

Dynamic Pricing for Client Recruitment in Federated Learning

Xuehe Wang, *Member, IEEE*, Shensheng Zheng, and Lingjie Duan, *Senior Member, IEEE*

Abstract—Though federated learning (FL) well preserves clients' data privacy, many clients are still reluctant to join FL given the communication cost and energy consumption in their mobile devices. It is important to design pricing compensations to motivate enough clients to join FL and distributively train the global model. Prior pricing mechanisms for FL are static and cannot adapt to clients' random arrival pattern over time. We propose a new dynamic pricing solution in closed-form by constructing the Hamiltonian function to optimally balance the client recruitment time and the model training time, without knowing clients' actual arrivals or training costs. During the client recruitment phase, we offer time-dependent monetary rewards per client arrival to trade off between the total payment and the FL model's accuracy loss. Such reward gradually increases when we approach to the recruitment deadline or have greater data aging, and we also extend the deadline if the clients' training time per iteration becomes shorter. Further, we extend to consider heterogeneous client types in training data size and training time per iteration. We successfully extend our dynamic pricing solution and develop an optimal algorithm of linear complexity to monotonically select client types for FL. Finally, we also show robustness of our solution against estimation error of clients' data sizes, and run numerical experiments to validate our results.

Index Terms—federated learning, dynamic pricing, incentive mechanism, incomplete information



1 INTRODUCTION

The Internet of Things (IoT) is fast developing and well connects many human clients when using their mobile devices. It is desirable to learn from the massive data generated on such mobile devices to train the client models (e.g., for personal advertisement or recommendation) [1]. To preserve clients' data privacy, federated learning (FL) is proposed to invite clients to iteratively update their computing results using local data, without sharing such data to the central server [2]. In an FL system (e.g., Gboard [3]), a learning task is implemented in two phases: client recruitment and model training. In the client recruitment phase, the central server waits for sequential arrivals of mobile clients to connect and participate; in the model training phase, clients obtain the central server's aggregation parameter feedback in last iteration and compute their local results for this iteration to update. In addition, in the model training phase, the central server well utilizes clients' computing power and local data to train the desired model [4].

Previous works mainly focus on the technological issues of FL. For example, to improve communication efficiency for FL, there are local updating [5], [6], [7], communication complexity reduction [8], [9] and compression schemes [10], [11]. To enhance the overall security and pri-

vacuity of FL systems, Fung et al. [12] propose FoolsGold that identifies poisoning sybils based on the diversity of client contributions in the distributed learning process. Liu et al. [13] further introduce a blockchain-based secure FL framework to prevent malicious or unreliable participants in FL. Hao et al. [14] propose an efficient and privacy-preserving federated deep learning protocol based on stochastic gradient descent method, by integrating the additively homomorphic encryption with differential privacy.

Most of their works assume that the clients will voluntarily participate in FL, which may not be realistic due to clients' training cost including computational energy consumption on model training and parameter transmission. In reality, many clients are reluctant to participate in model training if there is not enough compensation during the training process. Thus, to increase clients engagement and ensure training accuracy, it is important to design pricing compensations to motivate enough clients to join FL and distributively train the global model. There are some recent works (e.g., [15], [16]) discussing the incentive mechanism design in FL, by largely assuming that data quality and costs of clients are readily known. Under this complete information assumption, Zhan et al. [17] design an incentive mechanism to optimize the utilities of mobile clients and the accuracy of the training model by considering the different sensing and training capabilities of mobile clients. Some other researchers (e.g., Feng et al. [18] and Sarikaya et al. [19]) formulate a Stackelberg game to design pricing incentive against the clients' following responses. Reinforcement learning is also used to derive the optimal pricing strategy for the central server to recruit clients for training [20], [21], [22].

However, in many realistic FL applications such as the financial field (e.g., [23], [24]), client recruitment is by no means in one shot but lasts for a period of time, waiting

- X. Wang and S. Zheng are with the School of Artificial Intelligence, Sun Yat-sen University, Zhuhai 519082, China (E-mail: wangxuehe@mail.sysu.edu.cn, zhengshsh7@mail2.sysu.edu.cn). X. Wang is also with the Guangdong Key Laboratory of Big Data Analysis and Processing, Guangzhou 510006, China. L. Duan is with the Pillar of Engineering Systems and Design, Singapore University of Technology and Design, Singapore (E-mail: lingjie_duan@sutd.edu.sg).
- This work was supported by the National Natural Science Foundation of China (Grant No. 62206320) and the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 Grant (Award No. MOE-T2EP20121-0001).

for clients' arrivals and joining dynamically. Human clients have their own timing to observe the new FL task opportunity and some even need to complete ongoing tasks first (e.g., [3]). It is natural to consider our two-phase FL model including the recruitment and training phase. Prior works (e.g., [17], [18], [19], [20], [21]) simply decide fixed or one-shot pricing for client recruitment, by assuming all potential clients are always waiting for joining a new FL task. In our dynamic problem, one-shot or static pricing can easily lead to data over-sampling or under-sampling for training, and it is important to adaptively adjust the pricing compensation based on clients' actual arrival pattern and cost distribution to meet the data target for later model training. Moreover, the clients' training costs or even their arrival pattern are unknown to the FL system (e.g., [24]), as clients in general have different data sizes or distributions to contribute as well as different model training time per iteration. This incomplete information feature is largely overlooked in the existing incentive mechanisms for FL (e.g., [18], [19], [20], [21]). Only some very recent works (e.g., [25], [26]) consider the information uncertainty about clients' private training costs, they ignore the recruitment process and stick with the special case of static pricing strategy as an incentive mechanism. Therefore, their performances can be far from the optimum of our dynamic pricing as shown in Section 3.2 and Section 6 of this paper.

We are the first to design a dynamic pricing strategy to incentivize heterogeneous clients to participate in FL, under incomplete information about clients' private training costs and random arrivals. Moreover, for a finite time horizon, we need to balance the client recruitment time/process and the model training time/process per task, while the incentive works (e.g., [17], [18], [19], [25], [26]) above mainly focus on the first phase of client recruitment. Finally, when facing multiple types of clients with different data size and training time per iteration, the client-type selection should be further taken into consideration to trade off the total data size and global training iterations. Our key novelty and main contributions are summarized as follows:

- *Dynamic client recruitment in federated learning:* To our best knowledge, this paper is one of the first works studying how to motivate dynamically arriving clients to participate in the federated learning over time, without knowing their private training costs or even arrival pattern. Due to the non-trivial waiting time for enough clients, we jointly consider two phases to balance for each FL task: client recruitment and the model training by the involved clients, where the longer time for recruitment helps gain more training data yet leaves less time for FL convergence.
- *Dynamic pricing mechanism under incomplete information:* In the client recruitment phase, we decide time-dependent monetary rewards in closed-form by constructing the Hamiltonian function to balance the total payment to clients and model accuracy loss, where a higher pricing reward attracts more clients for data contribution to FL yet adds expense to the system. We prove that the central server should provide a higher price when approaching to the recruit-

ment deadline or with greater data aging, and show our dynamic pricing strategy always outperforms the static pricing strategy.

- *Threshold-based client recruitment policy to cope with the dynamic pricing scheme:* Though a longer client recruitment duration helps recruit more clients and enlarge the training dataset, it leaves less model training time. After deciding the dynamic pricing for any recruitment time, we systematically analyze the best partition in client recruitment time and model training time, which is proved to be threshold-based. As compared to model training, we relatively reserve less time for client recruitment given greater data aging or longer training time of each client.
- *Pricing extension to heterogeneous clients with robustness check:* We extend the pricing solution to consider heterogeneous client types in training data size and training time per iteration. Though a client type may have a greater data size, it also incurs greater computing time to accommodate in the synchronous training. We successfully extend our dynamic pricing solution and develop an optimal algorithm of linear complexity (with respect to the number of client types) to monotonically select client types for FL. Note that given a selected client type, the iteration duration is fixed and including client types with smaller data size and training time only creates updates within the iteration duration without reducing the number of global training iterations. Thus we prove that the optimal client-types should be selected monotonically to accelerate the model training. We also show robustness of our solution even if we have some errors in estimating clients' data size. We also run experiments to validate our conclusion.

The rest of this paper is organized as follows. The system model and problem formulation are given in Section 2. In Sections 3 and 4, we analyze the optimal dynamic pricing and data recruitment threshold for homogeneous clients. The extension to heterogeneous clients case and the robustness checking are discussed in Section 5. Experimental results are shown in Section 6. Section 7 concludes this paper.

2 SYSTEM MODEL AND PROBLEM FORMULATION

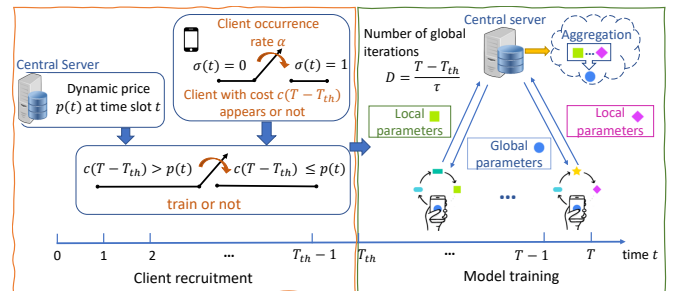


Fig. 1: Illustration of the two-phase model including the client recruitment phase and model training phase in federated learning.

We consider an FL platform that plans to complete a task in T time slots over the discrete time horizon. It first recruits dynamically arriving clients in the first T_{th} time slots

and then proceeds with model training at recruited clients' mobile devices using their local data in the rest $T - T_{th}$ slots, as shown in Fig. 1. Note that in client recruitment phase, the agreed clients just commit to help for the later model training, without incurring any cost. They only incur costs in computational energy consumption and parameter transmission during the model training phase. We first discuss homogeneous clients with identical data size and training time. The extension to heterogeneous clients with different data size and training time will be presented in Section 5. For ease of reading, we list the key notations in Table 1.

TABLE 1: Key notations and their physical meanings.

T	Total time horizon
T_{th}	Time threshold for client recruitment phase
D	Number of global iterations
N	Number of heterogeneous client types
c	Client's unit cost per training time $\in [0, b]$
α	Arrival rate of the clients in each time slot
r	Discount factor to indicate the freshness of data
s	Data size of the homogeneous clients
s_i	Data size of the i -th client type
τ	Training time per iteration for homogeneous clients
τ_i	Training time per iteration of the i -th client type
$p(t)$	Recruitment price of the homogeneous clients at time slot $t \in \{0, \dots, T_{th} - 1\}$
$p_i(t)$	Recruitment price of the i -th client type at time slot $t \in \{0, \dots, T_{th} - 1\}$
q_i	Percentage of type- i clients with $\sum_{i=1}^N q_i = 1$
$B(t)$	Resulting total data size at time slot $t \in \{0, \dots, T_{th}\}$
$U(T)$	Total expected cost of the homogeneous clients
$J(\mathbf{P}(t) T_{th}, \{1, \dots, j\})$	Total expected cost given client-types $\{1, \dots, j\}$ and recruitment threshold T_{th}

2.1 Client Recruitment Phase

For the client recruitment phase, the central server recruits data from clients for the FL task in any time slot $t = 0, \dots, T_{th} - 1$. As shown in Fig 1, at the beginning of each time slot $t \in \{0, \dots, T_{th} - 1\}$, the central server announces price $p(t)$ for training $T - T_{th}$ time slots. Then a client may appear randomly in this time slot and (if so) he further decides to help train the model or not by comparing the price offer $p(t)$ and his own total cost $c(T - T_{th})$, where c is the unit cost per training time [27]. Considering the clients appear according to a Poisson process, with an average number ν of client arrivals per unit time. Then, the probability of the random number of client arrivals $N(t)$ in the t -th time slot of the interval $[(t-1)\Delta, t\Delta)$ being equal to k is

$$Pr(N(t) = k) = \frac{e^{-\nu\Delta}(\nu\Delta)^k}{k!}, \quad (1)$$

with time duration Δ for each time slot. Note that $Pr(N(t) > 1)$ becomes trivial as long as the time duration Δ is small. Thus, each time slot's duration Δ is properly selected to be short such that it is almost sure to have at most one client arrival at a time. Then, whether a client appears at time slot t is represented as $\sigma(t)$:

$$\sigma(t) = \begin{cases} 1, & \text{if a client arrives in time slot } t, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

with the client arrival rate in each time slot $\alpha = \nu\Delta$.

To motivate clients' participation, we design pricing compensations by considering clients' private costs including computational energy consumption on model training and parameter transmission. We consider that the clients' private costs are i.i.d. according to a general cumulative distribution function (CDF) $F(c)$, $c \in [0, b]$, where the upper bound $b \geq 1$ is estimated from historical data. Though all potential users' costs follow the same random distribution as estimated from the group/mass data, their realized costs are different in general and unknown to the server. If a client with data size s and training time τ per iteration appears and accepts the price $p(t)$ at time slot t , i.e., $\sigma(t) = 1$ and $c(T - T_{th}) \leq p(t)$, he will help train the model using his local training dataset and return the updated local parameter within the training time τ , where τ represents the clients' training time (total computation and transmission time) per global iteration. The payoff of the participating client with private unit cost c is the difference between the price and his own training cost, i.e., $p(t) - c(T - T_{th})$. Then, the arrival client's payoff at time t is concluded as follows:

$$\Upsilon(t) = \begin{cases} p(t) - c(T - T_{th}), & \sigma(t) = 1 \ \& \ c \leq \frac{p(t)}{T - T_{th}}, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

The training efficiency is defined as the ratio between the data size and training time per client. Note that the small training efficiency for a client indicates that the training time is very large even for small data size. In this case, the platform should always set a trivial price $p(t)$ to be the upper bound $b(T - T_{th})$ to recruit arriving clients as soon as possible to save time for model training. With the strong computing capabilities nowadays, the training efficiency is generally not that small. Therefore, in this paper, we consider a more general case with non-trivial training efficiency (data size/training time) $\frac{s}{\tau} \geq 1$ to design dynamic pricing for client recruitment.

2.2 Model Training Phase

For the distributed model training phase, we consider the synchronous FL with the one-step local update, which means each client performs one step of mini-batch stochastic gradient descent (SGD) to update the model parameters in each round, and the server waits for all clients' local parameter updates and then sends the updated global parameter to all clients at the same time for next round's training. Note that most synchronous FL algorithms require the same client number to participate in each round [28], [29]. Given M clients agreed to participate, each participating client $n \in \{1, \dots, M\}$ uses its local dataset \mathcal{D}_n with data size s_n to train the model. Denote the collection of data samples in \mathcal{D}_n as $\{x_k, y_k\}_{k=1}^{s_n}$, where $x_k \in \mathbb{R}^d$ is the input sample vector and $y_k \in \mathbb{R}$ is the labeled output value for the sample x_k at client n . For a sample data $\{x_k, y_k\}$, the objective is to find the model parameter $\omega \in \mathbb{R}^d$ that predicts the output y_k based on x_k with the loss function $f_k(\omega)$, where $f_k(\omega)$ characterizes the difference between the predicted value $\hat{y}_k(x_k, \omega)$ and real output y_k . The loss function on the data set \mathcal{D}_n of client n is

$$F_n(\omega) = \frac{1}{s_n} \sum_{k \in \mathcal{D}_n} f_k(\omega).$$

At each iteration $t + 1$, client n updates its local parameter based on last global parameter ω_t sent by the central server:

$$\omega_{t+1}^n = \omega_t - \eta \nabla F_n(\omega_t),$$

where η is the learning rate, and sends ω_{t+1}^n back to the central server. The server averages the parameters sent back by M participating clients

$$\omega_{t+1} = \sum_{n=1}^M \frac{\varsigma_n}{\varsigma} \omega_{t+1}^n,$$

where $\varsigma = \sum_{n=1}^M \varsigma_n$ is the total data size, and sends the updated global parameter ω_{t+1} to all clients for next round's training.

The optimal model parameter ω^* that minimizes the global loss function is:

$$\omega^* = \arg \min_{\omega} f(\omega) = \arg \min_{\omega} \sum_{n=1}^M \frac{\varsigma_n}{\varsigma} F_n(\omega).$$

Note that there are $T - T_{th}$ time slots left for training and the number of training iterations is $D = \frac{T - T_{th}}{\tau}$. As in [30], [31], the accuracy loss after D global iterations is measured by the difference between the global loss with the predicted parameter ω_D and that with the optimal parameter ω^* , i.e., $f(\omega_D) - f(\omega^*)$, where $f(\omega)$ is the general convex objective function. When clients use one step of mini-batch SGD in each round and their datasets distribution is IID, the expected model accuracy loss is $O(\frac{1}{\sqrt{B(T_{th})D}} + \frac{1}{D})$ [30], [31], where $B(T_{th})$ is the total data size contributed by the participating clients at the end of the client recruitment phase. For finite time horizon T , the objective $U(T)$ of the central server is to find the optimal dynamic pricing $p(t), t \in \{0, \dots, T_{th} - 1\}$ and recruitment threshold T_{th} to minimize the total expected cost, which is the summation of the total expected payment to clients and the expected model accuracy loss:

$$U(T) = \min_{\substack{p(t), t \in \{0, \dots, T_{th} - 1\} \\ T_{th} \in \{1, \dots, T - 1\}}} \sum_{t=0}^{T_{th}-1} \zeta(p(t)) + \frac{1}{\sqrt{B(T_{th})D}} + \frac{1}{D}, \quad (4)$$

where $\zeta(p(t))$ is the expected payment at time t .

2.3 Problem Formulation

Based on the discussions in Sections 2.1 and 2.2, we use dynamic Bayesian Game to model our federated learning system's interaction with dynamically arriving clients without knowing their arrivals and private training costs.

- **Players:** the central server and random arriving clients.
- **Strategies:** For the central server, it decides the dynamic pricing $p(t), t \in \{0, \dots, T_{th} - 1\}$, recruitment threshold T_{th} (for heterogeneous clients case, it further decides the client-type choice). For the clients, they decide to help train the model or not.
- **Information set:** The central server does not know clients' arrivals and private costs, but knows the client arrival rate α in each time slot and the cost distribution $F(c)$.

- **Expected payoffs:** The client's payoff $\Upsilon(t)$ in (3) depends on his private training cost c and price $p(t)$. The central server's total cost $U(T)$ in (4) is the summation of the total payment to clients and model accuracy loss.

Note that a longer recruitment duration may help recruit more clients, but leaves less training time under finite time horizon T . Thus, the central server should balance the client recruitment time and model training time. Moreover, in the client recruitment phase, static pricing cannot adapt to clients' random arrival patterns and data aging over time. Hence, we need to use dynamic pricing to balance the total payment to clients and the accuracy loss under incomplete client information. By considering the above issues, we formulate the problem by two-stage:

- **Stage I:** The central server chooses the optimal client recruitment threshold T_{th} .
- **Stage II:** Given the optimal recruitment threshold T_{th} , the central server decides the optimal dynamic pricing $p(t), t \in \{0, \dots, T_{th} - 1\}$.

In the following, we use backward induction to first analyze the dynamic pricing in Stage II given the time threshold, then the optimal recruitment threshold in Stage I.

3 OPTIMAL DYNAMIC PRICING UNDER INCOMPLETE INFORMATION IN STAGE II

In this section, we study the central server's pricing strategy under incomplete information, i.e., the central server does not know the clients' arrivals during the client recruitment phase and the participating client's particular cost. We first discuss the design of the dynamic pricing strategy in Section 3.1. Then the discussion of the static strategy to serve as a benchmark is given in Section 3.2.

3.1 Analysis of Optimal Dynamic Pricing

In reality, when waiting for the central server's client recruitment, the value of the data may decrease, and the data collected earlier is not as fresh as the latest data contributed by recently arrived clients. For example, the real-time traffic information will become outdated and useless over time [32]. [33] shows the freshness of data is inversely proportional to the number of times used in the training process for vehicular networks. The FL server needs the fresh data continuously to include all the important data in the final model and improve the accuracy of the model over time domain [34]. Thus, the useful data size from each device reduces. We denote the data aging factor as r . Here we reasonably consider a non-trivial $0.5 < r < 1$ to model data aging, as a trivial r will lead to no recruitment till the deadline, and our dynamic pricing reduces to one-shot pricing.

For the homogeneous clients with identical data size s and training time τ , from the initial training data size $B(t = 0) = 0$, the probability that the data size $B(t + 1)$ at time $t + 1$ increases to $B(t) + s$ is $\alpha F(\frac{p(t)}{T - T_{th}})$, i.e., a client appears and accepts the price offer $p(t)$ at time t . The probability that the data size $B(t + 1)$ at time $t + 1$ remains $B(t)$ is $1 - \alpha F(\frac{p(t)}{T - T_{th}})$.

Consider uniform distribution of the clients' private costs¹, i.e., $F(c) = \frac{c}{b}$, $c \in [0, b]$, the dynamics of the expected data size $B(t)$ is given as:

$$\begin{aligned} B(t+1) &= r((B(t) + s)\alpha F(\frac{p(t)}{T - T_{th}})) \\ &\quad + B(t)(1 - \alpha F(\frac{p(t)}{T - T_{th}})) \quad (5) \\ &= r(B(t) + \frac{\alpha s}{b(T - T_{th})}p(t)). \end{aligned}$$

Note that r in Eq (5) indicates the data aging effect that the useful data size reduces over time. That is, we should not only consider the total data size, but also consider the freshness of the data at time slot t .

Since the probability that a client appears and accepts the price $p(t)$ at time t is $\frac{\alpha}{b(T - T_{th})}p(t)$, the expected payment to this client is $\frac{\alpha}{b(T - T_{th})}p^2(t)$. Note that the optimal price $p(t)$ should not exceed the maximum cost $b(T - T_{th})$ of the client as it is unnecessary for the provider to over-pay. Therefore, given any time threshold T_{th} , the objective function (4) of the central server can be rewritten as:

$$\begin{aligned} U(T) &= \min_{\substack{p(t) \leq b(T - T_{th}) \\ t \in \{0, \dots, T_{th} - 1\}}} \sum_{t=0}^{T_{th}-1} \frac{\alpha}{b(T - T_{th})} p^2(t) + \frac{1}{\sqrt{B(T_{th})D}} \\ &\quad + \frac{1}{D}, \quad (6) \end{aligned}$$

$$\text{s.t. } B(t+1) = r(B(t) + \frac{\alpha s}{b(T - T_{th})}p(t)) \quad (5)$$

We can see that a higher price in the client recruitment phase leads to smaller accuracy loss for FL, but cause higher payment to afford for the central server. It's not easy to solve the above problem by considering the huge number of price combinations over time, with computation complexity $O((b(T - T_{th})/\epsilon)^{T_{th}})$ increasing exponentially in T_{th} , where ϵ is the precision of searching for pricing in the range $[0, b]$. In the following proposition, we solve the dynamic pricing in closed-form by constructing the Hamiltonian function.

Proposition 3.1. The optimal closed-form dynamic pricing $p(t)$, $t \in \{0, \dots, T_{th} - 1\}$ is given by

$$p(t) = \left(\frac{b^3 \tau^3 D^2 r^{5T_{th} - 5t - 6} (1 - r^2)^3}{16 \alpha^3 s (1 - r^{2T_{th}})^3} \right)^{\frac{1}{5}}, \quad (7)$$

which is monotonically increasing in t , $p(t) \leq b(T - T_{th})$ holds for any $t \in \{0, \dots, T_{th} - 1\}$. The recruited clients' expected data size at the end of the client recruitment phase T_{th} is

$$B(T_{th}) = D^{-\frac{3}{5}} \left(\frac{\alpha s^2}{4b\tau} r^2 \sum_{i=1}^{T_{th}} r^{2(i-1)} \right)^{\frac{2}{5}}. \quad (8)$$

Proof: According to the problem (5)-(6), we have the discrete time Hamiltonian function as

$$\begin{aligned} H(t) &= \frac{\alpha}{b(T - T_{th})} p^2(t) + \lambda(t+1)((r-1)B(t) \\ &\quad + \frac{r\alpha s}{b(T - T_{th})} p(t)). \quad (9) \end{aligned}$$

1. Though more involved, our analysis and key results can be extended to some other distributions such as normal.

Since $\frac{\partial^2 H(t)}{\partial p^2(t)} > 0$, the Hamiltonian function $H(t)$ is convex in $p(t)$. Therefore, in order to find the optimal dynamic pricing that minimize the total expected cost $U(T)$ in (6), it is necessary to satisfy:

$$\frac{\partial H(t)}{\partial p(t)} = 0, \quad (10)$$

$$\lambda(t+1) - \lambda(t) = -\frac{\partial H(t)}{\partial B(t)}, \quad (11)$$

with boundary condition

$$\lambda(T_{th}) = \frac{\partial(\frac{1}{\sqrt{B(T_{th})D}} + \frac{1}{D})}{\partial B(T_{th})} = -\frac{1}{2} D^{-\frac{1}{2}} (B(T_{th}))^{-\frac{3}{2}}. \quad (12)$$

According to (11), we have $\lambda(t) = r\lambda(t+1)$. Then, based on the boundary condition, we can derive

$$\lambda(t) = -\frac{1}{2} r^{T_{th}-t} D^{-\frac{1}{2}} (B(T_{th}))^{-\frac{3}{2}}. \quad (13)$$

Based on (10) and (13),

$$p(t) = -\frac{rs}{2} \lambda(t+1) = \frac{s}{4} r^{T_{th}-t} D^{-\frac{1}{2}} (B(T_{th}))^{-\frac{3}{2}}. \quad (14)$$

Insert $p(t)$ in (14) into (5), for $t \in \{1, \dots, T_{th}\}$, we have

$$B(t) = \frac{\alpha s^2}{4b\tau} D^{-\frac{3}{2}} (B(T_{th}))^{-\frac{3}{2}} r^{T_{th}-t+2} \sum_{i=1}^t r^{2(i-1)}. \quad (15)$$

Thus, the total data size at the end of the client recruitment phase can be solved as in (8). According to (14) and (15), the optimal dynamic pricing $p(t)$ is obtained as in (7). Note that $\frac{b^3 \tau^3 D^2}{16 \alpha^3 s (\sum_{i=1}^{T_{th}} r^{2(i-1)})^3}$ will not change with t for any given T_{th} . When $r < 1$, $r^{5T_{th}-5t-6}$ increases with t , and thus $p(t)$ increases with t .

In the following, we will show that $p(t) \leq b(T - T_{th})$ is satisfied when $t \leq T_{th} - 2$. According to (7), $p(t) \leq b(T - T_{th})$ is equivalent to

$$16b^2 \alpha^3 (T - T_{th})^3 \frac{s}{\tau} \frac{(\sum_{i=1}^{T_{th}} r^{2(i-1)})^3}{r^{5T_{th}-5t-6}} \geq 1. \quad (16)$$

For $r < 1$, $\frac{(\sum_{i=1}^{T_{th}} r^{2(i-1)})^3}{r^{5T_{th}-5t-6}} > \frac{1}{r^{5T_{th}-5t-6}} > 1$ always holds when $t \leq T_{th} - 2$. Note that $b \geq 1$, $\frac{s}{\tau} \geq 1$, and $\alpha \geq 0.5$. Thus, (16) always holds when $t \leq T_{th} - 2$.

When $t = T_{th} - 1$, $p(T_{th} - 1) \leq b(T - T_{th})$ holds if

$$(1 - r^2)^3 \leq 2r(1 - r^{2T_{th}})^3, \quad (17)$$

which always holds if

$$(1 - r^2)^3 \leq 2r(1 - r^2)^3, \quad (18)$$

i.e., $r \geq 0.5$. Therefore, we can conclude that $p(t) \leq b(T - T_{th})$ is always satisfied when $t \leq T_{th} - 1$. ■

Proposition 3.1 shows that when time slot t approaches the recruitment deadline T_{th} or the data aging factor r is large, it is necessary to increase the price to ensure recruiting enough data to train the model.

3.2 Analysis of Optimal Static Pricing

Note that the static pricing strategy is to use a fixed price to recruit clients at all times. Then the objective function of the central server can be rewritten as:

$$U_{st}(T) = \min_{\substack{p \leq b(T-T_{th}) \\ t \in \{0, \dots, T_{th}-1\}}} \sum_{t=0}^{T_{th}-1} \frac{\alpha}{b(T-T_{th})} p^2 + \frac{1}{\sqrt{B(T_{th})D}} + \frac{1}{D}. \quad (19)$$

$$\text{s.t.} \quad B_{st}(t+1) = r(B_{st}(t) + \frac{\alpha s}{b(T-T_{th})} p), \quad (20)$$

where p is the static pricing given by the central server.

Similar to the dynamic price, we also solve the optimal static pricing in closed form.

Lemma 3.1. The optimal closed-form static pricing p^* is given by

$$p^* = \min\left\{\left(\frac{D^2 b^3 \tau^3 (1-r)}{16 T_{th}^2 \alpha^3 s r (1-r^{T_{th}})}\right)^{\frac{1}{5}}, b(T-T_{th})\right\}, \quad (21)$$

which is a special case of (7). The expected data size at the end of the client recruitment phase T_{th} is

$$B_{st}(T_{th}) = \frac{r(1-r^{T_{th}})\alpha s p^*}{(1-r)b\tau D}. \quad (22)$$

Proof: According to the equation (20), we can derive the dynamics of the expected data size $B_{st}(t)$:

$$B_{st}(t) = \frac{r(1-r^t)\alpha s p}{(1-r)b(T-T_{th})} = \frac{r(1-r^t)\alpha s p}{(1-r)b\tau D}. \quad (23)$$

Based on Eq (23), (22) is satisfied. Thus, we can get the objective function of the central server as:

$$U_{st}(T_{th}) = \frac{T_{th}}{b\tau D} p^2 + \sqrt{\frac{b\tau(1-r)}{\alpha s p r (r-r^{T_{th}})}} p^{-\frac{1}{2}} + \frac{1}{D}. \quad (24)$$

Since $\frac{\partial^2 H(t)}{\partial p^2(t)} = \frac{T_{th}}{b\tau D} > 0$, the objective function $U_{st}(T_{th})$ is convex in $p(t)$.

Therefore, in order to find the optimal static pricing that minimizes the total expected cost $U(T)$ in (24), it is necessary to satisfy:

$$\frac{\partial U_{st}(T_{th})}{\partial p} = 0. \quad (25)$$

Thus, the optimal closed-form static pricing p^* can be derived as (21). ■

The performance comparison of static pricing and dynamic pricing is shown in Section 6.

4 OPTIMAL RECRUITMENT THRESHOLD IN STAGE I

Under the optimal dynamic pricing in Proposition 3.1, a longer client recruitment time T_{th} results in a larger total data size $B(T_{th})$ in (8) at the cost of less training iteration number D . Therefore, in Stage I the central server should find the optimal recruitment threshold T_{th} to balance the

total data size and training time for cost minimization in finite time horizon T , i.e.,

$$T_{th}^* = \arg \min_{T_{th} \in \{1, \dots, T-1\}} U(T), \quad (26)$$

where the total expected costs $U(T)$ under the optimal dynamic pricing $p(t)$ in (7) is:

$$U(T) = (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b\tau}{\alpha s^2 r^2}\right)^{\frac{1}{5}} \left(\frac{1-r^2}{1-r^{2T_{th}}}\right)^{\frac{1}{5}} \left(\frac{\tau}{T-T_{th}}\right)^{\frac{1}{5}} + \frac{\tau}{T-T_{th}}. \quad (27)$$

In the following proposition, the optimal threshold T_{th} is derived in closed-form.

Proposition 4.1. The optimal threshold $T_{th}^* \in \mathbb{Z}^+$ depends on the training time τ per iteration and is given as follows:

- Given high training time per iteration ($\tau \geq \bar{\psi}^{\frac{5}{3}}$), the server decides $T_{th}^* = 1$ by recruiting clients in one time slot only and save more time for model training.
- Given low training time per iteration ($0 < \tau < \bar{\psi}^{\frac{5}{3}}$), the server decides $T_{th}^* = \arg \min_{T_{th}} (U(T)|_{T_{th}=\lfloor \bar{T}_{th} \rfloor}, U(T)|_{T_{th}=\lfloor \bar{T}_{th} \rfloor + 1}) \in \{1, \dots, T-1\}$, with \bar{T}_{th} as the unique solution to

$$\begin{aligned} & \frac{1}{5} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b\tau^2(1-r^2)}{\alpha s^2 r^2 (1-r^{2T_{th}})(T-T_{th})}\right)^{\frac{1}{5}} \\ & \times \left(\frac{2r^{2T_{th}} \ln(r)}{1-r^{2T_{th}}} + \frac{1}{T-T_{th}}\right) + \frac{\tau}{(T-T_{th})^2} = 0, \end{aligned} \quad (28)$$

where

$$\begin{aligned} \bar{\psi} &= \frac{1}{5} \left(\frac{b}{\alpha s^2 r^2}\right)^{\frac{1}{5}} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \times \\ & \left(2|\ln(r)| \frac{r^2}{1-r^2} (T-1)^{\frac{9}{5}} - (T-1)^{\frac{4}{5}}\right). \end{aligned} \quad (29)$$

Proof: Take the first-order derivative of (27) with respect to T_{th} , we have

$$\begin{aligned} \frac{\partial U(T)}{\partial T_{th}} &= \frac{1}{5} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b\tau^2(1-r^2)}{\alpha s^2 r^2 (1-r^{2T_{th}})(T-T_{th})}\right)^{\frac{1}{5}} \\ & \times \left(\frac{2r^{2T_{th}} \ln(r)}{1-r^{2T_{th}}} + \frac{1}{T-T_{th}}\right) + \frac{\tau}{(T-T_{th})^2}. \end{aligned} \quad (30)$$

Since $\frac{\partial^2 U(T)}{\partial T_{th}^2} > 0$ for any $0.5 \leq r < 1$, T_{th}^* can be obtained according to $\frac{\partial U(T)}{\partial T_{th}} = 0$. Note that $\frac{\partial U(T)}{\partial T_{th}}$ increases in T_{th} . Therefore, we consider the following three cases:

(i) if $\frac{\partial U(T)}{\partial T_{th}}|_{T_{th}=1} \geq 0$, i.e.,

$$\begin{aligned} \tau^{\frac{3}{5}} &\geq \frac{2}{5} \left(\frac{b}{\alpha s^2 r^2}\right)^{\frac{1}{5}} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) |\ln(r)| \frac{r^2}{1-r^2} (T-1)^{\frac{9}{5}} \\ & - \frac{1}{5} \left(\frac{b}{\alpha s^2 r^2}\right)^{\frac{1}{5}} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) (T-1)^{\frac{4}{5}} := \bar{\psi}, \end{aligned} \quad (31)$$

$\frac{\partial U(T)}{\partial T_{th}} \geq 0$ always holds for $T_{th} \in [1, T-1]$, which means $T_{th}^* = 1$.

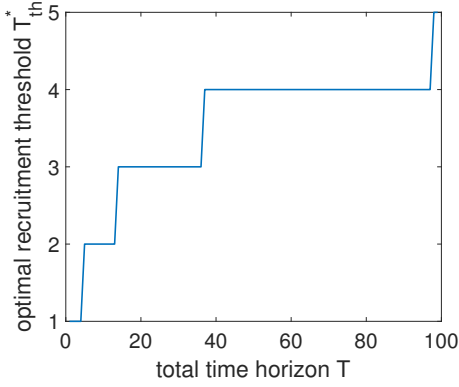


Fig. 2: Optimal recruitment threshold T_{th}^* versus total time horizon T when arrival rate $\alpha = 0.5$, data size $s = 2$, upper bound of private cost $b = 2$, discount factor $r = 0.6$ and training time per iteration $\tau = 0.16$.

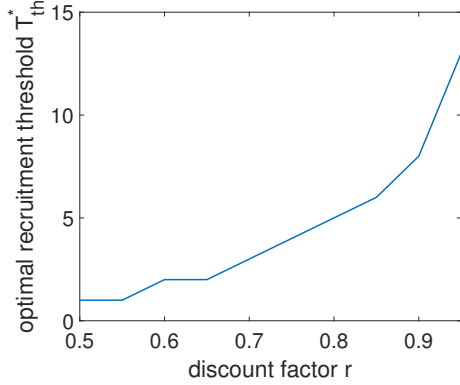


Fig. 3: Optimal recruitment threshold T_{th}^* versus discount factor r when total time horizon $T = 50$, arrival rate $\alpha = 0.5$, data size $s = 2$, upper bound of private cost $b = 2$ and training time per iteration $\tau = 0.16$.

(ii) If $\frac{\partial U(T)}{\partial T_{th}}|_{T_{th}=T-1} \leq 0$, i.e.,

$$\tau^{\frac{3}{5}} \leq \frac{2}{5} \left(\frac{b}{\alpha s^2 r^2} \right)^{\frac{1}{5}} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) |\ln(r)| \frac{r^{2(T-1)}(1-r^2)^{\frac{1}{5}}}{(1-r^{2(T-1)})^{\frac{6}{5}}} - \frac{1}{5} \left(\frac{b}{\alpha s^2 r^2} \right)^{\frac{1}{5}} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{1-r^2}{1-r^{2(T-1)}} \right)^{\frac{1}{5}} := \underline{\psi}, \quad (32)$$

$\frac{\partial U(T)}{\partial T_{th}} \leq 0$ always holds for $T_{th} \in [1, T-1]$, which means $T_{th}^* = T-1$.

However, in the following, we can show that $\underline{\psi}$ is always negative for $T \geq 2$ and $r < 1$. Thus, $\tau^{\frac{3}{5}} \leq \underline{\psi}$ doesn't exist.

According to (32), $\underline{\psi} \leq 0$ is equivalent to

$$(1 + 2|\ln(r)|)r^{2(T-1)} \leq 1. \quad (33)$$

Note that $\frac{\partial((1+2|\ln(r)|)r^{2(T-1)})}{\partial r} > 0$ always holds for $T \geq 2$ and $r < 1$, and $\lim_{r \rightarrow 1} (1 + 2|\ln(r)|)r^{2(T-1)} = 1$. Thus, (33) always holds, i.e., $\underline{\psi} \leq 0$.

(iii) if $\frac{\partial U(T)}{\partial T_{th}}|_{T_{th}=1} < 0$ and $\frac{\partial U(T)}{\partial T_{th}}|_{T_{th}=T-1} > 0$, i.e., $0 < \tau^{\frac{3}{5}} < \bar{\psi}$, since $\frac{\partial U(T)}{\partial T_{th}}$ increases in T_{th} , $T_{th}^* \in [1, \dots, T-1]$ can be obtained as the unique solution to $\frac{\partial U(T)}{\partial T_{th}} = 0$. Note that here T_{th}^* can still be equal to 1 as the solution to $\frac{\partial U(T)}{\partial T_{th}} = 0$ can be within the range $(1, 2)$. Thus, the integer T_{th}^* will be decided by comparing $U(T)|_{T_{th}=1}$ and $U(T)|_{T_{th}=2}$. ■

As shown in Proposition 4.1, if training time is high ($\tau \geq \bar{\psi}^{\frac{5}{3}}$), the FL training needs sufficient time to converge

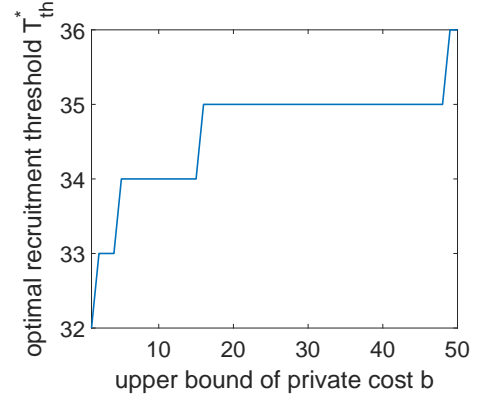


Fig. 4: Optimal recruitment threshold T_{th}^* versus upper bound of private cost b when total time horizon $T = 100$, arrival rate $\alpha = 0.5$, data size $s = 2$, discount factor $r = 0.6$ and training time per iteration $\tau = 0.16$.

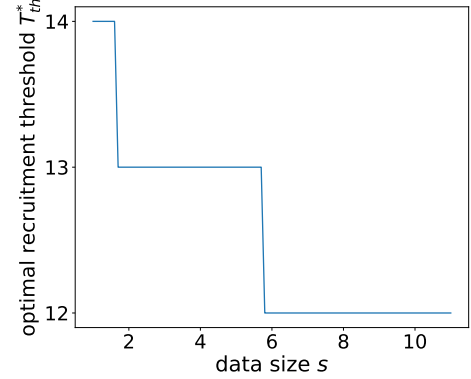


Fig. 5: Optimal recruitment threshold T_{th}^* versus data size s when total time horizon $T = 100$, arrival rate $\alpha = 0.5$, discount factor $r = 0.9$, upper bound of private cost $b = 5$ and training time per iteration $\tau = 0.9$.

and thus we decide the minimum recruitment time $T_{th}^* = 1$. As this requirement relaxes, we gradually increase the recruitment time T_{th}^* . Note that we obtain an expression for time partition T_{th}^* in closed-form, and do not face any issue for computational complexity.

To more clearly illustrate our results, Fig. 2 numerically shows that the optimal recruitment threshold T_{th}^* increases with the total time horizon T to relax the recruitment deadline. Moreover, Fig. 3 shows that the optimal recruitment threshold T_{th}^* increases with the discount factor r , which tells the data aging effect in (5). As the aging effect becomes weaker with greater r , the clients who arrived early in the recruitment phase still contribute a lot to the training dataset and we prolong the recruitment time T_{th}^* to accommodate more useful data. In addition, Fig. 4 shows that as the clients' costs for training increases, it takes more time for the server to recruit clients. Finally, Fig. 5 shows that as the client's data size s increases, the server can shorten the recruiting time and spend more time on model training to optimally balance the payment to clients and model accuracy.

5 EXTENSION TO HETEROGENEOUS CLIENTS

In previous sections, we have analyzed the optimal dynamic pricing and recruitment threshold for homogeneous clients with the identical data size and training time. In this section,

we will extend to N types of heterogeneous clients with different data sizes s_i and training time τ_i , $i \in \{1, \dots, N\}$. A client type may have multiple clients and will incur a longer iteration duration if provided with more data to compute [4]. Without loss of generality, we assume (s_i, τ_i) , $i \in \{1, \dots, N\}$ are sorted in ascending order, i.e., $s_1 < s_2 < \dots < s_N, \tau_1 < \tau_2 < \dots < \tau_N$. A client of type i with private unit cost c per training time will accept the price $p_i(t)$ at time slot t if the price offer can well compensate his cost with $c\tau_i D \leq p_i(t)$, $i = \{1, \dots, N\}$, where the training iteration number D now varies according to our selection of client types for FL and $\tau_i D$ is the total training time of type- i clients. In the synchronous FL setting, we set each iteration's duration to be the longest client computing time to wait for all the selected clients' updates. For example, if the central server recruits clients of types $i \in \{1, 2, 3\}$, the global iteration number is $D = \frac{T - T_{th}}{\tau_3}$, where the clients of types 1 and 2 wait for the type-3 clients' updates.

Given such latency in model training, not necessarily all types of clients will be invited to help train the global model. To include clients with larger amounts of data, the iteration duration time is also longer, and the left iteration number becomes smaller in the model training phase. Thus, it is necessary to select appropriate client types to balance the data size and global iteration number. However, the optimal client-type choosing is a combinatorial optimization problem that has very high computational complexity. The client-type choice will affect the following recruitment threshold as well as the dynamic price for each client type. The joint optimization of them is NP-hard. The time complexity to only check every combination of the client-type choice is $O(2^N)$ by increasing exponentially in N , not to mention the huge number of price combinations over time for each type- i clients, with computation complexity $O((b\tau_i D/\epsilon)^{T_{th}})$ increasing exponentially in T_{th} . By considering the above issues, we formulate the decision process of the central server as the following three stages:

- Stage I: The central server chooses the types of clients to recruit for model training.
- Stage II: Given the types of inviting clients, the central server decides the optimal recruitment threshold T_{th} .
- Stage III: Given the optimal recruitment threshold T_{th} and inviting types of clients, the central server decides the dynamic pricing $p_i(t)$, $t \in \{0, \dots, T_{th} - 1\}$ for each inviting type- i clients.

In the following, we use backward induction to analyze the above three-stage decision problem.

5.1 Optimal Dynamic Pricing in Stage III

In this section, given the inviting types of client in Stage I and recruitment threshold T_{th} in Stage II, we will analyze the optimal dynamic pricing $p_i(t)$, $t \in \{0, \dots, T_{th} - 1\}$ for each type- i clients. Without loss of generality, we denote the inviting types of clients as $\{1, \dots, \bar{N}\}$ with $\bar{N} \in \{1, \dots, N\}$, and later in Section 5.3, we prove the monotonic selection of client types. Denote the percentage of type- i clients as q_i with $\sum_{i=1}^{\bar{N}} q_i = 1$. For uniform distribution of the clients'

private costs, the probability that a type- i client appears and accepts the price offer $p_i(t)$ at time t is $\alpha q_i F(\frac{p_i(t)}{b\tau_i D}) = \frac{\alpha q_i}{b\tau_i D} p_i(t)$. By noting the data size s_i contributed by type- i client, the expected increase of data size at time t is $\sum_{i=1}^{\bar{N}} \frac{\alpha q_i s_i p_i(t)}{b\tau_i D}$. Starting from the initial training data size $B(t=0) = 0$, the dynamics of the expected data size $B(t)$ is given as (35).

Since the expected payment to type- i client at time t is $\frac{\alpha q_i}{b\tau_i D} p_i^2(t)$, the total expected payment at time t is $\sum_{i=1}^{\bar{N}} \frac{\alpha q_i}{b\tau_i D} p_i^2(t)$. Note that the optimal price $p_i(t)$ for each type- i clients should not exceed the maximum cost $b\tau_i D$ of this client-type as it is unnecessary for the provider to over-pay. Therefore, given the inviting types of clients $\{1, \dots, \bar{N}\}$ and recruitment threshold T_{th} , the central server aims to find the optimal dynamic pricing $p_i(t)$, $t \in \{0, \dots, T_{th} - 1\}$ for each type- i clients to minimize the total expected cost $J(\mathbf{P}(t) | T_{th}, \{1, \dots, \bar{N}\})$ consisting of the total payment to the clients and the expected accuracy loss for FL. That is,

$$\begin{aligned} & J(\mathbf{P}(t) | T_{th}, \{1, \dots, \bar{N}\}) \\ &= \min_{\substack{p_i(t) \leq b\tau_i D \\ t \in \{0, \dots, T_{th} - 1\}}} \sum_{t=0}^{T_{th}-1} \sum_{i=1}^{\bar{N}} \frac{\alpha q_i}{b\tau_i D} p_i^2(t) + \frac{1}{\sqrt{B(T_{th})D}} + \frac{1}{D} \\ &= \min_{\substack{p_i(t) \leq b\tau_i D \\ t \in \{0, \dots, T_{th} - 1\}}} \sum_{t=0}^{T_{th}-1} \left(\mathbf{P}(t)^\top \mathbf{W} \mathbf{P}(t) \right) + \frac{1}{\sqrt{B(T_{th})D}} + \frac{1}{D}, \end{aligned} \quad (34)$$

s.t.

$$B(t+1) = r \left(B(t) + \sum_{i=1}^{\bar{N}} \frac{\alpha q_i s_i p_i(t)}{b\tau_i D} \right) = rB(t) + \mathbf{Q} \mathbf{P}(t), \quad (35)$$

where

$$\begin{aligned} \mathbf{P}(t) &= [p_1(t), \dots, p_{\bar{N}}(t)]^\top \in \mathbb{R}^{\bar{N} \times 1} \\ \mathbf{Q} &= \left[\frac{r\alpha q_1 s_1}{b\tau_1 D}, \dots, \frac{r\alpha q_{\bar{N}} s_{\bar{N}}}{b\tau_{\bar{N}} D} \right] \in \mathbb{R}^{1 \times \bar{N}} \\ \mathbf{W} &= \begin{bmatrix} \frac{\alpha q_1}{b\tau_1 D} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{\alpha q_{\bar{N}}}{b\tau_{\bar{N}} D} \end{bmatrix} \in \mathbb{R}^{\bar{N} \times \bar{N}}. \end{aligned}$$

Similar to the analysis of Proposition 3.1, we have the following lemma.

Lemma 5.1. The optimal dynamic pricing $\mathbf{P}(t) \in \mathbb{R}^{\bar{N} \times 1}$ is

$$\mathbf{P}(t) = \min \left(\begin{bmatrix} s_1 \Gamma_t \\ \vdots \\ s_{\bar{N}} \Gamma_t \end{bmatrix}, \begin{bmatrix} b\tau_1 D \\ \vdots \\ b\tau_{\bar{N}} D \end{bmatrix} \right). \quad (36)$$

where $\Gamma_t = \left(\frac{b^3 D^2 r^{5T_{th} - 5t - 6} (1-r^2)^3}{16\alpha^3 (1-r^{2T_{th}})^3 (\sum_{i=1}^{\bar{N}} \frac{q_i s_i^2}{\tau_i})^3} \right)^{\frac{1}{5}}$, which is monotonically increasing in t .²

Proof: According to the problem (34)-(35), we can construct the Hamiltonian function as follows:

$$H(t) = \mathbf{P}(t)^\top \mathbf{W} \mathbf{P}(t) + \lambda(t+1) ((r-1)B(t) + \mathbf{Q} \mathbf{P}(t)). \quad (37)$$

2. If $s_j \Gamma_t > b\tau_j D$ for client type j from certain time slot t' , the optimal pricing is $p_j(t) = b\tau_j D$ for any $t \geq t'$.

Since $\frac{\partial^2 H(t)}{\partial p_i^2(t)} > 0, i \in \{1, \dots, \bar{N}\}$, the Hamiltonian function is convex in $p_i(t)$. Similar to the proof of Proposition 3.1, we have

$$\mathbf{P}(t) = \frac{1}{4} \mathbf{W}^{-1} \mathbf{Q}^\top D^{-\frac{1}{2}} (B(T_{th}))^{-\frac{3}{2}} r^{T_{th}-t-1}, \quad (38)$$

and

$$B(t) = \frac{1}{4} \mathbf{Q} \mathbf{W}^{-1} \mathbf{Q}^\top D^{-\frac{1}{2}} (B(T_{th}))^{-\frac{3}{2}} r^{T_{th}-t} \sum_{i=1}^t r^{2(i-1)}. \quad (39)$$

According to (39), $B(T_{th})$ is solved as:

$$B(T_{th}) = \left(\frac{1}{4} \mathbf{Q} \mathbf{W}^{-1} \mathbf{Q}^\top D^{-\frac{1}{2}} \sum_{i=1}^{T_{th}} r^{2(i-1)} \right)^{\frac{2}{5}}, \quad (40)$$

Insert (40) into (38) and note that $p_i(t) \leq b\tau_i D$ for any $i \in \{1, \dots, \bar{N}\}$, the optimal dynamic pricing $P(t)$ can be derived as (36). Since $r^{5T_{th}-5t-6}$ increases with t for any $0 < r < 1$, $P(t)$ is monotonically increasing with t .

Note that the Hamiltonian function is convex in $p_i(t), i \in \{1, \dots, \bar{N}\}$ for the problem (34)-(35). Thus, when $p_i(t) > b\tau_i D$, it's optimal to set $p_i(t) = b\tau_i D$. Moreover, by noting that $p_i(t), i \in \{1, \dots, \bar{N}\}$ increases with t , if $\Gamma_t s_j > b\tau_j D$ for client type j from certain time slot t' , the optimal pricing is $p_j(t) = b\tau_j D$ for any $t \geq t'$. ■

According to Lemma 5.1, if $s_i \Gamma_t \leq b\tau_i D$ holds for any client-type $i \in \{1, \dots, \bar{N}\}$, the resulting total expected cost is:

$$\begin{aligned} & J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\}) \\ &= (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b}{\alpha r^2} \right)^{\frac{1}{5}} \left(\frac{1-r^2}{1-r^{2T_{th}}} \right)^{\frac{1}{5}} \left(\sum_{i=1}^{\bar{N}} \frac{q_i s_i^2}{\tau_i} \right)^{-\frac{1}{5}} \left(\frac{\tau_{\bar{N}}}{T - T_{th}} \right)^{\frac{1}{5}} \\ & \quad + \frac{\tau_{\bar{N}}}{T - T_{th}}. \end{aligned} \quad (41)$$

Lemma 5.1 also shows that for the client type with large amount of data, higher dynamic prices are required to compensate for their higher training costs. Also, like homogeneous clients, the dynamic price increases over time due to data aging (that is, the value of data decreases over time).

5.2 Optimal Recruitment Threshold in Stage II

Based on the optimal dynamic pricing in Section 5.1, in Stage II we are ready to analyze the optimal recruitment threshold T_{th} given the types $\{1, \dots, \bar{N}\}$ of inviting clients in Stage I. We propose the following proposition to find the optimal threshold T_{th} for heterogeneous clients case.

Proposition 5.1. The optimal threshold $T_{th}^* \in \mathbb{Z}^+$ for heterogeneous clients depends on their training rate $\frac{s_i}{\tau_i}$ (data size/training time) and is given as follows:

- Given low training rate ($\frac{s_i}{\tau_i} \leq \frac{bD}{\Gamma_t}$ for any $i \in \{1, \dots, \bar{N}\}, t \in \{0, \dots, \tilde{T}_{th} - 1\}$), the server decides $T_{th}^* = \tilde{T}_{th}$.
- Given high training rate ($\frac{s_i}{\tau_i} > \frac{bD}{\Gamma_t}$ for certain $i \in \{1, \dots, \bar{N}\}, t \in \{0, \dots, \tilde{T}_{th} - 1\}$), the optimal recruitment threshold T_{th}^* can be obtained according to Algorithm 1.

where $\Gamma_t' = \left(\frac{b^3 D^2 r^{5\tilde{T}_{th}-5t-6} (1-r^2)^3}{16\alpha^3 (1-r^{2\tilde{T}_{th}})^3 (\sum_{i=1}^{\bar{N}} \frac{q_i s_i^2}{\tau_i})^3} \right)^{\frac{1}{5}}$ and $\tilde{T}_{th} = \arg \min_{T_{th}} (J_{\leq}(\mathbf{P}(t)|[\hat{T}_{th}], \{1, \dots, \bar{N}\}), J_{\leq}(\mathbf{P}(t)|[\hat{T}_{th}] + 1, \{1, \dots, \bar{N}\})) \in \{1, \dots, T-1\}$ with $\hat{T}_{th} \geq 1$ as the unique solution to

$$\begin{aligned} & \frac{1}{5} (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\sum_{i=1}^{\bar{N}} \frac{q_i s_i^2}{\tau_i} \right)^{-\frac{1}{5}} \left(\frac{b\tau_{\bar{N}}(1-r^2)}{\alpha r^2 (1-r^{2T_{th}})(T - T_{th})} \right)^{\frac{1}{5}} \\ & \times \left(\frac{2r^{2T_{th}} \ln(r)}{1-r^{2T_{th}}} + \frac{1}{T - T_{th}} \right) + \frac{\tau}{(T - T_{th})^2} = 0. \end{aligned} \quad (42)$$

Proof: Based on Lemma 5.1, we consider the following two cases based on whether $\Gamma_t s_i \leq b\tau_i D$:

(i) First, we consider the case that $\Gamma_t s_i \leq b\tau_i D$ is always satisfied for any $i \in \{1, \dots, \bar{N}\}, t \in \{0, \dots, T_{th} - 1\}$. According to $\mathbf{P}(t)$ in (36), we obtain the total expected cost $J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})$ in (41). We can check that $\frac{\partial^2 J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})}{\partial T_{th}^2} > 0$ is always satisfied, and thus the optimal T_{th}^* can be obtained based on the first-order conditions $\frac{\partial J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})}{\partial T_{th}} = 0$, which is given in (42).

Denote the solution to (42) as \hat{T}_{th} . Note that $\frac{\partial J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})}{\partial T_{th}} = 0$ increases in T_{th} . If $\frac{\partial J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})}{\partial T_{th}}|_{T_{th}=1} > 0$, we have $\hat{T}_{th} < 1$. Since $J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})$ is convex in T_{th} , the optimal threshold $T_{th}^* = \tilde{T}_{th} = \hat{T}_{th} = 1$. If $\frac{\partial J_{\leq}(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})}{\partial T_{th}}|_{T_{th}=1} \leq 0$, $\hat{T}_{th} \geq 1$ is the unique solution to (42). By noting that $T_{th} \in \mathbb{Z}^+$, the optimal threshold is $T_{th}^* = \tilde{T}_{th} = \arg \min_{T_{th}} (J_{\leq}(\mathbf{P}(t)|[\hat{T}_{th}], \{1, \dots, \bar{N}\}), J_{\leq}(\mathbf{P}(t)|[\hat{T}_{th}] + 1, \{1, \dots, \bar{N}\})) \in \{1, \dots, T-1\}$. By noting that Γ_t is a function of T_{th} , we define $\Gamma_t' = \left(\frac{b^3 D^2 r^{5T_{th}-5t-6} (1-r^2)^3}{16\alpha^3 (1-r^{2\tilde{T}_{th}})^3 (\sum_{i=1}^{\bar{N}} \frac{q_i s_i^2}{\tau_i})^3} \right)^{\frac{1}{5}}$ given \tilde{T}_{th} . Therefore, we can conclude that, if $\frac{s_i}{\tau_i} \leq \frac{bD}{\Gamma_t'}$ for any $i \in \{1, \dots, \bar{N}\}, t \in \{0, \dots, \tilde{T}_{th} - 1\}$, $p_i(t) = \Gamma_t' s_i \leq b\tau_i D$ is always satisfied and the optimal threshold $T_{th}^* = \tilde{T}_{th}$.

(ii) Then, we consider the case that $\Gamma_t' s_i > b\tau_i D$ for certain client type $i \in \{1, \dots, \bar{N}\}$ at time $t' \in \{0, \dots, \tilde{T}_{th} - 1\}$, the optimal pricing for client type i is $p_i(t) = b\tau_i D$ for any $t \geq t'$. Based on the data size updating dynamics (35), the total batch size $B(T_{th})$ at time T_{th} and the resulting total expected cost $J(\mathbf{P}(t)|T_{th}, \{1, \dots, j\})$ in (34) can be derived given the optimal dynamic pricing $\mathbf{P}(t)$ in (36). Then, similar to the above analysis, the optimal recruitment threshold T_{th}^* is obtained by checking the first-order condition as shown in Algorithm 1. ■

The procedure to find the optimal recruitment threshold T_{th}^* for heterogeneous clients is concluded in linear Algorithm 1. Note that if the solution to (42) is less than 1, $\hat{T}_{th} = 1$, and then we can calculate \tilde{T}_{th} and Γ_t' to check whether $\frac{s_i}{\tau_i} \leq \frac{bD}{\Gamma_t'}$ is satisfied for any $i \in \{1, \dots, \bar{N}\}, t \in \{0, \dots, T_{th} - 1\}$. If satisfied, $T_{th}^* = \tilde{T}_{th}$. Otherwise, the total expected cost $J(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})$ can be calculated according to the dynamics of data size $B(t)$ in (35) and optimal dynamic pricing $P(t)$ in (36), and then the optimal recruitment threshold T_{th}^* can be derived by checking the first-order condition $\frac{\partial J(\mathbf{P}(t)|T_{th}, \{1, \dots, \bar{N}\})}{\partial T_{th}}$.

Algorithm 1 Optimal recruitment threshold T_{th}^* and client-type choice for heterogeneous clients.

```

1: for  $j = 1 : N$  do
2:   Solve  $\hat{T}_{th}$  as the solution to (42)
3:   if  $\hat{T}_{th} < 1$  then
4:      $\hat{T}_{th} = 1$ 
5:   end if
6:   Calculate  $\tilde{T}_{th} = \arg \min_{T_{th}} (J_{\leq}(\mathbf{P}(\mathbf{t}) | [\hat{T}_{th}], \{1, \dots, \bar{N}\}),$ 
7:      $J_{\leq}(\mathbf{P}(\mathbf{t}) | [\hat{T}_{th}] + 1, \{1, \dots, \bar{N}\}))$  and  $\Gamma'_t$ 
8:   if  $\forall i \in \{1, \dots, j\}, t \in \{0, \dots, T_{th} - 1\}, \frac{s_i}{\tau_i} \leq \frac{bD}{\Gamma'_t}$  then
9:     return  $T_{th}^* = \tilde{T}_{th}$  and
        $J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\}) = J_{\leq}(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\})$ 
10:   else
11:   Calculate  $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, \dots, j\})$  according to the
       dynamics of data size  $B(t)$  in (35) and optimal
       dynamic pricing  $P(t)$  in (36)
12:   if  $\frac{\partial J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, \dots, j\})}{\partial T_{th}} |_{T_{th}=1} \geq 0$  then
13:     return  $T_{th}^* = 1$  and  $J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\})$ 
14:   else
15:     Solve  $\frac{\partial J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, \dots, j\})}{\partial T_{th}} = 0$ 
16:     return  $T_{th}^* = \arg \min (J(\mathbf{P}(\mathbf{t}) | [T_{th}], \{1, \dots, j\}),$ 
        $J(\mathbf{P}(\mathbf{t}) | [T_{th}] + 1, \{1, \dots, j\}))$  and
        $J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\})$ 
17:   end if
18: end if
19: end for
20:  $j^* = \arg \min (J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\}) | j = 1, \dots, N)$ 
21: return optimal client-type  $\{1, 2, \dots, j^*\}$ 

```

5.3 Optimal Client-Type Choice in Stage I

In this subsection, we will discuss the optimal types of clients to be invited in Stage I based on the optimal dynamic pricing and optimal recruitment threshold obtained above. Even though we have derived closed-form solutions to the optimal dynamic pricing and optimal recruitment threshold, the time complexity of finding the optimal client-type choice is still very high with $O(2^N)$ increasing exponentially in N . In the following sections, we consider the non-trivial case that $p_i(t) = s_i \Gamma_t \leq b \tau_i D$ for any $i \in \{1, \dots, N\}$ to reveal the analytical result in Proposition 5.2.

Proposition 5.2. For any N types of clients at the optimum, the central server monotonically chooses client types in set $\{1, 2, \dots, j^*\}$ with $j^* = \arg \min_{j \in \{1, \dots, N\}} J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\})$.

Proof: According to (41), the total expected cost for inviting type i clients only is

$$\begin{aligned}
& J(\mathbf{P}(\mathbf{t}) | T_{th}, \{i\}) \\
&= (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b\tau_i}{\alpha q_i s_i^2 r^2} \right)^{\frac{1}{5}} \left(\frac{1-r^2}{1-r^{2T_{th}}} \right)^{\frac{1}{5}} \left(\frac{\tau_i}{T-T_{th}} \right)^{\frac{1}{5}} \quad (43) \\
&+ \frac{\tau_i}{T-T_{th}}.
\end{aligned}$$

When there are two types of clients, i.e., $N = 2$, according to (43) and (41), we have $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{2\}) \geq J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2\})$ for any given T_{th} . Thus, inviting two types of clients $\{1, 2\}$ is always better than only inviting type-2 clients.

When there are three types of clients, i.e., $N = 3$, the possible client-type combinations are $\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}$ and $\{1, 2, 3\}$. According to (41), the total expected cost of inviting client types $\{1, 3\}$ is

$$\begin{aligned}
& J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 3\}) = (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b}{\alpha r^2} \right)^{\frac{1}{5}} \left(\frac{1-r^2}{1-r^{2T_{th}}} \right)^{\frac{1}{5}} \\
& \times \left(\frac{q_1 s_1^2}{\tau_1} + \frac{q_3 s_3^2}{\tau_3} \right)^{-\frac{1}{5}} \left(\frac{\tau_3}{T-T_{th}} \right)^{\frac{1}{5}} + \frac{\tau_3}{T-T_{th}}, \quad (44)
\end{aligned}$$

and the total expected cost of inviting client types $\{1, 2, 3\}$ is

$$\begin{aligned}
& J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2, 3\}) = (4^{-\frac{4}{5}} + 4^{\frac{1}{5}}) \left(\frac{b}{\alpha r^2} \right)^{\frac{1}{5}} \left(\frac{1-r^2}{1-r^{2T_{th}}} \right)^{\frac{1}{5}} \\
& \times \left(\sum_{i=1}^3 \frac{q_i s_i^2}{\tau_i} \right)^{-\frac{1}{5}} \left(\frac{\tau_3}{T-T_{th}} \right)^{\frac{1}{5}} + \frac{\tau_3}{T-T_{th}}. \quad (45)
\end{aligned}$$

Since $\frac{q_1 s_1^2}{\tau_1} + \frac{q_2 s_2^2}{\tau_2} + \frac{q_3 s_3^2}{\tau_3} > \frac{q_1 s_1^2}{\tau_1} + \frac{q_3 s_3^2}{\tau_3}$, $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2, 3\}) < J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 3\})$ for any given T_{th} . Similarly, we have $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2, 3\}) < J(\mathbf{P}(\mathbf{t}) | T_{th}, \{2, 3\})$ and $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2, 3\}) < J(\mathbf{P}(\mathbf{t}) | T_{th}, \{3\})$. Note that $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{2\}) \geq J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2\})$. Thus, when $N = 3$, we only need to compare the total expected costs $J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1\}), J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2\}), J(\mathbf{P}(\mathbf{t}) | T_{th}, \{1, 2, 3\})$ to find the optimal types of inviting clients.

Similar to the above analysis, for any N groups of clients, if client-type j is invited, all client-types $i \leq j$ should be invited for cost minimization. Thus, the central server only need to compare the optimal expected costs $J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\}), j \in \{1, \dots, N\}$ given the optimal recruitment threshold T_{th}^* and optimal dynamic pricing in (36). Thus, the optimal types of inviting clients is $\{1, 2, \dots, j^*\}$ with $j^* = \arg \min_{j \in \{1, \dots, N\}} J^*(\mathbf{P}(\mathbf{t}) | T_{th}^*, \{1, \dots, j\})$. ■

Given a selected client type j , the iteration duration is at least τ_j . Thus including any client type $i < j$ with smaller data size and training time only creates updates within the iteration duration without reducing the number of global training iterations. As shown in Proposition 5.2, we monotonically select the first j^* types of clients with smaller data sizes and training time to accelerate the model training. The optimal types of inviting clients is one of the following cases: $\{1\}, \{1, 2\}, \dots, \{1, 2, \dots, N\}$. This means the multiple client-type choice is monotonic, which can reduce the time complexity of finding the optimal client-type choice from $O(2^N)$ to $O(N)$ by enumerating only N subsets as shown in Algorithm 1.

5.4 Robustness to Data Size

In reality, our estimation of each client type may not be precise due to some noises, and we wonder our solution's robustness against estimation error of clients' data size. Assume the data size $s_i(t)$ contributed by a type- i client at time slot t faces a variable and bounded error from our estimation: $s_i(t) \in [s_i - \delta_i, s_i + \delta_i], 0 < \delta_i < s_i$, where s_i can be viewed as the mean of type- i clients' data size.

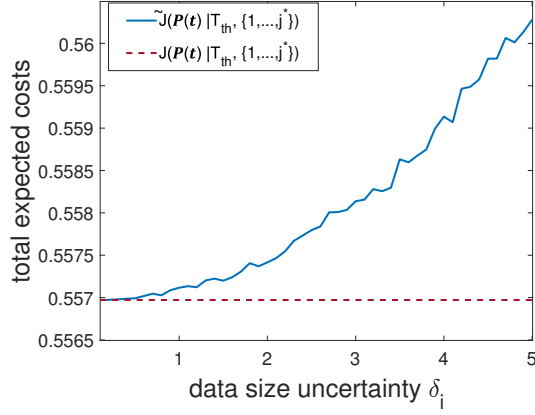


Fig. 6: Total expected cost $\tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ versus data size 's noise bound δ_i , $i \in \{1, \dots, j^*\}$.

Given the optimal client type choice $\{1, \dots, j^*\}$, by applying the dynamic pricing $p_i(t)$ in (36), the dynamics of data size $B(t)$ under uncertain client data size $s_i(t)$, $i \in \{1, \dots, j^*\}$ changes from (35) to

$$\tilde{B}(t+1) = r\tilde{B}(t) + r \sum_{i=1}^{j^*} \frac{\alpha q_i s_i(t)}{b\tau_i D} p_i(t), \quad (46)$$

and the resulting total expected cost is

$$\begin{aligned} \tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) &= \sum_{t=0}^{T_{th}-1} \sum_{i=1}^{j^*} \frac{\alpha q_i}{b\tau_i D} (p_i(t))^2 \\ &\quad + \frac{1}{\sqrt{\tilde{B}(T_{th})D}} + \frac{1}{D}. \end{aligned} \quad (47)$$

By comparing the total expected cost $\tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ under noisy data size $s_i(t)$ in (47) with the total expected cost $J(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ under no noise case where all type- i clients contribute the precisely data size s_i in (34), we have the following proposition.

Proposition 5.3. By applying the optimal dynamic pricing $P(t)$ in (36), the total expected cost objective $\tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ with noisy data size $s_i(t) \in [s_i - \delta_i, s_i + \delta_i]$, $i \in \{1, \dots, j^*\}$, $t \in \{0, \dots, T_{th} - 1\}$ satisfy

$$\tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) \leq J(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) + \Phi(\delta_i|i \in \{1, \dots, j^*\}), \quad (48)$$

where

$$\begin{aligned} \Phi(\delta_i|i \in \{1, \dots, j^*\}) &= \left(\frac{4b\tau_{j^*}(1-r^2)}{\alpha r^2(T-T_{th})(1-r^{2T_{th}})} \right)^{\frac{1}{5}} \\ &\quad \times \left(\frac{(\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i})^{\frac{3}{10}}}{(\sum_{i=1}^{j^*} \frac{q_i s_i (s_i - \delta_i)}{\tau_i})^{\frac{1}{2}}} - \left(\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i} \right)^{-\frac{1}{5}} \right), \end{aligned} \quad (49)$$

which increases with δ_i , $i \in \{1, \dots, j^*\}$.

Proof: According to (34), the total expected cost increases as the total data size $B(T_{th})$ decreases. Thus, given the opti-

mal dynamic pricing $p_i(t)$ in (36) and recruitment threshold T_{th} , the worst-case total cost is:

$$\begin{aligned} &\bar{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) \\ &= \max_{\substack{s_i(t) \in [s_i - \delta_i, s_i + \delta_i], \\ i \in \{1, \dots, j^*\}, \\ t \in \{0, \dots, T_{th} - 1\}}} \tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) \end{aligned} \quad (50)$$

which is achieved when $s_i(t) = s_i - \delta_i$ for any $i \in \{1, \dots, j^*\}$, $t \in \{0, \dots, T_{th} - 1\}$. In this case, according to (46) and $p_i(t)$ in (36), the total data size at time T_{th} is

$$\begin{aligned} \bar{B}(T_{th}) &= \left(\frac{\alpha^2}{16b^2 D^3 (\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i})^3} \right)^{\frac{1}{5}} \sum_{i=1}^{j^*} \frac{q_i s_i (s_i - \delta_i)}{\tau_i} \\ &\quad \times \left(\frac{r^2(1-r^{2T_{th}})}{1-r^2} \right)^{\frac{2}{5}}. \end{aligned} \quad (51)$$

The worst-case total cost given $p_i(t)$ in (36) and $\bar{B}(T_{th})$ in (51) is

$$\begin{aligned} &\bar{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) \\ &= \frac{\tau_{j^*}}{T-T_{th}} + \left(\frac{b\tau_{j^*}(1-r^2)}{\alpha r^2(T-T_{th})(1-r^{2T_{th}})} \right)^{\frac{1}{5}} \\ &\quad \times \left(4^{\frac{1}{5}} \frac{(\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i})^{\frac{3}{10}}}{(\sum_{i=1}^{j^*} \frac{q_i s_i (s_i - \delta_i)}{\tau_i})^{\frac{1}{2}}} + 4^{-\frac{4}{5}} \left(\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i} \right)^{-\frac{1}{5}} \right). \end{aligned} \quad (52)$$

By comparing the worst-case total cost $\bar{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ with the optimal total expected cost $J(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ under no noise case where all type- i clients contribute the precisely data size s_i , $i \in \{1, \dots, j^*\}$, we have

$$\begin{aligned} &\Phi(\delta_i|i \in \{1, \dots, j^*\}) \\ &= \bar{J}(T|i \in \{1, \dots, j^*\}) - J(T|i \in \{1, \dots, j^*\}) \\ &= \left(\frac{4b\tau_{j^*}(1-r^2)}{\alpha r^2(T-T_{th})(1-r^{2T_{th}})} \right)^{\frac{1}{5}} \\ &\quad \times \left(\frac{(\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i})^{\frac{3}{10}}}{(\sum_{i=1}^{j^*} \frac{q_i s_i (s_i - \delta_i)}{\tau_i})^{\frac{1}{2}}} - \left(\sum_{i=1}^{j^*} \frac{q_i s_i^2}{\tau_i} \right)^{-\frac{1}{5}} \right), \end{aligned}$$

where $J(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ is as given in (41). Note that $\tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\}) \leq \bar{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$. Then, (48) is obtained. ■

As shown in Proposition 5.3, the error term $\Phi(\delta_i|i \in \{1, \dots, j^*\})$ increases with upperbound error δ_i , $i \in \{1, \dots, j^*\}$. Besides worst-case analysis, we also provide the average-case analysis via simulations. As shown in Fig. 6, for randomly generated noisy data size $s_i(t) \in [s_i - \delta_i, s_i + \delta_i]$, $i \in \{1, \dots, j^*\}$ in each time slot t , the difference between $\tilde{J}(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ and $J(\mathbf{P}(t)|T_{th}, \{1, \dots, j^*\})$ increases with the noise bound δ_i , $i \in \{1, \dots, j^*\}$, which coincides with the changing trend of $\Phi(\delta_i|i \in \{1, \dots, j^*\})$.

6 NUMERICAL EXPERIMENT

In this section, we conduct simulation experiments to evaluate the performance of our proposed solution. For homogeneous clients with identical data size and training time, we discuss the static pricing benchmark versus our dynamic pricing and extend to our closed-form dynamic pricing to the truncated normal distribution of clients' private costs

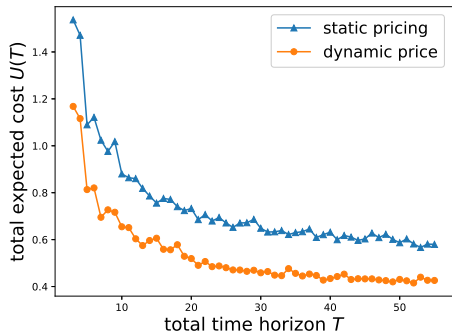


Fig. 7: Static Pricing versus Dynamic Pricing by comparing their total expected cost objectives on ResNet18.

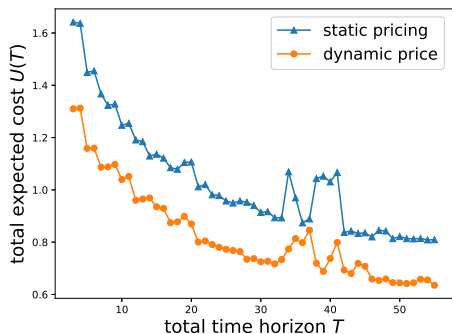


Fig. 8: Static Pricing versus Dynamic Pricing by comparing their total expected cost objectives on AlexNet.

with CDF. Then we consider heterogeneous clients with different data size and training time to show how different impacting factors affect optimal client-type choice. Default, we train the federated learning model on the CIFAR-10 dataset and use Resnet18 [35] as our neural network. We use a typical cross-entropy loss function and the accuracy to measure the quality of the model as in [1], [26]. We use the standard FL algorithm FedAVG [5], and the client uses the mini-batch SGD algorithm with an initial learning rate of 0.001.

Firstly, we examine the performance of our proposed dynamic pricing for homogeneous clients, by comparing the total expected cost $U(T)$ under the static pricing in (21) and our dynamic pricing in (7). We set parameter values to be $\alpha = 0.99, b = 1, s = 2.0, r = 0.7, \tau = 1.0$. The experimental results in Fig. 7 show that our proposed dynamic pricing strategy always outperforms the static pricing strategy, and the gap increases with time horizon T due to the accumulated advantage of dynamic price over time. In general, the total expected costs $U(T)$ in (6) for both static and dynamic pricing decrease over time. This is because the model accuracy loss decreases as the number of training iterations D increases due to the increased time horizon T , while the increase in the clients' payment is less than the accuracy loss. Moreover, we compare the performance of static pricing strategy and dynamic pricing strategy on the Cifar100 dataset with AlexNet in Fig. 8. It is shown that our dynamic pricing strategy is still very effective for different datasets and machine learning models.

Our results can also be applied to other distribution of clients' private costs. Consider the truncated normal

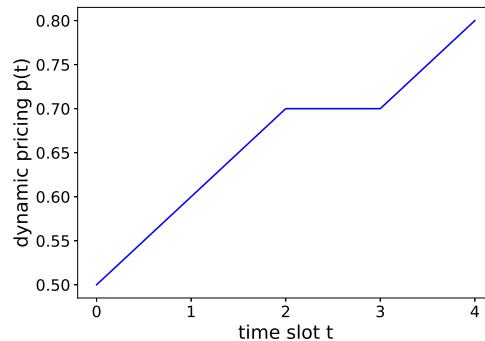


Fig. 9: Dynamic Price $P(t)$ under truncated normal distribution $F(c, a, b)$.

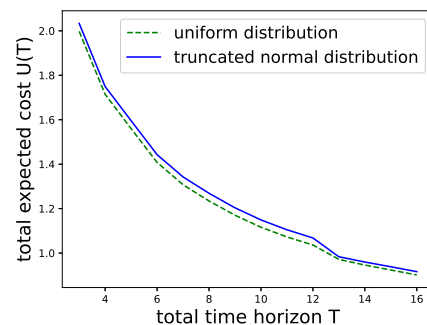


Fig. 10: Truncated normal CDF versus Uniform CDF by comparing their total expected cost objectives.

distribution with CDF as follows:

$$F(c, a, b) = \frac{\operatorname{erf}\left(\frac{c-\mu}{\sqrt{2}\sigma}\right) - \operatorname{erf}\left(\frac{a-\mu}{\sqrt{2}\sigma}\right)}{\operatorname{erf}\left(\frac{b-\mu}{\sqrt{2}\sigma}\right) - \operatorname{erf}\left(\frac{a-\mu}{\sqrt{2}\sigma}\right)}, c \in [a, b],$$

with clients' cost mean μ and variance σ . In this case, it is not easy to derive closed-form solution to dynamic pricing due to the nonlinear dynamics of data size. Consider $\alpha = 0.9, a = 0, b = 1, \mu = 0, 5, \sigma = 1.0, s = 1, r = 0.9, \tau = 0.9, T = 16, T_{th} = 5$ and $\epsilon = 0.1$. By taking exhaustive search, it is shown in Fig. 9 that the dynamic pricing $p(t)$ increases in t , which coincides with the results of uniform distribution in Proposition 3.1. However, the computational complexity is very high with $O((b(T - T_{th})/\epsilon)^{T_{th}} * T_{th})$. We further show in Fig. 10 that the gap of the total expected costs $U(T)$ by applying the optimal dynamic pricing via exhaustive search and our approximate closed-form dynamic pricing in (7) respectively is small, which indicates that it is feasible to apply our closed-form dynamic pricing to other distribution to alleviate the computational burden.

Moreover, we verify our dynamic pricing strategy's performance on more realistic Non-IID data settings. Let $\alpha = 0.9, b = 1, s_0 = 2.0, r = 0.99$ and $\tau = 1.98$. Refer to the work of Hsu et al. [36], we also assume that every client training examples are drawn independently with class labels following a categorical distribution over M classes parameterized by a vector \mathbf{q} ($q_i > 0, i \in \{1, \dots, M\}$ and $|\mathbf{q}|_1 = 1$). Then we generate $\mathbf{q} \sim \operatorname{Dir}(\alpha_d \mathbf{q})$ from a Dirichlet distribution, where \mathbf{q} characterizes a prior class distribution over M classes, and $\alpha_d > 0$ is a concentration parameter controlling the identicalness among clients. With $\alpha_d \rightarrow +\infty$, all

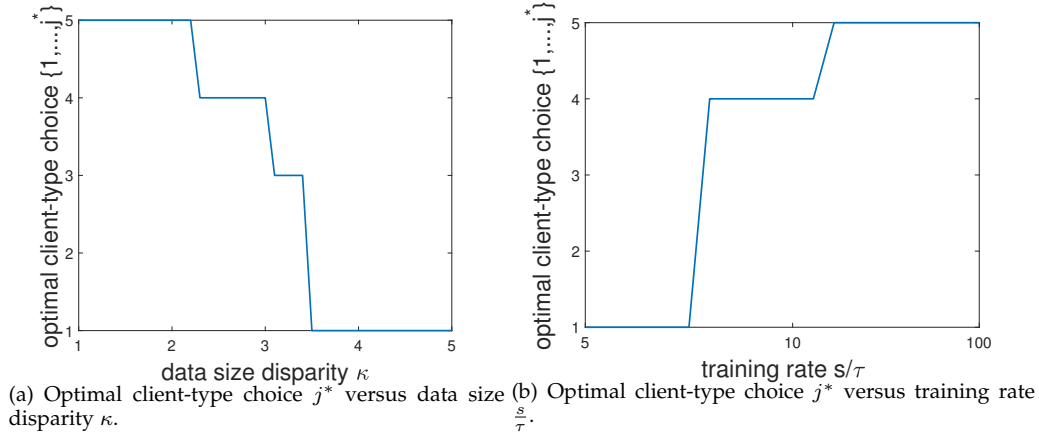


Fig. 12: Optimal client-type choice j^* with $\{1, \dots, j^*\}$ versus data size disparity κ and training rate $\frac{s}{\tau}$.

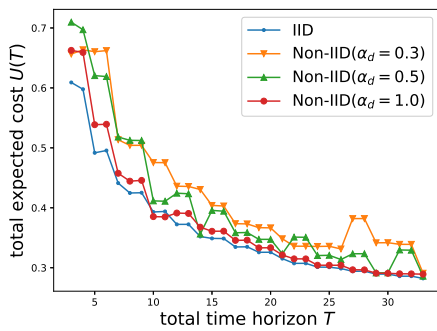


Fig. 11: The performances of our dynamic pricing under different Non-IID settings of different α_d degrees.

clients have identical distributions to the prior; with $\alpha_d \rightarrow 0$, on the other extreme, each client holds examples from only one class chosen at random [36]. Here, we run simulations on the MNIST dataset, and use a linear model with a fully connected layer of input channel 784 and output channel 10. We repeat all the experiments three times to take the average as the final results. Fig. 11 shows that the data performance under different Non-IID degrees. The trend of the expected server cost versus total time horizon T is consistent with the IID data case, which indicates that our theoretical results can be applied to the Non-IID data. When the Non-IID degree of data decreases, the expected cost becomes closer to that for the IID data case which can be viewed as the performance bound. Meanwhile, as the total time horizon T increases, the gap between Non-IID data and IID data also decreases. This is because increased recruitment allows more data to be recruited, which mitigates the impact of Non-IID data distribution.

Then, we will show how the optimal client-type choice changes with different factors. Consider $N = 5$ types of clients with uniform client distribution $\{q_i = \frac{1}{N}, i \in \{1, \dots, N\}\}$, different data size $\{s_i = s_0 + (i - 1)\kappa, i \in \{1, \dots, N\}\}$ and training time $\{\tau_i, i \in \{1, \dots, N\}\}$, where κ is the data size disparity. Note that a client's training time increases with the data size. According to [4], we set $\tau_i = \beta s_i$, where β is related to CPU-cycle frequency and transmission rate. Let $\alpha = 0.5, b = 1, s_0 = 1, r = 0.5$ and $T = 10$. When data size disparity $\kappa = 1$ and $\beta = 0.01$, the optimal client-

type choice is $\{1, 2, 3, 4, 5\}$ by inviting all types of clients. Starting from above setting, Fig 12 shows how the optimal client-type choice changes with different impacting factors. As the data size disparity κ between any two neighboring types increases from 1 to 5, it is shown in Fig. 12(a) that the optimal client-type choice decreases from $\{1, 2, 3, 4, 5\}$ including all types to $\{1\}$ with only type-1 clients with the smallest data size and training time per iteration. As κ increases in the synchronous FL running, the clients with smaller data and training time need to wait longer for those clients with larger dataset to complete, which results in less global training iterations and thus we drop higher client types. As the training rate $\frac{s_i}{\tau_i} = \frac{1}{\beta}$ for each client-type i increases, Fig. 12(b) shows that it is better to recruit more types of clients without worrying about their training time difference. Our simulations also show that we will include more client types given a longer time horizon T and more client types N . We skip the details here due to the page limit.

7 CONCLUSION

In this paper, we focus on the clients' incentive mechanism design in FL, by offering time-dependent monetary rewards per client arrival to trade-off between the total payment and the FL model's accuracy loss, under incomplete information about their random arrivals and private training costs. We jointly consider two phases including the client recruitment phase and model training phase to balance the total data size and training iterations. First, for homogeneous clients with identical data size and training time, we obtain a new dynamic pricing solution in closed-form to optimally balance the total payment to clients and the accuracy loss. Such pricing scheme gradually increases when close to recruitment deadline due to aging effect. Moreover, for heterogeneous clients with different data size and training time, we use a three-stage model to successfully extend our dynamic pricing solution. A linear algorithm is proposed to find the optimal client recruitment threshold and monotonically select client types for FL. Finally, we show the robustness of our solutions against estimation errors of clients' data size and run numerical experiments to validate our analytical results. In the future, we plan to extend our dynamic pricing analysis to asynchronous FL.

REFERENCES

- [1] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19 586–19 597, 2020.
- [2] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.
- [3] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018.
- [4] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. Paris, France: IEEE, 2019, pp. 1387–1395.
- [5] 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*, vol. 54. Volos, Greece: PMLR, 2017, pp. 1273–1282.
- [6] V. Smith, S. Forte, M. Chenxin, M. Takáč, M. I. Jordan, and M. Jaggi, "Cocoa: A general framework for communication-efficient distributed optimization," *Journal of Machine Learning Research*, vol. 18, p. 230, 2018.
- [7] C. Briggs, Z. Fan, and P. Andras, "Federated learning with hierarchical clustering of local updates to improve training on non-iid data," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–9.
- [8] H. Wang, S. Guo, Z. Qu, R. Li, and Z. Liu, "Error-compensated sparsification for communication-efficient decentralized training in edge environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 1, pp. 14–25, 2021.
- [9] F. Wu, S. He, S. Guo, Z. Qu, H. Wang, W. Zhuang, and J. Zhang, "Sign bit is enough: a learning synchronization framework for multi-hop all-reduce with ultimate compression," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022, pp. 193–198.
- [10] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.
- [11] S. Caldas, J. Konečný, H. B. McMahan, and A. Talwalkar, "Expanding the reach of federated learning by reducing client resource requirements," *arXiv preprint arXiv:1812.07210*, 2018.
- [12] C. Fung, C. J. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *arXiv preprint arXiv:1808.04866*, 2018.
- [13] Y. Liu, J. Peng, J. Kang, A. M. Ilyyasu, D. Niyato, and A. A. Abd El-Latif, "A secure federated learning framework for 5g networks," *IEEE Wireless Communications*, vol. 27, no. 4, pp. 24–31, 2020.
- [14] M. Hao, H. Li, G. Xu, S. Liu, and H. Yang, "Towards efficient and privacy-preserving federated deep learning," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. Shanghai, China: IEEE, 2019, pp. 1–6.
- [15] H. Yu, Z. Liu, Y. Liu, T. Chen, M. Cong, X. Weng, D. Niyato, and Q. Yang, "A fairness-aware incentive scheme for federated learning," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 2020, pp. 393–399.
- [16] Y. Zhan, J. Zhang, Z. Hong, L. Wu, P. Li, and S. Guo, "A survey of incentive mechanism design for federated learning," *IEEE Transactions on Emerging Topics in Computing*, 2021.
- [17] Y. Zhan, P. Li, K. Wang, S. Guo, and Y. Xia, "Big data analytics by crowdlearning: Architecture and mechanism design," *IEEE Network*, vol. 34, no. 3, pp. 143–147, 2020.
- [18] S. Feng, D. Niyato, P. Wang, D. I. Kim, and Y.-C. Liang, "Joint service pricing and cooperative relay communication for federated learning," in *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (Green-Com) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. Atlanta, GA, USA: IEEE, 2019, pp. 815–820.
- [19] Y. Sarikaya and O. Erceetin, "Motivating workers in federated learning: A stackelberg game perspective," *IEEE Networking Letters*, vol. 2, no. 1, pp. 23–27, 2019.
- [20] 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.
- [21] Y. Zhan and J. Zhang, "An incentive mechanism design for efficient edge learning by deep reinforcement learning approach," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. Toronto, ON, Canada: IEEE, 2020, pp. 2489–2498.
- [22] 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.
- [23] Webank, "Application of fate in credit risk management of small and micro enterprises," <https://cn.fedai.org/cases/utilization-of-fate-in-risk-management-of-credit-in-small-and-micro-enterprises/>, 2020.
- [24] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, "Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2438–2455, 2019.
- [25] 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.
- [26] 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)*. Volos, Greece: IEEE, 2020, pp. 1–8.
- [27] AWS, "Amazon EC2 pricing," <https://aws.amazon.com/ec2/pricing>, 2021.
- [28] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.
- [29] J. Jiang and L. Hu, "Decentralised federated learning with adaptive partial gradient aggregation," *CAA Transactions on Intelligence Technology*, vol. 5, no. 3, pp. 230–236, 2020.
- [30] 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, pp. 1–38, 2012.
- [31] 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.
- [32] J. Pei, K. Zhong, M. A. Jan, and J. Li, "Personalized federated learning framework for network traffic anomaly detection," *Computer Networks*, vol. 209, p. 108906, 2022.
- [33] N. Cha, Z. Du, C. Wu, T. Yoshinaga, L. Zhong, J. Ma, F. Liu, and Y. Ji, "Fuzzy logic based client selection for federated learning in vehicular networks," *IEEE Open Journal of the Computer Society*, vol. 3, pp. 39–50, 2022.
- [34] J. Xu and H. Wang, "Client selection and bandwidth allocation in wireless federated learning networks: A long-term perspective," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1188–1200, 2020.
- [35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Las Vegas, Nevada: IEEE, 2016, pp. 770–778.
- [36] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.



Xuehe Wang (S'15-M'16) received her Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore in 2016. She is an Associate Professor of School of Artificial Intelligence, Sun Yat-sen University, China. Before that, she was an Assistant Professor of Infocomm Technology Cluster with Singapore Institute of Technology from 2019 to 2021, and a postdoctoral research fellow with the Pillar of Engineering Systems and Design, Singapore University of Technology and Design

from 2015 to 2019. Her research interests cover multi-agent systems, control theory, network economics and game theory.



Shensheng Zheng received his B.E. degree in School of Computer Engineering and Science from Shanghai University, Shanghai, China, in 2021. He is now a M.S. student with the School of Artificial Intelligence, Sun Yat-sen University, China. His research interests include federated learning and control theory.



Lingjie Duan (S'09-M'12-SM'17) received the Ph.D. degree from The Chinese University of Hong Kong in 2012. He is an Associate Professor of Engineering Systems and Design with the Singapore University of Technology and Design (SUTD). In 2011, he was a Visiting Scholar at University of California at Berkeley, Berkeley, CA, USA. His research interests include network economics and game theory, cognitive and green networks, and energy-harvesting wireless communications. He is an Associate Editor of IEEE Transactions on Mobile Computing. He was an Editor of IEEE Transactions on Wireless Communications. He was also an Editor of IEEE Communications Surveys and Tutorials. He also served as a Guest Editor of the IEEE Journal on Selected Areas in Communications Special Issue on Human-in-the-Loop Mobile Networks, as well as IEEE Wireless Communications Magazine. He received the SUTD Excellence in Research Award in 2016 and the 10th IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2015.