,3] \times 10^8$ cycles/bit
x_n (number of data samples)	$U \sim [500, 1000]$
κ (switched capacitance parameter)	10^{-28}
B_n (bandwidth of user n)	30 kHz
σ (background noise)	-100 dBm
ξ (size of local model)	0.16 Mbits
p_n (transmission power)	$U \sim [100, 200]$ mW
d_n (The distance of user n)	$U \sim [100, 200]$ m
ν_n (local accuracy)	$U \sim [0.5, 0.9]$
ε (global accuracy)	0.5



(a) The utility of BS with different payment value (b) The utilities of users with different frequencies.

Fig. 2. The impact of different strategies.

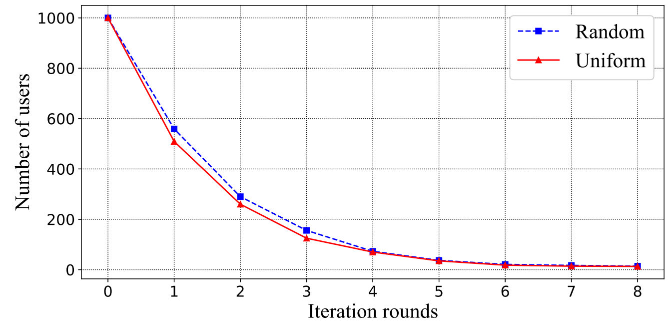
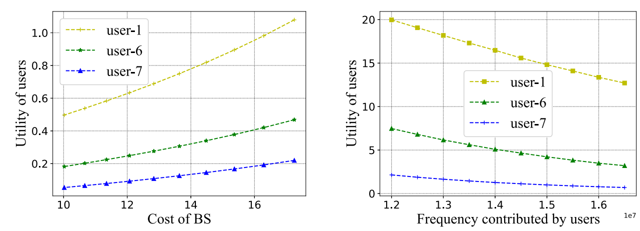


Fig. 3. Number of users with each iteration.



(a) The utilities of users with different payment value (b) The utilities of users with different frequency

Fig. 4. The utilities of users with different strategies.

V. PERFORMANCE ANALYSIS

A. Experimental Setup

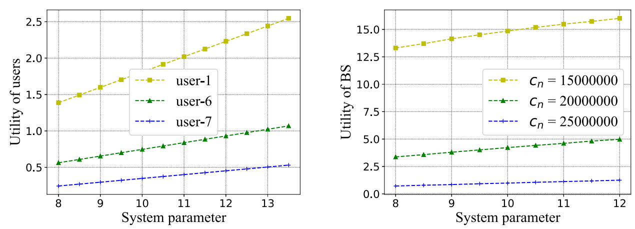
In this section, we conduct simulation experiments to validate our GIMA algorithm. We consider a cellular network that includes one BS and multiple users. The users are randomly distributed in the BS's coverage area. The parameter settings are as follows. The size of the FL model is 0.16 Mbits. The number of data samples is uniformly distributed between [500,1000]. The effective switched capacitance parameter κ is 10^{-28} [32]. The channel bandwidth allocated to each user is 30 kHz and the transmission power is uniformly distributed between [100, 200] mW. The channel gain is set to d_n^{-3}, where d_n is the distance between user n and the BS. The local accuracy is uniformly distributed between [0.5, 0.9] and the global accuracy is 0.5. The background noise is -100 dBm [39]. The detailed description of the parameters are listed in Table II. Unless otherwise noted, subsequent experiments were conducted adhering to the aforementioned parameters.

B. Parameter Analysis

Fig. 2 shows that there exist the unique Stackelberg equilibrium strategies for the BS and the users. We set $N = 30$ to execute the experiment. At the Stackelberg equilibrium point, the optimal payment is 13.7 and the utility value is 26.4. In order to validate the existence of the unique Stackelberg equilibrium, we vary the payment value from 0 to 50 with fixed user number and users' strategies. As illustrated in Fig. 2(a), the curve of the BS's utility is convex, and the highest point of the curve is the best response of the Stackelberg equilibrium. We also validate the unique Nash equilibrium among the users. As shown in Fig. 2(b), for each user, we vary the frequency with fixed strategies of other users and the payment of the BS. It is noted that the utility curve of each user is convex. In this case, the users can maximize the utility by determining the strategies of computing resources.

Fig. 3 shows that the remaining number of users after several iterations. In the case of uniform distribution of user parameters, half of the users are eliminated based on Theorem 4. In our simulated experiments, we obtain the parameters from a domain randomly, and we compare the remaining user number of our GIMA algorithm with the uniform distribution method. We can see that the two ways of parameter selection have the same downward trend in terms of the number of remaining users. The results also demonstrate that the GIMA algorithm can reach the Stackelberg equilibrium quickly. This shows the scalability of our GIMA algorithm, which is important for real-world and large-scale system.

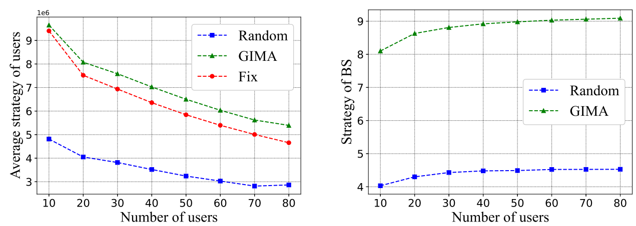
Fig. 4 shows the variations of the users' utilities with different strategies of BS and users. Fig. 4(a) shows that the utility value of the users with different payment from the BS. We illustrate the variation in utility for three users, i.e., user-1, user-6 and user-7, as the payment value changes from 10 to 17. Obviously, all the users' utility valve is increasing with the increase of the payment. Fig. 4(b) shows that the utility value of the users with different computing frequency. It is clear that the utilities of the users decrease as the frequency increase. This is because more frequency contribution results in more consumption of energy. Therefore, the utilities of all users are monotonically decrease. As can be observed, the slope of user-1 is greater than that of user-6, which in turn is greater than that of user-7. This is a consequence of the competitive dynamics among the users. After the BS provides a payment, each user obtains their respective optimal solutions. However,



(a) The utilities of users with different λ

(b) The utility of BS with different λ

Fig. 5. The utilities of BS and users with different λ.



(a) Average frequency with different numbers of users

(b) Payment of BS with different numbers of users

Fig. 6. The strategies of BS and users with different number of users.

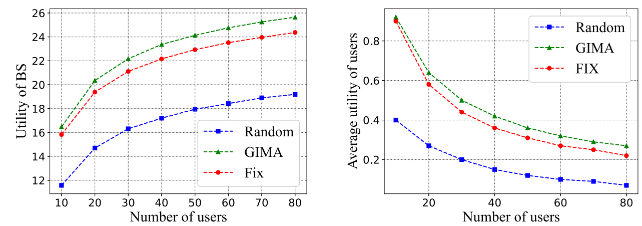
due to the inequalities $b_1 c_1 < b_6 c_6 < b_7 c_7$, user-1 is in a more advantageous position in the competition, enabling it to achieve higher utility.

Fig. 5 shows the impact of satisfaction level λ on the utilities of BS and users. Firstly, we illustrate the variation in utility for the users with the fixed payment value η, and the satisfaction level is varied from 10 to 17.5. It can be observed that the utilities of users increase with the increase of satisfaction level λ in Fig. 5(a). Actually, as the satisfaction level increases further, the BS is willing to pay more for the computing resources. Furthermore, the user-1 contributes the most computing resources, and with the increase of the satisfaction level, the growth speed of the user-1 is the fastest in terms on the utility value. Fig. 5(b) shows that the utility of BS increases with the level of satisfaction. In order to show the impact of data samples' density on the utility of the BS, we set the same data samples' density for all the users. As the value of the data samples' density increases, we can see that the utility value of BS is decreasing. This is because the users need more computing resources to process one data sample, as the density value of the data sample increases.

C. Comparison Experiment

In this section, we perform comparison experiments using the following two comparison algorithms to further evaluate the superiority of our proposed GIMA algorithm.

- *Random:* The BS publishes an FL training task with a random η, then, all the users to determine their strategies with the payment η. We first reach the Stackelberg equilibrium based on our GIMA algorithm, and we determine the optimal strategies η^* and $\boldsymbol{f}^*$ of the BS and the users respectively. Then, we randomly choose a payment from 0 to η^*. The users redetermine their strategies with the payment η^*.



(a) utility of BS with different numbers of devices

(b) Average utility of users with different numbers of devices

Fig. 7. The utilities of BS and users with different number of users.

- *Fix:* The BS publishes an FL training task with a fixed $\eta = 10$, then, all the users reach the Nash equilibrium based on the payment η.

Fig. 6 shows the strategies of BS and users with different number of users. Fig. 6(a) illustrates that the average frequency of users for the three algorithms decreases as the number of users increases. This is because more and more users participate in the FL training and divide the payment as the number of users raises. Therefore, the average amount of payment gained by users become smaller and they are not willing to contribute more computing resources. In addition, it is noted that the user's average frequency of our GIMA algorithm is always highest. This also demonstrates that our proposed GIMA algorithm has good performance to inspire users to contribute computing resources.

Fig. 6(b) shows the BS's strategies of the GIMA algorithm and Random algorithm. We can see that our GIMA algorithm provide the highest payment to the users. In addition, with the user number increase, the growth speed of both algorithms is getting slower and slower. This is because when the computing resources is already satisfied, the BS cannot increase its expenses indefinitely to ensure its revenue.

Fig. 7 shows the utilities of the BS and the users. Fig. 7(a) shows the BS's utility of the Random algorithm, the GIMA algorithm and the Fix algorithm with different user number. We can see that the BS utilities of our GIMA algorithm is the largest among the three algorithms. With the number of users increases, the gap of the utility between the GIMA algorithm and the other two algorithm is getting wider. Furthermore, it is noted that the growth speed of our GIMA algorithm is slower than linear growth. Actually, as the user number increases further, more and more users participate in the FL training, and the BS can obtain large amount of computing resources. However, in order to obtain the maximum utility, the amount of computing resources needed for BS will stabilize.

Fig. 7(b) illustrates the users' average utilities of the three algorithms with different user numbers. We can see that as the user numbers increases, the users' average utilities decreases. This is because with the number of users increases, more and more users participate in the FL training while the growth speed of the payment is flat. Therefore, the average utilities of the users become smaller. It is obvious that the BS's utility of our GIMA algorithm is the largest among the three algorithms, and the gap of the users' average utilities between the GIMA algorithm and the other two algorithm is getting wider.

The main factors influencing the superior performance of GIMA are its foundation in the two-stage Stackelberg game, which inherently allows it to balance utility maximization for both users and the BS by systematically considering their respective cost-benefit trade-offs in a hierarchical decision-making process. This design effectively encourages user participation and maximizes the utility of the BS, thereby providing a robust solution to our optimization problems.

VI. Conclusion

In this paper, we investigate an MEC system cooperated with FL for an recommendation system in the coverage area of BS. The BS publishes FL tasks with payment, and the users compete for profit by contributing computing resources and local data. We design two utility functions for the users and BS. We also formulate the utility maximize problem with a two-stage Stackelberg game, and theoretically prove the existence of Stackelberg equilibrium. Then, the GIMA algorithm is proposed to obtain the equilibrium strategies for the users and BS in a limited number of iterations. In the evaluation part, through simulation experiments, we illustrate that the GIMA algorithm can converge quickly and achieve higher utility value compared with other incentive methods. For our future work, we will consider the impact of freshness of the data and inspire the users to provide fresh data to train a more accurate model. In our future work, we will further consider the insufficient data sample and study the adaptive framework of incentive mechanism for FL. Moreover, we will explore various incentive mechanism design approaches from the economic perspective, such as reverse auction.

References

[1] M. Majid et al., "Applications of wireless sensor networks and Internet of Things frameworks in the industry revolution 4.0: A systematic literature review," *Sensors*, vol. 22, no. 6, p. 2087, 2022.

[2] P. K. R. Maddikunta et al., "Industry 5.0: A survey on enabling technologies and potential applications," *J. Ind. Inf. Integr.*, vol. 26, Mar. 2022, Art. no. 100257.

[3] C. K. Wu, C.-T. Cheng, Y. Uwate, G. Chen, S. Mumtaz, and K. F. Tsang, "State-of-the-art and research opportunities for next-generation consumer electronics," *IEEE Trans. Consum. Electron.*, early access, Dec. 27, 2022, doi: 10.1109/TCE.2022.3232478.

[4] Y. Chen, J. Zhao, Y. Wu, J. Huang, and X. Shen, "QoE-aware decentralized task offloading and resource allocation for end-edge-cloud systems: A game-theoretical approach," *IEEE Trans. Mobile Comput.*, vol. 23, no. 1, pp. 769–784, Jan. 2024.

[5] D. Ayata, Y. Yaslan, and M. E. Kamasak, "Emotion based music recommendation system using wearable physiological sensors," *IEEE Trans. Consum. Electron.*, vol. 64, no. 2, pp. 196–203, May 2018.

[6] P. Tiwari, A. Lakhan, R. H. Jhaveri, and T.-M. Gronli, "Consumer-centric Internet of Medical Things for cyborg applications based on federated reinforcement learning," *IEEE Trans. Consum. Electron.*, early access, Feb. 7, 2023, doi: 10.1109/TCE.2023.3242375.

[7] D. Zeng, L. Gu, S. Pan, J. Cai, and S. Guo, "Resource management at the network edge: A deep reinforcement learning approach," *IEEE Netw.*, vol. 33, no. 3, pp. 26–33, May/Jun. 2019.

[8] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.

[9] Y. Chen, J. Hu, J. Zhao, and G. Min, "QoS-aware computation offloading in LEO satellite edge computing for IoT: A game-theoretical approach," *Chin. J. Electron.*, vol. 33, pp. 1–12, Jul. 2023.

[10] H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, "Federated learning of deep networks using model averaging," 2016, *arXiv:1602.05629*.

[11] P. Boobalan et al., "Fusion of federated learning and Industrial Internet of Things: A survey," *Comput. Netw.*, vol. 212, Jul. 2022, Art. no. 109048.

[12] T. Wang, J. Jin, X. Gu, Y. Zhang, and X. Chen, "PREEN: An aggregation mechanism for privacy-preserving smart-metering communications," *IEEE Trans. Consum. Electron.*, early access, Nov. 28, 2022, doi: 10.1109/TCE.2022.3225088.

[13] F. Liu, J. Huang, and X. Wang, "Joint task offloading and resource allocation for device-edge-cloud collaboration with subtask dependencies," *IEEE Trans. Cloud Comput.*, vol. 11, no. 3, pp. 3027–3039, Jul.–Sep. 2023.

[14] Y. Chen, J. Zhao, J. Hu, S. Wan, and J. Huang, "Distributed task offloading and resource purchasing in NOMA-enabled mobile edge computing: Hierarchical game theoretical approaches," *ACM Trans. Embed. Comput. Syst.*, early access, May 16, 2023, doi: 10.1145/3597023.

[15] A. Hard et al., "Federated learning for mobile keyboard prediction," 2018, *arXiv:1811.03604*.

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

[17] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[18] C. T. Dinh et al., "Federated learning over wireless networks: Convergence analysis and resource allocation," *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 398–409, Feb. 2020.

[19] S. Wang et al., "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 63–71.

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

[21] A. K. Bashir et al., "Federated learning for the healthcare metaverse: Concepts, applications, challenges, and future directions," *IEEE Internet Things J.*, vol. 10, no. 24, pp. 21873–21891, Dec. 2023.

[22] S. Pandya et al., "Federated learning for smart cities: A comprehensive survey," *Sustain. Energy Technol. Assess.*, vol. 55, Feb. 2023, Art. no. 102987.

[23] T. Qi, F. Wu, C. Wu, Y. Huang, and X. Xie, "Privacy-preserving news recommendation model learning," 2020, *arXiv:2003.09592*.

[24] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, "Three approaches for personalization with applications to federated learning," 2020, *arXiv:2002.10619*.

[25] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.

[26] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6G: Applications, challenges, and opportunities," *Engineering*, vol. 8, pp. 33–41, Jan. 2022.

[27] W. Wang et al., "BSIF: Blockchain-based secure, interactive, and fair mobile crowdsensing," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 12, pp. 3452–3469, Dec. 2022.

[28] W. Wang, G. Srivastava, J. C.-W. Lin, Y. Yang, M. Alazab, and T. R. Gadekallu, "Data freshness optimization under CAA in the UAV-aided MECN: A potential game perspective," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 12912–12921, Nov. 2023.

[29] Y. Yang et al., "Mixed game-based AoI optimization for combating COVID-19 with AI bots," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 11, pp. 3122–3138, Nov. 2022.

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

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

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

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

[34] Y. Jiao, P. Wang, D. Niyato, B. Lin, and D. I. Kim, "Toward an automated auction framework for wireless federated learning services market," *IEEE Trans. Mobile Comput.*, vol. 20, no. 10, pp. 3034–3048, Oct. 2021.

[35] T. H. T. Le, N. H. Tran, Y. K. Tun, Z. Han, and C. S. Hong, "Auction based incentive design for efficient federated learning in cellular wireless networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2020, pp. 1–6.

[36] M. Shafiq, R. Yadav, A. R. Javed, and S. A. H. Mohsin, "CoopGBFS: A federated learning and game-theoretic based approach for personalized security, recommendation in 5G beyond IoT environments for consumer electronics," *IEEE Trans. Consum. Electron.*, early access, Aug. 15, 2023, doi: 10.1109/TCE.2023.3305508.

[37] W. Y. B. Lim et al., "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020.

[38] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained IoT devices," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 1–24, Jan. 2022.

[39] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1935–1949, Mar. 2021.

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

[41] T. H. T. Le et al., "An incentive mechanism for federated learning in wireless cellular network: An auction approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 4874–4887, Aug. 2021.

[42] T. Mahn, H. Al-Shatri, and A. Klein, "Distributed algorithm for energy efficient joint cloud and edge computing with Splittable tasks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2019, pp. 1–7.

[43] S. Bi, L. Huang, and Y.-J. A. Zhang, "Joint optimization of service caching placement and computation offloading in mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 7, pp. 4947–4963, Jul. 2020.

[44] Y. Zhan, Y. Xia, J. Zhang, T. Li, and Y. Wang, "An incentive mechanism design for mobile crowdsensing with demand uncertainties," *Inf. Sci.*, vol. 528, pp. 1–16, Aug. 2020.

[45] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1732–1744, Jun. 2016.