Enabling Long-Term Cooperation in Cross-Silo Federated Learning: A Repeated Game Perspective

Ning Zhang, *Student Member, IEEE,* Qian Ma, *Member, IEEE,* and Xu Chen, *Senior Member, IEEE*

Abstract—Cross-silo federated learning (FL) is a distributed learning approach where clients of the same interest train a global model cooperatively while keeping their local data private. The success of a cross-silo FL process requires active participation of many clients. Different from cross-device FL, clients in cross-silo FL are usually organizations or companies which may execute multiple cross-silo FL processes repeatedly due to their time-varying local data sets, and aim to optimize their long-term benefits by selfishly choosing their participation levels. While there has been some work on incentivizing clients to join FL, the analysis of clients' long-term selfish participation behaviors in cross-silo FL remains largely unexplored. In this paper, we analyze the selfish participation behaviors of heterogeneous clients in cross-silo FL. Specifically, we model clients' long-term selfish participation behaviors as an infinitely repeated game, with the stage game being a selfish participation game in one cross-silo FL process (SPFL). For the stage game SPFL, we derive the unique Nash equilibrium (NE), and propose a distributed algorithm for each client to calculate its equilibrium participation strategy. We show that at the NE, clients fall into at most three categories: (i) *free riders* who do not perform local model training, (ii) a unique *partial contributor* (if exists) who performs model training with part of its local data, and (iii) *contributors* who perform model training with all their local data. The existence of free riders has a detrimental effect on achieving a good global model and sustaining other clients' long-term participation. For the long-term interactions among clients, we derive a cooperative strategy for clients which minimizes the number of free riders while increasing the amount of local data for model training. We show that enforced by a punishment strategy, such a cooperative strategy is a subgame perfect Nash equilibrium (SPNE) of the infinitely repeated game, under which some clients who are free riders at the NE of the stage game choose to be (partial) contributors. We further propose an algorithm to calculate the optimal SPNE which minimizes the number of free riders while maximizing the amount of local data for model training. Simulation results show that our proposed cooperative strategy at the optimal SPNE can effectively reduce the number of free riders by up to 98.8% and increase the amount of local data for model training by up to 96%.

Index Terms—Cross-silo federated learning, selfish participation, free rider, long-term cooperation

1 Introduction

1.1 Background and Motivations

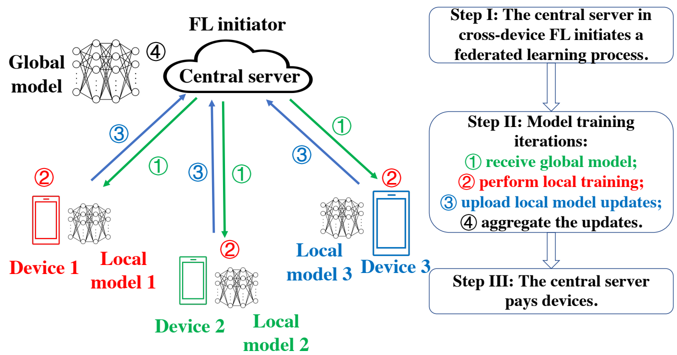
The rapid development of 5G and Internet of Things (IoT) technologies accelerates the generation of massive amount of user data [1], such as users' finance data in different banks and patients' clinical data in different hospitals. User data is of paramount importance for artificial intelligence (AI). With sufficient user data, banks can develop AI models for customized financial advice and risk control services, and hospitals can train AI-assisted diagnosis and treatment models. Traditional machine learning approaches usually collect large amounts of raw data and train AI models on a central server, which may lead to privacy leakage. In order to protect data privacy, Google proposed federated learning (FL) [2], which is a distributed learning approach without sharing raw data.

In FL, a central server coordinates the model training of many clients [3]. In the local training steps, each client trains the model by using its local data, and then sends the updated model to the central server. In the global aggregation steps, the central server aggregates the updates and sends the aggregated global model back to clients for the next iteration. The iterations stop until a predefined stopping criterion is satisfied.

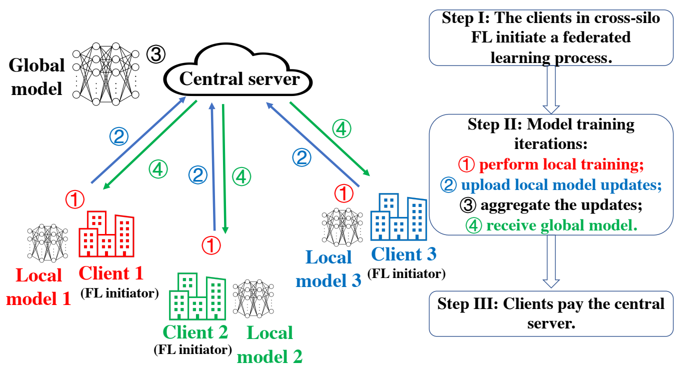
Depending on the initiator of the FL process and the type of clients, FL is classified into two types: cross-device FL and cross-silo FL [4]. In cross-device FL, as shown in Fig. 1(a), the central server initiates the FL process, and clients are devices who perform local training using their local data. The central server pays each client a reward as the incentive for local training. In cross-silo FL, as shown in Fig. 1(b), clients who are usually companies or organizations of the same interest, initiate the FL process and pay the central server for global aggregation.

In this paper, we study cross-silo FL, which has a wide range of practical applications. For example, WeBank and Swiss Re cooperatively perform data analysis for finance and insurance services [5]. NVIDIA Clara helps hospitals with different data sets train AI models by FL for mammogram assessment [6]. Owkin collaborates with medical institutions and pharmaceutical companies to conduct distributed biomedical data analysis [7]. MELLODDY uses cross-silo FL to speed up drug research [8]. In cross-silo FL, many clients who have the same interest (e.g., training models for finance service or medical diagnosis) cooperatively train a global model by performing model training

- N. Zhang and Q. Ma are with the School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou 510006, China. E-mail: zhangn87@mail2.sysu.edu.cn, maqian25@mail.sysu.edu.cn.
- X. Chen is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510006, China. E-mail: chenxu35@mail.sysu.edu.cn.



(a) cross-device FL



(b) cross-silo FL

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

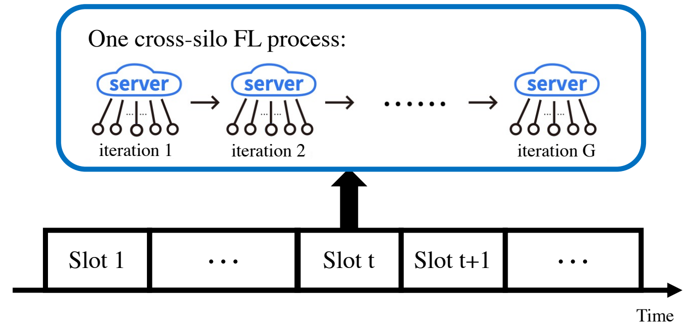


Fig. 2. The cross-silo FL processes in an infinite time horizon

with their local data, and hence usually can achieve a better model compared with the ones trained only locally [4]. Despite the popular applications of cross-silo FL seen in practice, there is little work analyzing clients' participation behaviors in cross-silo FL theoretically, which is the focus of this paper.

In cross-silo FL, since clients usually belong to different entities (e.g., companies or organizations), each client may behave selfishly to maximize its own benefit. Specifically, each client selfishly chooses its participation level (i.e., the amount of local data) for the local training steps. The total amount of local data chosen by all clients will affect the global model accuracy, while the model training leads to costs to clients. In practice, clients have heterogeneous valuations for global model accuracy [9], and will make a tradeoff between the achieved global model accuracy and the incurred cost. For example, clients with high valuations for global model accuracy may choose a high participation level to achieve a good global model accuracy. On the other hand, clients with low valuations for global model accuracy may choose not to perform local training to reduce the cost, and we call such clients as *free riders*. This motivates us to study clients' selfish participation behaviors in cross-silo FL through a game-theoretic approach. In this paper, we aim at addressing the following fundamental question in cross-silo FL.

Key Question 1: *How do heterogeneous clients selfishly choose their participation levels in cross-silo FL?*

Different from cross-device FL, cross-silo FL usually involves long-term repeated interactions among clients. For example, MELLODDY project is a long-term project which involves the repeated interactions among 17 partners. One reason is that clients' local data may change over time, and hence clients will perform multiple cross-silo FL processes repeatedly to adapt the global model to the time-varying local data sets. For example, hospitals constantly admit new patients and collect treatment data of these cases. Furthermore, different from cross-device FL where the central server chooses different devices to perform different FL tasks and devices aim to maximize their short-term benefits, in cross-silo FL, the same set of clients initiate FL processes repeatedly and train the global model cooperatively. Moreover, clients (e.g., organizations or companies) are usually far-sighted and aim to maximize their long-term benefits. Therefore, it is necessary to analyze clients' long-term interactions in cross-silo FL. However, the existence of free riders is detrimental to the global model for the following two reasons. First, when many clients choose to be free riders, the amount of local data for model training is small, which leads to a bad global model accuracy. Second, the behavior of free riders is unfair to the clients who use their local data to perform local trainings, which will disrupt clients' long-term cooperation. To sustain clients' long-term cooperation in cross-silo FL, we need to reduce the number of free riders. This motivates us to address the second fundamental question in cross-silo FL.

Key Question 2: *How to minimize the number of free riders to sustain clients' long-term cooperation in cross-silo FL?*

1.2 Contributions

In this paper, we analyze clients' participation behaviors in cross-silo FL through a game-theoretic approach. We consider an infinite time horizon which is divided into many time slots, as shown in Fig. 2. We model the interactions among clients in the infinite time horizon as a repeated game with the stage game being the selfish participation game in one cross-silo FL process (SPFL) in each time slot. Specifically, in each time slot, clients perform one cross-silo FL process which proceeds in several iterations of local training steps and global aggregation steps. Each time slot can be one month for banks [10] or one week for hospitals [11]. Since the local data may change dynamically over time (e.g., banks update the finance data sets monthly or even weekly [10], and hospitals update the clinical data sets weekly [11]), clients need to repeatedly perform cross-silo

FL processes to adapt the global model to the dynamic local data. We summarize our main contributions as follows.

- *Novel Game Analysis in Cross-Silo FL:* To the best of our knowledge, this is the first paper that provides a comprehensive game-theoretic analysis of the long-term participation behaviors of heterogeneous clients in cross-silo FL from the repeated game perspective.
- *Selfish Participation Game in One Cross-Silo FL Process (SPFL):* For the stage game SPFL, we derive the unique Nash equilibrium (NE), and propose a distributed algorithm for each client to calculate its equilibrium participation strategy. We show that at NE, clients fall into at most three categories: (i) clients with low valuations for global model accuracy being *free riders* who do not perform local model training, (ii) a unique client (if exists) being the *partial contributor* who performs model training with part of its local data, and (iii) clients with high valuations being *contributors* who perform model training with all their local data.
- *Infinitely Repeated Game in the Long-Term Cross-Silo FL Processes:* In the infinitely repeated game, we derive a subgame perfect Nash equilibrium (SPNE), i.e., a cooperative strategy enforced by a punishment strategy, which achieves the minimum number of free riders while increasing the amount of local data for model training. Deriving the cooperative strategy is challenging since clients are heterogeneous and their decisions are coupled in a highly non-linear manner. We further propose an algorithm to calculate the optimal SPNE which minimizes the number of free riders while maximizing the amount of local data for model training.
- *Simulation Results:* We conduct extensive numerical evaluations and derive useful insights. First, clients' selfish participation behaviors significantly hamper the scalability of cross-silo FL, and at the NE of the stage game SPFL, the number of free riders increases with the number of clients, the fraction of clients with low valuations, the number of training iterations, the size of local data set, and the computation cost. Second, the cooperative strategy at the optimal SPNE can effectively reduce the number of free riders by up to 98.8%, and increase the amount of local data for model training by up to 96%.

1.3 Related work

FL has drawn researchers' attention in recent years. McMahan *et al.* propose and illustrate the effectiveness of FL [12]. Kairouz *et al.* classify FL into cross-device FL and cross-silo FL [4]. In the following, we discuss related work regarding FL optimization, incentive mechanism design, the free rider issue, and cross-silo FL.

FL Optimization: Some papers aim to optimize different objectives in FL, such as the learning time, the model accuracy, and the energy consumption of local devices. Zhu *et al.* present the framework of broadband analog aggregation to minimize the latency of federated edge learning [13]. Wang *et al.* present the framework of Favor to reduce the number of communication rounds [14]. Wang *et al.* propose a control algorithm for FL with non-i.i.d. data to achieve the minimum loss function and a better global model accuracy [15]. Mo *et al.* optimize the energy efficiency by balancing the energy tradeoff between communication and computation in FL process [16]. Luo *et al.* propose a sampling based algorithm to minimize the total cost with marginal overhead [17].

Incentive Mechanism Design: There are many papers focusing on the incentive mechanism design for FL. For example, Ding *et al.* in [18] consider contract design for the central server in FL under multi-dimensional private information. Kang *et al.* in [19] design an incentive mechanism based on reputation and contract theory to motivate high-reputation clients to participate in the learning process.

Free Rider Issue: Free riders are those who benefit from resources or public goods but refuse to pay for them [20], which have been extensively studied in peer-to-peer systems [21] [22] [23]. Several papers study how to identify free riders in FL. Lin *et al.* reveal the free-riding attack in federated learning and propose a novel anomaly detection technique using autoencoders to identify free riders [24]. Huang *et al.* design a protection mechanism named Gradient Auditing to detect and punish the behavior of free-riding [25]. Fraboni *et al.* propose to establish a routine practice to inspect clients' distribution for the detection of free-rider attacks in federated learning [26]. Although these papers notice the existence of free riders in FL, they only focus on identifying free riders rather than addressing the problem effectively.

Cross-silo FL: There have been some researches analyzing cross-silo FL. Chen *et al.* propose FOCUS for cross-silo FL to solve the problem of noisy local labels [27]. Heikkila *et al.* combine additively homomorphic secure summation protocols with differential privacy in cross-silo FL to learn complex model while guaranteeing privacy of local data [28]. Marfoq *et al.* design cross-silo FL topologies to maximize the number of completed rounds per time unit [29]. Majeed *et al.* build a cross-silo horizontal federated model for traffic classification [30]. Zhang *et al.* present a system solution BatchCrypt for cross-silo FL to reduce the encryption and communication overhead caused by additively homomorphic encryption [31]. Tang *et al.* propose an incentive mechanism considering the public goods feature of cross-silo FL [9]. However, none of the above papers analyze the long-term selfish participation behaviors of clients in cross-silo FL.

In summary, existing works mainly focus on cross-device FL, and none of these papers studies clients' long-term selfish participation behaviors in cross-silo FL. As far as we know, this is the first work that analyzes the long-term participation behaviors of heterogeneous clients in cross-silo FL and proposes a strategy to reduce the number of free riders through the game-theoretic approach.

We organize our paper as follows. In Section 2, we present the system model. Then we analyze the stage game SPFL in Section 3. We analyze the infinitely repeated game in Section 4. We present simulation results in Section 5 and conclude in Section 6. Due to space constraints, we relegate all the proofs to the supplementary material.

2 System Model

We consider a set $\mathcal{N} = \{1, 2, \ldots, N\}$ of clients (e.g., companies or organizations) with the same interest participating in the cross-silo FL processes. Each client has some local data which may change over time, and hence clients will perform multiple cross-silo FL processes repeatedly to adapt the global model to the time-varying local data sets. Furthermore, clients are far-sighted and aim to optimize their long-term benefits. Therefore, we model clients' long-term interactions in cross-silo FL processes in the infinite time horizon which is divided into time slots. In the following, we first describe a cross-silo FL process in one time slot, then model the cost of clients in a cross-silo FL process, and finally show the interactions among clients in two time scales.

2.1 A Cross-Silo FL Process in One Time Slot

As shown in Fig. 2, each time slot corresponds to one cross-silo FL process which proceeds in training iterations. We first introduce the objective of cross-silo FL, and then describe the iteration process.

In a cross-silo FL process, clients of the same interest cooperatively train a global model represented by a parameter vector $\boldsymbol{w}$. Each client $n \in \mathcal{N}$ has a local data set $\mathcal{D}_n$, where the number of local data samples is $D_n \triangleq |\mathcal{D}_n|$.[1] Each client can choose a subset $\mathcal{X}_n \subseteq \mathcal{D}_n$ of local data for model training. Let x_n denote the size of the chosen subset of local data, i.e., $x_n \triangleq |\mathcal{X}_n|$, and let $\boldsymbol{d}_{ni}$ denote the i-th data sample in set $\mathcal{X}_n$. The cross-silo FL aims to find the optimal global model represented by $\boldsymbol{w}^*$ that minimizes the global loss function [4]:

$$L(\boldsymbol{w}) = \sum_{n \in \mathcal{N}} \frac{x_n}{\sum_{n' \in \mathcal{N}} x_{n'}} L_n(\boldsymbol{w}). \tag{1}$$

Here $L_n(\boldsymbol{w})$ is the local loss function of client n, which can be calculated as:

$$L_n(\boldsymbol{w}) = \frac{1}{x_n} \sum_{\boldsymbol{d}_{ni} \in \mathcal{X}_n} l(\boldsymbol{w}; \boldsymbol{d}_{ni}). \tag{2}$$

Here $l(\boldsymbol{w}; \boldsymbol{d}_{ni})$ is the loss function for data sample $\boldsymbol{d}_{ni}$ under $\boldsymbol{w}$.

To achieve the optimal global model $\boldsymbol{w}^*$, the cross-silo FL proceeds in training iterations. One widely adopted algorithm to derive $\boldsymbol{w}^*$ is the FedAvg algorithm [12].[2] In each training iteration r, clients perform local model trainings over the previous global model $\boldsymbol{w}^{r-1}$ with the chosen subsets of local data by using the mini-batch stochastic gradient descent (SGD) method [34]. Clients derive the updated local models $\boldsymbol{w}_n^r, \forall n \in \mathcal{N}$, and send them[3] to the central server for global aggregation. The central server derives the updated global model $\boldsymbol{w}^r = \sum_{n \in \mathcal{N}} \frac{x_n}{\sum_{n' \in \mathcal{N}} x_{n'}} \boldsymbol{w}_n^r$ [12]. The above cross-silo FL process causes costs to clients, which we will introduce in detail next.

1. Although different clients have different local data, we assume that the data is i.i.d. across all clients [32].

2. In cross-silo FL, we assume that all clients participate in training iterations [4], and clients perform synchronous update scheme [4] [33].

3. We assume that clients will truthfully report their local models to the central server, and the central server can verify clients' contributions (e.g., by the Trusted Execution Environments proposed in [35]).

TABLE 1
Key Notations

Symbol	Physical Meaning
$\mathcal{N}$	The set of clients participating in cross-silo FL
$\mathcal{D}_n$	The set of local data of client $n \in \mathcal{N}$
D_n	The number of local data samples in set $\mathcal{D}_n$
$\mathcal{X}_n$	The subset of local data that client n chooses to perform model training
x_n	The size of the chosen subset $\mathcal{X}_n$
$\boldsymbol{x}$	The vector of clients' chosen strategies
B	The total amount of chosen local data, i.e., $B = \sum_{n \in \mathcal{N}} x_n$
$\boldsymbol{d}_{ni}$	The i-th data sample in set $\mathcal{X}_n$
$\boldsymbol{w}$	The parameter vector of the global model
$L(\boldsymbol{w})$	The global loss function under $\boldsymbol{w}$
$L_n(\boldsymbol{w})$	The local loss function of client n under $\boldsymbol{w}$
$l(\boldsymbol{w}; \boldsymbol{d}_{ni})$	The loss function for data sample $\boldsymbol{d}_{ni}$ under $\boldsymbol{w}$
G	The number of iterations in a cross-silo FL process
$A(\boldsymbol{x})$	The model accuracy loss under $\boldsymbol{x}$
$\mathcal{E}(x_n)$	The computation cost of client n
C	The communication cost of each client
p	The payment that each client pays the central server
ρ_n	Client n's valuation for the model accuracy
$F_n(x_n, \boldsymbol{x}_{-n})$	The total cost of each client $n \in \mathcal{N}$

2.2 Cost of Clients in A Cross-Silo FL Process

In cross-silo FL, clients incur the following costs: the model accuracy loss, the computation cost, the communication cost, and the payment to the central server.

2.2.1 Model Accuracy Loss

The purpose of clients to perform cross-silo FL is to achieve a global model with good accuracy, i.e., a global model with a small accuracy loss [32]. The global model accuracy loss can be calculated as $L(\boldsymbol{w}^G) - L(\boldsymbol{w}^*)$, where $L(\boldsymbol{w}^G)$ and $L(\boldsymbol{w}^*)$ are the global losses under parameters $\boldsymbol{w}^G$ and $\boldsymbol{w}^*$, respectively, and G is the number of training iterations in a cross-silo FL process. According to [36], the expected global model accuracy loss is bounded by $O(1/\sqrt{BG} + 1/G)$, where B is the total batch size that clients use for model trainings, i.e., $B = \sum_{n \in \mathcal{N}} x_n$.[4] Hence, the expected global model accuracy loss decreases with the amount of local data $\sum_{n \in \mathcal{N}} x_n$ for model trainings and the number of training iterations G. In this paper, we denote the model accuracy loss of each client $n \in \mathcal{N}$ as $A(x_n, \boldsymbol{x}_{-n})$, which depends on all clients' chosen amounts of local data for model trainings and can be calculated as follows:[5]

$$A(x_n, \boldsymbol{x}_{-n}) = \frac{1}{\sqrt{(x_n + \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'})G}} + \frac{1}{G}, \tag{3}$$

where $\boldsymbol{x}_{-n} = \{x_1, x_2, \ldots, x_{n-1}, x_{n+1}, \ldots, x_N\}$.

4. Note that in federated learning, the central server knows clients' chosen amounts of local data for model trainings, i.e., the values of $x_n, \forall n \in \mathcal{N}$, which are the weights in the global aggregation steps [9] [32] [37].

5. In this paper, we assume a complete information setting where each client $n \in \mathcal{N}$ knows other clients' chosen amounts of local data, i.e., the values of $x_{n'}, \forall n' \in \mathcal{N}, n' \neq n$. This information can be announced by the central server to clients. Note that our proposed algorithms later in the paper which calculate the equilibrium strategy for each client do not require such information. The analysis in the complete information setting provides useful insights for practical cross-silo FL systems. For the analysis of the incomplete information setting, we will leave it as future work.

2.2.2 Computation Cost

The local model training steps consume local computation resources. In a cross-silo FL process, the CPU energy consumption of each client $n \in \mathcal{N}$, which depends on its chosen amount of local data x_n for model training, can be calculated as [38]

$$\mathcal{E}(x_n) = \frac{\varsigma_n}{2}\mu_n \vartheta_n^2 x_n, \tag{4}$$

where ς_n is a coefficient depending on the client's computing chip architecture, μ_n is the number of CPU cycles of client n to perform the model training on one local data sample, and ϑ_n is the CPU processing speed (in cycles per second) of client n [39] [40]. We assume that ς_n, μ_n and ϑ_n are the same for each client [40], and denote $E = \frac{\varsigma_n}{2}\mu_n\vartheta_n^2$ for simplicity. Therefore, we can calculate the computation cost of each client $n \in \mathcal{N}$ as $\mathcal{E}(x_n) = Ex_n$.

2.2.3 Communication Cost

In a cross-silo FL process, clients send their local models $\boldsymbol{w}_n^r, \forall n \in \mathcal{N}$, to the central server for aggregation, and receive the updated global model $\boldsymbol{w}^r$ for the next training iteration, for all $r = 1, 2, \ldots, G$. Since clients in cross-silo FL are companies or organizations, the data transmission between clients and the central server can be through either the wired networks (e.g., through the high-speed wired connections [41]) or the wireless networks (e.g., using transmission protocols TDMA or OFDMA [37]). We assume that each client has the same communication resource [42], and the parameters of local models have the same size. In this case, clients experience the same communication cost (e.g., transmission delay), which depends on the number of iterations between clients and the central server [40]. We denote the communication cost of each client as C, which is a constant in our paper.

2.2.4 Payment to the Central Server

Clients need to pay the central server for global model aggregation. We assume that the payment of each client is p.

In summary, we define the total cost of each client $n \in \mathcal{N}$ as follows,

$$F_n(x_n, \boldsymbol{x}_{-n}) = \rho_n A(x_n, \boldsymbol{x}_{-n}) + \mathcal{E}(x_n) + C + p. \tag{5}$$

Here ρ_n denotes client n's valuation for the model accuracy, which describes how important the model accuracy is to the client [9]. For example, ρ_n can be a bank's customer churn per unit of model accuracy loss when using the risk control service trained by cross-silo FL, or a pharmaceutical company's unit revenue loss when using the drug research model trained by cross-silo FL. Without loss of generality, we assume that $\rho_1 \leq \rho_2 \leq \cdots \leq \rho_N$.

2.3 Interactions among Clients in Two Time Scales

In this subsection, we consider the interactions among clients in two time scales as shown in Fig. 2. Specifically, we describe the behaviors of clients in each time slot and in the infinite time horizon, respectively.

We first analyze the interactions among clients in a single time slot. As in [10] [11], the length of a time slot can be a month for banks or a week for hospitals. One time slot corresponds to one cross-silo FL process which proceeds in several iterations of local training steps and global aggregation steps. At the beginning of the time slot, each client n chooses the amount of local data x_n for model training (i.e., its participation level in the cross-silo FL process), to minimize its total cost calculated in (5), considering the participation behaviors of other clients. When clients are myopic and only care about their costs in the current time slot, we model their participation behavior as a selfish participation game in one cross-silo FL process (SPFL), to be introduced in detail in Section 3.

We then analyze the interactions among clients in the infinite time horizon which is divided into many time slots. The interactions among clients in the infinite time horizon are inherently a repeated process. Specifically, at the beginning of each time slot t, clients receive the global model $\boldsymbol{w}^{t-1}$ in the previous time slot, and then perform a cross-silo FL process using their local data (as described in the selfish participation game SPFL) to derive the global model $\boldsymbol{w}^t$. Each client's local data set changes over time, i.e., $\mathcal{D}_n^{t-1} \neq \mathcal{D}_n^t, \forall n \in \mathcal{N}$, where $\mathcal{D}_n^{t-1}$ and $\mathcal{D}_n^t$ are the local data sets of client n in time slot $t-1$ and time slot t, respectively. We assume that the number of local data samples remains unchanged, i.e., $|\mathcal{D}_n^{t-1}| = |\mathcal{D}_n^t|, \forall n \in \mathcal{N}$.[6] In the infinite time horizon, clients are far-sighted, and each client n chooses the amount of local data for model training to minimize its long-term discounted total cost. We model clients' long-term selfish participation behaviors as an infinitely repeated game with the stage game being SPFL, which will be introduced in detail in Section 4.

3 Stage Game Analysis

In this section, we analyze the stage game, i.e., the selfish participation game in one cross-silo FL process (SPFL). We derive the unique Nash equilibrium of the stage game SPFL, and design a distributed algorithm for each client to compute its equilibrium participation strategy.

3.1 Stage Game Modeling

In the cross-silo FL process, clients selfishly choose their participation levels to minimize their own costs. We model the behaviors of clients in each time slot as a selfish participation game as follows:

Game 1 (Selfish Participation Game in a Cross-silo FL Process (SPFL))**.**

- *Players: the set $\mathcal{N}$ of clients.*
- *Strategies: each client $n \in \mathcal{N}$ chooses the amount of local data $x_n \in [0, D_n]$ for model training.*
- *Objectives: each client $n \in \mathcal{N}$ aims to minimize its total cost $F_n(x_n, \boldsymbol{x}_{-n})$ defined in* (5).

Here D_n is the total number of data samples in client n's local data set $\mathcal{D}_n$, i.e., $D_n = |\mathcal{D}_n|$.

6. For banks or hospitals, the number of users does not change much from month to month, so we assume that the number of local data samples remains the same.

3.2 Nash Equilibrium of SPFL

Next we define the Nash equilibrium (NE) of SPFL and derive the unique NE later.

Definition 1 (Nash Equilibrium)**.** *A Nash equilibrium of Game 1 (SPFL) is a strategy profile $\boldsymbol{x}^* = \{x_n^* : \forall n \in \mathcal{N}\}$ such that for each client $n \in \mathcal{N}$,*

$$F_n(x_n^*, \boldsymbol{x}_{-n}^*) \leq F_n(x_n, \boldsymbol{x}_{-n}^*), \text{ for all } x_n \in [0, D_n].$$

The NE is a strategy profile where each client's strategy is the best response to other clients' strategies. In Lemma 1, we characterize the best response of each client $n \in \mathcal{N}$ that minimizes its total cost given all other clients' strategies $\boldsymbol{x}_{-n}$.

Lemma 1. *The best response of each client $n \in \mathcal{N}$ in Game 1 (SPFL) is*

$$x_n^{\mathrm{BR}}(\boldsymbol{x}_{-n}) = \min\left\{D_n, \max\left\{\sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'}, 0\right\}\right\}. \tag{6}$$

Proof. See Appendix A in the supplementary material. □

Client n's best response $x_n^{\mathrm{BR}}(\boldsymbol{x}_{-n})$ increases with its valuation ρ_n for model accuracy, while decreases with the number of iterations G, the computation cost coefficient E, and the total amount of local data $\sum_{n' \in \mathcal{N}, n' \neq n} x_{n'}$ that other clients choose for model training. Intuitively, when client n has a high valuation for the global model accuracy, client n will choose a large amount of local data for model training to achieve a good global model. On the contrary, when the number of iterations G is large, clients can achieve a good global model with a small amount of local data as shown in (3). Similarly, when the total amount of local data from other clients is large, the accuracy loss in (3) is small and client n will choose a small amount of local data for model training to reduce the computation cost. When the computation cost coefficient E is large, client n will choose a small amount of local data for training to reduce the computation cost.

Next we derive the NE of Game 1. For simplicity of analysis, we assume that $D_n = D, \forall n \in \mathcal{N}$. For each client $n \in \mathcal{N}$, we define $h_n \triangleq \sqrt[3]{\frac{\rho_n^2}{4GE^2}}$.

Theorem 1. *A Nash equilibrium $\boldsymbol{x}^*$ always exists in Game 1, and falls into one of the following two cases.*

- *Case I: If there exists a critical client $k \in \mathcal{N}$ that satisfies $(N-k)D \leq h_k \leq (N+1-k)D$, then the NE $\boldsymbol{x}^* = \{x_n^* : \forall n \in \mathcal{N}\}$ is:*

$$x_n^* = \begin{cases} 0, & \text{if } n < k; \\ h_k - (N-k)D, & \text{if } n = k; \\ D, & \text{if } n > k. \end{cases} \tag{7}$$

- *Case II: If the critical client k in Case I does not exist, then there must exist a client $m \in \mathcal{N}$ that satisfies $h_m < (N-m)D < h_{m+1}$, and in this case, the NE $\boldsymbol{x}^* = \{x_n^* : \forall n \in \mathcal{N}\}$ is:*

$$x_n^* = \begin{cases} 0, & \text{if } n \leq m; \\ D, & \text{if } n > m. \end{cases} \tag{8}$$

Proof. See Appendix B in the supplementary material. □

Theorem 1 shows that the NE of Game 1 falls into two cases depending on whether a critical client k exists or not. If such a critical client k exists, clients can be divided into three categories at equilibrium: (i) clients whose valuations for global model accuracy are lower than that of the critical client k choose to be *free riders* that do not perform local model training, (ii) the unique critical client k chooses to be a *partial contributor* who performs model training with part of its local data, and (iii) clients whose valuations are higher than that of the critical client k choose to be *contributors* who perform model training with all their local data. If the critical client k does not exist, clients can be divided into two categories at equilibrium: (i) clients with low valuations choose to be free riders, and (ii) clients with high valuations choose to be contributors. Note that at equilibrium of Game 1, it is not possible that all clients choose to be free riders, as shown in the following corollary.

Corollary 1. *At equilibrium of Game 1, the client with the highest valuation for global model accuracy chooses a positive amount of local data for model training, i.e., $x_N^* > 0$.*

The significance of the above result is to establish that at equilibrium of Game 1, there will always be a positive amount of local data chosen by clients to perform model training in cross-silo FL. Specifically, the client N who has the highest valuation for global model accuracy will never choose to be a free rider. Even if all other clients choose to be free riders at equilibrium, client N will choose a positive amount of local data for model training, otherwise its model accuracy loss in (3) and its total cost in (5) will go to infinity.

At equilibrium, however, clients with low valuations for global model accuracy may choose to be free riders. This is a result of the tradeoff between the model accuracy loss and the computation cost. Specifically, given the global model aggregated from the local models of the (partial) contributors, clients with low valuations choose not to perform local trainings to avoid the computation cost. The behaviors of free riders only minimize their own costs, but cannot improve the global model accuracy. As discussed in Section 1.1, the existence of free riders hampers clients' long-term participation in cross-silo FL. Later in Section 4, we will analyze the interactions among clients in the infinite time horizon, and design a cooperative strategy to reduce the number of free riders.

Next we discuss the uniqueness of the NE of Game 1.

Theorem 2. *Game 1 admits a unique Nash equilibrium if $\rho_1 < \rho_2 < \cdots < \rho_N$.*

Proof. See Appendix C in the supplementary material. □

Here $\rho_1 < \rho_2 < \cdots < \rho_N$ is a mild constraint which can be satisfied in most cases in practice [43].[7] Next we will design a distributed algorithm to calculate the equilibrium participation strategy for each client.

7. When the constraint $\rho_1 < \rho_2 < \cdots < \rho_N$ is not satisfied, the Nash equilibrium may not be unique. Consider an example with $N = 2$ clients, and $\rho_1 = \rho_2$, $h_1 = h_2 = 10$, $D_1 = D_2 = 10$. In this example, $(x_1^*, x_2^*) = (4, 6)$ is a Nash equilibrium of Game 1, and $(x_1^*, x_2^*) = (5, 5)$ is also a Nash equilibrium of Game 1.

Algorithm 1: A Distribute Algorithm to Compute Equilibrium Strategy for Each Client in Game 1

Input: $\rho_n, \forall n \in \mathcal{N}, E, G, N, D$
Output: $x_n^*, \forall n \in \mathcal{N}$

```
1  Each client n ∈ N reports the value of ρ_n to the central server.
2  The central server sorts the clients in an ascending order of ρ_n, ∀n ∈ N, and sends each client its index n as well as the total number of clients N.
3  for n = 1 : N do
4      Calculate h_n = ∛(ρ_n^2 / (4GE^2));
5      if h_n < (N − n)D then
6          x_n^* = 0;
7      else if h_n > (N − n + 1)D then
8          x_n^* = D;
9      else
10         x_n^* = h_n − (N − n)D.
11     end
12 end
```

3.3 A Distributed Algorithm to Compute Equilibrium Strategy

In this section, we design a distributed algorithm (in Algorithm 1) for clients to compute the equilibrium strategy of Game 1. In the distributed algorithm, each client calculates its equilibrium participation strategy based on its own information, without knowing other clients' participation decisions or parameter information (e.g., the valuation parameters for global model accuracy).

To calculate the equilibrium strategy, each client $n \in \mathcal{N}$ needs to know the total number of clients N and the ranking of its valuation parameter ρ_n in all clients (i.e., the index n), which can be obtained by reporting its valuation parameter ρ_n to the central server for ordering[8] (Lines 1-2 in Algorithm 1). The equilibrium strategy of each client $n \in \mathcal{N}$ depends on the value of h_n. If $h_n < (N-n)D$, client n chooses to be a free rider, i.e., $x_n^* = 0$ (Lines 5-6 in Algorithm 1). If $h_n > (N-n+1)D$, client n chooses to be a contributor, i.e., $x_n^* = D$ (Lines 7-8 in Algorithm 1). Otherwise, client n chooses to be a partial contributor, i.e., $x_n^* = h_n - (N-n)D$ (Lines 9-10 in Algorithm 1). Note that clients do not need to know whether the critical client k exists at equilibrium, and Algorithm 1 covers both cases of the equilibrium described in Theorem 1.

The complexity of Algorithm 1 is $\mathcal{O}(N)$, and hence it is scalable with the number of clients in cross-silo FL.

As discussed in Section 2.3, the interactions among clients in the infinite time horizon are inherently a repeated process. Since clients' local data may change over time, they need to adapt the global model to the local data sets constantly. In the next section, we analyze clients' long-term selfish participation behaviors.

4 Repeated Game Analysis

In this section, we analyze the long-term interactions among clients in cross-silo FL processes. We model clients' selfish participation behaviors as a repeated game in the infinite time horizon, with the stage game being the SPFL (i.e., Game 1). We first model the repeated game, and then derive a cooperative participation strategy that can minimize the number of free riders while increasing the amount of local data for model training. We derive a subgame perfect Nash equilibrium (SPNE) of the repeated game which enforces the cooperative participation strategy by a punishment strategy. Finally, we propose an algorithm to calculate the optimal SPNE that can minimize the number of free riders while maximizing the amount of local data for model training.

8. We assume that each client $n \in \mathcal{N}$ will truthfully report its valuation parameter ρ_n to the central server. How to incentivize clients to truthfully reveal their information [44] [45] is beyond the scope of this paper.

4.1 Repeated Game Modeling

As shown in Fig. 2, we consider clients' interactions in the infinite time horizon which consists of infinitely many time slots. Each time slot corresponds to one cross-silo FL process. In the infinite time horizon, each client chooses its participation level (i.e., the amount of local data for model training) for each cross-silo FL process to minimize its long-term discounted total cost.

We first calculate the long-term discounted total cost for each client. We denote the participation strategy profile of all clients in time slot t as $\boldsymbol{x}^t = \{x_n^t : \forall n \in \mathcal{N}\}$. Recall that for each client $n \in \mathcal{N}$, its local data set changes over time while keeping the size unchanged, i.e., $\mathcal{D}_n^{t-1} \neq \mathcal{D}_n^t, |\mathcal{D}_n^{t-1}| = |\mathcal{D}_n^t| = D_n$. Therefore, we have $x_n^t \in [0, D_n]$ for each time slot t. We denote the strategy profile history up to time slot t as

$$s^t \triangleq \{\boldsymbol{x}^0, \boldsymbol{x}^1, \ldots, \boldsymbol{x}^{t-1}\}.$$

Then for each client $n \in \mathcal{N}$, its long-term discounted total cost is

$$\mathcal{C}_n(s^\infty) = \sum_{t=0}^{\infty} \delta_n^t F_n(\boldsymbol{x}^t), \tag{9}$$

where $\delta_n \in [0, 1)$ is the discount factor [46] [47] of client n, and $F_n(\boldsymbol{x}^t)$ is the cost of client n in time slot t calculated in (5).

We model the interactions among clients in the infinite time horizon as a repeated game as follows.

Game 2 (Repeated Game in Infinitely Many Cross-Silo FL Processes)**.**

- *Players: the set $\mathcal{N}$ of clients.*
- *Strategies: each client $n \in \mathcal{N}$ chooses the amount of local data $x_n^t \in [0, D_n]$ for model training in each time slot t.*
- *Histories: the strategy profile history s^t till time slot t.*
- *Objectives: each client $n \in \mathcal{N}$ aims to minimize its long-term discounted total cost $\mathcal{C}_n(s^\infty)$ defined in (9).*

We aim to find the subgame perfect Nash equilibrium (SPNE) of Game 2. According to Folk Theorem [47], under proper discount factors $\delta_n, \forall n \in \mathcal{N}$, any feasible and individually rational participation strategy profile can be a subgame perfect Nash equilibrium of the infinitely repeated game (Game 2). Since the success of cross-silo FL depends on the active long-term participation of clients, we will characterize the SPNE that can minimize the number of free riders while increasing the amount of local data chosen by clients to perform model training. Next in Section 4.2, we derive a cooperative participation strategy in each cross-silo FL process, i.e., a strategy profile where clients cooperate to minimize the number of free riders. Later in Section 4.3, we show that the cooperative participation strategy can be enforced as the SPNE of the repeated game by a punishment strategy.

4.2 Cooperative Participation Strategy

Now we derive the cooperative participation strategy $\boldsymbol{x}^{coop,t}$ in each time slot t. The cooperative participation strategy aims to minimize the number of free riders while reducing the cost of each client compared with the cost at NE of the stage game SPFL, i.e., $F_n(\boldsymbol{x}^{coop,t}) < F_n(\boldsymbol{x}^{*,t})$ for all $n \in N$. Here $\boldsymbol{x}^{*,t}$ is the NE of the stage game SPFL in time slot t. In this subsection, we focus on a time slot t. When there is no confusion, we ignore the time index and write $\boldsymbol{x}^{coop,t}$ and $\boldsymbol{x}^{*,t}$ as $\boldsymbol{x}^{coop}$ and $\boldsymbol{x}^{*}$, respectively.

We first analyze the cooperative participation strategy for the case where the critical client k exists at the NE of Game 1 (as shown in Case I in Theorem 1). For each client $n \in \mathcal{N}$, we define $B_n \triangleq \frac{2E\sqrt{G((N-k)D+x_k^*)^3}}{k-n}$ and $O_n \triangleq \frac{ED\sqrt{G}}{\frac{1}{\sqrt{(N-k)D+x_k^*}}-\frac{1}{\sqrt{(N-n)D+x_k^*}}}$, where x_k^* is the participation strategy of the critical client k at the NE of Game 1. In this case, we derive the cooperative strategy as follows.

Theorem 3. *When the critical client k exists at the NE of Game 1 (as shown in Case I in Theorem 1), we find the client l which satisfies*

$$l = \min\{n \in \mathcal{N} : n < k, \text{ and } \rho_n > B_n\}. \tag{10}$$

Then the cooperative participation strategy $\boldsymbol{x}^{coop} = \{x_n^{coop} : \forall n \in \mathcal{N}\}$ that minimizes the number of free riders is:

$$x_n^{coop} = \begin{cases} 0, & \text{if } n < l; \\ x^{coop}, & \text{if } l \le n < k; \\ x_k^*, & \text{if } n = k; \\ D, & \text{if } n > k. \end{cases} \tag{11}$$

Here x^{coop} is a cooperative participation level that satisfies $x^{coop} \le x_l^{th}$, and x_l^{th} is the maximum amount of local data that client l can choose under the cooperative strategy, calculated as follows,

$$x_l^{th} = \begin{cases} x_l^{th}(\rho_l), & \text{if } B_l < \rho_l \le O_l; \\ D, & \text{if } \rho_l > O_l. \end{cases} \tag{12}$$

Here $x_l^{th}(\rho_l)$ is the unique non-zero solution to the following implicit equation:

$$\rho_l = \frac{E\sqrt{G}x_l^{th}}{\frac{1}{\sqrt{(N-k)D+x_k^*}} - \frac{1}{\sqrt{(N-k)D+x_k^*+(k-l)x_l^{th}}}}. \tag{13}$$

Proof. See Appendix D in the supplementary material. □

Theorem 3 shows that when the critical client k exists at the NE of Game 1, we can derive a cooperative participation strategy under which clients fall into at most four categories: (i) the free riders at the NE of Game 1 who satisfy $\rho_n \le B_n$ still choose to be *free riders* under the cooperative strategy, (ii) the free riders at the NE of Game 1 who satisfy $\rho_n > B_n$ choose to be *converted contributors* who perform model training with a positive amount[9] of local data x^{coop}, (iii) the unique critical client k chooses to be a *partial contributor* who performs model training with the amount of local data x_k^*, and (iv) the contributors at the NE of Game 1 still choose to be *contributors* who perform model training with all their local data.

Compared with the NE of Game 1 in Theorem 1, Theorem 3 shows that under the cooperative participation strategy, we have the following improved results. First, the number of free riders is reduced from $k-1$ to $l-1$ where $l \le k$. Second, the amount of local data chosen by the converted contributors increases from 0 to $x^{coop} > 0$. Furthermore, the number of free riders under the cooperative strategy is non-increasing with the valuation parameters $\{\rho_n : n \in \mathcal{N}, n < k\}$. The following corollary shows that under certain conditions, the cooperative strategy can reduce the number of free riders to 0.

Corollary 2. *Under the cooperative strategy in Theorem 3, no client chooses to be a free rider if*

$$\rho_1 > B_1 \triangleq \frac{2E\sqrt{G((N-k)D+x_k^*)^3}}{k-1}.$$

Furthermore, all clients choose to be contributors, i.e., $x_n^{coop} = D, \forall n \in \mathcal{N}$, if

$$\rho_1 > O_1 \triangleq \frac{ED\sqrt{G}}{\frac{1}{\sqrt{(N-k)D+x_k^*}} - \frac{1}{\sqrt{(N-1)D+x_k^*}}}.$$

The above result shows that when client 1's valuation parameter ρ_1 is high, all clients choose to be (converted/partial) contributors under the cooperative participation strategy, and there is no free rider in cross-silo FL.

Next, we analyze the cooperative participation strategy for the case where the critical client k does not exist at the NE of Game 1 (as shown in Case II in Theorem 1). For each client $n \in \mathcal{N}$, we define $B'_n \triangleq \frac{2E\sqrt{G(N-m)^3D^3}}{m-n+1}$ and $O'_n \triangleq \frac{ED\sqrt{G}}{\frac{1}{\sqrt{(N-m)D}}-\frac{1}{\sqrt{(N-n+1)D}}}$. Here m is the index of the client who satisfies $h_m < (N-m)D < h_{m+1}$. In this case, we derive the cooperative strategy as follows.

Theorem 4. *When there is a client m that satisfies $h_m < (N-m)D < h_{m+1}$ at the NE of Game 1 (as shown in Case II in Theorem 1), we find the client l which satisfies*

$$l = \min\{n \in \mathcal{N} : n \le m, \text{ and } \rho_n > B'_n\}. \tag{14}$$

Then the cooperative participation strategy $\boldsymbol{x}^{coop} = \{x_n^{coop} : \forall n \in \mathcal{N}\}$ that minimizes the number of free riders is:

$$x_n^{coop} = \begin{cases} 0, & \text{if } n < l; \\ x^{coop}, & \text{if } l \le n \le m; \\ D, & \text{if } n > m. \end{cases} \tag{15}$$

Here x^{coop} is a cooperative participation level that satisfies $x^{coop} \le x_l^{th}$, and x_l^{th} is the maximum amount of local data that client l can choose under the cooperative strategy, calculated as follows,

9. We assume that for fairness, converted contributors choose the same amount of local data for model training. Fairness is an important problem in federated learning [48], which is beyond the scope of this paper.

$$x_l^{th} = \begin{cases} x_l^{th}(\rho_l), & \text{if } B_l' < \rho_l \le O_l'; \\ D, & \text{if } \rho_l > O_l'. \end{cases} \tag{16}$$

Here $x_l^{th}(\rho_l)$ is the unique non-zero solution to the following implicit equation:

$$\rho_l = \frac{E\sqrt{G}x_l^{th}}{\frac{1}{\sqrt{(N-m)D}} - \frac{1}{\sqrt{(N-m)D+(m-l+1)x_l^{th}}}}. \tag{17}$$

Proof. See Appendix E in the supplementary material. □

Theorem 4 shows that when the critical client k does not exist at the NE of Game 1, we can derive a cooperative participation strategy under which clients fall into at most three categories: (i) the free riders at the NE of Game 1 who satisfy $\rho_n \le B_n'$ still choose to be *free riders* under the cooperative strategy, (ii) the free riders at the NE of Game 1 who satisfy $\rho_n > B_n'$ choose to be *converted contributors* who perform model training with a positive amount of local data x^{coop}, and (iii) the contributors at the NE of Game 1 still choose to be *contributors*.

Similar as Theorem 3, Theorem 4 shows that compared with the NE of Game 1 in Theorem 1, we have the following improved results under the cooperative participation strategy. First, the number of free riders is reduced from m to $l-1$, where $l-1 \le m$. Second, the amount of local data chosen by the converted contributors increases from 0 to $x^{coop} > 0$. Furthermore, the number of free riders under the cooperative strategy is non-increasing with the valuation parameters $\{\rho_n : n \in \mathcal{N}, n \le m\}$. The following corollary shows that under certain conditions, the cooperative strategy can reduce the number of free riders to 0.

Corollary 3. *Under the cooperative strategy in Theorem 4, no client chooses to be a free rider if*

$$\rho_1 > B_1' \triangleq \frac{2E\sqrt{G((N-m)D)^3}}{m}.$$

Furthermore, all clients choose to be contributors, i.e., $x_n^{coop} = D, \forall n \in \mathcal{N}$, if

$$\rho_1 > O_1' \triangleq \frac{ED\sqrt{G}}{\frac{1}{\sqrt{(N-m)D}} - \frac{1}{\sqrt{ND}}}.$$

Similar as Corollary 2, Corollary 3 shows that when client 1's valuation parameter ρ_1 is high, all clients choose to be (converted/partial) contributors under the cooperative participation strategy, and there is no free rider in cross-silo FL.

In summary, Theorem 3 and Theorem 4 characterize the cooperative participation strategies that can minimize the number of free riders while increasing the amount of local data for model training in cross-silo FL, for both cases of the NE of Game 1 in Theorem 1. We next discuss how to enforce the cooperative participation strategy as the SPNE of the repeated game by a punishment strategy.

4.3 Subgame Perfect Nash Equilibrium

In this section, we aim to derive the condition under which clients behave according to the cooperative participation strategy $\boldsymbol{x}^{coop}$ at the SPNE of the repeated game.

In the infinitely repeated game, we will enforce the cooperative participation strategy $\boldsymbol{x}^{coop}$ as the SPNE by a punishment strategy $\boldsymbol{x}^{pun}$. In this paper, we adopt Friedman punishment [49], where clients play the NE of the stage game SPFL in Theorem 1 if any client deviates from the cooperative participation strategy (i.e., $\boldsymbol{x}^{pun} = \boldsymbol{x}^*$). Note that although the cooperative participation strategy can reduce the cost of each client in each time slot, i.e., $F_n(\boldsymbol{x}^{coop,t}) < F_n(\boldsymbol{x}^{*,t}), \forall n \in \mathcal{N}$, it is not an equilibrium strategy of the selfish participation game SPFL. In other words, given other clients' strategies under $\boldsymbol{x}^{coop}$, a client who is a (converted/partial) contributor has the incentive to decrease its chosen amount of local data for model training to reduce its total cost. To punish the client who deviates from the cooperative participation strategy, other clients resort to the NE of the stage game SPFL.

We use the one-stage deviation principle [50] to characterize clients' behaviors at the SPNE. Specifically, each client $n \in \mathcal{N}$ plays the cooperative participation strategy at the SPNE, if its long-term discounted total cost under cooperation is no larger than that under deviation, i.e.,

$$\sum_{t=0}^{\infty} \delta_n^t F_n(\boldsymbol{x}^{coop}) \le F_n(\boldsymbol{x}_n^{least}) + \sum_{t=1}^{\infty} \delta_n^t F_n(\boldsymbol{x}^{pun}). \tag{18}$$

Here $\boldsymbol{x}_n^{least}$ is the participation strategy profile under which other clients keep cooperation while client n deviates to minimize its own cost in one time slot. Hence we have

$$\delta_n \ge \delta_n^{th}(\boldsymbol{x}^{coop}) \triangleq \frac{F_n(\boldsymbol{x}^{coop}) - F_n(\boldsymbol{x}_n^{least})}{F_n(\boldsymbol{x}^{pun}) - F_n(\boldsymbol{x}_n^{least})}, \tag{19}$$

where $\delta_n^{th}(\boldsymbol{x}^{coop})$ is the threshold discount factor of client n to play the cooperative strategy profile $\boldsymbol{x}^{coop}$. If each client $n \in \mathcal{N}$ is patient enough, i.e., the discount factor satisfies $\delta_n \ge \delta_n^{th}(\boldsymbol{x}^{coop})$, all clients will play the cooperative participation strategy at the SPNE. For simplicity, we assume that all clients have the same discount factor, i.e., $\delta_n = \delta, \forall n \in \mathcal{N}$, as in [46] [51].

Next we show how to enforce the cooperative strategies derived in Theorem 3 and Theorem 4 as the SPNE in the repeated game.

Theorem 5. *Consider the following strategy profile: all clients choose the cooperative participation strategy $\boldsymbol{x}^{coop}$ (calculated in Theorem 3 or Theorem 4) until a client deviates, and then all clients play the NE $\boldsymbol{x}^*$ of the stage game SPFL (calculated in Theorem 1) in all future time slots. Such a strategy profile is the SPNE of Game 2 if*

$$\delta \ge \delta^{th}(\boldsymbol{x}^{coop}) \triangleq \max\{\delta_n^{th}(\boldsymbol{x}^{coop}) : \forall n \in \mathcal{N}\}. \tag{20}$$

Proof. See Appendix F in the supplementary material. □

Theorem 5 shows that when clients are patient enough, i.e., $\delta \ge \delta^{th}(\boldsymbol{x}^{coop})$, clients play the cooperative strategy $\boldsymbol{x}^{coop}$ at the SPNE, and no client has the incentive to decrease its chosen amount of local data.

Note that the cooperative strategy $\boldsymbol{x}^{coop}$ characterized in Theorem 3 and Theorem 4 is not unique, since the converted contributors can choose any cooperative participation level $x^{coop} \le x_l^{th}$. Therefore, the SPNE in Theorem 5 is not unique. Next we will design an algorithm to calculate the

Algorithm 2: An Algorithm to Calculate the Optimal SPNE of Game 2

Input: $\rho_n, \forall n \in \mathcal{N}, E, C, G, N, D, p$.

Output: The optimal SPNE $\boldsymbol{x}^{os} = \{x_n^{os} : \forall n \in \mathcal{N}\}$

1 **if** $\exists k \in \mathcal{N}$ *such that* $(N-k)D \leq h_k \leq (N+1-k)D$ **then**

2   Find the client l where $l = \min\{n \in \mathcal{N} : 1 \leq n \leq k-1, \rho_n > B_n\}$;

3   Compute the maximum amount of local data x_l^{th} that the converted contributors can choose by solving (13);

4   Solve the optimization problem (22);

5   Announce to each client $n \in \mathcal{N}$ the optimal SPNE:

$$x_n^{os} = \begin{cases} 0, & \text{if } n < l; \\ x_{cc}^{os}, & \text{if } l \leq n < k; \\ x_k^*, & \text{if } n = k; \\ D, & \text{if } n > k. \end{cases}$$

6 **else if** $\exists m \in \mathcal{N}$ *such that* $h_m < (N-m)D < h_{m+1}$ **then**

7   Find the client l where $l = \min\{n \in \mathcal{N} : 1 \leq n \leq m, \rho_n > B_n^{'}\}$;

8   Compute the maximum amount of local data x_l^{th} that the converted contributors can choose by solving (17);

9   Solve the optimization problem (24);

10   Announce to each client $n \in \mathcal{N}$ the optimal SPNE:

$$x_n^{os} = \begin{cases} 0, & \text{if } n < l; \\ x_{cc}^{os}, & \text{if } l \leq n \leq m; \\ D, & \text{if } n > m. \end{cases}$$

11 **end**

optimal SPNE that can minimize the number of free riders while maximizing the amount of local data for model training.

4.4 Optimal SPNE

In this section, we aim to derive the optimal SPNE that can minimize the number of free riders while maximizing the amount of local data for model training. In the following, we first define the optimal SPNE of Game 2, and then present the procedures (in Algorithm 2) to compute the optimal SPNE.

We first analyze the optimal SPNE $\boldsymbol{x}^{os} = \{x_n^{os} : \forall n \in \mathcal{N}\}$ for the case where the critical client k exists at the NE of Game 1 (as shown in Case I in Theorem 1). Since the optimal SPNE minimizes the number of free riders, it follows the solution structure $\boldsymbol{x}^{coop}$ shown in (11) in Theorem 3, and hence the optimal SPNE $\boldsymbol{x}^{os}$ can be written as:

$$x_n^{os} = \begin{cases} 0, & \text{if } n < l; \\ x_{cc}^{os}, & \text{if } l \leq n < k; \\ x_k^*, & \text{if } n = k; \\ D, & \text{if } n > k. \end{cases} \tag{21}$$

Here x_{cc}^{os} is converted contributors' strategy under the optimal SPNE which satisfies $0 \leq x_{cc}^{os} \leq x_l^{th}$, and x_l^{th} is defined in (12), (13). We can see from (21) that only converted contributors' strategy x_{cc}^{os} in the optimal SPNE $\boldsymbol{x}^{os}$ is unknown.

Now we derive x_{cc}^{os}. Since $\boldsymbol{x}^{os}$ is the SPNE of Game 2, it satisfies the one-stage deviation principle described in Theorem 5, i.e., $\delta \geq \delta^{th}(\boldsymbol{x}^{os})$. Furthermore, the optimal SPNE maximizes the amount of local data for model training, i.e., $(k-l)x_{cc}^{os} + x_k^* + (N-k)D$. In summary, x_{cc}^{os} is the optimal solution to the following optimization problem with respect to x_{cc}:

$$\begin{aligned} \max \quad & (k-l)x_{cc} + x_k^* + (N-k)D \\ \text{s.t.} \quad & 0 \leq x_{cc} \leq x_l^{th}, \\ & \delta \geq \delta^{th}(\boldsymbol{x}^{os}). \end{aligned} \tag{22}$$

Although the above optimization problem has a simple linear objective with respect to x_{cc}, we cannot derive the closed-form expression for the optimal solution x_{cc}^{os} since the constraint $\delta \geq \delta^{th}(\boldsymbol{x}^{os})$ is complicated due to the following two reasons. First, similar to (20), $\delta^{th}(\boldsymbol{x}^{os})$ takes the maximum value among $\delta_n^{th}(\boldsymbol{x}^{os}), \forall n \in \mathcal{N}$, and the max operator is a non-differentiable function. Second, similar to (19), $\delta_n^{th}(\boldsymbol{x}^{os})$ involves not only $\boldsymbol{x}^{os}$ but also $\boldsymbol{x}_n^{least}$, and hence the function $\delta_n^{th}(\boldsymbol{x}^{os})$ changes with x_{cc} in a highly non-linear manner (as shown in Appendix F in the supplementary material). In order to find the optimal solution x_{cc}^{os}, we can use greedy algorithms to solve the univariate optimization problem (22).

Next we analyze the optimal SPNE $\boldsymbol{x}^{os} = \{x_n^{os} : \forall n \in \mathcal{N}\}$ for the case where the critical client k does not exist at the NE of Game 1 (as shown in Case II in Theorem 1). Similarly, the optimal SPNE follows the solution structure $\boldsymbol{x}^{coop}$ shown in (15) in Theorem 4, and hence can be written as:

$$x_n^{os} = \begin{cases} 0, & \text{if } n < l; \\ x_{cc}^{os}, & \text{if } l \leq n \leq m; \\ D, & \text{if } n > m. \end{cases} \tag{23}$$

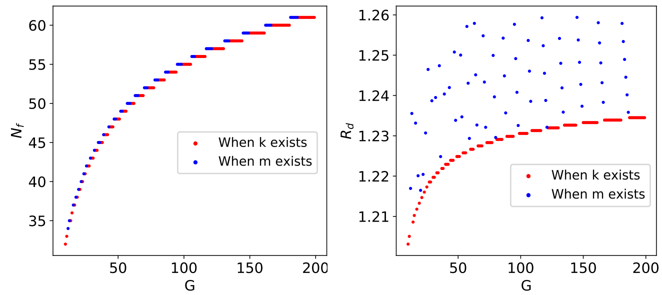
Here x_{cc}^{os} is converted contributors' strategy under the optimal SPNE which satisfies $0 \leq x_{cc}^{os} \leq x_l^{th}$, and x_l^{th} is defined in (16), (17). We can see from (23) that only converted contributors' strategy x_{cc}^{os} in the optimal SPNE $\boldsymbol{x}^{os}$ is unknown. We can derive x_{cc}^{os} by solving the following optimization problem with respect to x_{cc}:

$$\begin{aligned} \max \quad & (m-l+1)x_{cc} + (N-m)D \\ \text{s.t.} \quad & 0 \leq x_{cc} \leq x_l^{th}, \\ & \delta \geq \delta^{th}(\boldsymbol{x}^{os}). \end{aligned} \tag{24}$$

Similarly, we can use greedy algorithms to solve the above univariate optimization problem.

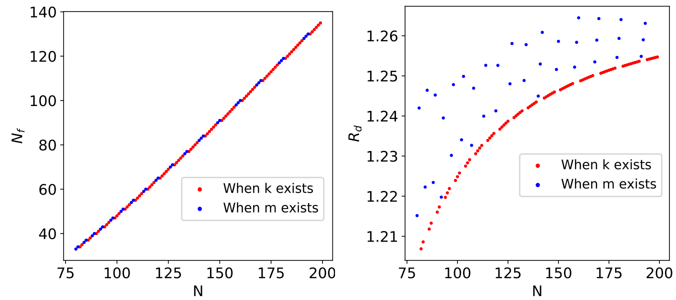
Algorithm 2 shows the procedures for the central server to calculate and announce the optimal SPNE of Game 2. To calculate the optimal SPNE of Game 2, the central server first needs to check whether the critical client k exists (Line 1 and Line 6 in Algorithm 2). Then the central server calculates the minimum number of free riders in cross-silo FL under the optimal SPNE, which depends on the value of each client's valuation parameter ρ_n and the value of B_n or B_n' (Line 2 and Line 7 in Algorithm 2). The central server then calculates the maximum amount of local data x_l^{th} that the converted contributors can choose under the optimal SPNE, by solving (13) or (17) (Line 3 and Line 8 in Algorithm 2). Finally the central server solves the optimization problem (22) or (24) and announces the optimal SPNE strategy x_n^{os} to each client $n \in \mathcal{N}$ (Line 4-5 and Line 9-10 in Algorithm 2).

The complexity of Algorithm 2 is $\mathcal{O}(N)$. Under the optimal SPNE, there will be $l-1$ free riders, and the amount of local data for model training is $(k-l)x_{cc}^{os} + x_k^* + (N-k)D$ when the critical client k exists or $(m-l+1)x_{cc}^{os} + (N-m)D$ when the critical client k does not exist.



(a) (b)

Fig. 3. (a) N_f under different G; (b) R_d under different G.



(a) (b)

Fig. 4. (a) N_f under different N; (b) R_d under different N.

5 Simulation Results

In this section, we evaluate the performance of our proposed optimal SPNE strategy in terms of two important metrics: the number of reduced free riders N_f and the total data contribution ratio R_d. The number of reduced free riders is the difference between the number of free riders at the NE of the stage game and the number of free riders at the optimal SPNE of the repeated game. That is, $N_f = k - l$ when the critical client k exists, and $N_f = m - l + 1$ when the critical client k does not exist. The total data contribution ratio is the ratio between the total amount of local data for model training at the optimal SPNE of the repeated game and that at the NE of the stage game, i.e., $R_d = \sum_{n\in\mathcal{N}} x_n^{os} / \sum_{n\in\mathcal{N}} x_n^*$. We will discuss the impact of several system parameters on N_f and R_d, including the iteration number G, the number of clients N, the total number of data D of each client, the computation cost parameter E, and the discount factor δ. We will also discuss the impact of the distribution of clients' valuation parameters on the number of reduced free riders.

For the parameter setting, we assume that there are $N = 100$ clients participating in the cross-silo FL processes, and each client has a local data set with $D = 10000$ local data samples. Furthermore, we assume that the number of iterations in a time slot $G = 50$, the coefficient of computation cost $E = \sqrt{\frac{12.5}{(50.5\times 10000)^3}}$, the communication cost $C = 0.002$, and the payment to the central server $p = 0.002$. We assume that the valuation for global model accuracy of client n is $\rho_n = n \times \frac{100}{N}$, and the discount factor of each client is $\delta = 0.6$.

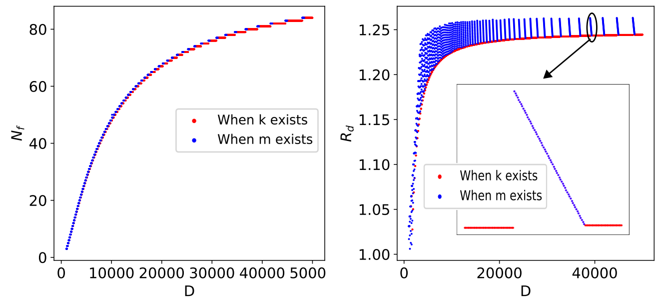
The impact of the number of iterations G. We show how the number of iterations G affects the number of reduced free riders N_f and the data contribution ratio R_d in Fig. 3.

Fig. 3(a) shows that the number of reduced free riders increases with G. In Fig. 3, the red dots represent the case where the critical client k exists, and the blue dots represent the case where the critical client k does not exist. A larger G means more iterations in one cross-silo FL process, in which case clients can achieve a global model with a higher model accuracy. As G increases, at the NE of the stage game SPFL, contributors with low valuations for the global model accuracy will change to be free riders, because the global model trained by contributors with high valuations under a large G is good enough. So the number of free riders at the NE of the stage game increases with the number of iterations G in cross-silo FL. Our proposed cooperative strategy under the optimal SPNE can reduce the number of free riders effectively, under which some free riders at the NE of the stage game changes to be converted contributors. The curve is a stepwise curve, where each step (i.e., the interval of G values) corresponds to a fixed number of free riders at the NE of the stage game SPFL. When the critical client k exists at equilibrium, its equilibrium strategy x_k^* (calculated in Theorem 1) decreases with G, until $x_k^* = 0$ in which case, the number of free riders at equilibrium increases by one. When the critical client k does not exist, the number of free riders at equilibrium does not change with G, until a contributor changes to be a free rider.

Fig. 3(b) shows how the data contribution ratio R_d changes with G. Specifically, when the critical client k exists, the data contribution ratio R_d generally increases with G (the red dots). As we discussed in Fig. 3(a), the number of free riders at the NE of the stage game increases with G, and there will be less local data for model training. Our cooperative strategy under the optimal SPNE can increase the amount of chosen local data since converted contributors choose a positive amount of local data for model training. Since the equilibrium strategy x_k^* of the critical client k decreases with G, convert contributors will choose more local data for model training to reduce their costs, and hence the total data chosen by clients increases. When the critical client k does not exist, under the same number of free riders, the data contribution ratio R_d decreases with G (the blue dots). The reason is that under a larger G, convert contributors will choose less local data for model training to reduce the computation cost while achieving the same global model accuracy. Under the same number of contributors, the total amount of training data chosen by converted contributors when the critical client k exists is less than that when the critical client k does not exist, thus the blue dots are above the red dots under the same number of contributors.

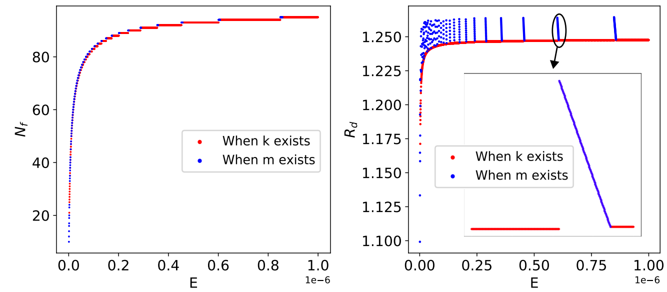
In summary, when the cross-silo FL process can achieve a good global model through a large number of training iterations G, the selfish participation behaviors among clients in cross-silo FL will lead to more free riders. Our proposed cooperative strategy can effectively reduce the number of free riders in clients' repeated interactions by up to 98%. It can also increase the amount of local data chosen by clients to perform model training by up to 26%.

The impact of the number of clients N. We show how the number of clients N affects the number of reduced free



(a) (b)

Fig. 5. (a) N_f under different D; (b) R_d under different D.



(a) (b)

Fig. 6. (a) N_f under different E; (b) R_d under different E.

riders N_f and the data contribution ratio R_d in Fig. 4.

Fig. 4(a) shows that the number of reduced free riders increases with N. The reason is that as N increases, the global model trained by the contributors with high valuation parameters is good enough, and hence the contributors with low valuations may become free riders to reduce their total costs. More clients will then choose to be free riders with the increase of N. It's interesting to note that as N increases, the ratio N_f/N increases. When the global model is good enough, if a new client joins the cross-silo FL process, it is very likely to choose to be a free rider.

Fig. 4(b) shows how the data contribution ratio R_d changes with N. Specifically, when the critical client k exists, the data contribution ratio R_d generally increases with N. As N increases, new clients who participate in the training process are more likely to be free riders at the NE of the stage game, while the cooperative strategy can make the new clients choose to be converted contributors and thus increase the data contribution ratio. When k does not exist, under the same number of free riders, the data contribution ratio decreases with N. The reason is that under the same number of free riders, a larger N indicates more contributors, and hence a larger amount of local data for model training, which leads to a global model with a higher model accuracy at the NE of stage game. Therefore, the converted contributors under the cooperative strategy choose a smaller amount of local data for model training to reduce the computation cost.

In summary, both the number of free riders and the proportion of free riders increase with the number of clients in the cross-silo FL system. Hence the selfish participation behavior hinders the scalability of the cross-silo FL system. Our proposed cooperative strategy solves this problem effectively.

The impact of the total number of local data samples D in each client's local data set. We show how the parameter D affects the number of reduced free riders N_f and the data contribution ratio R_d in Fig. 5.

Fig. 5(a) shows that the number of reduced free riders increases with D. When D is small, to achieve a good global model with a high model accuracy requires many clients to choose to be contributors to perform model training, and hence there are few free riders at the NE of the stage game. When D is large, a small number of contributors can train a good global model since each client has a large amount of local data to perform model training, and hence there are many free riders at the NE of the stage game. Our proposed cooperative strategy can effectively reduce the number of free riders.

Fig. 5(b) shows how the data contribution ratio R_d changes with D. Specifically, when the critical client k exists, the data contribution ratio R_d generally increases with D. When D is small, the curve increases quickly with D, and when D is large, R_d increases slowly in a zigzag pattern. When D is small, there are few free riders at the NE of the stage game, and hence there is no much room to increase the amount of local data for model training. As D increases, the cooperative strategy allows the converted contributors to choose a positive amount of local data to perform model training, which increases the amount of chosen data by up to 26%. The subfigure in Fig. 5(b) shows the zigzag shape of the curve under the same number of free riders. Note that the cooperative strategy of the critical client k is x_k^* which decreases with D as calculated in Theorem 1, while the cooperative strategy x^{coop} of the converted contributors increases with D. Hence the total amount of local data remains unchanged at the beginning of the curve. When k does not exist, under the same number of free riders, the data contribution ratio decreases with D. The reason is that a larger D leads to less accuracy loss, so the converted contributors will use less data for training to reduce the computation cost.

In summary, when each client in cross-silo FL has more local data samples, the selfish participation behavior leads to more free riders. Our proposed cooperative strategy can effectively reduce the number of free riders by up to 98.8%, and increase the data contribution by up to 26%.

The impact of the computation cost coefficient E. We show how the parameter E affects the number of reduced free riders N_f and the data contribution ratio R_d in Fig. 6.

Fig. 6(a) shows that the number of reduced free riders increases with E with a diminishing marginal return effect. When E is small, performing local model training incurs a small computation cost, and hence there are few free riders at the NE of the stage game. When E is large, clients experience a huge computation cost when performing model training, and hence many clients with low valuations for model accuracy choose to be free riders at the NE of the stage game to avoid computation cost. Our proposed cooperative strategy can effectively reduce the number of free riders.

Fig. 6(b) shows how the data contribution ratio R_d changes with E, and the trend of the curve is similar to

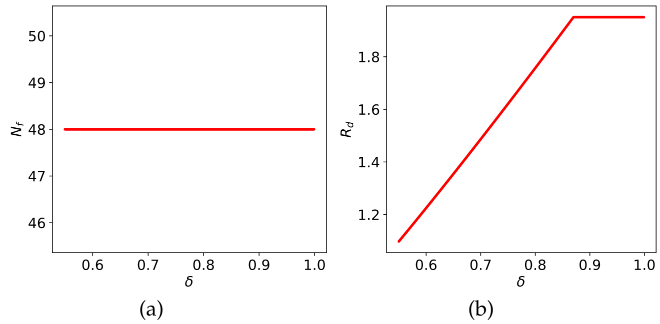


Fig. 7. (a) N_f under different δ; (b) R_d under different δ.

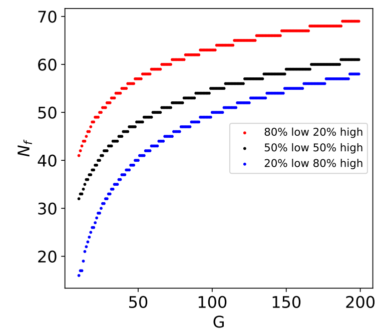


Fig. 8. N_f under three different distributions of ρ and different G

that in Fig. 5(b). Specifically, when the critical client k exists, the data contribution ratio R_d generally increases with E. When E is small, R_d increases quickly with E, and when E is large, R_d increases slowly in a zigzag pattern. When E is small, there are few free riders at the NE of the stage game, and hence there is no much room to increase the amount of local data for model training. As E increases, the cooperative strategy allows the converted contributors to choose a positive amount of local data to perform model training, which increases the amount of chosen data by up to 26.4%. The subfigure in Fig. 6(b) shows the zigzag shape of the curve under the same number of free riders. The reason is similar to that of the subfigure of Fig. 5(b).

The impact of the discount factor δ. We show how the discount factor δ affects the number of reduced free riders N_f and the data contribution ratio R_d in Fig. 7.

Fig. 7(a) shows that the discount factor δ does not affect the number of reduced free riders. The reason is that the number of free riders at the NE of the stage game and the number of free riders under the cooperative strategy depend on the valuation parameters $\{\rho_n : \forall n \in \mathcal{N}\}$, the bounds $\{B_n : \forall n \in \mathcal{N}\}$, and the local data set sizes $\{D_n : \forall n \in \mathcal{N}\}$, which are independent of δ.

Fig. 7(b) shows that the data contribution ratio first increases with δ, and then remains to be a constant when δ is larger than 0.87. A larger discount factor δ indicates that clients are more patient, and hence they are more willing to choose a larger amount of local data to reduce the long-term discounted total cost. When clients are patient enough, i.e., $\delta \geq 0.87$, all clients choose to be contributors who perform model training with all their local data, in which case the data contribution ratio is $R_d = 1.96$.

The impact of the distributions of ρ. Fig. 8 shows how the number of reduced free riders changes with G under three distributions of ρ: 80% low, 20% high; 50% low, 50% high; and 20% low, 80% high. Here 80% low means the valuation parameters of 80 clients are uniformly distributed in $(0, 50]$, 20% high means the valuation parameters of 20 clients are uniformly distributed in $(50, 100]$. We can see that there will be more free riders at the NE of the stage game when more clients have low valuations for global model accuracy. Our cooperative strategy can effectively reduce the number of free riders.

6 Conclusion

In this work, we analyze clients' selfish participation behaviors in cross-silo FL. We model the interactions among clients in a single cross-silo FL process as a selfish participation game and derive the unique Nash equilibrium for it. We propose a distributed algorithm for clients to calculate the equilibrium participation strategy. We show that clients' selfish participation behaviors lead to free riders at equilibrium. The existence of free riders hampers clients' long-term participation in cross-silo FL. We model clients' interactions in the long-term cross-silo FL processes as a repeated game in the infinite time horizon. We derive the optimal SPNE that can minimum the number of free riders while maximizing the amount of local data for model training, which is a cooperative strategy enforced by a punishment strategy. We propose an algorithm to calculate the optimal SPNE strategy. Simulation results show that the cooperative strategy can reduce the number of free riders effectively by up to 98.8% and increase the amount of data for training by up to 96%. For future work, one direction is to design proper incentive mechanisms such as differentiated pricing policy to incentivize clients to choose higher participation levels. It is also interesting to analyze the economic interactions between the central server and clients in practical cross-silo FL markets. Another direction is to analyze the interactions among clients in the incomplete information scenario where clients do not know the participation strategies of other clients or in the asymmetric information scenario where the valuation for global model accuracy is the private information of each client.

References

[1] H. Li, K. Ota, and M. Dong, "Learning iot in edge: Deep learning for the internet of things with edge computing," *IEEE network*, vol. 32, no. 1, pp. 96–101, 2018.

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

[3] Y. Huang, L. Chu, Z. Zhou, L. Wang, J. Liu, J. Pei, and Y. Zhang, "Personalized cross-silo federated learning on non-iid data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

[4] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[5] L. Insurance, "Swiss re and chinese firm webank partner to explore use of ai in reinsurance," https://www.lifeinsuranceinternational.com/news/swiss-re-webank/, Accessed Apr. 11, 2021.

[6] E. research project, “Mammogram assessment with nvidia clara federated learning,” https://blogs.nvidia.com/blog/2020/04/15/federated-learning-mammogram-assessment/, Accessed Apr. 11, 2021.

[7] Owkin, “Owkin solutions,” https://owkin.com/solutions, Accessed Apr. 11, 2021.

[8] C. Hale, “The melloddy project,” https://www.fiercebiotech.com/special-report/melloddy-project, Accessed Apr. 11, 2021.

[9] M. Tang and V. W. Wong, “An incentive mechanism for cross-silo federated learning: A public goods perspective.”

[10] P. J. Miller and D. M. Chin, “Using monthly data to improve quarterly model forecasts,” *Federal Reserve Bank of Minneapolis Quarterly Review*, vol. 20, pp. 16–28, 1996.

[11] N. Van Leersum, H. Snijders, D. Henneman, N. Kolfschoten, G. Gooiker, M. Ten Berge, E. Eddes, M. Wouters, R. Tollenaar, W. Bemelman *et al.*, “The dutch surgical colorectal audit,” *European Journal of Surgical Oncology (EJSO)*, vol. 39, no. 10, pp. 1063–1070, 2013.

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1273–1282.

[13] G. Zhu, Y. Wang, and K. Huang, “Broadband analog aggregation for low-latency federated edge learning,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 491–506, 2019.

[14] H. Wang, Z. Kaplan, D. Niu, and B. Li, “Optimizing federated learning on non-iid data with reinforcement learning,” in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1698–1707.

[15] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, “Adaptive federated learning in resource constrained edge computing systems,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[16] X. Mo and J. Xu, “Energy-efficient federated edge learning with joint communication and computation design,” *arXiv preprint arXiv:2003.00199*, 2020.

[17] B. Luo, X. Li, S. Wang, J. Huang, and L. Tassiulas, “Cost-effective federated learning design,” *arXiv preprint arXiv:2012.08336*, 2020.

[18] N. Ding, Z. Fang, and J. Huang, “Optimal contract design for efficient federated learning with multi-dimensional private information,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 186–200, 2020.

[19] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, “Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory,” *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 700–10 714, 2019.

[20] W. J. Baumol, “Welfare economics and the theory of the state,” in *The encyclopedia of public choice*. Springer, 2004, pp. 937–940.

[21] T. Locher, P. Moore, S. Schmid, and R. Wattenhofer, “Free riding in bittorrent is cheap,” 2006.

[22] M. Feldman and J. Chuang, “Overcoming free-riding behavior in peer-to-peer systems,” *ACM sigecom exchanges*, vol. 5, no. 4, pp. 41–50, 2005.

[23] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica, “Free-riding and whitewashing in peer-to-peer systems,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 5, pp. 1010–1019, 2006.

[24] J. Lin, M. Du, and J. Liu, “Free-riders in federated learning: Attacks and defenses,” *arXiv preprint arXiv:1911.12560*, 2019.

[25] J. Huang, R. Talbi, Z. Zhao, S. Boucchenak, L. Y. Chen, and S. Roos, “An exploratory analysis on users’ contributions in federated learning,” *arXiv preprint arXiv:2011.06830*, 2020.

[26] Y. Fraboni, R. Vidal, and M. Lorenzi, “Free-rider attacks on model aggregation in federated learning,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 1846–1854.

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

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

[29] O. Marfoq, C. Xu, G. Neglia, and R. Vidal, “Throughput-optimal topology design for cross-silo federated learning,” *arXiv preprint arXiv:2010.12229*, 2020.

[30] U. Majeed, L. U. Khan, and C. S. Hong, “Cross-silo horizontal federated learning for flow-based time-related-features oriented traffic classification,” in *2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, 2020, pp. 389–392.

[31] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, “Batchcrypt: Efficient homomorphic encryption for cross-silo federated learning,” in *2020 {USENIX} Annual Technical Conference ({USENIX}{ATC} 20)*, 2020, pp. 493–506.

[32] N. Ding, Z. Fang, and J. Huang, “Incentive mechanism design for federated learning with multi-dimensional private information,” in *2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*. IEEE, 2020, pp. 1–8.

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

[34] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao, “Optimal distributed online prediction using mini-batches.” *Journal of Machine Learning Research*, vol. 13, no. 1, 2012.

[35] X. Zhang, F. Li, Z. Zhang, Q. Li, C. Wang, and J. Wu, “Enabling execution assurance of federated learning at untrusted participants,” in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1877–1886.

[36] M. Li, T. Zhang, Y. Chen, and A. J. Smola, “Efficient mini-batch training for stochastic optimization,” in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 661–670.

[37] D. Conway-Jones, T. Tuor, S. Wang, and K. K. Leung, “Demonstration of federated learning in a resource-constrained networked environment,” in *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2019, pp. 484–486.

[38] 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 *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 1387–1395.

[39] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, “Performance optimization of federated learning over wireless networks,” in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[40] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, “Federated learning for internet of things: Recent advances, taxonomy, and open challenges,” *arXiv preprint arXiv:2009.13012*, 2020.

[41] G. H. Lee and S.-Y. Shin, “Federated learning on clinical benchmark data: Performance assessment,” *Journal of medical Internet research*, vol. 22, no. 10, p. e20891, 2020.

[42] W.-T. Chang and R. Tandon, “Communication efficient federated learning over multiple access channels,” *arXiv preprint arXiv:2001.08737*, 2020.

[43] Y. Tian, Y. Liu, and P. Chen, “Bayesian analysis for a mixture of log-normal distributions,” *Department of Mathematics Northwest University*, vol. 6, no. 1, pp. 65–71, 2010.

[44] J. Gao, L. Zhao, and X. Shen, “Network utility maximization based on an incentive mechanism for truthful reporting of local information,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 7523–7537, 2018.

[45] G. Gao, M. Xiao, J. Wu, L. Huang, and C. Hu, “Truthful incentive mechanism for nondeterministic crowdsensing with vehicles,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 12, pp. 2982–2997, 2018.

[46] L. Song, Y. Xiao, and M. Van Der Schaar, “Demand side management in smart grids using a repeated game framework,” *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1412–1424, 2014.

[47] S. Shakkottai and R. Srikant, “Economics of network pricing with multiple isps,” *IEEE/ACM Transactions On Networking*, vol. 14, no. 6, pp. 1233–1245, 2006.

[48] M. Mohri, G. Sivek, and A. T. Suresh, “Agnostic federated learning,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 4615–4625.

[49] J. W. Friedman, “A non-cooperative equilibrium for supergames,” *The Review of Economic Studies*, vol. 38, no. 1, pp. 1–12, 1971.

[50] E. Hendon, H. J. Jacobsen, and B. Sloth, “The one-shot-deviation principle for sequential rationality,” *Games and Economic Behavior*, vol. 12, no. 2, pp. 274–282, 1996.

[51] G. J. Mailath and L. Samuelson, *Repeated games and reputations: long-run relationships*. Oxford university press, 2006.

APPENDIX A
PROOF OF LEMMA 1

Take the first-order derivative of (5) and let it to be 0, that is,

$$E - \frac{\rho_n}{2\sqrt{G(x_n + \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'})^3}} = 0. \tag{25}$$

So we can get,

$$x_n = \sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'}. \tag{26}$$

Since the second-order derivative of (5) satisfies

$$\frac{3\rho_n}{4\sqrt{G(x_n + \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'})^5}} > 0, \tag{27}$$

the total cost of each client n, calculated in (5), decreases monotonically in the interval $(-\infty, \sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'})$ and increases monotonically in the interval $(\sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'}, +\infty)$.

Since the strategy of client n is in the range $[0, D_n]$, we know that the best response of each client n to minimize its total cost is:

$$x_n^{BR}(\boldsymbol{x}_{-n}) = \min \left\{ D_n, \max \left\{ \sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'}, 0 \right\} \right\}. \tag{28}$$

APPENDIX B
PROOF OF THEOREM 1

We will prove that under the Nash equilibrium, no client can get a lower cost by changing its strategy while keeping other clients' strategies unchanged. Here we denote $\sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sum_{n' \in \mathcal{N}, n' \neq n} x_{n'})$ as x_n^{opt} for convenience and x_n^{opt} is the stationary point of the cost function of client n.

When the critical client k exists, for client k, the stationary point of client k's cost function is $x_k^{opt} = \sqrt[3]{\frac{\rho_k^2}{4GE^2}} - (N-k)D$, because $(N-k)D \leq \sqrt[3]{\frac{\rho_k^2}{4GE^2}} \leq (N-k+1)D$, we can get $0 \leq x_k^{opt} \leq D$, the best response of client k is $x_k^{BR}(\boldsymbol{x}_{-k}) = \sqrt[3]{\frac{\rho_k^2}{4GE^2}} - (N-k)D$.

Similarly for client $k-1$, $x_{k-1}^{opt} = \sqrt[3]{\frac{\rho_{k-1}^2}{4GE^2}} - (N-k)D - x_k^{BR}(\boldsymbol{x}_{-k}) = \sqrt[3]{\frac{\rho_{k-1}^2}{4GE^2}} - \sqrt[3]{\frac{\rho_k^2}{4GE^2}} < 0$, the best response of client $k-1$ is $x_{k-1}^{BR}(\boldsymbol{x}_{-(k-1)}) = 0$. For client $k-2, k-3, \cdots, 1$, similarly we can get $x_1^{BR}(\boldsymbol{x}_{-1}) = x_2^{BR}(\boldsymbol{x}_{-2}) = \cdots = x_{k-2}^{BR}(\boldsymbol{x}_{-(k-2)}) = 0$.

For client $k+1$, $x_{k+1}^{opt} = \sqrt[3]{\frac{\rho_{k+1}^2}{4GE^2}} - (N-k-1)D - x_k^{BR}(\boldsymbol{x}_{-k}) = \sqrt[3]{\frac{\rho_{k+1}^2}{4GE^2}} - \sqrt[3]{\frac{\rho_k^2}{4GE^2}} + D > D$, the best response of client $k+1$ is $x_{k+1}^{BR}(\boldsymbol{x}_{-(k+1)}) = D$. For client $k+2, k+3, \cdots, N$, similarly we can get $x_{k+2}^{BR}(\boldsymbol{x}_{-(k+2)}) = x_{k+3}^{BR}(\boldsymbol{x}_{-(k+3)}) = \cdots = x_N^{BR}(\boldsymbol{x}_{-N}) = D$.

When the critical client k does not exist. There must exist a client m as mentioned in Case 2 in Theorem 1.

For client m, $x_m^{opt} = \sqrt[3]{\frac{\rho_m^2}{4GE^2}} - (N-m)D$. Because $\sqrt[3]{\frac{\rho_m^2}{4GE^2}} < (N-m)D < \sqrt[3]{\frac{\rho_{m+1}^2}{4GE^2}}$, $x_m^{opt} < 0$, so the best response of client m is $x_m^{BR}(\boldsymbol{x}_{-m}) = 0$.

For client $m-1$, $x_{m-1}^{opt} = \sqrt[3]{\frac{\rho_{m-1}^2}{4GE^2}} - (N-m)D < \sqrt[3]{\frac{\rho_m^2}{4GE^2}} - (N-m)D < 0$, the best response of client $m-1$ is $x_{m-1}^{BR}(\boldsymbol{x}_{-(m-1)}) = 0$. Similarly, $x_1^{BR}(\boldsymbol{x}_{-1}) = x_2^{BR}(\boldsymbol{x}_{-2}) = \cdots = x_m^{BR}(\boldsymbol{x}_{-m}) = 0$.

For client $m+1$, $x_{m+1}^{opt} = \sqrt[3]{\frac{\rho_{m+1}^2}{4GE^2}} - (N-m-1)D > D$, the best response of client $m+1$ is $x_{m+1}^{BR}(\boldsymbol{x}_{-(m+1)}) = D$.

For client $m+2$, $x_{m+2}^{opt} = \sqrt[3]{\frac{\rho_{m+2}^2}{4GE^2}} - (N-m-2)D - x_{m+1}^{BR}(\boldsymbol{x}_{-(m+1)}) = \sqrt[3]{\frac{\rho_{m+2}^2}{4GE^2}} - \sqrt[3]{\frac{\rho_{m+1}^2}{4GE^2}} + D > D$, the best response of client $m+2$ is $x_{m+2}^{BR}(\boldsymbol{x}_{-(m+2)}) = D$. Similarly, $x_{m+1}^{BR}(\boldsymbol{x}_{-(m+1)}) = x_{m+2}^{BR}(\boldsymbol{x}_{-(m+2)}) = \cdots = x_N^{BR}(\boldsymbol{x}_{-N}) = D$.

We can see that under this strategy, each client's strategy is the best response to other clients' strategies, so this strategy is the Nash equilibrium of Game 1.

APPENDIX C
PROOF OF THEOREM 2

We will prove Theorem 2 by mathematical induction. For convenience, we denote $\sqrt[3]{\frac{\rho_n^2}{4GE^2}}$ as h_n. We will prove that no matter what initial strategies clients choose, they strategies will eventually converge to the Nash equilibrium.

(1) There are two clients y_1, y_2, $y_1 < y_2$, whose initial strategy is a and b respectively, a, b can be any value in $[0, D]$. For client y_1, he will calculate the stationary point of his cost function $x_{y_1}^{opt} = h_{y_1} - b$, now we suppose $h_{y_1} - b \in [0, D]$, so the best response of client y_1 is $x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) = h_{y_1} - b$. Then client y_2 will calculate the stationary point of his cost function $x_{y_2}^{opt} = h_{y_2} - h_{y_1} + b$, here we still assume $h_{y_2} - h_{y_1} + b \in [0, D]$, so the best response of client y_2 is $x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}) = h_{y_2} - h_{y_1} + b$. Then client y_1 will update his best response $x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) = h_{y_1} - h_{y_2} + h_{y_1} - b$. Client y_2 will update his best response $x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}) = h_{y_2} - h_{y_1} + h_{y_2} - h_{y_1} + b$. As this process goes on, we can see that $x_{y_1}^{BR}(\boldsymbol{x}_{-y_1})$ continues to decrease from $h_{y_1} - b$ and $x_{y_2}^{BR}(\boldsymbol{x}_{-y_2})$ continues to increase from b until one client reaches the boundary, i.e., 0 or D. Let's assume that client y_1 reaches the boundary 0 first, i.e., $x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) = 0, x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}) < D$. So $x_{y_2}^{opt} = h_{y_2} - 0 = h_{y_2} = x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}), x_{y_1}^{opt} = h_{y_1} - h_{y_2} < 0, x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) = 0$. It eventually converges to $x_{y_1}^* = 0, x_{y_2}^* = h_{y_2}$. Similarly, if y_2 reaches the boundary D first, i.e., $x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) > 0, x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}) = D)$. So $x_{y_1}^{opt} = h_{y_1} - D > 0$, $x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) = h_{y_1} - D$, $x_{y_2}^{opt} = h_{y_2} - h_{y_1} + D > D$, $x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}) = D$. It eventually converges to $x_{y_1}^* = h_{y_1} - D, x_{y_2}^* = D$. It illustrates that no matter which client choose which value, it will always converge to the Nash equilibrium under the condition $\rho_1 < \rho_2$.

(2) Suppose when there are n clients who satisfies $\rho_1 < \rho_2 < \cdots < \rho_N$, there initial strategies are $a_1, a_2, \cdots, a_N \in [0, D]$ respectively. Their strategy will eventually converge to the Nash equilibrium:

$x_1^* = x_2^* = \cdots = x_{k-1}^* = 0, x_k^* = h_k - (N-k)D, x_{k+1}^* = x_{k+2}^* = \cdots = x_N^* = D$ (Similarly if k don't exist)

(3) When there are $n+1$ clients who satisfies $\rho_1 < \rho_2 < \cdots < \rho_N < \rho_{N+1}$, their initial strategies are $a_1, a_2, \cdots, a_N, a_{N+1} \in [0, D]$. According to (2), the last N clients will eventually converge to the Nash equilibrium beacuse they satisfy the condition $\rho_2 < \cdots < \rho_N < \rho_{N+1}$. When client y_1 adds to the list, let's see what will happens from the converge point of the last N clients.

We have discussed that there may have two situations. We first consider that there exists a critical client $k+1$ in the interval. The last N clients will eventually converge to the form

$$x_n^* = \begin{cases} 0, & \text{if } y_2 \leq n < y_{k+1}; \\ h_{k+1} - (N-k)D, & \text{if } n = y_{k+1}; \\ D, & \text{if } y_{k+1} < n \leq N. \end{cases} \tag{29}$$

When clients y_1 adds to the list, $x_{y_1}^{opt} = h_{y_1} - (N-k)D - x_{y_{k+1}}^*$. According to (2), we know that $x_{y_2}^{opt} = h_{y_2} - (N-k)D - x_{y_{k+1}}^* < 0$. We know that $h_{y_2} > h_{y_1}$, so $x_{y_1}^{opt} < 0, x_{y_1}^{BR}(\boldsymbol{x}_{-y_1}) = 0$. Then calculate the best response of the other n clients, $x_{y_2}^{opt} = h_{y_2} - (N-k)D - x_{y_{k+1}}^* < 0$, $x_{y_2}^{BR}(\boldsymbol{x}_{-y_2}) = 0, \cdots, x_{y_{k+1}}^{opt} = h_{y_{k+1}} - (N-k)D \in [0, D]$, $x_{y_{k+1}}^{BR}(\boldsymbol{x}_{-y_{k+1}}) = h_{y_{k+1}} - (N-k)D$, $x_{y_{k+2}}^{opt} = h_{y_{k+2}} - (N-k)D - x_{y_{k+1}}^* + D > D$, $x_{y_{k+2}}^{BR}(\boldsymbol{x}_{-y_{k+2}}) = D, \cdots, x_{y_{N+1}}^{opt} = h_{y_{N+1}} - (N-k)D - x_{y_{k+1}}^* + D > D$, $x_{y_{N+1}}^{BR}(\boldsymbol{x}_{-y_{N+1}}) = D$, we eventually get

$$x_n^* = \begin{cases} 0, & \text{if } y_1 \leq n < y_{k+1}; \\ h_{k+1} - (N-k)D, & \text{if } n = y_{k+1}; \\ D, & \text{if } y_{k+1} < n \leq N. \end{cases} \tag{30}$$

Similarly, when we consider that there does not exist the critical client $k+1$ in the interval. The game will eventually converge to

$$x_n^* = \begin{cases} 0, & \text{if } y_1 \leq n \leq y_{m+1}; \\ D, & \text{if } y_{m+1} < n \leq N. \end{cases} \tag{31}$$

According to (3), when there are $n+1$ clients who satisfy $\rho_1 < \rho_2 < \cdots < \rho_N < \rho_{N+1}$, no matter no matter what initial strategies clients choose, their strategies will eventually converge to the unique Nash equilibrium. So far, we have proved the uniqueness of Nash equilibrium.

Appendix D
Proof of Theorem 3.

First, we prove a lemma that guarantees $B_n \leq O_n$ and $B_n^{'} \leq O_n^{'}$.

Lemma 2. *For $k \in \mathcal{N}$, we have $B_n \leq O_n$, and for $m \in \mathcal{N}$, we have $B_n^{'} \leq O_n^{'}$.*

Proof. Set the function $f(k)$:

$$f(k) = 3Dk + 2\sqrt{\frac{((N-k)D + x_k^*)^3}{(N-n)D + x_k^*}}. \tag{32}$$

Take the first derivative of $f(k)$:

$$f'(k) = 3D - 3D\sqrt{\frac{(N-k)D + x_k^*}{(N-n)D + x_k^*}} \geq 0 \ k \in \mathcal{N}. \tag{33}$$

The minimum value of $f(k)$ in the domain $[n, N]$ is $f(n)$:

$$f(n) = (2N+n)D + 2x_k^*. \tag{34}$$

Then we know that $f(k) \geq f(n)$, so

$$3Dk + 2\sqrt{\frac{((N-k)D + x_k^*)^3}{(N-n)D + x_k^*}} \geq (2N+n)D + 2x_k^*. \tag{35}$$

Deal with the inequality (35), we can get

$$\frac{ED\sqrt{G}}{\frac{1}{\sqrt{(N-k)D+x_k^*}} - \frac{1}{\sqrt{(N-n)D+x_k^*}}} \geq \frac{2E\sqrt{G((N-k)D + x_k^*)^3}}{k-n}. \tag{36}$$

□

When the critical client k does not exist, the process is similarly to the above, and we will not repeat it.

Then we will prove that under the cooperative strategy, each client's cost is less than that at the NE of Game 1.

We denote $F_l(\boldsymbol{x}^{coop}) - F_l(\boldsymbol{x}^*)$ as $H_l(x^{coop})$, which is the difference between the cost under cooperative strategy and the cost at the NE of Game 1. If $H_l(x^{coop}) < 0$, the cost of each client under cooperative strategy is less than that at the NE of Game 1. Specifically, $H_l(x^{coop}) = \frac{\rho_l}{\sqrt{G}}(\frac{1}{\sqrt{(k-l)x^{coop} + x_k^* + (N-k)D}} - \frac{1}{\sqrt{(N-k)D + x_k^*}}) + Ex^{coop}$.

Take the first derivative of $H_l(x^{coop})$:
$H_l^{'}(x^{coop}) = E - \frac{\rho_l(k-l)}{2\sqrt{G((k-l)x^{coop} + x_k^* + (N-k)D)^3}}$.

When $\rho_l > B_l$, $H_l^{'}(0) < 0, H_l(0) = 0, H_l^{'}(+\infty) > 0$. It's easy to know that $H_l^{'}(x^{coop})$ is a monotonically increasing function of x^{coop}, so the curve of $H_l(x^{coop})$ is a curve that monotonically decreases and then monotonically increases, like a parabola with an upper opening. When $\rho_l > O_l, H_l(D) > 0$, the unique intersection of $H_l(x^{coop})$ and the x^{coop} axis at $(0, +\infty)$ is larger than D, so the maximum amount of local data that client l can choose under the cooperative strategy is $x_l^{th} = D$. The $k-l$ converted contributors can choose any same strategy on $(0, D]$ as their cooperative strategy. This strategy can reduce their cost. When $B_l < \rho_l \leq O_l$, $H_l(D) \geq 0$, the unique intersection of $H(x^{coop})$ and the x^{coop} axis at $[0, +\infty)$ is the solution of the implicit function (13), we set it as $x_l^{th}(\rho_l)$. So $x_l^{th} = x_l^{th}(\rho_l)$, the $k-l$ converted contributors can choose any strategy on $(0, x_l^{th}(\rho_l))$ as their cooperative strategy. This strategy can reduce their cost.

When $\rho_l \leq B_l$, it means $H_l^{'}(0) \geq 0, H_l(0) = 0$, $H_l(x^{coop})$ is a monotonically increasing function of x^{coop}, so there is no intersection at $(0, +\infty)$, all free riders should choose 0 as their strategy. If $\rho_1 \leq B_1, \rho_2 \leq B_2, \cdots, \rho_{k-1} \leq B_{k-1}$, it means that there will not exist a cooperative strategy that can reduce the cost of each client at the same time, so all client will not cooperate and play the NE of the stage game. The reason of this situation is clients has low valuation for the accuracy of the global model. With the increase of training data, the local model is good enough and the global model has small improvement. The reduction of accuracy loss is very small (i.e., the difference of accuracy loss between the NE of stage game and cooperative strategy of repeated game), while the impact of computation cost is very significant. Any increase in data will lead to the increase of total cost.

Now we will illustrate why each client can reduce their cost under the cooperative strategy.

According to Theorem 3, clients $1, 2, \cdots, l-1$ will choose 0 as their strategy. Because $(k-l)x^{coop} + x_k^* + (N-k)D > (N-k)D + x_k^*$, we can get

$$H_1(x^{coop}) = \frac{\rho_1}{\sqrt{G}}(\frac{1}{\sqrt{(k-l)x^{coop} + x_k^* + (N-k)D}} - \frac{1}{\sqrt{(N-k)D + x_k^*}}) < 0. \tag{37}$$

Similarly, $H_2(x^{coop}) < 0, \cdots, H_{l-1}^{coop} < 0$.

Because $\rho_l > B_l$, we can get

$$H_l(x^{coop}) = \frac{\rho_l}{\sqrt{G}}(\frac{1}{\sqrt{(k-l)x^{coop} + x_k^* + (N-k)D}} - \frac{1}{\sqrt{(N-k)D + x_k^*}}) + Ex^{coop} \leq 0. \tag{38}$$

Because $\rho_l < \rho_{l+1} < \cdots < \rho_{k-1}$, we can get

$$H_{l+1}(x^{coop}) = \frac{\rho_{l+1}}{\sqrt{G}}(\frac{1}{\sqrt{(k-l)x^{coop} + x_k^* + (N-k)D}} - \frac{1}{\sqrt{(N-k)D + x_k^*}}) + Ex^{coop} < H_l(x^{coop}) < 0. \tag{39}$$

Similarly, $H_{l+2}(x^{coop}) < 0, \cdots, H_{k-1}(x^{coop}) < 0$.

For client k, his cooperative strategy is x_k^*, we can get

$$H_k(x^{coop}) = \frac{\rho_k}{\sqrt{G}}(\frac{1}{\sqrt{(k-l)x^{coop} + x_k^* + (N-k)D}} - \frac{1}{\sqrt{(N-k)D + x_k^*}}) < 0. \tag{40}$$

For client $k+1$, his cooperative strategy is D, we can get

$$H_{k+1}(x^{coop}) = \frac{\rho_{k+1}}{\sqrt{G}}(\frac{1}{\sqrt{(k-l)x^{coop} + x_k^* + (N-k)D}} - \frac{1}{\sqrt{(N-k)D + x_k^*}}) < 0. \tag{41}$$

Similarly, $H_{k+2}(x^{coop}) < 0, \cdots, H_N(x^{coop}) < 0$.

Appendix E
Proof of Theorem 4.

We will prove that under the cooperative strategy, each client can reduce their cost under the cooperative strategy.

Similarly to the proof of Theorem 3, $F_l(\boldsymbol{x}^{coop}) - F_l(\boldsymbol{x}^*) = H_l(x^{coop})$ which is the difference between the cost under cooperative strategy and the cost at the NE of the Game 1. Specifically, $H_l(x^{coop}) = \frac{\rho_l}{\sqrt{G}}(\frac{1}{\sqrt{(m-l+1)x^{coop} + (N-m)D}} - \frac{1}{\sqrt{(N-m)D}}) + Ex^{coop}$.

Take the first derivative of $H_l(x^{coop})$: $H_l^{'}(x^{coop}) = E - \frac{(m-l+1)\rho_l}{2\sqrt{G((N-m)D + (m-l+1)x_1^{coop})^3}}$.

When $\rho_l > B_l^{'}$, $H_l^{'}(0) < 0, H_l(0) = 0, H_l^{'}(+\infty) > 0$. It's easy to know that $H_l^{'}(x^{coop})$ is a monotonically increasing function of x^{coop}, so the curve of $H_l(x^{coop})$ is a curve that monotonically decreases and then monotonically increases, like a parabola with an upper opening. When $\rho_l > O_l^{'}, H_l(D) < 0$, the unique intersection of $H_l(x^{coop})$ and the x^{coop} axis at $(0, +\infty)$ is larger than D, so the maximum amount of local data that client l can choose under the cooperative strategy is $x_l^{th} = D$. The $m-l+1$ converted contributors can choose any same strategy on $(0, D]$ as their cooperative strategy. This strategy can reduce their cost. When $B_l^{'} < \rho_l \leq O_l^{'}$, $H_l(D) \geq 0$, the unique intersection of $H_l(x^{coop})$ and the x^{coop} axis at $(0, +\infty)$ is the solution of the implicit function (17), we set it as $x_l^{th}(\rho_l)$, so $x_l^{th} = x_l^{th}(\rho_l)$. The $m-l+1$ converted contributors can choose any strategy on $(0, x_l^{th}(\rho_l))$ as their cooperative strategy. This strategy can reduce their cost.

When $\rho_l \leq B_l^{'}$, it means $H_l^{'}(0) \geq 0, H_l(0) = 0$, $H_l(x^{coop})$ is a monotonically increasing function of x^{coop}, so there is no intersection at $(0, +\infty)$, all free riders should choose 0 as their strategy. If $\rho_1 \leq B_1^{'}, \rho_2 \leq B_2^{'}, \cdots, \rho_m \leq B_m^{'}$, it means that there will not exist a cooperative strategy to reduce the cost of each client at the same time, so all client will not cooperate and play the NE of Game 1. The reason is the same as in the proof of Theorem 3

According to Theorem 4, clients $1, 2, \cdots, l-1$ will choose 0 as their strategy. Because $(m-l+1)x^{coop} + (N-m)D > (N-m)D$, we can get

$$H_1(x^{coop}) = \frac{\rho_1}{\sqrt{G}}(\frac{1}{\sqrt{(m-l+1)x^{coop} + (N-m)D}} - \frac{1}{\sqrt{(N-m)D}}) < 0. \tag{42}$$

Similarly, $H_2(x^{coop}) < 0, \cdots, H_{l-1}(x^{coop}) < 0$.

Because $\rho_l > B_l^{'}$, we can get

$$H_l(x^{coop}) = \frac{\rho_l}{\sqrt{G}}(\frac{1}{\sqrt{(m-l+1)x^{coop} + (N-m)D}} - \frac{1}{\sqrt{(N-m)D}}) + Ex^{coop} \leq 0. \tag{43}$$

Because $\rho_l < \rho_{l+1} < \cdots < \rho_m$, we can get

$$H_{l+1}(x^{coop}) = \frac{\rho_{l+1}}{\sqrt{G}}(\frac{1}{\sqrt{(m-l+1)x^{coop} + (N-m)D}} - \frac{1}{\sqrt{(N-m)D}}) + Ex^{coop} < H_l(x^{coop}) < 0. \tag{44}$$

Similarly, $H_{l+2}(x^{coop}) < 0, \cdots, H_m(x^{coop}) < 0$.

For client $m+1$, his cooperative strategy is D, we can get

$$H_{m+1}(x^{coop}) = \frac{\rho_{m+1}}{\sqrt{G}}(\frac{1}{\sqrt{(m-l+1)x^{coop} + (N-m)D}} - \frac{1}{\sqrt{(N-m)D}}) < 0. \tag{45}$$

Similarly, $H_{m+2}(x^{coop}) < 0, \cdots, H_N(x^{coop}) < 0$.

Appendix F
Proof of Theorem 5

Here we assume cooperative free riders choose x^{coop} as their cooperative strategy.

For client n, when client n utilizes the cooperation of other clients to achieve the least cost at current time slot. The cost function of client n is

$$\begin{aligned} &F_n(\boldsymbol{x}_n^{least}) = \\ &\frac{\rho_n}{\sqrt{G}}(\frac{1}{\sqrt{(k-l-1)x^{coop} + x_k^* + (N-k)D + x}} \\ &+ \frac{1}{\sqrt{G}}) + Ex + C + p. \end{aligned} \tag{46}$$

Set $F_n(\boldsymbol{x}_n^{least})' = 0$, we can get

$$\begin{aligned} x &= \sqrt[3]{\frac{\rho_n^2}{4GE^2}} - (k-l-1)x^{coop} - (N-k)D - x_k^* \\ &\leq \sqrt[3]{\frac{\rho_n^2}{4GE^2}} - \sqrt[3]{\frac{\rho_k^2}{4GE^2}} - (k-l-1)x^{coop} \\ &< 0. \end{aligned} \tag{47}$$

Similarly, for client $l+1, \cdots, k-1$, the point at which the cost function minimizes is 0. So we can calculate the threshold discount factor $\delta_l^{th}(\boldsymbol{x}^{coop})$ of client l:

$$\begin{aligned} \delta_l^{th}(\boldsymbol{x}^{coop}) &= \frac{F_l(\boldsymbol{x}^{coop}) - F_l(\boldsymbol{x}_n^{least})}{F_l(\boldsymbol{x}^{pun}) - F_l(\boldsymbol{x}_n^{least})} \\ &= \frac{\frac{1}{\sqrt{(k-l)x^{coop}+x_k^*+(N-k)D}} - \frac{1}{\sqrt{(k-l-1)x^{coop}+x_k^*+(N-k)D}}}{\frac{1}{\sqrt{x_k^*+(N-k)D}} - \frac{1}{\sqrt{(k-l-1)x^{coop}+x_k^*+(N-k)D}}} \\ &+ \frac{Ex^{coop}\sqrt{G}}{\rho_l(\frac{1}{\sqrt{x_k^*+(N-k)D}} - \frac{1}{\sqrt{(k-l-1)x^{coop}+x_k^*+(N-k)D}})}. \end{aligned} \tag{48}$$

It's easy to see that the threshold discount factor of $\delta_l^{th}(\boldsymbol{x}^{coop})$ decreases with the increase of ρ_l, so we can know that the threshold discount factor of client l is larger than the threshold discount factor of other converted contributors, i.e., $\delta_l^{th}(\boldsymbol{x}^{coop}) > \delta_{l+1}^{th}(\boldsymbol{x}^{coop}) > \cdots > \delta_{k-1}^{th}(\boldsymbol{x}^{coop})$.

For client k, similar to client l, we can get the point at which the cost function of client k minimizes, which we denote as x_k^{least}:

$$x_k^{least} = \begin{cases} 0, & \text{if } x_k^* - (k-l)x^{coop} \leq 0; \\ x_k^* - (k-l)x^{coop}, & \text{if } 0 < x_k^* - (k-l)x^{coop} < D. \end{cases} \tag{49}$$

The cost function of client k under this strategy profile is:

$$\begin{aligned} F_k(\boldsymbol{x}_k^{least}) = &\frac{\rho_k}{\sqrt{G}}(\frac{1}{(k-l)x^{coop} + x_k^{least} + (N-k)D} \\ &+ \frac{1}{\sqrt{G}}) + Ex_k^{least} + C + p. \end{aligned} \tag{50}$$

Then we calculate the threshold discount factor of client k:

$$\delta_k^{th}(\boldsymbol{x}^{coop}) = \frac{F_k(\boldsymbol{x}^{coop}) - F_k(\boldsymbol{x}_k^{least})}{F_k(\boldsymbol{x}^{pun}) - F_k(\boldsymbol{x}_k^{least})}. \tag{51}$$

For client $k+1$, we can get the point at which the cost function of client $k+1$ minimizes, which we denote as x_{k+1}^{least}, here we denote $\sqrt[3]{\frac{\rho_{k+1}^2}{4GE^2}} - (k-l)x^{coop} - x_k^* - (N-k-1)D$ as x_{k+1}^{min} for convenience,

$$x_{k+1}^{least} = \begin{cases} 0, & \text{if } x_{k+1}^{min} \leq 0; \\ x_{k+1}^{min}, & \text{if } 0 < x_{k+1}^{min} < D; \\ D, & \text{if } x_{k+1}^{min} \geq D. \end{cases} \tag{52}$$

The cost function of client $k+1$ under this strategy profile is:

$$\begin{aligned} F_{k+1}(\boldsymbol{x}^{least}) = &\frac{\rho_{k+1}}{\sqrt{G}}(\frac{1}{(k-l)x^{coop} + x_k^* + (N-k-1)D + x_{k+1}^{least}} \\ &+ \frac{1}{\sqrt{G}}) + Ex_{k+1}^{least} + C + p. \end{aligned} \tag{53}$$

Then we calculate the threshold discount factor of client $k+1$:

$$\delta_{k+1}^{th}(\boldsymbol{x}^{coop}) = \frac{F_{k+1}(\boldsymbol{x}^{coop}) - F_{k+1}(\boldsymbol{x}_{k+1}^{least})}{F_{k+1}(\boldsymbol{x}^{pun}) - F_{k+1}(\boldsymbol{x}_{k+1}^{least})}. \tag{54}$$

Similarly, for client $k+2, \cdots, N$, we can get their threshold discount factor $\delta_{k+2}^{th}(\boldsymbol{x}^{coop}), \cdots, \delta_N^{th}(\boldsymbol{x}^{coop})$ respectively.

When the critical client k does not exist, the process is similarly to the above, and we will not repeat it.