

OPTION: An Online Pricing Strategy for Asynchronous Federated Learning Against Free-Riding Attacks

Bangqi Pan^{1, 2}, Jianfeng Lu^{2, 3*}, Shuqin Cao², Xiao Zhang⁴, Gang Li⁵, Guanghui Wen⁶

¹School of Computer Science and Technology, Wuhan University of Science and Technology, China

²Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, Wuhan University of Science and Technology, China

³Key Laboratory of Social Computing and Cognitive Intelligence (Dalian University of Technology), Ministry of Education, China

⁴School of Computer Science, South-Central Minzu University, China

⁵College of Computer Science, Inner Mongolia University, China

⁶School of Automation, Southeast University, China

{panbq, lujianfeng, shuqincao}@wust.edu.cn, xiao.zhang@mail.scuec.edu.cn, gli@imu.edu.cn, ghwen@seu.edu.cn

Abstract

Asynchronous Federated Learning (AFL) is acclaimed for accelerating collaborative training on heterogeneous systems by eliminating the wait for stragglers. While current solutions focus on improving convergence amidst update delays, they neglect how delayed aggregation fosters free-riding attacks, allowing malicious clients to easily extract the global model without contribution. This behavior results in significant fairness issues and performance degradation. To address this challenge, we propose OPTION, the first online pricing strategy tailored to mitigate free-riding in AFL. OPTION establishes an economic model in which access to model updates is purchased using credits earned from verified contributions. Specifically, OPTION values each model update according to its marginal performance gain and training cost, and subsequently necessitates a download fee from each client based on the Hotelling model to prevent zero-cost acquisition. Moreover, OPTION rewards clients for successful updates under non-arbitrage constraints, effectively balancing individual utility and task budget. To maximize the average model performance while satisfying these conditions, OPTION leverages the Lyapunov drift framework and a probabilistic sampling-based algorithm to optimize the pricing parameters. Extensive experimental results on three real-world datasets demonstrate that OPTION effectively mitigates free-riding attacks in AFL, increases the number of valid updates by at least 23.97%, and achieves a model accuracy improvement of at least 3.01% compared to state-of-the-art baselines.

Introduction

Federated Learning (FL) enables collaborative model training with decentralized data while preserving privacy. However, its deployment is challenged by client heterogeneity, a problem especially pronounced in wireless environments such as the Internet of Things, with its diverse computation resources, or mobile health, where user participation varies (Kairouz et al. 2021; Wang et al. 2022). In recent years,

Asynchronous Federated Learning (AFL) has emerged, allowing clients to participate in aggregation asynchronously, enabling the server to proceed without waiting for slower clients (Xu et al. 2023). Despite its advantages, AFL's asynchronous and decentralized nature exposes it to security threats, particularly free-riding attacks. In these attacks, malicious or non-cooperative clients obtain the global model without contributing valuable parameters (Wang et al. 2024; Zhang et al. 2021), which not only undermines the model's performance but also exacerbates fairness and motivation challenges (Fraboni, Vidal, and Lorenzi 2021).

Current research on free-riding primarily focuses on Synchronous FL (SFL) and regulates client behavior through incentive mechanisms. For instance, (Zhang, Ma, and Chen 2022) employed game theory to study client behavior and discourage free-riders after detecting attacks. Additionally, (Tang, Peng, and Wong 2024) developed a blockchain-based incentive mechanism that accelerates convergence speed by rewarding participating clients. Although these approaches mitigate free-riding to some extent in SFL, they inherently assume that the server dominates global aggregation. Therefore, existing work on free-riding in SFL is not applicable to scenarios, where clients update parameters online.

Different from SFL, AFL typically provides clients with greater flexibility in determining the timing and frequency of model updates. Preventing free-riding in AFL still faces the following significant challenges.

- **Non-periodical aggregation.** In AFL, clients asynchronously upload their local updates, facing participation costs that fluctuate in real time with volatile network conditions where available upload bandwidth can plummet unexpectedly, and connections may suffer from severe packet loss and jitter (Korhonen and Wang 2005). This dual unpredictability renders existing incentive mechanisms, which often rely on synchronous rounds or static pricing, largely ineffective (Deng et al. 2021; Zhang, Lin, and Zhang 2022; Wang et al. 2021; Zhu et al. 2023). Therefore, developing a mechanism that can dynamically price contributions in this unpredictable

*Corresponding author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

environment is a critical open problem.

- **Divergent model versions.** Existing methods focus on maximizing the global model performance in SFL (Shi et al. 2024). Although the server publishes a same global model, the update cost varies across clients due to differences in their local model versions in AFL. This disparity presents a significant challenge for pricing, as the value provided to each client is difficult to quantify. Extensive economic research suggests that sustainable development can only be achieved by maximizing social utility (Asheim, Mitra, and Tungodden 2017). Therefore, it is essential to monitor the performance of each client’s local model, which introduces additional optimization objectives and increases computational complexity.
- **Long-term coupled constraints.** Another fundamental challenge is satisfying the coupled, long-term constraints of individual rationality (Tang and Wong 2021) and budget balance (Su et al. 2022). The budget is fixed for the entire training horizon, not per-round, creating temporal interdependencies that couple all payment decisions. This structure, along with the need to deter free-riding, makes the online allocation problem exceptionally difficult. Consequently, finding a mechanism with strong performance guarantees, let alone proving its optimality, represents a significant hurdle.

With this in mind, we propose a novel Online Pricing Strategy for parameter informatION, named **OPTION**, which prices the global model at different aggregation rounds to eliminate potential free-riding behavior in AFL. Specifically, **OPTION** divides the finite aggregation period into multiple aggregation rounds and employ the Hotelling model to determine download fees for accessing the global model. Regarding asynchronous updating, **OPTION** considers the heterogeneous update costs arising from different versions of local models, effectively reducing the unfair profits and eliminating free-riders in the AFL system. Additionally, **OPTION** rewards clients who successfully upload model parameters under a non-arbitrage constraint, incentivizing participation while balancing individual client utilities. To maximize model performance under these conditions, **OPTION** builds an online algorithm, which employs the Lyapunov drift framework and probabilistic sampling to optimize the strategy in dynamic wireless networks.

Our main contributions are summarized as follows.

- Methodologically, we propose an online pricing strategy to eliminate free-riders in AFL. To the best of our knowledge, this is the first study to focus on free-riding in AFL. We establish a non-arbitrage constraint based on the law of one price while simultaneously ensuring individual rationality and budget balance, and employ the block coordinate descent method to optimize the pricing strategy.
- Theoretically, we devise virtual queues to accommodate the dynamic costs in AFL and prove the convergence of the queue lengths. In particular, we derive a closed-form solution for the optimal pricing policy in the bi-convex case and extend our approach to non-convex scenarios using a saturation function. Furthermore, we analyze the

computational overhead of our proposed algorithm, **OPTION**, and prove its polynomial-time complexity.

- Experimentally, we conduct extensive experiments on three real-world datasets, including comparisons against three baselines. The detailed results indicate that **OPTION** not only maintains system’s budget balance but also increases the gradient update amount for clients in AFL by at least 23.97%, shorts the execution time by 75.42%, and reduces average model accuracy loss by up to 3.01% compared with state-of-the-art baselines.

Related Work

In this section, we analyze two aspects of defense strategies for free-riding: detection and defense.

Free-riding attack detection

Free-riding attack detection is an essential issue in extensive FL applications, such as medical FL (Ottun et al. 2022), Internet of Things (Wang et al. 2021). For instance, (Lin, Du, and Liu 2019) proposed a detection method based on the deep auto-encoding Gaussian mixture model to identify abnormal model parameters. On this basis, (Xu et al. 2022) designed a client reputation based approach to identify free-riders by evaluating their contributions to the global model.

Although the above methods can identify free-riders in SFL, they are unsuitable for AFL due to its variable update frequency and unbalanced parameter information. In this paper, we integrate attack detection with online incentive strategies to eliminate free-riders in the AFL system.

Free-riding attack defense

Incentive mechanisms are frequently employed to mitigate the negative externalities caused by selfish clients in FL (Zhan et al. 2021). For example, (Hu and Gong 2020) designed an incentive mechanism based on game theory, which models the interaction between selfish clients and servers as a two-stage Stackelberg game and solves the equilibrium of the game to compensate for client’s privacy costs. Furthermore, (Murhekar et al. 2023) analyzed the Nash equilibrium of collaborative training between clients and derived the equilibrium solution through dynamics. To maximize social welfare, they also designed a budget balanced incentive mechanism to reward clients for their updates.

The above mechanisms hold potential for addressing free-riding issues in SFL but is not applicable to AFL due to their inability to tackle the challenges of multiple gradient versions and dynamic communication environment. Contrarily, **OPTION** is designed to overcome these limitations. It introduces an online optimization framework to AFL, ensuring that our online pricing strategy can adapt to changing and complex aggregation scenarios.

Problem Formulation

In this section, we first model AFL and client utility, and then we formalize the optimal pricing problem.

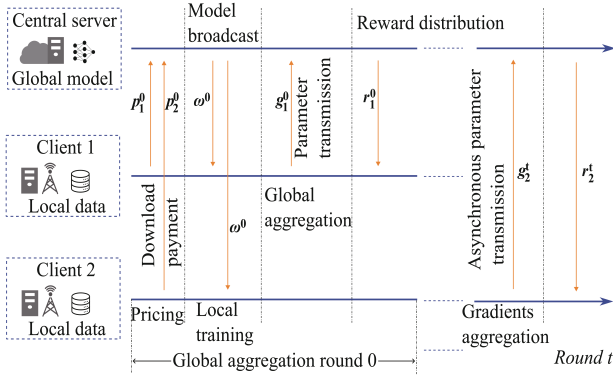


Figure 1: The system model of OPTION.

System Model

AFL procedure The AFL system consists of a central server and a set of clients $\mathcal{N} = \{1, 2, \dots, N\}$. As shown in Figure 1, asynchronous aggregation runs within a discrete time range consisting of multiple rounds in set \mathcal{T} (Chen, Sun, and Jin 2019). In round t , the server broadcasts the global model parameters ω^t to the selected clients, and each client initializes the local model with the received parameters. Each client $n \in \mathcal{N}$ updates its local model based on the gradient of the regularized loss function $l(\omega_n^t)$ as follows:

$$g_n^t = \nabla l(\omega_n^t) + \kappa g_n^{t-1}, \quad (1)$$

where g_n^{t-1} is the local gradient of client n in round $t-1$, and κ is the decay coefficient (Zheng et al. 2017). The gradients g_n^{t-1} that were not transmitted in previous rounds are aggregated with the current gradient. We define $s^t = \{s_1^t, \dots, s_N^t\}$ as the transmission provisioning indicator of the central server, where $s_n^t = 1$ represents that client n is selected in round t , and $s_n^t = 0$ otherwise.

After local training, client n transmits its local gradient to the central server according to the aggregation provisioning. Then, client n initializes local gradient as $g_n^t = 0$, if client n transmit its gradient. Here, we use the Holtling model Δv_n^t to represent the version difference between global model ω^t and local model ω_n^t (Long et al. 2024), where $v_n^t = \{1, \dots, T\}$ represents the version of the global model held by client n at round t .

Client utility In each round, the utilities of the clients participating in gradient uploads are influenced by training costs $c_t = \{c_1^t, \dots, c_N^t\}$, model download payment $\mathbf{p}_t = \{p_1^t, \dots, p_N^t\}$, and upload rewards $\mathbf{r}_t = \{r_1^t, \dots, r_N^t\}$, defined as $u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t)$. The transmission cost is an exogenous variable of the uplink (Yu et al. 2010), while the download costs and upload rewards are endogenous variables specified by OPTION. It is necessary to ensure that the utility of clients' update gradients is non-negative.

Definition 1. (Individual Rationality) In each round t , individual rationality is satisfied if and only if for any selected client n participating in the model update, there is

$$u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t) \geq 0. \quad (2)$$

Since local training and transmission incur costs, satisfying individual rationality requires that rewards exceed payments. According to (Tang and Wong 2021), the setting of rewards needs to consider the balance of system budget.

Definition 2. (Budget Balance) Given the task budget B for model updates, budget balance means that B covers the gap between payments p_n^t and rewards r_n^t , i.e.,

$$B - \sum_{t \in \mathcal{T}} \sum_{n \in \mathcal{N}} (r_n^t - p_n^t) \geq 0. \quad (3)$$

Non-arbitrage Pricing Problem

In addition to individual rationality and budget balance, we also consider the non-arbitrage constraint, which prevents clients from gaining benefits through free-riding. According to the law of one price (Fackler and Goodwin 2001), the non-arbitrage constraint requires that the payment for clients to download the model equals updating cost. That is to say, there is no cost lower than \mathbf{p}_t for client $n \in \mathcal{N}$ to obtain the global model ω^{t+1} in the next round.

Definition 3. (Non-arbitrage Pricing) Given update cost in the market $M_t(\Delta v_n^t)$ of client n in round t , non-arbitrage pricing is satisfied if and only if

$$p_n^t - M_t(\Delta v_n^t) \geq 0. \quad (4)$$

Taking incentive compatibility into account (Ding, Gao, and Huang 2023), we are unable to mandate clients to disclose their truth update costs. Instead, the required update costs $M_t(\Delta v_n^t)$ can be estimated based on the improvement in the model's test accuracy (See Appendix). We focus on social performance, measured by each client's local model performance. We define $E_t(s^t)$ as the global model performance for each round, and v_n^t as the model version for each client. The pricing problem in AFL can be formulated as

$$\begin{cases} \mathbf{P}_1 : \min_{\mathbf{p}, \mathbf{r}, \mathbf{s}} F_T \triangleq \sum_{n \in \mathcal{N}} E_n^T(\mathbf{s}, \mathbf{p}, \mathbf{r}), & (5a) \\ \text{s.t. } p_n^t - M_t(\Delta v_n^t) \geq 0, \forall n \in \mathcal{N}, & (5b) \\ u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t) \geq 0, \forall n \in \mathcal{N}, & (5c) \\ B - \sum_{t \in \mathcal{T}} \sum_{n \in \mathcal{N}} (r_n^t - p_n^t) \geq 0, & (5d) \end{cases}$$

where the term $E_n^T(\mathbf{s}, \mathbf{p}, \mathbf{r})$ in Eq. (5a) is the loss of the model that the client n received from the server in round T . Our objective function F_T balances two competing goals: 1) learning a high-quality model sequence (maximizing $E_t(s_t)$), and 2) ensuring the system efficiently delivers these models to clients to minimize model staleness. This prevents a scenario where a high-quality model is generated but rarely used by clients in a timely manner. In problem \mathbf{P}_1 , we define tensor $\mathbf{p} = [p_1, \dots, p_T]$, $\mathbf{r} = [r_1, \dots, r_T]$ and $\mathbf{s} = [s_1, \dots, s_T]$ for convenience. \mathbf{P}_1 is a mixed-integer nonlinear optimization problem, where exact solutions are computationally prohibitive and impractical for aggregation.

Design of OPTION

We first construct an online optimization to approximate \mathbf{P}_1 . Then, we obtain a suboptimal solution using a greedy algorithm, analyze the performance of the pricing strategy, and derive a closed form solution under linear conditions.

Online Pricing and Queue Stability

We use an online optimization framework to address the challenges posed by long-term individual rationality and no arbitrage constraints. We outline the global model's long-term performance evolution, transform individual rationality constraints into a queue stability problem, and present approximate per-round solutions.

Convergence of AFL. To analyze the changes of Eq. (5a) in each round, we approximate model performance using the upper bound of convergence error. Let $\nabla l(\mathbf{x})$ denote the gradient of the function at point \mathbf{x} , we make the following typical assumptions, as outlined in (Li et al. 2020) and (Chen et al. 2020).

Assumption 1. For all aggregated parameters \mathbf{x} and \mathbf{y} of the global model, we assume:

- (*L-smoothness*) There exists a Lipschitz constant $L > 0$ that satisfies $\|\nabla l(\mathbf{x}) - \nabla l(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$;
- (ε -strongly convexity) There exists a constant $\varepsilon > 0$ that satisfies $l(\mathbf{x}) - l(\mathbf{y}) \geq \langle \nabla l(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{\varepsilon}{2}\|\mathbf{x} - \mathbf{y}\|^2$;
- (*Bounded gradient divergence*) There exists a constant ρ , such that $\mathbb{E}[\|g_n^t\|^2] \leq L\|\nabla l(\boldsymbol{\omega})\|^2\rho^2$, and there exists $\varrho > 0$ such that $\nabla l(\boldsymbol{\omega})^\top \mathbb{E}[g_n^t] \geq \varrho\|\nabla l(\boldsymbol{\omega})\|^2$.

With Assumption 1, we can obtain the convergence upper bound of AFL process, as stated in the following theorem.

Theorem 1. Under Assumption 1, the convergence error of AFL after T rounds is upper bounded by

$$\mathbb{E}[l(\boldsymbol{\omega}^T) - l(\boldsymbol{\omega}^*)] \leq E_T(s_T) := \prod_{t=0}^{T-1} \left[1 - 2\varepsilon\hat{\eta}^t \left(\frac{\hat{\eta}^t \rho^2 L S^2}{2} - \varrho \sum_{n \in \mathcal{N}} s_n^t \mathbb{P}[h_n^t = 1] \right) \right] \times [l(\boldsymbol{\omega}^0) - l(\boldsymbol{\omega}^*)], \quad (6)$$

where $\hat{\eta}^t = \max\{\eta_n^t | n \in \mathcal{N}\}$ and $S = \max_{t \in \mathcal{T}} \sum_{n \in \mathcal{N}} s_n^t$. The Boolean variable $h_n^t \in \{0, 1\}$ indicates whether client n uploads parameters in round t , where $h_n^t = 1$ represents that g_n^t is uploaded and 0 represents that g_n^t is not uploaded.

Evolution of model performance. Regarding the changes in global model performance after each aggregation round, based on Theorem 1, we have the following conclusion.

Corollary 1. If there exists $\xi \leq 0$ such that $\sum_{n \in \mathcal{N}} p_n^t \geq \xi$ and $\eta_n^t < \eta^t := \frac{2\varepsilon\xi}{L\rho^2 S^2} [\max\{\lambda_n^t | n \in \mathcal{N}\}]^{-1}$, we can represent the evolution of Eq. (5a) from round t to $t+1$ as

$$F_{t+1} = F_t - E_t(s_t) \sum_{n \in \mathcal{N}} \theta_n^{t+1}(\mathbf{p}_{t+1}, \mathbf{r}_{t+1}) \times \left[1 - 2\varepsilon\hat{\eta}^t \left(\frac{\hat{\eta}^t \rho^2 L S^2}{2} - \varrho \sum_{n \in \mathcal{N}} s_n^t \mathbb{P}[h_n^t = 1] \right) \right], \quad (7)$$

where $\eta := \min\{\eta_n^t, |n \in \mathcal{N}, t \in [0, T-1]\}$ and $\hat{\eta} := \max\{\eta^t, |t \in [0, T-1]\}$. The term $\theta_n^t(\mathbf{p}, \mathbf{r})$ in Eq. (7) is the probability that the client n receives the global model parameters in round t , satisfying $\theta_n^t(\mathbf{p}_t, \mathbf{r}_t) = \mathbb{P}[v_n^t = t]$.

Based on Eq. (7), we observe that the model performance in round $t+1$ is influenced by the participation rate. In an asynchronous system, a client n may suffer from network delays or computational straggling, causing it to work on a stale model version (i.e., $v_n^t < t$). Therefore, optimizing F_{t+1} can be achieved by optimizing $\sum_{n \in \mathcal{N}} \mathbb{P}[v_n^t = t]$, as shown in Eq. (7).

Optimization problem in round t . We transform the budget balance in Eq. (5d), individual rationality in Eq. (5c), and Non-arbitrage constraint in Eq. (5b) into a queue stability problem to design pricing.

- (*Utility queue*) For the individual utility of client n , $\Gamma_n^0 = \max\{0, -u_n^0(c_0, \mathbf{p}_0, \mathbf{r}_0)\}$ and

$$\Gamma_n^{t+1} = \max\{0, \Gamma_n^t - u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t)\}, \forall t \in \mathcal{T}. \quad (8)$$

Eq. (8) calculates the distance between the utility of different clients and individual rationality in each round.

- (*Budget queue*) To satisfy the budget balance, we first define the average budget as $\bar{B} = \frac{B}{T}$, and then use a queue to capture excess expenditures in each round as

$$\Lambda^{t+1} = \max\left\{0, \Lambda^t - \bar{B} + \sum_{n \in \mathcal{N}} (r_n^t - p_n^t)\right\}. \quad (9)$$

- (*Arbitrage queue*) To satisfy the non-arbitrage constraint, we further define the arbitrage queue. The length of the queue captures the potential free-riding probability as

$$\Theta_n^{t+1} = \Theta_n^t + \sum_{n \in \mathcal{N}} |p_n^t - M_t(\Delta v_n^t)|. \quad (10)$$

The above queues characterize the cumulative violation of constraints Eqs. (5b) - (5d). This means that by ensuring the stability of these queues, the solution to the optimization problem will adhere to these constraints over time, despite occasional violations in individual periods. Therefore, we use Lyapunov drift framework to approximate \mathbf{P}_1 and handle the constraints by adding penalty terms. Similar to (Han et al. 2023), we define a constant μ to indicate the importance of the target compared to queue stability. For each round of model training, we simultaneously optimize the download pricing p_n^t at round t and the upload pricing r_n^{t+1} from the previous round to achieve the optimal model performance at round $t+1$. Combining Eqs. (8) - (10), we approximate \mathbf{P}_1 as the following Lyapunov optimization:

$$\mathbf{P}_2 : \min_{\mathbf{p}, \mathbf{r}, \mathbf{s}} \mu [F_{t+1} - F_t] - \sum_{n \in \mathcal{N}} \Gamma_n^t u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t) + \sum_{n \in \mathcal{N}} \Theta_n^t |p_n^t - M_t(\Delta v_n^t)| + \Lambda^t \left[\sum_{n \in \mathcal{N}} (r_n^t - p_n^t) - \bar{B} \right], \forall t \in \mathcal{T}. \quad (11)$$

Algorithm 1: Online pricing and client selection

Input: $\mathcal{N}, \mathcal{T}, c, L, S, \varrho$, and ρ
Output: $\mathbf{p}, \mathbf{r}, \mathbf{s}$

```

1: Initialize  $\Lambda^t \leftarrow U, \Theta^t \leftarrow U, \Gamma_n^t \leftarrow U, n \in \mathcal{N}$ .
2: for  $t = 0 : T - 1$  do
3:   Obtain  $\mathbf{p}^t$  by solving Problem  $\mathbf{P}_2$  under  $\mathbf{s}^t = \mathbf{1}_N$ .
4:   for each  $n$  in  $\mathcal{N}$  do
5:     Compute  $s_n^t$  according to Eq. (12).
6:     Sample  $x_n^t$  following Bernoulli distribution ( $s_n^t$ ).
7:     if  $x_n^t = 1$  then
8:       Transmit the global model parameters  $\omega_t$  to
         client  $n$  and set  $v_n^t = t$ .
9:     else
10:      Set  $g_n^{t+1} = g_n^t$ .
11:    end if
12:  end for
13:  Obtain  $\mathbf{r}^t$  by solving Problem  $\mathbf{P}_2$ .
14: end for

```

In Problem \mathbf{P}_2 , our goal is to jointly optimize the global model performance and queue stability based on the weight coefficient μ . In each round, we set the pricing based on the real-time system status. Since \mathbf{P}_2 is a non-linear mixed-integer programming problem, it is difficult to solve directly.

Download Payment and Client Selection

To address this challenge, we employ an alternating optimization. In each round, we first optimize model pricing based on the real-time system state to ensure queue stability. Once the model pricing is determined, we further optimize the client selection strategy using a greedy algorithm to enhance the global model performance for the next round. To this end, we propose the online pricing and client selection algorithm. As shown in Algorithm 1, we initialize the queue length with a constant U (line 1). In round t , we use alternating optimization and a greedy algorithm to determine download pricing, upload pricing, and device selection.

Download payment. To ensure continuous improvement in queue stability, we temporarily assume $\mathbf{s}^t = \mathbf{1}_N$ to optimize \mathbf{p}^t . In round t , we sequentially optimize model configurations \mathbf{p}^t and \mathbf{r}^t to improve \mathbf{P}_2 (line 2), enhancing the expectation of satisfying long-term individual rationality, budget balance, and non-arbitrage constraints.

Client selection. Even with known \mathbf{p} , \mathbf{P}_2 remains a non-linear integer programming problem and is NP-hard. To address this issue, we introduce a greedy client selection algorithm. The heuristic of the algorithm is based on the principle that efficient clients contribute to model performance improvement. Specifically, we consider Holtling line Δv_n^t and cost c_n^t of each client, and greedily allocate selection probabilities. In line 5, we use the Softmax function to determine the selection probability for client n as

$$\mathbb{P}[s_n^t = 1] = \frac{\exp(-\Delta v_n^t - c_n^t)}{\sum_{m \in \mathcal{N}} \exp(-\Delta v_m^t - c_m^t)}. \quad (12)$$

Based on the determined value of s_n^t , Algorithm 1 decides the global model broadcasting decision and model aggrega-

tion in lines 6-10. At the end of round t , each client updates queue Λ^t , Θ^t , and Γ_n^t , $n \in \mathcal{N}$ for subsequent rounds.

The construction of Eq. (12) is based on the following reasons: 1) OPTION increases the selection probability for clients with lower update costs. Lower communication costs save the budget, allowing more flexibility for subsequent optimization and budget balancing. Thus, we prioritize selecting clients with lower update costs based on the current round's cost c . 2) OPTION assigns a higher selection probability to clients with older model versions to reduce the unfairness in client selection, addressing the non-IID problem. 3) OPTION uses the Softmax function to calculate selection probability, effectively handling unbounded input parameters, i.e., model version and communication cost.

Theoretical Analysis of OPTION

In this section, we analyze the violation of Eqs. (5b) - (5d), the closed-form solution of the optimal pricing, and the convergence and overhead of OPTION when solving \mathbf{P}_2 .

Assumption 2. Similar to (Wang et al. 2023), we first make the assumption about the upper bound of convergence error. We assume that the convergence error $K(\mathbf{p}_d, \mathbf{p}_u, t)$ has an upper bound D .

Theorem 2. *When Assumption 2 holds, there exists a bound on constraint violation of of Eq. (5b) - (5d) for round t as $O\left(\sqrt{\frac{1}{T^2} + \frac{1}{T} - \frac{1}{T}}\right)$, i.e, there exists a positive number k such that $\forall t \in [0, T - 1]$,*

$$\sum_{n \in \mathcal{N}} \Gamma_n^t + \Lambda_n^t + \Theta_n^t \leq k \left(\sqrt{\frac{1}{T^2} + \frac{1}{T} - \frac{1}{T}} \right). \quad (13)$$

Closed-form solution We now derive the closed-form solution for lines 3 and 5 in Algorithm 1 to improve the efficiency of solving \mathbf{P}_2 under linear utilities. For nonlinear $u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t)$, we can still obtain the pricing strategy through a linear search over \mathbf{P}_1 . We define the utility of client n in round t is

$$u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t) := \alpha (r_n^{t+1} - p_n^t) + x_n^t c_n^t, \quad (14)$$

where α is a positive coefficient. Then the probability of the client n uploading local parameters in round t is

$$\mathbb{P}[h_n^t = 1] := \frac{u_n^t(c_t, \mathbf{p}_t, \mathbf{r}_t) - u_{min}}{\beta}, \quad (15)$$

where β is the normalization coefficient, which can be computed by empirical formula $\beta = u_{max} - u_{min}$ ($u_{max} = \alpha r_n^{t+1} + x_n^t c_n^t$ and $u_{min} = \bar{B} - M_t(\Delta v_n^t)$). Eq. (15) implies that clients with higher expected utility are more likely to upload gradients.

Based on Eq. (14), the second derivative of Eq. (11) with respect to \mathbf{p}^t is positive. Thus, \mathbf{P}_2 is convex with respect to \mathbf{p}_n^t . Similarly, the objective and constraints are linear with respect to \mathbf{r}_n^t , indicating that the problem \mathbf{P}_2 is convex in these variables. Therefore, \mathbf{P}_2 is biconvex. Based on the above assumptions, we can directly calculate the client's reward and the download payment for the next round upon receiving the parameters, as stated in the following theorem.

Theorem 3. Given $s_n^{t+1} = 1$, the optimal solution to p_n^t is

$$p_n^t = \frac{1}{\mu\alpha^2 G_t} \left[\frac{\alpha}{\sum_{n \in \mathcal{N}} \alpha \Gamma_n^t - \Lambda^t + \Theta^t} \right]^2 (\alpha \Gamma_n^t - \Lambda^t), \quad (16)$$

and the optimal solution to r_n^t is

$$r_n^t = \sqrt{\frac{\mu p_n^t G_t}{\alpha \Gamma_n^t - \Lambda^t}} + \frac{c_n^t + M_t (\Delta v_n^t) - \bar{B}}{\alpha}. \quad (17)$$

where $G_t = E_t - 2\varepsilon\hat{\eta}^t E_t \sum_{n \in \mathcal{N}} s_n^t \mathbb{P}[h_n^t = 1] + \varepsilon(\hat{\eta}^t)^2 \rho^2 L S^2 E_t$.

Extension to non-convex objective. To further refine our predictive model, we now consider the case where Assumption 1 does not hold.. We introduce a weight $\kappa_n > 0$ for each client n , which can represent factors such as its dataset size ($\kappa_n = |\mathcal{D}_n|$), data quality, or other importance metrics. The E_t of a selection vector s_t is then determined by the cumulative weight of its members, which is adapted as:

$$E_t(s_t) = E_{\max} \left(1 - e^{-\gamma \sum_{n \in \mathcal{N}} \kappa_n s_n^t} \right), \quad (18)$$

where E_{\max} is the optimal performance when $\omega^t = \omega^*$. By setting $\kappa_n = |\mathcal{D}_n|$, the scheduler will naturally prioritize clients with larger datasets, aligning the selection strategy with the principles of methods like FedAvg.

Proposition 1. The computational complexity of *OPTION* is $O(TN^2)$ when E_t is computed via Eq. (7), but increases to $O(TN^3)$ under Eq. (18) due to client-specific weighting.

Performance Evaluation

Experiment Setup

Datasets and settings. We use three standard real-world datasets to evaluate performance of *OPTION*, which are widely used in AFL related research work: (1) MNIST (Deng 2012), (2) CIFAR-10 (Krizhevsky, Hinton et al. 2009), and (3) CINIC-10 (Darlow et al. 2018).

We configure an AFL platform with 80 clients and a central server ($N = 80$), assigning a total budget $B = 40000$ and a per-round update cost $\bar{B} = 200$. Following (Wang et al. 2023), we set a variable transmission cost c for each client, following a normal distribution $c_n^t \sim \mathcal{N}[1, 100]$. We determine the constant parameters in the convergence error through gradient updating and considered the influence of different penalty coefficients ($\mu = 0.01$ or $\mu = 0.001$). We evaluate the linear utility of each client and set $\alpha = 0.5$ and $\beta = 5$. More specific parameter settings are detailed in the Appendix. Experiments were performed on a machine with an NVIDIA GTX 1080 Ti GPU, an Intel Core i9-12900K CPU, and 32 GB RAM.

Baselines. We compare our approach with three baselines:

- **NCX-OPTION:** is an extended model based on the non-convex objective function in Eq. (18), where the pricing strategy is solved using block coordinate descent.

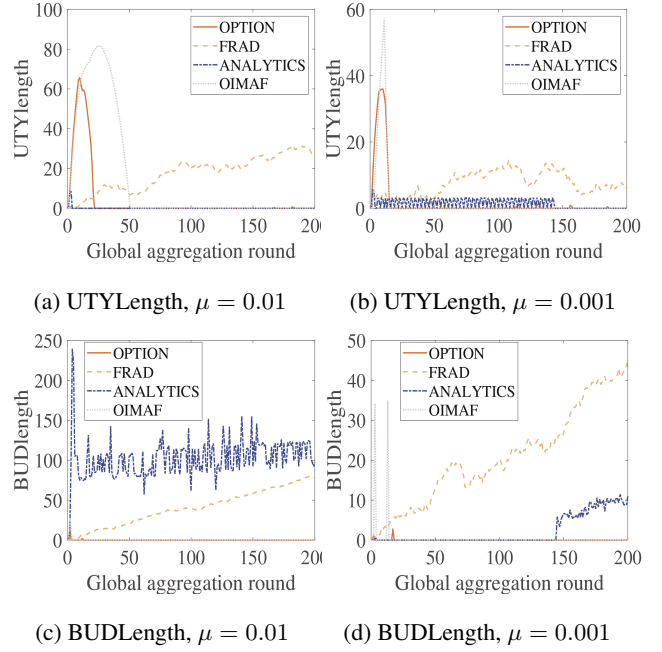


Figure 2: Queue length over aggregation rounds.

- **FRAD:** a mechanism for detecting free-riding attacks in SFL, leveraging client contributions and reputation to ensure system robustness (Wang et al. 2024).
- **ANALYTICS:** a method that balances payments and FL model accuracy loss by rewarding clients with analytic monetary incentives (Wang, Zheng, and Duan 2024).
- **OIMAF:** the only existing AFL incentive mechanism, which ensures individual rationality based on clients' contributions in real-time (Li et al. 2024).

Metrics. We use three types of metrics to compare *OPTION* with baselines: (1) The length of the utility queue (**UTYLength**) and the budget queue (**BUDLength**), representing the constraint violation degree of the optimization problem; (2) The number of devices not uploading updates (**Offline device**), which is determined by the number of clients that satisfy individual rationality and no arbitrage constraints; (3) **Average test accuracy (Avg-ACC)**, global test accuracy (GI-ACC) and training loss of AFL model.

Experimental Results

Queue length of constraints Figures 2(a) and 2(b) depict the UTYLength under four strategies. In Figures 2(a) and 2(b), the UTYLength of all strategies increases monotonically with μ . This is because $\frac{1}{\mu}$ can be seen as a punitive factor that constrains individual rationality. After training, the UTYLength of *OPTION* is equal to 0, while the UTYLength of *FRAD* is less ideal. This phenomenon occurs primarily because *FRAD*'s reward strategy relies on reputation theory and cannot adaptively adjust pricing based on historical data. In contrast, *OPTION* maximizes the long-term individual rationality through stable utility queue.

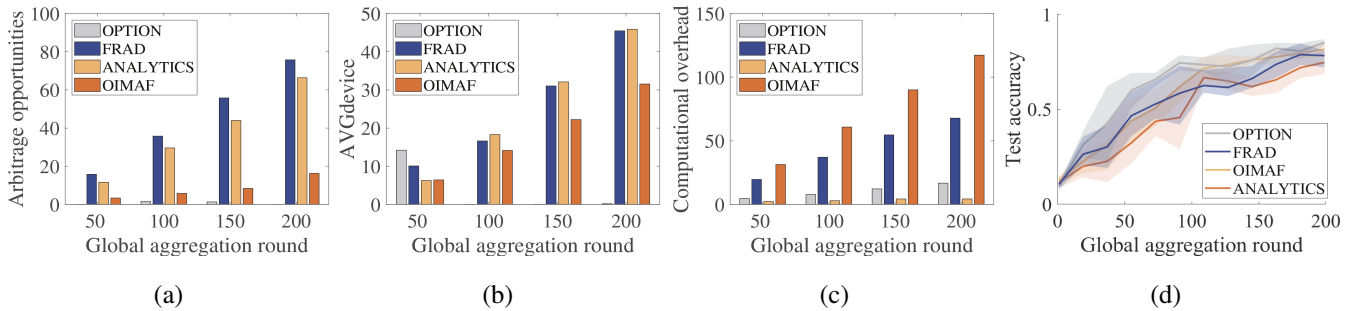


Figure 3: Effectiveness and overhead analysis: (a) sum of the violation of non-arbitrage constraint; (b) offline device; (c) global test accuracy on CIFAR-10; (d) cumulative calculation time of pricing strategy.

Pricing Methods	MNIST		CIFAR-10		CINIC-10	
	Gl-Acc. (%)	Avg-Acc. (%)	Gl-Acc. (%)	Avg-Acc. (%)	Gl-Acc. (%)	Avg-ACC. (%)
OPTION	93.76 ± 0.31	93.41 ± 0.25	85.25 ± 0.45	81.65 ± 0.52	40.95 ± 1.12	32.90 ± 1.34
NCX-OPTION	91.27 ± 0.42	91.43 ± 0.38	82.33 ± 0.61	75.73 ± 0.73	42.47 ± 0.98	33.77 ± 1.05
FRAD	90.93 ± 0.55	87.59 ± 0.67	78.35 ± 0.88	68.74 ± 0.91	24.49 ± 2.01	23.92 ± 2.62
OIMAF	92.42 ± 0.33	89.87 ± 0.41	81.08 ± 0.59	73.34 ± 0.77	36.92 ± 1.21	29.33 ± 1.45
ANALYTICS	91.90 ± 0.39	90.68 ± 0.44	72.21 ± 0.71	69.32 ± 0.85	31.63 ± 1.33	27.50 ± 1.51

Table 1: Model performance in AFL system (Accuracy in %).

Figures 2(c) and 2(d) provide illustrations of BUDLength. From Figure 2(c), we observe that both OPTION and OIMAF ensure the stability of the budget queue within a given training round, as the queue length does not infinitely change over time. However, in any round of the two graphs, the budget deficit of OIMAF is significantly higher than our OPTION. As shown in Figure 2(d), although OPTION’s queue length did not fully converge by the final round, the constraint violation remained below 5. In contrast, the BUDLength of baselines exceed at least 20.

Free-riding mitigation and overhead analysis Figure 3(a) plots the aggregate arbitrage queue length, which indicates the difficulty of free-riding. OPTION consistently maintains queue stability, reflecting its ability to create an arbitrage-free environment, whereas the baseline’s queue fluctuates, allowing for potential free-riding. Figures 3(b) depict the average number of offline devices. It is evident that update frequency of the client is relatively high in OPTION, with the next best online pricing improving by 23.97%. Consequently, OPTION exhibits the best global model accuracy and rapid convergence, as illustrated in Figure 3(c). Regarding computational overhead in Figure 3(d), OPTION simplifies the calculation process using a closed-form solution, thereby reducing the computation time of its strategy by 75.42% and 85.78% compared to FRAD and OIMAF, respectively.

Average model performance Table 1 presents the performance analysis of OPTION and its extended models based on the non-convex objective function in Eq. (18) on three real-world datasets. It is evident that OPTION outperforms the baselines on three datasets, with an improvement of at most 10.91% in global test accuracy. The main reason be-

hind this phenomenon is that OPTION uses queue stability to characterize accumulated constraint violations, which improves the update probability of clients and thus achieves higher average model performance. Table 1 shows the average performance of client models at round 200. Unlike the global model, this is influenced by the parameter download status and is more meaningful for AFL. Experimental data in Table 1 also show that OPTION improves the Avg-ACC by at least 3.01%. The improvement is attributed not only to the higher final accuracy but also to the inclusion of training performance in the optimization objective (Eq. (5a)), which benefits clients that did not fully participate in updates. Furthermore, the non-linear extension of OPTION achieved the highest average accuracy on the CINIC-10 dataset, which demonstrates the effectiveness of using Eq. (18) as the objective function for the alternating optimization.

Conclusion

In this paper, we have proposed a novel online pricing strategy tailored for AFL systems, named OPTION, to eliminate potential free-riding behavior during model aggregation rounds. OPTION required clients to pay a model download fee based on the Hotelling model, ensuring fairness within the system. Additionally, it compensated clients who successfully uploaded model updates under non-arbitrage constraints, balancing the individual utility and the training of local model. To optimize the pricing strategy and maximize model performance, an online optimization algorithm was introduced, which utilized the Lyapunov drift framework and a probabilistic sampling-based greedy algorithm. Extensive experimental results on three real-world datasets demonstrated that OPTION effectively reduced the number of free riders while improving model accuracy.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 62325304, 62372343, 62402352, 62562050, U22B2046, 62072411), in part by the Natural Science Foundation of Jiangsu Province of China (No. BK20253020), and in part by the Key Research and Development Program of Zhejiang Province (No. 2025C01055), and in part by the Open Fund of Key Laboratory of Social Computing and Cognitive Intelligence (Dalian University of Technology), Ministry of Education (No. SCCI2024TB02).

References

- Asheim, G. B.; Mitra, T.; and Tungodden, B. 2017. Sustainable recursive social welfare functions. In *The Economics of the Global Environment: Catastrophic Risks in Theory and Policy*, 165–190. Springer.
- Chen, Y.; Ning, Y.; Slawski, M.; and Rangwala, H. 2020. Asynchronous online federated learning for edge devices with non-iid data. In *2020 IEEE International Conference on Big Data (Big Data)*, 15–24. IEEE.
- Chen, Y.; Sun, X.; and Jin, Y. 2019. Communication-efficient federated deep learning with layerwise asynchronous model update and temporally weighted aggregation. *IEEE Transactions on Neural Networks and Learning Systems*, 31(10): 4229–4238.
- Darlow, L. N.; Crowley, E. J.; Antoniou, A.; and Storkey, A. J. 2018. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*.
- Deng, L. 2012. The mnist database of handwritten digit images for machine learning research [best of the web]. *Signal Processing Magazine*, 29(6): 141–142.
- Deng, Y.; Lyu, F.; Ren, J.; Chen, Y.-C.; Yang, P.; Zhou, Y.; and Zhang, Y. 2021. Fair: Quality-aware federated learning with precise user incentive and model aggregation. In *INFOCOM*, 1–10. IEEE.
- Ding, N.; Gao, L.; and Huang, J. 2023. Joint participation incentive and network pricing design for federated learning. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, 1–10. IEEE.
- Fackler, P. L.; and Goodwin, B. K. 2001. Chapter 17 Spatial price analysis. In *Marketing, Distribution and Consumers*, volume 1 of *Handbook of Agricultural Economics*, 971–1024. Elsevier.
- Fraboni, Y.; Vidal, R.; and Lorenzi, M. 2021. Free-rider attacks on model aggregation in federated learning. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 1846–1854. PMLR.
- Han, P.; Wang, S.; Jiao, Y.; and Huang, J. 2023. Federated Learning While Providing Model as a Service: Joint Training and Inference Optimization. *arXiv preprint arXiv:2312.12863*.
- Hu, R.; and Gong, Y. 2020. Trading data for learning: Incentive mechanism for on-device federated learning. In *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 1–6. IEEE.
- Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2): 1–210.
- Korhonen, J.; and Wang, Y. 2005. Effect of packet size on loss rate and delay in wireless links. In *IEEE Wireless Communications and Networking Conference, 2005*, volume 3, 1608–1613. IEEE.
- Krizhevsky, A.; Hinton, G.; et al. 2009. *Learning multiple layers of features from tiny images*. Toronto, ON, Canada.
- Li, G.; Cai, J.; He, C.; Zhang, X.; and Chen, H. 2024. Online incentive mechanism designs for asynchronous federated learning in edge computing. *IEEE Internet of Things Journal*, 11(5): 7787–7804.
- Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2: 429–450.
- Lin, J.; Du, M.; and Liu, J. 2019. Free-riders in federated learning: Attacks and defenses. *arXiv preprint arXiv:1911.12560*.
- Long, Q.; Liu, Y.; Tang, S.; and Peng, J. 2024. Effects of Payment Interoperability and Noninteroperability on Platform Competition. *IEEE Transactions on Engineering Management*, 71: 13918–13935.
- Murhekar, A.; Yuan, Z.; Ray Chaudhury, B.; Li, B.; and Mehta, R. 2023. Incentives in Federated Learning: Equilibria, Dynamics, and Mechanisms for Welfare Maximization. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 17811–17831. Curran Associates, Inc.
- Ottun, A.-R.; Mane, P. C.; Yin, Z.; Paul, S.; Liyanage, M.; Pridmore, J.; Ding, A. Y.; Sharma, R.; Nurmi, P.; and Flores, H. 2022. Social-aware federated learning: Challenges and opportunities in collaborative data training. *Internet Computing*, 27(2): 36–44.
- Shi, Z.; Zhang, L.; Yao, Z.; Lyu, L.; Chen, C.; Wang, L.; Wang, J.; and Li, X.-Y. 2024. Fedfaim: A model performance-based fair incentive mechanism for federated learning. *IEEE Transactions on Big Data*, 10(6): 1038–1050.
- Su, L.; Zhou, R.; Wang, N.; Fang, G.; and Li, Z. 2022. An Online Learning Approach for Client Selection in Federated Edge Learning under Budget Constraint. In *Proceedings of the 51st International Conference on Parallel Processing*, 1–11.
- Tang, M.; Peng, F.; and Wong, V. W. 2024. A blockchain-empowered incentive mechanism for cross-silo federated learning. *IEEE Transactions on Mobile Computing*, 23(10): 9240–9253.
- Tang, M.; and Wong, V. W. 2021. An Incentive Mechanism for Cross-Silo Federated Learning: A Public Goods Perspective. In *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 1–10.

Wang, B.; Li, H.; Liu, X.; and Guo, Y. 2024. Frad: Free-rider attacks detection mechanism for federated learning in aiot. *IEEE Internet of Things Journal*, 11(3): 4377 – 4388.

Wang, S.; Perazzone, J.; Ji, M.; and Chan, K. S. 2023. Federated learning with flexible control. In *INFOCOM*, 1–10. IEEE.

Wang, X.; Zheng, S.; and Duan, L. 2024. Dynamic pricing for client recruitment in federated learning. *IEEE/ACM Transactions on Networking*, 32(2): 1273 – 1286.

Wang, Y.; Su, Z.; Luan, T. H.; Li, R.; and Zhang, K. 2021. Federated learning with fair incentives and robust aggregation for UAV-aided crowdsensing. *IEEE Transactions on Network Science and Engineering*, 9(5): 3179–3196.

Wang, Z.; Zhang, Z.; Tian, Y.; Yang, Q.; Shan, H.; Wang, W.; and Quek, T. Q. 2022. Asynchronous federated learning over wireless communication networks. *IEEE Transactions on Wireless Communications*, 21(9): 6961–6978.

Xu, C.; Qu, Y.; Xiang, Y.; and Gao, L. 2023. Asynchronous federated learning on heterogeneous devices: A survey. *Computer Science Review*, 50: 100595.

Xu, J.; Xu, Z.; Lin, J.; and She, W. 2022. Double security guarantee: Protecting user privacy and model security in qos prediction. In *2022 IEEE International Conference on Services Computing (SCC)*, 140–145. IEEE.

Yu, H.; Gao, L.; Li, Z.; Wang, X.; and Hossain, E. 2010. Pricing for uplink power control in cognitive radio networks. *IEEE Transactions on Vehicular Technology*, 59(4): 1769–1778.

Zhan, Y.; Li, P.; Guo, S.; and Qu, Z. 2021. Incentive mechanism design for federated learning: Challenges and opportunities. *IEEE network*, 35(4): 310–317.

Zhang, J.; Ge, C.; Hu, F.; and Chen, B. 2021. RobustFL: Robust federated learning against poisoning attacks in industrial IoT systems. *IEEE Transactions on Industrial Informatics*, 18(9): 6388–6397.

Zhang, N.; Ma, Q.; and Chen, X. 2022. Enabling long-term cooperation in cross-silo federated learning: A repeated game perspective. *IEEE Transactions on Mobile Computing*, 22(7): 3910–3924.

Zhang, S. Q.; Lin, J.; and Zhang, Q. 2022. A multi-agent reinforcement learning approach for efficient client selection in federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 9091–9099.

Zheng, S.; Meng, Q.; Wang, T.; Chen, W.; Yu, N.; Ma, Z.-M.; and Liu, T.-Y. 2017. Asynchronous stochastic gradient descent with delay compensation. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, 4120–4129. PMLR.

Zhu, Y.; Liu, Z.; Wang, P.; and Du, C. 2023. A dynamic incentive and reputation mechanism for energy-efficient federated learning in 6g. *Digital Communications and Networks*, 9(4): 817–826.