Submission and Formatting Instructions for International Conference on Machine Learning (ICML 2025)

Anonymous Authors[1]

Abstract

This document provides a basic paper template and submission guidelines. Abstracts must be a single paragraph, ideally between 4–6 sentences long. Gross violations will trigger corrections at the camera-ready phase.

1. Introduction

With the development of Internet of Things(IoT), many end devices are generating plenty of data each day. Due to the growing storage and computing power, it becomes more appealing to store data locally and push more compution function to them. It motivates the application of federated learning, which supports local storage and local training on each device without violating their privacy.

A federated learning system comprises a parameter server and a large number of clients(i.e. end devices). The server maintains a global model while each client owns local statistic. During each iteration, clients download global model from the central server and makes local updates on their private data. Then the server will aggregate the updated models from the clients and generate a new global model. The training process will terminate when the accuracy of global model has reached the preset threshold. During the process, since clients share not their private data but the trained local model with the server, the federated learning system can protect clients' privacy efficiently.

Despite various advantages in enabling edge computing while protecting data privacy, it still faces two bottlenecks: 1) *Staleness of Data:* In most existing works, clients are assumed to hold a static dataset and use the same dataset to train local model over the time horizon. But in reality, there are many highly time-sensitive tasks with streaming data. That is, data are generated continuously and data valuation largely depends on timeliness of data (Xiao et al., 2023). Under this circumstance, outdated data used for training may deteriorate model parameters and reduce model service quality. 2) *Incentive Mechanism:* Incentive mechanism is attached great importance for stimulting clients to participate the training task. That's because clients consume various resources including computing capacity, communication bandwidth and battery charge while training. Besides, collecting fresh data frequently to meet the requirement of highly time-sensitive tasks also cause extra cost. Considering huge resource expenditure, clients may be not reluctant to provide fresh data or even reject to participate in federated learning tasks without economic compensation.

Efforts have been made in incentive mechanism design ranging from game theory (Lim et al., 2020) (Zhang et al., 2022) (Huang et al., 2024) to auction theory (Jiao et al., 2020) and contract theory (Kang et al., 2019) (Ding et al., 2020). However, most of these studies don't take into consideration the negative impact of outdated data on model performance. A few works have studied AoI-concerned incentive mechanisms against mobile crowdsensing (Xiao et al., 2023) (Wang et al., 2021). But they can't be applied in the context of federated learning directly. In the federated learning field, some papers discuss the AoI-concerned incentive mechanism, focusing on controlling dynamic prices or update frequency (Wang & Duan, 2019) (Wu et al., 2023). But none of them explore to manipulate local data update by decaying stale data and collecting new data simultaneously considering value and volume of present data.

Motivated by the above considerations, we propose an incentive mechanism in federated learning aware of data staleness, called FedStream. Different from previous works, this system allows that the server coordinates with clients to manipulate their local data update in a novel scheme. Specifically, the server controls two variables: payment intented to encourage clients to collect new data and conservation rate intented to force clients to abandon outdated data. It targets to minimize model accuracy loss with plenty of fresh data and as low monetary payment as possible. For a particular client, it controls the volume of new data collected from local data stream dynamically according to the server's strategy. Before each communication round, it updates loal dataset by abandoning parts of outdated data as the server

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

required and collecting some new data according to its own optimal strategy, then followed by local training. The objective of a client is to maximize the balance between payment and various costs.

There are three key challenges in FedStream as follows. First, the server faces a two-variable optimization problem where greater reward incurs more fresh data for model training but increases monetary payment, while smaller conservation rate forces to abandon more stale data but decreases data volume at the same time. It's critical to minimize the server's cost by elaborately determining both of them. Besides, the client faces a long-term optimization problem with dynamic constraint of update strategy. How to derive the optimal strategy for a client to maximize its utility is of significance and challenging. Second, capturing the converged model performance before training is an important step to determine the optimal strategy of server. Intuitively, global model performance can be impacted by data volume and data staleness simultaneously. Specially, more fresh data will lead to better model performance, particularly in highly time-sensitive tasks. However, there is lack of quantitative relationship between global model performance and the freshness of data. Third, to derive clients' optimal strategies, they need to estimate their expected benefit before training, which depends on reward by the server and respective contributed share. However, in many real-world scenarios, the contributed share is unknown since clients cannot communicate with each other. Unknown information makes it challenging to make decision on client side.

To overcome the above challenges, we first derive the convergence upper bound of FedStream for the server, which reveals the relationship between converged model performance and data staleness. In addition, we introduce a mean-field term, which is similar to the method adopted in [?], to estimate the unknown information for clients. Based on the above two approaches, the optimization problems on both server and client side can be constructed respectively. Then we use a Stackelberg game to model the interaction between both sides, where the server acts as a leader and clients are regarded as followers. Lastly, with the aid of backward reduction approach, we derive the optimal solution for this Stackelberg game.

The main contributions in this paper are summarized as follows:

- We propose a new federated learning system aware of data staleness with the local data update scheme at the core. To fit the proposed update scheme, we define a novel concept of DoS to measure the degree of data staleness. Based on the definition, we conduct convergence analysis and secure the upper bound of converged upper bound for FedStream.
- On the basis of convergence upper bound, we construct the optimization problems for both the server and clients, where the server controls reward and conservation rate to minimize total cost, and clients control volume of new data to maximize their balance.
- We model the interaction between the server and clients as a Stackelberg game. Under the backward reduction, we derive clients' optimal strategy with Hamilton equation. Based on this ,we dervie the server's strategy by adopting a search algorithm.
- We carry out extensive experiments on two dataset: MNIST and FMNIST, to illustrate the extraordinary performance of FedStream.

The remainder of the paper is organized as follows: In Section 2, we introduce related work. We provide problem formulation and corresponding convergence analysis in Section 3. System model and methodology are demonstrated in Section 4 and 5 respectively. Then we conduct experiments in Section 6. The paper is concluded in Section 7. Proofs of theorems and remarks are moved to the Appendix.

2. Related Work

2.1. Incentive Mechanism

Incentive mechanism designed for federated learning has been widely investigated in previous works. They can be assorted into three catagories: (1) *game theory:* (Lim et al., 2020) proposes a hierarchical incentive mechanism based on coalitional game theory approach, where multiple workers can form various federations. (Zhang et al., 2022) builds a incentive mechanism utilizing repeated game theory to enable long-term cooperation among participants in cross-silo federated learning. (Huang et al., 2024) designs a novel incentive framework based on Stackelberg game to model the collaboration behaviour among server and clients in federated learning with difference privacy. (2) *auction theory:* (Jiao et al., 2020) designs two auction mechanisms for the federated learning platform to maximize the social welfare of the federated learning services market. (3) *contract theory:* (Kang et al., 2019) proposes an effective incentive mechanism combining reputation with contract theory to motivate high-reputation mobile devices with high-quality data to participate in model learning. (Ding et al., 2020) presents an analytical study on the server's optimal incentive mechanism design by contract theory, in the presence of users' multi-dimensional private information. However, the above studies don't take into consideration the negative impact of outdated data on model performance and they can't be applied in highly time-sensitive tasks.

2.2. Data Staleness Optimization

Many efforts have been devoted to data staleness optimization. For example, (Tripathi & Modiano, 2021) consider the problem of minimizing age of information in general single-hop and multihop wireless networks. (Fang et al., 2021) devises a joint preprocessing and transmission policy to minimize the average AoI and the energy consumption at the IoT device. Only a few works among them study the AoI optimization with economic consideration. For example, (Xiao et al., 2023) investigates the incentive mechanism design in MCS systems that take the freshness of collected data and social benefits into concerns. (Wang et al., 2021) considers a general multi-period status acquisition system, aiming to maximize the aggregate social welfare and ensure the platform freshness. However, they can't be applied in the context of federated learning. In the federated learning field, (Wang & Duan, 2019) proposes dynamic pricing for the server to offer age-dependent monetary returns and encourages clients to sample information at different rates over time. (Wu et al., 2023) aims to minimize the loss of global model for FL with a limited budget by determining a client selection strategy under time-sensitive scenarios. But none of them explore to manipulate local data update by decaying stale data and collecting new data simultaneously considering value and volume of present data.

3. Problem Formulation

3.1. Federated Learning with Data Stream

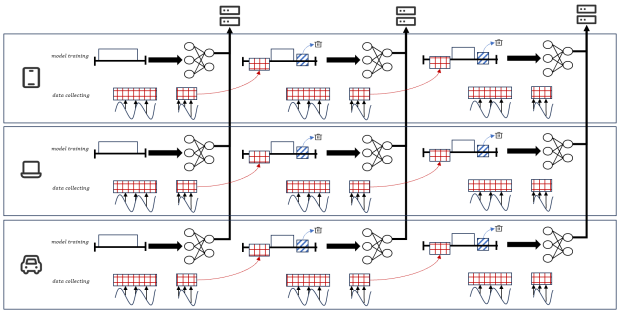


Figure 1. Framework

We assume that there is a central server and N clients in the federated learning system and they are arranged to conduct T rounds in the training task. Different from the static local dataset in previous works, each client k has a local data stream $\mathcal{D}_k$ which generates new data points randomly and dynamically over the time horizon. In addition, due to the storage limit, each client maintains a buffer to store data points sampled from its data stream $\mathcal{D}_k$ and used for FL model training. We denote the stored data points in the buffer of client k at round t as $\mathcal{D}_k(t)$ and $|\mathcal{D}_k(t)| = D_k(t)$. During each round, the buffer of client k can be updated by decaying part of stale data points and sampling new ones from its local data stream, and then used for local model training. Therefore, the updating strategy of client k at round t can be modeled as

$$D_k(t+1) = \theta D_k(t) + \Delta_k(t), \tag{1}$$

where $\theta \in [0, 1]$ is the conservation rate of stale data points and it effects the degree of staleness. Besides, $\Delta_k(t)$ is the amount of newly collected data points during round t.

Note that considering the uncertainty of data stream, data collection cannot be completed immediately, which means the traditional "collecting-training" paradigm may lead to severe time latency for federated learning task. To improve the task efficiency, data collection is designed to proceed in parallel with model training in our framework, and the newly collected data points during this round will be used to update buffer and train model for the next round, as depicted in Fig 1.

Then, the loss function of client k based on global model $w(t)$ using stored data points $\mathcal{D}_k(t)$ at round t can be represented as

$$F_k(w(t), \mathcal{D}_k(t)) = \frac{1}{D_k(t)} \sum_{j=1}^{D_k(t)} f(w(t), x_k^j), \tag{2}$$

where $f(w(t), x_k^j)$ is the loss function of each data point $\{x_k^j, y_k^j\} \in \mathcal{D}_k(t)$. Then client k updates its local model by

$$w_k(t+1) = w(t) - \eta \nabla F_k(w(t), \mathcal{D}_k(t)), \tag{3}$$

where η is the learning rate and $\nabla F_k(w(t), \mathcal{D}_k(t))$ is the loss gradient of client k at round t. Until each client completes its local update and sends $w_k(t+1)$ to the central server, the central server will aggregate them by

$$w(t+1) = \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} w_k(t+1), \tag{4}$$

where $\mathcal{D}(t) = \cup_{k=1}^{N} \mathcal{D}_k(t)$ and $D(t) = \sum_{k=1}^{N} D_k(t)$. Subsequently, the central server launches the new global model $w(t+1)$ to each client for the next round's training. The global loss function at round t can be represented as

$$F(w(t), \mathcal{D}(t)) = \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} F_k(w(t), \mathcal{D}_k(t)). \tag{5}$$

The ultimate goal is to find optimal parameters $w(t), t \in [0, \cdots, T-1]$ to minimize the global loss function in each round t, which can be expressed as

$$\arg\min_{w(t)} F(w(t), \mathcal{D}(t)) = \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} F_k(w_k(t), \mathcal{D}_k(t)). \tag{6}$$

3.2. Convergence Analysis for Federated Learning with Data Stream

Model performance convergence analysis for FedStream is provided in this section. In practice, it's difficult to measure the model performance of FedStream directly due to data dynamity, data heterogeneity and so on. To overcome the challenge, previous works have leveraged the convergence bound of the expected difference between the training loss and the optimal loss to capture the model performance. In this section, we adopt this method and provide a convergence upperbound for federated learning with data stream, considering the impact of data quantity, data heterogeneity and data staleness on final model performance. Before that, the concept of degree of staleness(DoS) for data points is introduced in Definition 3.1.

Definition 3.1. We denote $S_k(t)$ as the degree of staleness(DoS) for client k's data samples at round t. In FedStream, the recursive definition is provided as

$$S_k(t) = \begin{cases} \frac{\theta D_k(t-1)}{D_k(t)}(S_k(t-1)+1) + \frac{\Delta_k(t-1)}{D_k(t)}, & t > 0; \\ 1, & t = 0, \end{cases} \tag{7}$$

where $S_k(t)$ is the weighted sum of previous data points' DoS and the new data points' DoS. The previous data points' DoS should be updated by adding to 1 once stepping into the next time slot, while the new data points' DoS is set to be 1.

Remark 3.2. $S_k(t)$ increases with conservation rate θ and decreases with increment $\Delta_k(t)$, which reflects the fact that less stale data samples and more fresh ones contribute to DoS reduction.

Remark 3.3. The general formula of $S_k(t)$ can be further provided as

$$S_k(t) = \sum_{\tau=0}^{t} \frac{\theta^{t-\tau} D_k(\tau)}{D_k(t)}. \tag{8}$$

Next, we introduce some assumptions on local loss function $F_k(w)$ which have been widely used in previous works before the presentation of convergence analysis.

Assumption 3.4. $F_k(w)$ is ρ-Lipschitz, i.e., $F_k(w) - F_k(w') \leq \rho \|w - w'\|_2$.

Assumption 3.5. $F_k(w)$ is β-Lipschitz smooth, i.e., $\|\nabla F_k(w) - \nabla F_k(w')\| \leq \beta \|w - w'\|_2$.

Assumption 3.6. $F_k(w)$ is μ-strong contex, i.e., $F_k(w)$ satisfies $F_k(w) - F_k(w^*) \leq \frac{1}{2\mu}\|\nabla F_k(w)\|_2^2$.

Assumption 3.7. The stochastic gradient is unbiased and variance-bound, that is, $E[\nabla F_k(w(t)|D_k(t))] = \nabla F_k(w(t)|\mathcal{D}_k)$ and $E\|\nabla F_k(w(t)|D_k(t)) - \nabla F_k(w(t)|\mathcal{D}_k)\|^2 \leq \frac{\psi^2}{D_k(t)}$.

Assumption 3.8. The data heterogeneity is bounded, i.e, $\|\nabla F_k(w(t)|\mathcal{D}_k) - \nabla F(w(t)|D(t)\| \leq \delta_{k,t}$.

Assumption 3.9. The expected square norm of stochastic gradient is bounded, i.e., $E\|\nabla F_k(w(t)|D_k(t))\|^2 \leq G_k^2 + S_k(t)\sigma^2$, where σ is the sensitivity coefficient of client k to the freshness of data samples.

Theorem 3.10. *Under Assumptions 3.4-3.9, with $\eta \leq \frac{1}{2\beta}$, $\xi \geq 2$, the convergence upperbound after T rounds can be formulated as*

$$\begin{aligned} &E[F(w(T)|D(T)) - F(w^*)] \\ &\leq \underbrace{\kappa_1^T E[F(w(0)|D(0)) - F(w^*)]}_{(1)} \\ &+ \sum_{t=0}^{T-1} \kappa_1^{T-1-t} \Bigg[\underbrace{\kappa_2 \frac{N\psi^2}{D(t)}}_{(2)} + \underbrace{\kappa_3 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} S_k(t)\sigma^2}_{(3)} + \underbrace{\kappa_4 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} \overline{\delta_k}^2}_{(4)} + \end{aligned} \tag{9}$$

where $\Omega_t = F(w(t+1)|D(t+1)) - F(w(t+1)|D(t))$, *and* $\overline{\delta_k} \triangleq \max_{0 \leq t \leq T-1} \delta_{k,t}$ *with* $\delta_{k,t} = \|\nabla F_k(w(t)|\mathcal{D}_k) - F(w(t)|D(t))\|$. *In addition,* $\kappa_1 = 1 + 4\mu\beta\eta^2 - 2\mu\eta$, $\kappa_2 = 2\beta\eta^2$, $\kappa_3 = \beta\eta^2$ *and* $\kappa_4 = 2\xi\beta\eta^2 + \frac{1}{2}\xi\eta$.

The detailed proof is provided in Appendix A.1.

Remark 3.11. In the fifth term of Theorem 3.10, Ω_t captures the expected difference of global loss function based on the same global model $w(t)$ between total stored data points at current round $\mathcal{D}(t)$ and at previous round $\mathcal{D}(t-1)$. Note that Ω_t is time-varying and it measures the influence of data dynamity has on model performance, and describes the generalization of global model towards new data points. The more fluctuating data dynamity is, the larger the convergence bound is and the worse the global model performs.

Remark 3.12. In the fourth term of Theorem 3.10, $\delta_{k,t}$ captures the gradient gap between the global loss function and the local loss function of client k at the round t. It measures the degree of data heterogeneity for client k at round t. Note that due to data dynamity, $\delta_{k,t}$ is time-varying, too. Thus, we define $\overline{\delta_k}$ to represent the upperbound of $\delta_{k,t}$ across the time slots $t \in [0, T-1]$. The fourth term describes the impact of global data heterogeneity degree on model performance. According to this term, within single round t, the convergence bound would be reduced if a client with smaller $\overline{\delta_k}$ stores more data points for training than others.

4. System Model

In this section, we design an incentive mechanism for federated learning with data stream, called FedStream. This

section starts with the workflow of FedStream in 4.1. Then the cost function of central server and the utility function of clients will be demonstrated in 4.2 and 4.3 respectively. Ultimately, we model this federated learning system interaction with a two-stage Stackelberg Game.

4.1. Overview

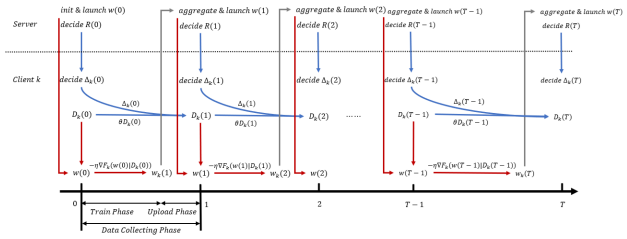


Figure 2. Timeline

Assume that there is a central server and N clients, each client holds a local data stream and a buffer with stored data points sampled from its local data stream. The central server needs to conduct T rounds in its FL task. At each round t, the central server launches the global model $w(t)$ to all clients. Considering clients may be reluctant to collect data and train model actively due to various expenditure, the central server also provides some payment R to them.

After downloading global model from the central server, each client begins to train model $w(t)$ using data points stored in buffer, i.e. $\mathcal{D}_k(t)$. Moreover, stimulated by the payment R, clients attempt to sample some new data points from their local data stream. The amount of newly sampled data points $\Delta_k(t)$ is determined by themselves based on the central server's payment R and other clients' decisions. It deserves to be noted repeatedly that model training and data sampling proceed in parallel and that the newly sampled data points are prepared for the next round's buffer updating and model training.

When all clients upload their local models to the central server, it aggregates their local model and obtains a new global model $w(t+1)$. In round $t+1$, the central server will launch new global model $w(t+1)$ and payment R. For each client k, it adds new data points sampled during last round, i.e., $\Delta_k(t)$ into buffer and generates $\mathcal{D}_k(t+1)$ at first, then they use $\mathcal{D}_k(t+1)$ to train $w(t+1)$. Meanwhile, they decide the amount of newly sampled data points $\Delta_k(t+1)$ and sample for round $t+2$.

For ease of demonstration, we define $\boldsymbol{D_k} = \{D_k(0), D_k(1), \cdots, D_k(T-1)\}$ and it's the decision sequence of client k's buffer size during the training period. Furthermore, we define $\boldsymbol{D} = \{\boldsymbol{D_1}, \boldsymbol{D_2}, \cdots, \boldsymbol{D_N}\}$ as the composition of all clients' buffer size sequence throughout the training. In our framework, both R and $\boldsymbol{D}$ need to be pitched precisely. In the next section, we will construct a rigorous model to unveil the behaviour of both the central server and all clients. Then based on this model, we try to explore the optimal payment R for central server and optimal buffer vector $\boldsymbol{D_k}$ for each client $k \in \mathcal{N}$.

4.2. Cost Function of Central Server

From the prespective of the central server, on the one hand, it needs to stimulate clients to participate in model training to achieve a better model performance. On the other hand, it needs to control the payment to clients at the same time. Hence, the expenditure of the central server comprises two parts: accuracy loss of model and payment to clients. In reality, it's extremely hard to obtain accurate form of model accuracy loss directly. Yet we can approximate it with the convergence upperbound provided in Theorem 1. Note that (1) and (5) in Theorem 1 cannot be controlled by the central server, hence we extract the valid part of (2), (3) and (4) to the cost function. In addition, total payment during the training period offered to clients can be represented as TR. Then, the cost function of the central server can be constructed as the sum of accuracy loss of model and total payment to clients as

$$U(\boldsymbol{D}, R, \theta) = \sum_{t=0}^{T-1} \left[(1-\gamma)\kappa_1^{T-1-t} \left(\kappa_2 \frac{N\psi^2}{D(t)} + \kappa_3 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} S_k(t)\sigma^2 + \kappa \right. \right. \tag{10}$$

where $\gamma \in [0,1]$ is a factor to strike the balance between model performance and cost payment. When γ trends to 1, the central server pays more attention to enhancing the model performance. Otherwise, the central server prefers to save cost payment. By observing (10), the cost function of central server is dominated by $\boldsymbol{D}$ and R simultaneously. In the cost function, a greater payment R encourages clients to sample more data points and maintains a greater buffer for the next round, which helps to reduce accuracy loss and enhance model performance. But the total payment to clients increase at the same time. There's a trade off in the cost function. Hence, the optimization problem on the central server's side can be modeled as

$$\min_{R,\theta} U(\boldsymbol{D}, R, \theta), \tag{11}$$

where the central server's target is to minimize its cost function $U(\boldsymbol{D}, R, \theta)$ by navigating the optimal payment R and decay factor θ.

4.3. Utility Function of Clients

For a client who participates into the federated learning task, it needs to spend various types of resource for data sampling and model training. Here, we use $\alpha_k \Delta_k(t)^2$ and $\beta_k D_k(t)^2$ to depict the expenditure for data collecting and training, respectively, where α_k is unit cost for data collecting and

β_k is the counterpart for model training. It should be noted that we choose the quadratic form of $\Delta_k(t)^2$ and $D_k(t)^2$ to reflect the fact that a client's sampling and training expenditure should convexly increase with its increment and scale, respectively. Therefore, the cost can be expressed as

$$C_k(t) = \alpha_k \Delta_k(t)^2 + \beta_k D_k(t)^2. \tag{12}$$

Meanwhile, a client can obtain some payment. The payment received by a client is proportional to its contribution to model performance and inversely proportional to others' contribution. In FedStream, a client's contribution should consider three aspects: privacy budget ε_k, buffer size $D_k(t)$ and data heterogeneity $\overline{\delta_k}$. Specially, it is proportional to $D_k(t)$, as well as inversely proportional to $\overline{\delta_k}$. Thus, the payment received by a client can be formulated as

$$P_k(t) = \frac{\overline{\delta_k}^{-1} D_k(t)}{\sum_{i=1}^{N} \overline{\delta_i}^{-1} D_i(t)} R, \tag{13}$$

where $\sum_{i=1}^{N} \overline{\delta_i}^{-1} D_i(t)$ can be considered as the total contribution of all clients make to model performance. Then, the utility function of client k is constructed as the sum of gap between payment and cost over the time horizon:

$$U_k(\boldsymbol{D_k}, \boldsymbol{D_{-k}}, R, \theta) = \sum_{t=0}^{T-1} \left(P_k(t) - C_k(t) \right), \tag{14}$$

where $\boldsymbol{D_{-k}} = \boldsymbol{D} \setminus \boldsymbol{D_k}$ and it's the set of all clients' buffer size except for client k. The utility of client k is dominated by its own buffer size vector $\boldsymbol{D_k}$, other clients' buffer size vector $\boldsymbol{D_{-k}}$ and the central server's payment R. For client k, maintaining a great buffer helps to secure more payment from the central server, but incurs more expenditure in data sampling and model training at the same time. There's also a trade off in the utility function. Hence, the optimization problem on the client's side can be modeled as

$$\begin{aligned} &\max_{\substack{\boldsymbol{\Delta_k}=[\Delta_k(t)] \\ t\in[0,\cdots,T-1]}} U_k(\boldsymbol{D_k}, \boldsymbol{D_{-k}}, R, \theta), \\ &\text{s.t.} D_k(t+1) = \theta D_k(t) + \Delta_k(t), \end{aligned} \tag{15}$$

where client k's target is to maximize its utility function $U_k(\boldsymbol{D_k}, \boldsymbol{D_{-k}}, R, \theta)$ by navigating the optimal buffer size vector $\boldsymbol{D_k}$ under the given constraint.

4.4. Stackelberg Game Formulation

Based on the discussions in 4.2 and 4.3, we use a two-stage Stackelberg Game to model the interaction between central server and clients in FedStream as

$$\begin{aligned} &\text{Stage I}: \min_{R,\theta} U(\boldsymbol{D}, R, \theta), \\ &\text{Stage II}: \max_{\substack{\boldsymbol{\Delta_k}=[\Delta_k(t)] \\ t\in[0,\cdots,T-1]}} U_k(\boldsymbol{D_k}, \boldsymbol{D_{-k}}, R, \theta), \\ &\text{s.t.} D_k(t+1) = \theta D_k(t) + \Delta_k(t). \end{aligned} \tag{16}$$

Before each round, the central server launches an optimal payment R to clients in Stage I. Then in Stage II, according to R announced by the central server, each client $k \in \mathcal{N}$ decides the newly sampled data point $\Delta_k(t)$. In this game, the server targets to minimize its cost function while each client attempts to maximize its utility function. Thus, the optimal solution is feasible if and only if it's mutually optimal for both central server and clients.

5. Methodology

5.1. Challenges

There are some challenges when solving the problem directly. First, **online uncertainty for stage I**. Noting that in FedStream, all data arrive dynamically. The influence of data dynamity can be reflected in the convergence upper-bound of FedStream with unstable value of hyperparameters including Ω_t and $\Upsilon_{k,t}$ across the time. Hence, they need to be observed and measured at the beginning of each round in a real time way, which means offline algorithm is not feasible when dealing with the optimization problem in stage I. Second, **correlated decisions for stage II**. Due to the "decay and accumulation" mechanism, the buffer size of a client is not independent across the time slots. Decisions in history will affect future decisions. For example, $D_k(t)$ of client k in round t updates on the basis of $D_k(t-1)$ and $\Delta_k(t-1)$ in round $t-1$. In this condition, we need to take the relationship between adjactory times slot into account, rather than treating the optimization problem in stage II as a single slot optimization problem. Third, **unknown information for stage II**. By observing the objective function in stage II, it contains the total contribution of $\sum_{i=1}^{N} \delta_{i,t}^{-1} D_i(t)$, which is the sum of all clients' contribution in round t. This term is needed when deriving each client's optimal increment $\Delta_k(t)$ and buffer size $D_k(t)$ through the objective function. But in reality, the total contribution can only be secured after each client has decided its own increment and contribution to the model performance. In general, there's an endless loop.

To deal with these challenges, we introduce a mean field term to estimate the unknown part in the objective function of each client at first. Then in stage II, we apply the Hamilton equation to handle the problem of correlated decisions. In addition, in response to the problem of online uncertainty, we design an online decision-making algorithm with real-time hyperparameters estimation for stage I.

5.2. Introduction of Mean Field Term

Before dealing with the above optimization problem, we first introduce a mean field term to estimate the total contribution of $\sum_{i=1}^{N} \overline{\delta_i}^{-1} D_i(t)$ at round t. Noting that this term is the sum of each client's contribution. The change of client i 's

buffer size $D_i(t)$ will affects its contribution, thereby affects the total contribution as a whole, which in turn affects the objective function of client i as well as its choice of $\Delta_i(t)$ and $D_i(t)$ at round t. There's a close loop. But in reality, the total contribution keeps unknown to each client before they decide their newly sampled data size, which makes it challenging to solve the optimization problem on the clients' side. Thus, we introduce a mean field term $\phi(t)$ to estimate the total contribution of $\sum_{i=1}^{N} \overline{\delta_i}^{-1} D_i(t)$ at round t, which can be formulated as

$$\phi(t) = \sum_{i=1}^{N} \overline{\delta_i}^{-1} D_i(t). \tag{17}$$

We define $\boldsymbol{\phi} = [\phi(0), \phi(1), \cdots, \phi(T-1)]$ and it's used to estimate the total contribution in distinct rounds. From a mathematics perspective, $\phi(t)$ is a given function no matter how $D_i(t)$ changes. Based on this, the payment received by client k can be rewritten as

$$P_k(t) = \frac{\overline{\delta_k}^{-1} D_k(t)}{\phi(t)} R. \tag{18}$$

The optimization problem of client k can be reformulated as

$$\max_{\substack{\boldsymbol{\Delta_k}=[\Delta_k(t)] \\ t\in[0,\cdots,T-1]}} \sum_{t=0}^{T-1} \left(\frac{\overline{\delta_k}^{-1} D_k(t)}{\phi(t)} R - \alpha_k \Delta_k(t)^2 - \beta_k D_k(t)^2 \right), \quad \text{s.t.} D_k(t+1) = \theta D_k(t) + \Delta_k(t). \tag{19}$$

5.3. Optimal Strategy of Clients in Stage II

In this stage, provided the payment R launched by the central server, we try to derive the optimal increment $\Delta_k(t)$ and buffer size $D_k(t)$ for client k at round t. However, $D_k(t)$ is not independent across the time slots and it relys on the last round's buffer size $D_k(t-1)$ and increment $\Delta_k(t)$. Thus, we apply the Hamilton equation, a kind of optimal control method, to capture the dependence and solve the optimization problem in stage II. Then we have the following proposition.

Proposition 5.1. *For any client k at arbitray round t, the optimal increment $\Delta_k(t)$ is*

$$\Delta_k(t) = \begin{cases} \max\left\{ \frac{1}{2\alpha_k} \sum_{\tau=t+1}^{T-1} \theta^{\tau-t-1} \left(\frac{\overline{\delta_k}^{-1} R}{\phi(\tau)} - 2\beta_k D_k(\tau) \right), 0 \right\}, & t \in [0, T-2]; \\ 0, & t = T-1. \end{cases} \tag{20}$$

Furthermore, the optimal buffer size $D_k(t)$ is

$$D_k(t) = \begin{cases} D_0, & t = 0; \\ \theta^t D_k(0) + \sum_{\tau=0}^{t-1} \theta^{t-1-\tau} \Delta_k(\tau), & t \in [1, T-1]. \end{cases} \tag{21}$$

The detailed proof is provided in Appendix A.2.

Remark 5.2. (30) reveals that provided the central server's payment R and mean field term $\phi(t)$, the optimal increment of client k at round t, i.e. $\Delta_k(t)$ is affected by the following buffer size of $D_k(\tau), \tau \in [t+1, \cdots, T]$, which in turn affects $D_k(\tau)$ according to (31). Therefore, a close loop exists between $\Delta_k(t)$ and $D_k(t)$, where greater $\delta_k(t)$ leads to greater $D_k(t)$, which in turn inhibits $\Delta_k(t)$.

Remark 5.3. A stationary state exists in an infinite time horizon.

5.4. Optimal Strategy of Server in Stage I

Searching Algorithm: In this stage, provided the optimal buffer size of all clients across the time slots, i.e. $D_k(t), k \in [1, N], t \in [0, \cdots, T-1]$, we try to derive the optimal payment R and decay factor θ on the central server's side. However, as mentioned above, it's extremely hard to derive a close-form solution of (R, θ) for the central server directly because the objective function in stage I is complex. Thus, we apply a searching algorithm to explore the optimal strategy (R, θ).

Parameter Estimation: In scenario of FL task with continuous data stream, parameters such as $\delta_{k,t}$ is time-varying over the time horizon due to the uncertain arrival pattern, hence the central server cannot predict the future information about data distribution in advance before training, which impedes the development of the optimal strategy (R, θ) for the central server. To address this problem, we can estimate them by conducting a simple federated learning procedure, which is similar to [18][45]. Namely, the central server will conduct a light-weight training experiment with all clients and estimates $\delta_{k,t}$ using the uploaded gradient. Note that this experiment is not involved in monetary payment. Then the central server adopts the maximum as $\overline{\delta_k}$ and solves the optimization problem in stage I.

5.5. Estimation of Mean Field Term

Ultimately, we try to estimate $\phi(t), t \in [0, \cdots, T-1]$, the mean field term introduced in section 5.1 with fixed point algorithm. According to (31), $D_k(t)$ is a function of $\Delta_k(\tau), \tau \in [0, \cdots, t-1]$. By inserting (30) into (31), $D_k(t)$ is actually a function of mean field terms $\phi(t)$ and buffer size $D_k(t), t \in [0, \cdots, T-1]$. We define this function as

$$D_k(t) = \Psi_{k,t}(\phi(0), \phi(1), \cdots, \phi(T-1), D_k(0), D_k(1), \cdots, D_k(T-1)).$$

Furthermore, according to (26), $\phi(t)$ is a function of $D_k(t), k \in [1, \cdots, N]$ at time t, so it can be derived that $D_k(t)$ is a function of $D_k(t), k \in [1, \cdots, N], t \in$

Algorithm 1 `Client-Strategy`

Input: ϕ, R, θ.
Output: $\Delta_k(t), D_k(t), t \in [0, \cdots, T-1], k \in [1, \cdots, N]$.
Initialize: $D_k^0(t), t \in [0, \cdots, T-1], k \in [1, \cdots, N], j = 0, \epsilon$.
repeat
  for $t = 0$ **to** $T-1$ **do**
    for $k = 1$ **to** N **do**
      Calculate $\Delta_k^j(t)$ using ϕ, R, θ and $D_k^j(t)$ according to (17).
      Calculate $D_k^j(t)$ using $\Delta_k^j(t)$ according to (18).
    end for
  end for
  $j \leftarrow j + 1$.
until $D_k^{j+1}(t) - D_k^j(t) \leq \epsilon, k \in [1, \cdots, K], t \in [0, \cdots, T-1]$.

$[0, \cdots, T-1]$. That is

$$\begin{aligned} D_k(t) = \Psi_{k,t}(&D_1(0), D_1(1), \cdots, D_1(T-1), \\ &D_2(0), D_2(1), \cdots, D_2(T-1), \\ &\cdots, \\ &D_N(0), D_N(1), \cdots, D_N(T-1)). \end{aligned} \tag{22}$$

For ease of reading, we denote the parameter matrix in (39) as A. Then we have $D_k(t) = \Psi_{k,t}(A)$. To summarize all data size $D_k(t)$ over the time horizon $t \in [0, \cdots, T-1]$ and $k \in [1, \cdots, N]$, we have the following vector function as

$$\begin{aligned} A =&\Psi(A) \\ =&(\Psi_{1,0}(A), \Psi_{1,1}(A), \cdots, \Psi_{1,T-1}(A) \\ &\Psi_{2,0}(A), \Psi_{2,1}(A), \cdots, \Psi_{2,T-1}(A) \\ &\cdots, \\ &\Psi_{K,0}(A), \Psi_{K,1}(A), \cdots, \Psi_{K,T-1}(A)), \end{aligned} \tag{23}$$

which is a mapping from A to A. Then we have the following proposition.

Proposition 5.4. *Ψ has a fixed point.*

Based on proposition 2, we can apply fixed point algorithm to find optimal $D_k(t), t \in [0, \cdots, T], k \in [1, \cdots, N]$, which will be demonstrated in the later section.

5.6. Description of Algorithm

6. Experiment

In this section, we evaluate the performance of proposed FedStream by numerical experiments.

Algorithm 2 `Server-Strategy`

Input: ϕ.
Output: R, θ.
def Func(R, θ):
  $D_k(t) \leftarrow$ `Client-Strategy`(ϕ, R, θ).
  Calculate *res* using $D_k(t)$ according to (16).
  return *res*.
$R^*, \theta^* \leftarrow$ *PSO(Func)*.
return R^*, θ^*.

Algorithm 3 `Estimate-MFT`

Input: None.
Output: $\phi_k(t), k \in [1, \cdots, N], t \in [0, \cdots, T-1]$.
Initialize: $\phi_k^0(t), k \in [1, \cdots, N], t \in [0, \cdots, T-1], j = 0, \epsilon$.
repeat
  $R, \theta \leftarrow$ `Server-Strategy`(ϕ).
  $D_k(t) \leftarrow$ `Client-Strategy`(ϕ, R, θ).
  for $t = 0$ **to** $T-1$ **do**
    Calculate $\phi_k^j(t)$ using $D_k^j(t)$ according to (20).
  end for
  $j \leftarrow j + 1$.
until $D_k^{j+1}(t) - D_k^j(t) \leq \epsilon, k \in [1, \cdots, K], t \in [0, \cdots, T-1]$.

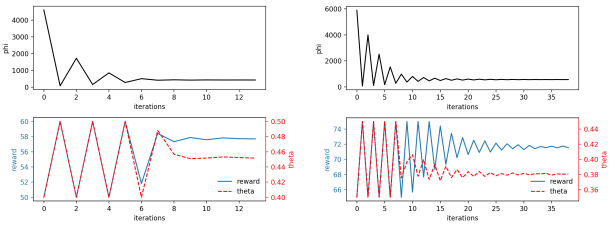
6.1. Experimental Setup

Datasets and Models: We conduct our experiments in two widely used real datasets: MNIST and FMNIST. The MNIST dataset contains 60,000 training samples and 10,000 test samples for handwrittern digit recognization. The FMNIST dataset comprises 50,000 training samples and 10,000 test samples for fashion item recognization. To simulate the impact of DoS in practical training, we mislabel parts of local buffer data of each client before each time slot, which is similar to the method adopted in (Xu et al., 2024). The proportion of mislabeling depends on time sensitivity coefficient σ. If we set a greater σ, which corresponds to a higher time-sensitive FL task, then the proportion of mislabeling will rise and vice versa. Besides, to simulate the setting of local data stream $\mathcal{D}$, we assign the dataset into all clients beforehand in a IID method. During the task, each client will sample new data from local dataset to its buffer according to the optimal strategy. Lastly, we use CNN as the training model in our experiments, which is made up of two sets of convolution layers and max pooling layers, and then two fully-connected layers and a RELU layer.

Hyperparameters Settings: We consider a federated learning setting with the number of rounds $T = 20$. In each round, there are 10 clients participating in the training. For a particular client k, it holds an initial buffer data with volume $D_k(0) = 700$ by default. During training, the unit price of

Algorithm 4 `Main Procedure`

Input: T, N.
Output: $w(T)$.
Initialize: $w(0)$.
$\phi \leftarrow$ `Estimate-MFT()`.
$R, \theta \leftarrow$ `Server-Strategy`(ϕ).
$\Delta, D \leftarrow$ `Client-Strategy`(ϕ, R, θ).
for $t = 0$ **to** $T - 1$ **do**
  Server distributes global model $w(t)$ to clients.
  for each client k **in parallel do**
    Abandon $\theta D_k(t-1)$ data points randomly from buffer: $\widetilde{D}_k(t) = D_k(t-1) - \theta D_k(t-1)$.
    Collect $\Delta_k(t-1)$ data points from local data stream: $D_k(t) = \widetilde{D}_k(t) + \Delta_k(t-1)$.
    Execute local update with global model $w(t)$: $w_k(t+1) = w(t) - \eta \nabla F_k(w(t)|D_k(t))$.
    Upload local model $w_k(t+1)$ back to the server.
  end for
  Server aggregates local model $w_i(t+1), i \in [N]$: $w(t+1) = \sum_{i=1}^{N} \frac{D_i(t+1)}{\sum_{i=1}^{N} D_i(t+1)} w_i(t+1)$.
end for
return $w(T)$.



$N = 3, T = 5$ $\quad$ $N = 5, T = 10$

Figure 3. The iteration of ϕ, R, θ and Δ

data collection and training is set to be $\alpha_k \sim U(10^{-4}, 10^{-3})$ and $\beta_k \sim U(5^{-6}, 5^{-5})$ respectively. Each client conducts one local update at a round with learning rate $\eta = 10^{-2}$. In addition, the server's cost balance factor is $\gamma = 10^{-4}$, with time sensitivity coefficient of $\sigma = 1$ and strategy space of $\theta \in [0, 1]$, $R \in [0, 100]$.

As for the hyperparameters of κ_2, κ_3 and κ_4, we take the similar method adopted in (Wang et al., 2019) and (Huang et al., 2024), both of which estimate these values with a simple FL training procedure. By conducting the simple procedure, we obtain $\kappa_2 = 1$ and $\kappa_3 = \kappa_4 = 10^{-2}$.

6.2. Experimental Results

Iteration of Mean-Field Term:

Effect of Client's Strategy on its Utility: In this paragraph, we analyse the effect of increment strategy $\Delta_k(t)$ on

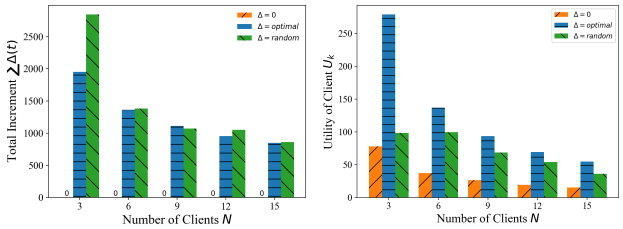


Figure 4. Comparative analysis of total increment $\sum_{t=0}^{T-1} \Delta_k(t)$ (**left**) and utility function U_k (**right**) against a particular client k versus number of clients N across three increment strategies, including $\Delta_k(t) = 0$, optimal and random.

clients' utility and the results are plotted in Figure 4. For comparison, we implement two auxiliary strategies: zero strategy and random strategy. For a particular client k, zero strategy means it doesn't collect any new data through the training procedure, while random strategy refers that it collects new data randomly in each time slot. Note that in order to keep comparison meaningful, the total amount of new data collected over the time horizon under random strategy is set to be roughly consistent with that under optimal strategy. We can find optimal strategy helps client k secure the highest utility compared with other two strategies. Provided clients are selfish, this indicates that each of them will follow the optimal strategy, thereby the mutually best response strategies are reached simultaneously, which meets the requirement the Nash equilibrium solution of Stage II. In addition, we can find under the same increment strategy, the client's utility U_k decreases with number of clients N. The underlying reason is that clients expansion may intensify the competition for a fixed reward among them, thereby leads to reward reduction and utility reduction.

Effect of Server's Strategy on Average Buffer Data: In this paragraph, we study the effect of server's strategy of both reward R and conservation rate θ on average buffer data from three terms: average increment $\Delta(t)$, average data size $D(t)$ and average data staleness $S(t)$. The results are depicted in Figure 5, 6 and 9 respectively. Specially, we fix the reward $R_0 = 61.62$ and consider four types of strategies: $\pi_1 = (R_0, 0.1)$, $\pi_2 = (R_0, 0.5)$, $\pi_3 = (R_0, 0.7)$, $\pi_4 = (R_0, 0.9)$. Besides, we also fix $\theta_0 = 0.3$ and consider another four strategies: $\pi_5 = (30, \theta_0)$, $\pi_6 = (60, \theta_0)$, $\pi_7 = (100, \theta_0)$, $\pi_8 = (140, \theta_0)$.

According to Figure 5, provided a fixed R, $\Delta(t)$ decreases with θ. The underlying realistic meaning is that the more valuable previous data are, the greater θ is set by central server to conserve previous data, thereby the less necessarily central server recruits new data. In contrast, given a fixed θ, $\Delta(t)$ increases with R. That's because a greater reward can effectively stimulate clients to collect more new data for training. In addition, we notice that $\Delta(t)$ steps into a stationary state after several communication rounds no mat-

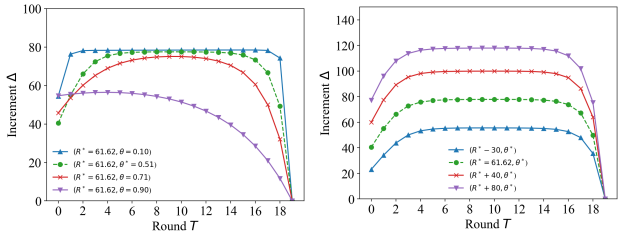


Figure 5. Comparative analysis of increment $\Delta_k(t)$ against a particular client k versus communication rounds t across various value of θ ranging from 0 to 1 (**left**) and R ranging from 0 to 150 (**right**).

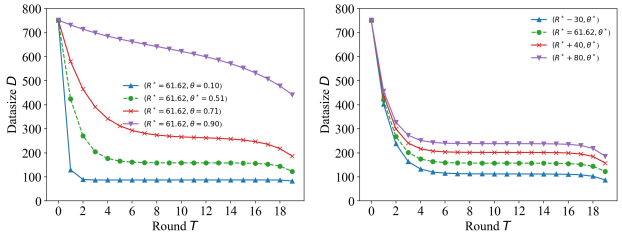


Figure 6. Comparative analysis of data size $D_k(t)$ against a particular client k versus communication rounds t across various value of θ ranging from 0 to 1 (**left**) and R ranging from 0 to 150 (**right**).

ter the strategy π, and then converges to 0 before the end of training. The observation of stationary state is substantiated by remark 5.3, wherein it is illustrated that a stationary state exists in an dynamic circumstance where clients adopts updated method based on (1). And the convergence trend to 0 is also in line with the boundary condition proposed in (20).

For data volume $D(t)$, as shown in Figure 6, the volume of buffer data $D(t)$ increases with both θ and R since greater θ means to conserve more previous data and greater R means to collect more new data from local data stream. Besides, the stationary state of $D(t)$ keeps pace with that of $\Delta(t)$ according to (21).

The observation against staleness $S(t)$ is not the same as that against volume. In Figure 9, we can find as θ grows, $S(t)$ soars to an unacceptable level because of rapid accumulation of deteriorated previous data. Yet it seldom effected by the change of R. Thus, the observation demonstrates that $S(t)$ is dominated by θ rather than R. It's not trivial to strike the balance between data volume and staleness by adjusting θ for a better model performance.

Effect of Server's Strategy on its Cost: In this paragraph, we analyse the effect of server's strategy (R, θ) on its cost. Under the hyperparameters settings in experimental setup, the optimal strategy is $\pi^* = (61.62, 0.51)$. For comparison, we fix the optimal reward R^* and consider three types of strategies: $\pi_1 = (R^*, \theta^* - 0.4), \pi_2 = (R^*, \theta^* + 0.2), \pi_3 = (R^*, \theta^* + 0.4)$. Besides, we also fix the optimal theta θ^* and consider another three types of strategies: $\pi_4 = (R^* -$

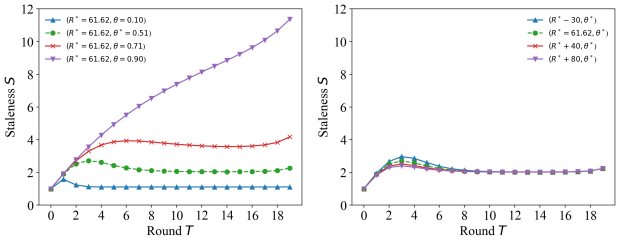


Figure 7. Comparative analysis of data staleness $S_k(t)$ against a particular client k versus communication rounds t across various value of θ ranging from 0 to 1 (**left**) and R ranging from 0 to 150 (**right**).

$30), \pi_5 = (R^*, \theta^* + 40), \pi_6 = (R^*, \theta + 80)$.

The effect of θ and R on test accuracy and cost is shown in Figure 10. We can find that π^*(solid line) outperforms π_1, π_2 and π_3(dotted line) in terms of test accuracy on the MNIST and CIFAR-10 datasets. The underlying reason is that strategy π_2, π_3 with improperly great θ contributes to huge but stale buffer data, while π_1 with improperly small θ contributes to fresh but tiny ones. Both of them leads to worse model performance. Under the same payment to clients, π^* reach the lowest cost compared with other strategies.

In addition, π_5 and π_6(dotted line) surpass the optimal strategy of π^*(solid line) with respect to test accuracy across all datasets because richer payment encourages clients to collect more fresh data, thereby enhance the model performance. However, π^* still reach the lowest cost, which indicates that our proposed strategy has the ability to strike the trade-off between payment and accuracy loss simultaneously. The above two experiments verify the optimality of our proposed strategy.

Effect of Hyperparameters: In this paragraph, we explore the result of effect of hyperparameters such as time-sensitive coefficient σ and initial volume of data $D(0)$ on exprements.

For generalization, we set three task modes: low time sensitivity, normal sensitivity and high sensitivity. The low time sensitivity mode usually corresponds to weak-dynamic scenarios where the data distribution is stable, such as long-term agriculture analysis or chronic disease analysis, while the high time sensitivity mode corresponds to strong-dynamic scenarios where the data distribution is unstable, such as real-time personalized recommendation and smart transporation. To simulate the above three modes in the experiments, we set the time sensitivity coefficient of σ as $0.1, 0.3$ and 0.7 respectively. The optimal strategies under various σ are shown in Figure 12. As σ increases, the optimal reward R rises with optimal conservation rate θ drops. It matches the intution that when facing tasks with higher time sensitivity, the value of previous data decrease. Server

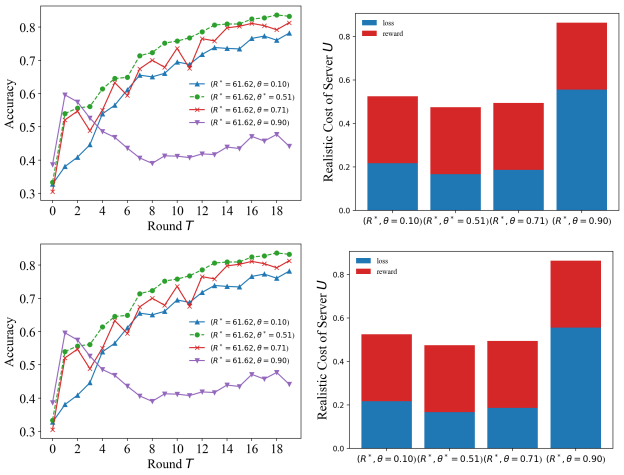


Figure 8. Comparative analysis of test accuracy (**left**) and cost (**right**) versus communication rounds t across various value of θ ranging from 0 to 1(**top**) and R ranging from 0 to 140(**bottom**) with CNN over MNIST. Left-Top: Test accuracy vs. Rounds, Right-Top: Cost vs. Rounds

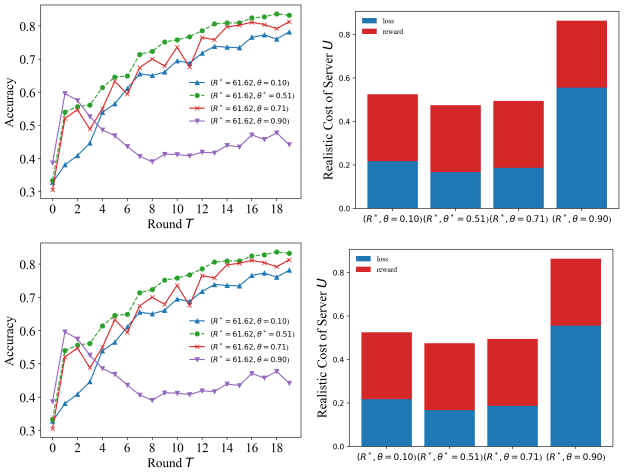


Figure 9. Comparative analysis of test accuracy (**left**) and cost (**right**) versus communication rounds t across various value of θ ranging from 0 to 1(**top**) and R ranging from 0 to 140(**bottom**) with CNN over CIFAR-10. Left-Top: Test accuracy vs. Rounds, Right-Top: Cost vs. Rounds

need to guide clients to abandon previous data as well as to collect new data in time. This illustrates that our proposed strategy can be applied in various different scenarios.

Next, we exhibite the effect of initial data size $D_k(0)$ on data volume. There is a key question about *whether the volume of data keeps dropping along with communication rounds*. It seems elusive from the perspective of intution. For comparison, We sampe three initial data volume: $D^1(0) = 50$, $D^2(0) = 100$ and $D^3(0) = 700$. The results is shown in Figure 12. We can observe that $D(0)$ only makes differneces on a few rounds at the begining of training, then they converge to the same stationary state and drop synchronously. From the aspect of mathematics, stationary state doesn't depend on $D(0)$ as demonstrated in remark 5.3. Therefore, the answer to the above question is that the trend of curve of data volumn banks on $D(0)$ and stationary state at the same time. If $D(0)$ is greater than stationary state, then $D(t)$ keeps rising before reaching the stationary state, and vice versa.

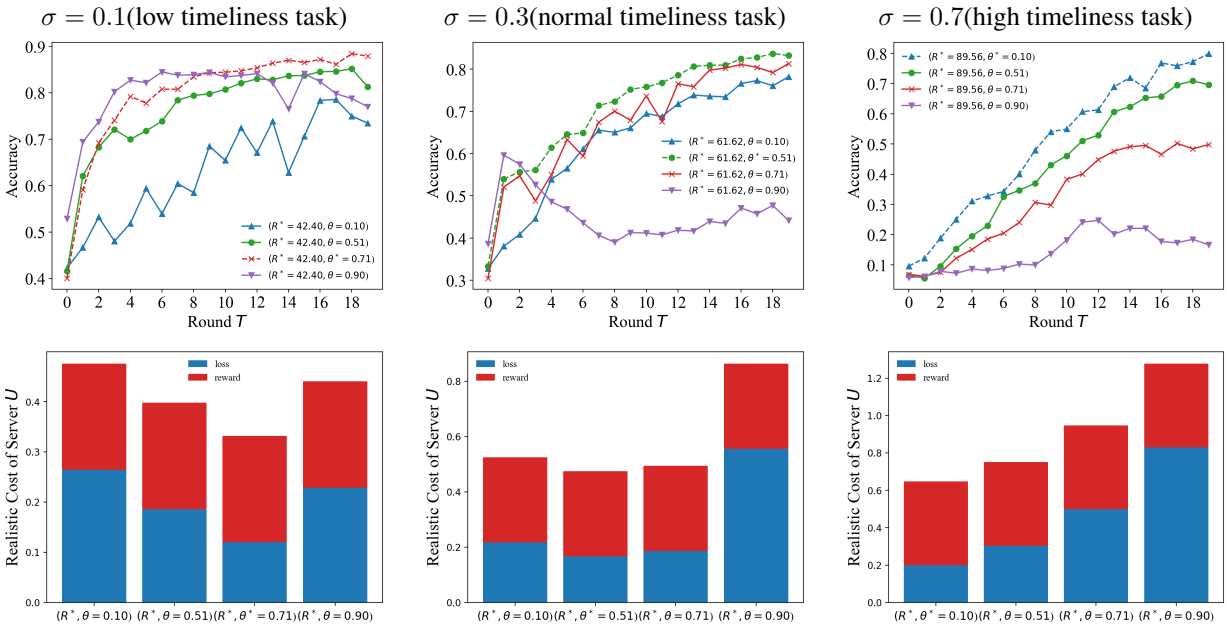


Figure 10. A 3×2 image matrix with column titles and aligned content.

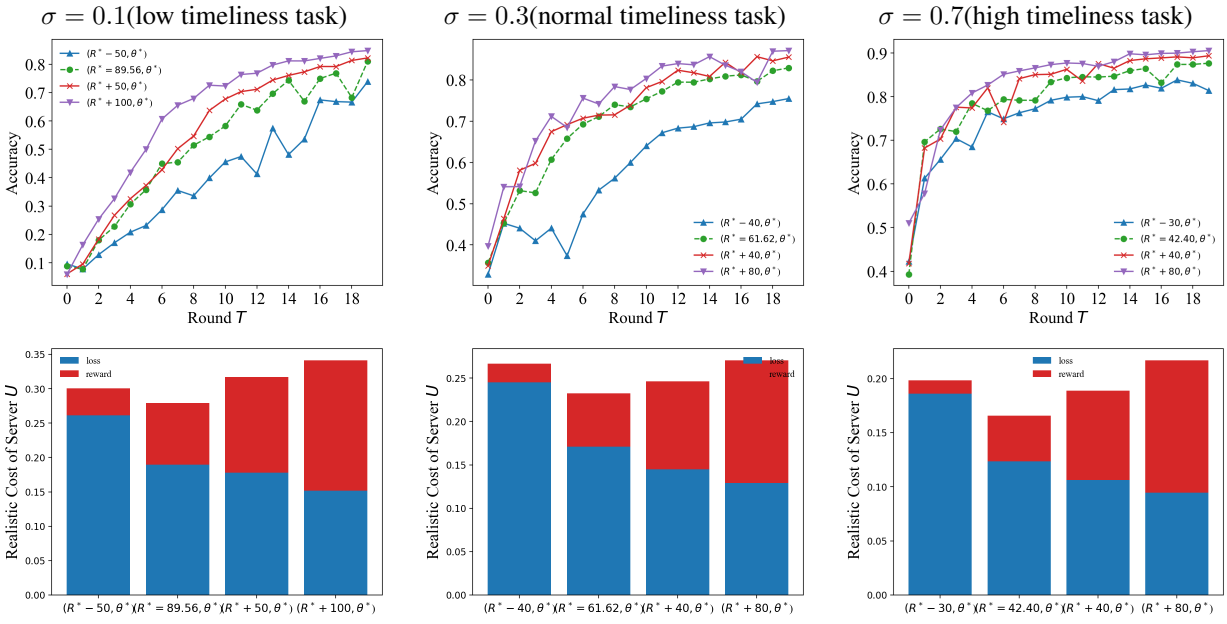


Figure 11. A 3×2 image matrix with column titles and aligned content.

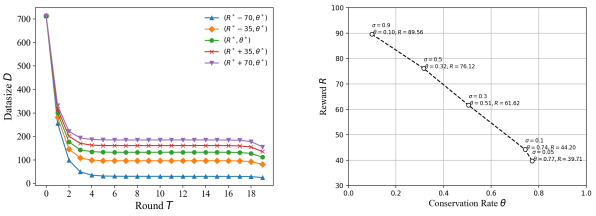


Figure 12. Comparative analysis of increment $\Delta_k(t)$ against a particular client k versus communication rounds t across various value of θ ranging from 0 to 1 (**left**) and R ranging from 0 to 100 (**right**).

7. Electronic Submission

Submission to ICML 2025 will be entirely electronic, via a web site (not email). Information about the submission process and LaTeX templates are available on the conference web site at:

`http://icml.cc/`

The guidelines below will be enforced for initial submissions and camera-ready copies. Here is a brief summary:

- Submissions must be in PDF.
- If your paper has appendices, submit the appendix together with the main body and the references **as a single file**. Reviewers will not look for appendices as a separate PDF file. So if you submit such an extra file, reviewers will very likely miss it.
- Page limit: The main body of the paper has to be fitted to 8 pages, excluding references and appendices; the space for the latter two is not limited in pages, but the total file size may not exceed 10MB. For the final version of the paper, authors can add one extra page to the main body.
- **Do not include author information or acknowledgements** in your initial submission.
- Your paper should be in **10 point Times font**.
- Make sure your PDF file only uses Type-1 fonts.
- Place figure captions *under* the figure (and omit titles from inside the graphic file itself). Place table captions *over* the table.
- References must include page numbers whenever possible and be as complete as possible. Place multiple citations in chronological order.
- Do not alter the style template; in particular, do not compress the paper format by reducing the vertical spaces.
- Keep your abstract brief and self-contained, one paragraph and roughly 4–6 sentences. Gross violations will require correction at the camera-ready phase. The title should have content words capitalized.

7.1. Submitting Papers

Anonymous Submission: ICML uses double-blind review: no identifying author information may appear on the title page or in the paper itself. Section 8.3 gives further details.

Authors must provide their manuscripts in **PDF** format. Furthermore, please make sure that files contain only embedded Type-1 fonts (e.g., using the program `pdffonts` in linux or using File/DocumentProperties/Fonts in Acrobat). Other fonts (like Type-3) might come from graphics files imported into the document.

Authors using **Word** must convert their document to PDF. Most of the latest versions of Word have the facility to do this automatically. Submissions will not be accepted in Word format or any format other than PDF. Really. We're not joking. Don't send Word.

Those who use **LaTeX** should avoid including Type-3 fonts. Those using `latex` and `dvips` may need the following two commands:

```
dvips -Ppdf -tletter -G0 -o paper.ps paper.dvi
ps2pdf paper.ps
```

It is a zero following the "-G", which tells dvips to use the config.pdf file. Newer TeX distributions don't always need this option.

Using `pdflatex` rather than `latex`, often gives better results. This program avoids the Type-3 font problem, and supports more advanced features in the `microtype` package.

Graphics files should be a reasonable size, and included from an appropriate format. Use vector formats (.eps/.pdf) for plots, lossless bitmap formats (.png) for raster graphics with sharp lines, and jpeg for photo-like images.

The style file uses the `hyperref` package to make clickable links in documents. If this causes problems for you, add `nohyperref` as one of the options to the `icml2025` usepackage statement.

7.2. Submitting Final Camera-Ready Copy

The final versions of papers accepted for publication should follow the same format and naming convention as initial submissions, except that author information (names and affiliations) should be given. See Section 8.3.2 for formatting instructions.

The footnote, "Preliminary work. Under review by the

International Conference on Machine Learning (ICML). Do not distribute." must be modified to "*Proceedings of the 42^{nd} International Conference on Machine Learning*, Vancouver, Canada, PMLR 267, 2025. Copyright 2025 by the author(s)."

For those using the **LaTeX** style file, this change (and others) is handled automatically by simply changing `\usepackage{icml2025}` to

```
\usepackage[accepted]{icml2025}
```

Authors using **Word** must edit the footnote on the first page of the document themselves.

Camera-ready copies should have the title of the paper as running head on each page except the first one. The running title consists of a single line centered above a horizontal rule which is 1 point thick. The running head should be centered, bold and in 9 point type. The rule should be 10 points above the main text. For those using the **LaTeX** style file, the original title is automatically set as running head using the `fancyhdr` package which is included in the ICML 2025 style file package. In case that the original title exceeds the size restrictions, a shorter form can be supplied by using

```
\icmltitlerunning{...}
```

just before `\begin{document}`. Authors using **Word** must edit the header of the document themselves.

8. Format of the Paper

All submissions must follow the specified format.

8.1. Dimensions

The text of the paper should be formatted in two columns, with an overall width of 6.75 inches, height of 9.0 inches, and 0.25 inches between the columns. The left margin should be 0.75 inches and the top margin 1.0 inch (2.54 cm). The right and bottom margins will depend on whether you print on US letter or A4 paper, but all final versions must be produced for US letter size. Do not write anything on the margins.

The paper body should be set in 10 point type with a vertical spacing of 11 points. Please use Times typeface throughout the text.

8.2. Title

The paper title should be set in 14 point bold type and centered between two horizontal rules that are 1 point thick, with 1.0 inch between the top rule and the top edge of the page. Capitalize the first letter of content words and put the rest of the title in lower case.

8.3. Author Information for Submission

ICML uses double-blind review, so author information must not appear. If you are using LaTeX and the `icml2025.sty` file, use `\icmlauthor{...}` to specify authors and `\icmlaffiliation{...}` to specify affiliations. (Read the TeX code used to produce this document for an example usage.) The author information will not be printed unless `accepted` is passed as an argument to the style file. Submissions that include the author information will not be reviewed.

8.3.1. Self-Citations

If you are citing published papers for which you are an author, refer to yourself in the third person. In particular, do not use phrases that reveal your identity (e.g., "in previous work (Langley, 2000), we have shown ...").

Do not anonymize citations in the reference section. The only exception are manuscripts that are not yet published (e.g., under submission). If you choose to refer to such unpublished manuscripts (Author, 2021), anonymized copies have to be submitted as Supplementary Material via OpenReview. However, keep in mind that an ICML paper should be self contained and should contain sufficient detail for the reviewers to evaluate the work. In particular, reviewers are not required to look at the Supplementary Material when writing their review (they are not required to look at more than the first 8 pages of the submitted document).

8.3.2. Camera-Ready Author Information

If a paper is accepted, a final camera-ready copy must be prepared. For camera-ready papers, author information should start 0.3 inches below the bottom rule surrounding the title. The authors' names should appear in 10 point bold type, in a row, separated by white space, and centered. Author names should not be broken across lines. Unbolded superscripted numbers, starting 1, should be used to refer to affiliations.

Affiliations should be numbered in the order of appearance. A single footnote block of text should be used to list all the affiliations. (Academic affiliations should list Department, University, City, State/Region, Country. Similarly for industrial affiliations.)

Each distinct affiliations should be listed once. If an author has multiple affiliations, multiple superscripts should be placed after the name, separated by thin spaces. If the authors would like to highlight equal contribution by multiple first authors, those authors should have an asterisk placed after their name in superscript, and the term "*Equal contribution" should be placed in the footnote block ahead of the list of affiliations. A list of corresponding authors and their emails (in the format Full Name <email@domain.com>)

can follow the list of affiliations. Ideally only one or two names should be listed.

A sample file with author names is included in the ICML2025 style file package. Turn on the `[accepted]` option to the stylefile to see the names rendered. All of the guidelines above are implemented by the LaTeX style file.

8.4. Abstract

The paper abstract should begin in the left column, 0.4 inches below the final address. The heading 'Abstract' should be centered, bold, and in 11 point type. The abstract body should use 10 point type, with a vertical spacing of 11 points, and should be indented 0.25 inches more than normal on left-hand and right-hand margins. Insert 0.4 inches of blank space after the body. Keep your abstract brief and self-contained, limiting it to one paragraph and roughly 4–6 sentences. Gross violations will require correction at the camera-ready phase.

8.5. Partitioning the Text

You should organize your paper into sections and paragraphs to help readers place a structure on the material and understand its contributions.

8.5.1. Sections and Subsections

Section headings should be numbered, flush left, and set in 11 pt bold type with the content words capitalized. Leave 0.25 inches of space before the heading and 0.15 inches after the heading.

Similarly, subsection headings should be numbered, flush left, and set in 10 pt bold type with the content words capitalized. Leave 0.2 inches of space before the heading and 0.13 inches afterward.

Finally, subsubsection headings should be numbered, flush left, and set in 10 pt small caps with the content words capitalized. Leave 0.18 inches of space before the heading and 0.1 inches after the heading.

Please use no more than three levels of headings.

8.5.2. Paragraphs and Footnotes

Within each section or subsection, you should further partition the paper into paragraphs. Do not indent the first line of a given paragraph, but insert a blank line between succeeding ones.

You can use footnotes[1] to provide readers with additional information about a topic without interrupting the flow of the paper. Indicate footnotes with a number in the text where the point is most relevant. Place the footnote in 9 point type at the bottom of the column in which it appears. Precede the first footnote in a column with a horizontal rule of 0.8 inches.[2]

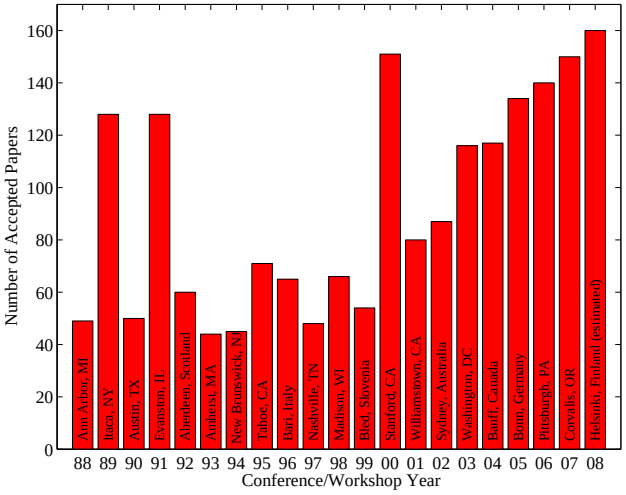


Figure 13. Historical locations and number of accepted papers for International Machine Learning Conferences (ICML 1993 – ICML 2008) and International Workshops on Machine Learning (ML 1988 – ML 1992). At the time this figure was produced, the number of accepted papers for ICML 2008 was unknown and instead estimated.

8.6. Figures

You may want to include figures in the paper to illustrate your approach and results. Such artwork should be centered, legible, and separated from the text. Lines should be dark and at least 0.5 points thick for purposes of reproduction, and text should not appear on a gray background.

Label all distinct components of each figure. If the figure takes the form of a graph, then give a name for each axis and include a legend that briefly describes each curve. Do not include a title inside the figure; instead, the caption should serve this function.

Number figures sequentially, placing the figure number and caption *after* the graphics, with at least 0.1 inches of space before the caption and 0.1 inches after it, as in Figure 13. The figure caption should be set in 9 point type and centered unless it runs two or more lines, in which case it should be flush left. You may float figures to the top or bottom of a column, and you may set wide figures across both columns (use the environment `figure*` in LaTeX). Always place

[1] Footnotes should be complete sentences.

[2] Multiple footnotes can appear in each column, in the same order as they appear in the text, but spread them across columns and pages if possible.

Algorithm 5 Bubble Sort

Input: data x_i, size m
repeat
  Initialize $noChange = true$.
  for $i = 1$ **to** $m - 1$ **do**
    if $x_i > x_{i+1}$ **then**
      Swap x_i and x_{i+1}
      $noChange = false$
    end if
  end for
until $noChange$ is $true$

Table 1. Classification accuracies for naive Bayes and flexible Bayes on various data sets.

DATA SET	NAIVE	FLEXIBLE	BETTER?
BREAST	95.9± 0.2	96.7± 0.2	√
CLEVELAND	83.3± 0.6	80.0± 0.6	×
GLASS2	61.9± 1.4	83.8± 0.7	√
CREDIT	74.8± 0.5	78.3± 0.6	
HORSE	73.3± 0.9	69.7± 1.0	×
META	67.1± 0.6	76.5± 0.5	√
PIMA	75.1± 0.6	73.9± 0.5	
VEHICLE	44.9± 0.6	61.5± 0.4	√

two-column figures at the top or bottom of the page.

8.7. Algorithms

If you are using LaTeX, please use the "algorithm" and "algorithmic" environments to format pseudocode. These require the corresponding stylefiles, algorithm.sty and algorithmic.sty, which are supplied with this package. Algorithm 5 shows an example.

8.8. Tables

You may also want to include tables that summarize material. Like figures, these should be centered, legible, and numbered consecutively. However, place the title *above* the table with at least 0.1 inches of space before the title and the same after it, as in Table 1. The table title should be set in 9 point type and centered unless it runs two or more lines, in which case it should be flush left.

Tables contain textual material, whereas figures contain graphical material. Specify the contents of each row and column in the table's topmost row. Again, you may float tables to a column's top or bottom, and set wide tables across both columns. Place two-column tables at the top or bottom of the page.

8.9. Theorems and such

The preferred way is to number definitions, propositions, lemmas, etc. consecutively, within sections, as shown below.

Definition 8.1. A function $f : X \to Y$ is injective if for any $x, y \in X$ different, $f(x) \neq f(y)$.

Using Theorem 8.1 we immediate get the following result:

Proposition 8.2. *If f is injective mapping a set X to another set Y, the cardinality of Y is at least as large as that of X*

Proof. Left as an exercise to the reader. □

Theorem 8.3 stated next will prove to be useful.

Lemma 8.3. *For any $f : X \to Y$ and $g : Y \to Z$ injective functions, $f \circ g$ is injective.*

Theorem 8.4. *If $f : X \to Y$ is bijective, the cardinality of X and Y are the same.*

An easy corollary of Theorem 8.4 is the following:

Corollary 8.5. *If $f : X \to Y$ is bijective, the cardinality of X is at least as large as that of Y.*

Assumption 8.6. The set X is finite.

Remark 8.7. According to some, it is only the finite case (cf. Theorem 8.6) that is interesting.

8.10. Citations and References

Please use APA reference format regardless of your formatter or word processor. If you rely on the LaTeX bibliographic facility, use `natbib.sty` and `icml2025.bst` included in the style-file package to obtain this format.

Citations within the text should include the authors' last names and year. If the authors' names are included in the sentence, place only the year in parentheses, for example when referencing Arthur Samuel's pioneering work (1959). Otherwise place the entire reference in parentheses with the authors and year separated by a comma (Samuel, 1959). List multiple references separated by semicolons (Kearns, 1989; Samuel, 1959; Mitchell, 1980). Use the 'et al.' construct only for citations with three or more authors or after listing all authors to a publication in an earlier reference (Michalski et al., 1983).

Authors should cite their own work in the third person in the initial version of their paper submitted for blind review. Please refer to Section 8.3 for detailed instructions on how to cite your own papers.

Use an unnumbered first-level section heading for the references, and use a hanging indent style, with the first line of

the reference flush against the left margin and subsequent lines indented by 10 points. The references at the end of this document give examples for journal articles (Samuel, 1959), conference publications (Langley, 2000), book chapters (Newell & Rosenbloom, 1981), books (Duda et al., 2000), edited volumes (Michalski et al., 1983), technical reports (Mitchell, 1980), and dissertations (Kearns, 1989).

Alphabetize references by the surnames of the first authors, with single author entries preceding multiple author entries. Order references for the same authors by year of publication, with the earliest first. Make sure that each reference includes all relevant information (e.g., page numbers).

Please put some effort into making references complete, presentable, and consistent, e.g. use the actual current name of authors. If using bibtex, please protect capital letters of names and abbreviations in titles, for example, use {B}ayesian or {L}ipschitz in your .bib file.

Accessibility

Authors are kindly asked to make their submissions as accessible as possible for everyone including people with disabilities and sensory or neurological differences. Tips of how to achieve this and what to pay attention to will be provided on the conference website `http://icml.cc/`.

Software and Data

If a paper is accepted, we strongly encourage the publication of software and data with the camera-ready version of the paper whenever appropriate. This can be done by including a URL in the camera-ready copy. However, **do not** include URLs that reveal your institution or identity in your submission for review. Instead, provide an anonymous URL or upload the material as "Supplementary Material" into the OpenReview reviewing system. Note that reviewers are not required to look at this material when writing their review.

Acknowledgements

Do not include acknowledgements in the initial version of the paper submitted for blind review.

If a paper is accepted, the final camera-ready version can (and usually should) include acknowledgements. Such acknowledgements should be placed at the end of the section, in an unnumbered section that does not count towards the paper page limit. Typically, this will include thanks to reviewers who gave useful comments, to colleagues who contributed to the ideas, and to funding agencies and corporate sponsors that provided financial support.

Impact Statement

Authors are **required** to include a statement of the potential broader impact of their work, including its ethical aspects and future societal consequences. This statement should be in an unnumbered section at the end of the paper (co-located with Acknowledgements – the two may appear in either order, but both must be before References), and does not count toward the paper page limit. In many cases, where the ethical impacts and expected societal implications are those that are well established when advancing the field of Machine Learning, substantial discussion is not required, and a simple statement such as the following will suffice:

"This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here."

The above statement can be used verbatim in such cases, but we encourage authors to think about whether there is content which does warrant further discussion, as this statement will be apparent if the paper is later flagged for ethics review.

References

Author, N. N. Suppressed for anonymity, 2021.

Ding, N., Fang, Z., and Huang, J. Optimal contract design for efficient federated learning with multi-dimensional private information. *IEEE Journal on Selected Areas in Communications*, 39(1):186–200, 2020.

Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification*. John Wiley and Sons, 2nd edition, 2000.

Fang, M., Wang, X., Xu, C., Yang, H. H., and Quek, T. Q. Computing-aided update for information freshness in the internet of things. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–7. IEEE, 2021.

Huang, G., Wu, Q., Sun, P., Ma, Q., and Chen, X. Collaboration in federated learning with differential privacy: A stackelberg game analysis. *IEEE Transactions on Parallel and Distributed Systems*, 2024.

Jiao, Y., Wang, P., Niyato, D., Lin, B., and Kim, D. I. Toward an automated auction framework for wireless federated learning services market. *IEEE Transactions on Mobile Computing*, 20(10):3034–3048, 2020.

Kang, J., Xiong, Z., Niyato, D., Xie, S., and Zhang, J. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal*, 6(6): 10700–10714, 2019.

Kearns, M. J. *Computational Complexity of Machine Learning*. PhD thesis, Department of Computer Science, Harvard University, 1989.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Lim, W. Y. B., Xiong, Z., Miao, C., Niyato, D., Yang, Q., Leung, C., and Poor, H. V. Hierarchical incentive mechanism design for federated machine learning in mobile networks. *IEEE Internet of Things Journal*, 7(10):9575–9588, 2020.

Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. (eds.). *Machine Learning: An Artificial Intelligence Approach, Vol. I*. Tioga, Palo Alto, CA, 1983.

Mitchell, T. M. The need for biases in learning generalizations. Technical report, Computer Science Department, Rutgers University, New Brunswick, MA, 1980.

Newell, A. and Rosenbloom, P. S. Mechanisms of skill acquisition and the law of practice. In Anderson, J. R. (ed.), *Cognitive Skills and Their Acquisition*, chapter 1, pp. 1–51. Lawrence Erlbaum Associates, Inc., Hillsdale, NJ, 1981.

Samuel, A. L. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3(3):211–229, 1959.

Tripathi, V. and Modiano, E. Age debt: A general framework for minimizing age of information. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–6. IEEE, 2021.

Wang, S., Tuor, T., Salonidis, T., Leung, K. K., Makaya, C., He, T., and Chan, K. Adaptive federated learning in resource constrained edge computing systems. *IEEE journal on selected areas in communications*, 37(6):1205–1221, 2019.

Wang, X. and Duan, L. Dynamic pricing for controlling age of information. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pp. 962–966. IEEE, 2019.

Wang, Z., Gao, L., and Huang, J. Taming time-varying information asymmetry in fresh status acquisition. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2021.

Wu, C., Xiao, M., Wu, J., Xu, Y., Zhou, J., and Sun, H. Towards federated learning on fresh datasets. In *2023 IEEE 20th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, pp. 320–328. IEEE, 2023.

Xiao, M., Xu, Y., Zhou, J., Wu, J., Zhang, S., and Zheng, J. Aoi-aware incentive mechanism for mobile crowdsensing using stackelberg game. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2023.

Xu, Y., Xiao, M.-J., Wu, C., Wu, J., Zhou, J.-R., and Sun, H. Age-of-information-aware federated learning. *Journal of Computer Science and Technology*, 39(3):637–653, 2024.

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

In the appendix, the complete proofs of theoretic results provided in the main text and more contents about experiment will be exhibited in detail.

A. Proof of Theoretic Results

A.1. Proof of Collary 3.2

Proof. According to the definition of DoS, we have

$$\begin{aligned}
S_k(t+1) &= S_k(t)\frac{\theta D_k(t)}{D_k(t+1)} + 1 = S_k(t)\left(1 - \frac{\Delta_k(t)}{D_k(t+1)}\right) + 1 \\
&= \left(S_k(t-1)\frac{\theta D_k(t-1)}{D_k(t)} + 1\right)\frac{\theta D_k(t)}{D_k(t+1)} + 1 \\
&= S_k(t-1)\frac{\theta^2 D_k(t-1)}{D_k(t+1)} + \frac{\theta D_k(t)}{D_k(t+1)} + 1 \\
&= S_k(t-2)\frac{\theta^3 D_k(t-2)}{D_k(t+1)} + \frac{\theta^2 D_k(t-1)}{D_k(t+1)} + \frac{\theta D_k(t)}{D_k(t+1)} + 1 \\
&= \cdots \\
&= S_k(0)\frac{\theta^{t+1} D_k(0)}{D_k(t+1)} + \frac{\theta^t D_k(1)}{D_k(t+1)} + \cdots + \frac{\theta^2 D_k(t-1)}{D_k(t+1)} + \frac{\theta D_k(t)}{D_k(t+1)} + 1 \\
&= \sum_{\tau=0}^{t+1} \frac{\theta^{t+1-\tau} D_k(\tau)}{D_k(t+1)}.
\end{aligned} \tag{24}$$

□

A.2. Proof of Theorem 3.10

Proof. Let's pay attention to the round $t+1$.

$$E[F(w(t+1)|\mathcal{D}) - F(w(t)|\mathcal{D})]$$

$$\leq \underbrace{E\left\langle \nabla F(w(t)|\mathcal{D}), w(t+1) - w(t)|\mathcal{D}\right\rangle}_{A1} + \underbrace{\frac{\beta}{2} E\|w(t+1) - w(t)|\mathcal{D}\|^2}_{A2} \tag{25}$$

(26)

We focus on bounding $A1$ at first:

$$
\begin{aligned}
&E\left\langle \nabla F(w(t)|D(t)), w(t+1)-w(t)|D(t)\right\rangle \\
=&E\left\langle \nabla F(w(t)|D(t)), (-\eta)\nabla F(w(t)|D(t))\right\rangle \\
=&E\left\langle \nabla F(w(t)|D(t)), \sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\left(-\eta\nabla F_k(w_k(t)|D_k(t))\right)\right\rangle \\
=&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}E\left\langle \nabla F(w(t)|D(t)), \nabla F_k(w(t)|D_k(t))\right\rangle \\
=&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\left\langle \nabla F(w(t)|D(t)), \nabla F_k(w(t)|\mathcal{D}_k)\right\rangle \\
=&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\frac{\|\nabla F(w(t)|D(t))\|^2+\|\nabla F_k(w(t)|\mathcal{D}_k)\|^2-\|\nabla F(w(t)|D(t))-\nabla F_k(w(t)|\mathcal{D}_k)\|^2}{2} \\
\leq&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\frac{2\|\nabla F(w(t)|D(t))\|\|\nabla F_k(w(t)|\mathcal{D}_k)\|-\delta_k(t)^2}{2} \\
=&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\frac{2\|\nabla F(w(t)|D(t))\|(\|\nabla F(w(t)|D(t))\|-\delta_k(t))-\delta_k(t)^2}{2} \\
\leq&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\frac{2(\|\nabla F(w(t)|D(t))\|^2-\delta_k(t)\|\nabla F(w(t)|D(t))\|)-\delta_k(t)^2}{2} \\
\leq&(-\eta)\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\frac{2(\|\nabla F(w(t)|D(t))\|^2-\rho\delta_k(t))-\delta_k(t)^2}{2} \\
=&(-\eta)\|\nabla F(w(t)|D(t))\|^2+\rho\eta\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)+\frac{\eta}{2}\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2
\end{aligned}
\tag{27}
$$

Then, we focus on bounding $A2$:

$$
\begin{aligned}
&\frac{\beta}{2}E\|w(t+1)-w(t)|D(t)\|^2 \\
=&\frac{\beta}{2}E\|(-\eta)\nabla F(w(t)|D(t))\|^2 \\
\leq&\beta\eta^2E\|\nabla F(w(t)|D(t))\|^2 \\
=&\beta\eta^2E\left\|\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\nabla F_k(w(t)|D_k(t))\right\|^2 \\
\leq&\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}E\|\nabla F_k(w(t)|D_k(t))\|^2 \\
\leq&\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\left(2\|\nabla F(w(t)|D(t))\|^2+2\delta_k(t)^2+\frac{\psi^2}{D_k(t)}+S_k(t)\sigma^2\right) \\
=&2\beta\eta^2\|\nabla F(w(t)|D(t))\|^2+2\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2+\beta\eta^2\frac{N\psi^2}{D(t)}+\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2.
\end{aligned}
\tag{28}
$$

Combining $A1$ and $A2$, we have:

$$\begin{aligned}
&E[F(w(t+1)|D(t)) - F(w(t)|D(t))] \\
\leq&(2\beta\eta^2 - \eta)E\|\nabla F(w(t)|D(t))\|^2 + 2\beta\eta^2\frac{N\psi^2}{D(t)} + +\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2 \\
&+ \rho\eta\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t) + (2\beta\eta^2 + \frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2 \\
\leq&\underbrace{(2\beta\eta^2 - \eta)E\|\nabla F(w(t)|D(t))\|^2}_{B1} + \underbrace{2\beta\eta^2\frac{N\psi^2}{D(t)}}_{B2} \\
&+ \underbrace{\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2}_{B3} + \underbrace{\xi(2\beta\eta^2 + \frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2}_{B4}
\end{aligned} \tag{29}$$

where $\xi \geq 2$.

Then we bound $B1$. We set $\eta < \frac{1}{2\beta}$, then $2\beta\eta^2 - \eta < 0$. According to Polyak Lojasiewicz condition from 3) in assumption,

$$E[F(w(t)|D(t)) - F(w^*)] \leq \frac{1}{2\mu}E\|\nabla F(w(t)|D(t))\|_2^2 \tag{30}$$

Then we have

$$2\mu(2\beta\eta^2 - \eta)E[(F(w(t)|D(t)) - F(w^*))] \geq (2\beta\eta^2 - \eta)E\|\nabla F(w(t)|D(t))\|_2^2 \tag{31}$$

Combining $B1$, $B2$, $B3$ and $B4$, we have:

$$\begin{aligned}
&E[F(w(t+1)|D(t)) - F(w(t)|D(t))] \\
\leq&2\mu(2\beta\eta^2 - \eta)E[F(w(t)|D(t)) - F(w^*)] + 2\beta\eta^2\frac{N\psi^2}{D(t)} \\
&+ \beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2 + \xi(2\beta\eta^2 + \frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2
\end{aligned} \tag{32}$$

Therefore,

$$\begin{aligned}
&E[F(w(t+1)|D(t+1)) - F(w(t)|D(t))] \\
=&E[F(w(t+1)|D(t)) - F(w(t)|D(t))] \\
&+ E[F(w(t+1)|D(t+1)) - F(w(t+1)|D(t))] \\
\leq&2\mu(2\beta\eta^2 - \eta)E[F(w(t-1)|D(t)) - F(w^*|D(t))] \\
&+ 2\beta\eta^2\frac{N\psi^2}{D(t)} + \beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2 + \xi(2\beta\eta^2 + \frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2 \\
&+ E[F(w(t+1)|D(t+1)) - F(w(t+1)|D(t))]
\end{aligned} \tag{33}$$

Adding $E[F(w(t)|D(t)) - F(w^*)]$ on both sides, we get

$$\begin{aligned}
&E[F(w(t+1)|D(t+1)) - F(w^*)] \\
\leq&(1+4\mu\beta\eta^2-2\mu\eta)E[F(w(t)|D(t)) - F(w^*)] \\
&+2\beta\eta^2\frac{N\psi^2}{D(t)}+\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2+\xi(2\beta\eta^2+\frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2 \\
&+E[F(w(t)|D(t+1)) - F(w(t)|D(t))]
\end{aligned} \tag{34}$$

Then use (74) recursively

$$\begin{aligned}
&E[F(w(t+1)|D(t+1)) - F(w^*)] \\
\leq&(1+4\mu\beta\eta^2-2\mu\eta)E[F(w(t)|D(t)) - F(w^*)] \\
&+2\beta\eta^2\frac{N\psi^2}{D(t)}+\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2+\xi(2\beta\eta^2+\frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2 \\
&+E[F(w(t+1)|D(t+1)) - F(w(t+1)|D(t))] \\
\leq&(1+4\mu\beta\eta^2-2\mu\eta)^2E[F(w(t-1)|D(t-1)) - F(w^*)] \\
&+(1+4\mu\beta\eta^2-2\mu\eta)\left[2\beta\eta^2\frac{N\psi^2}{D(t-1)}+\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t-1)}{D(t-1)}S_k(t-1)\sigma^2\right. \\
&\left.+\xi(2\beta\eta^2+\frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t-1)}{D(t-1)}\delta_k(t-1)^2+E[F(w(t)|D(t)) - F(w(t)|D(t-1))]\right] \\
&+\left[2\beta\eta^2\frac{N\psi^2}{D(t)}+\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}S_k(t)\sigma^2\right. \\
&\left.+\xi(2\beta\eta^2+\frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t)}{D(t)}\delta_k(t)^2+E[F(w(t+1)|D(t+1)) - F(w(t+1)|D(t))]\right] \\
\leq&\cdots \\
\leq&(1+4\mu\beta\eta^2-2\mu\eta)^{t+1}E[F(w(0)|D(0)) - F(w^*)] \\
&+\sum_{r=0}^{t}(1+4\mu\beta\eta^2-2\mu\eta)^r\left[2\beta\eta^2\frac{N\psi^2}{D(t-r)}+\beta\eta^2\sum_{k=1}^{N}\frac{D_k(t-r)}{D(t-r)}S_k(t-r)\sigma^2\right. \\
&\left.+\xi(2\beta\eta^2+\frac{\eta}{2})\sum_{k=1}^{N}\frac{D_k(t-r)}{D(t-r)}\delta_k(t-r)^2+E[F(w(t+1-r)|D(t+1-r)) - F(w(t+1-r)|D(t-r))]\right]
\end{aligned} \tag{35}$$

For ease of representation, let $\kappa_1 = 1+4\mu\beta\eta^2-2\mu\eta$, $\kappa_2 = 2\beta\eta^2$, $\kappa_3 = \beta\eta^2$ and $\kappa_4 = 2\xi\beta\eta^2+\frac{1}{2}\xi\eta$. Let $\Omega_t =$

$$
\begin{aligned}
&E[F(w(t+1)|D(t+1)) - F(w(t+1)|D(t))] \\
&\qquad E[F(w(T+1)|D(T+1)) - F(w^*)] \\
\leq &\kappa_1^{T+1} E[F(w(0)|D(0)) - F(w^*)] \\
&+ \sum_{t=0}^{T} \kappa_1^t \left[\kappa_2 \frac{N\psi^2}{D(T-t)} + \kappa_3 \sum_{k=1}^{N} \frac{D_k(T-t)}{D(T-t)} S_k(T-t)\sigma^2 + \kappa_4 \sum_{k=1}^{N} \frac{D_k(T-t)}{D(T-t)} \delta_k(T-t)^2 + \Omega_{T-t} \right] \\
= &\kappa_1^{T+1} E[F(w(0)|D(0)) - F(w^*)] \\
&+ \sum_{t=0}^{T} \kappa_1^{T-t} \left[\kappa_2 \frac{N\psi^2}{D(t)} + \kappa_3 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} S_k(t)\sigma^2 + \kappa_4 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} \delta_k(t)^2 + \Omega_t \right]
\end{aligned} \tag{36}
$$

Therefore,

$$
\begin{aligned}
&E[F(w(T)|D(T)) - F(w^*)] \\
= &\kappa_1^{T} E[F(w(0)|D(0)) - F(w^*)] \\
&+ \sum_{t=0}^{T-1} \kappa_1^{T-1-t} \left[\kappa_2 \frac{N\psi^2}{D(t)} + \kappa_3 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} S_k(t)\sigma^2 + \kappa_4 \sum_{k=1}^{N} \frac{D_k(t)}{D(t)} \delta_k(t)^2 + \Omega_t \right].
\end{aligned} \tag{37}
$$

Proof ended. □

A.3. Proof of Proposition 5.1

Proof. To find the optimal buffer size $D_k(t)$ for client k, we can construct the Hamilton equation $H_k(t)$ as

$$
\begin{aligned}
H_k(t) = &\frac{\overline{\delta_k}^{-1} D_k(t)}{\phi(t)} R - \alpha_k \Delta_k(t)^2 - \beta_k D_k(t)^2 \\
&+ \lambda_k(t+1) \left((\theta - 1) D_k(t) + \Delta_k(t) \right).
\end{aligned} \tag{38}
$$

Then we have

$$
\frac{\partial H_k(t)}{\partial \Delta_k(t)} = -2\alpha_k \Delta_k(t) + \lambda_k(t+1) = 0, \tag{39}
$$

$$
\Delta_k(t) = \frac{1}{2\alpha_k} \lambda_k(t+1), \tag{40}
$$

with

$$
\frac{\partial^2 H_k(t)}{\partial \Delta_k(t)^2} = -2\alpha_k < 0. \tag{41}
$$

Moreover,

$$
\frac{\partial H_k(t)}{\partial D_k(t)} = \frac{\overline{\delta_k}^{-1} R}{\phi(t)} - 2\beta_k D_k(t) + \lambda_k(t+1)(\theta - 1) = \lambda_k(t) - \lambda_k(t+1), \tag{42}
$$

$$
\lambda_k(t) = \theta \lambda_k(t+1) + \frac{\overline{\delta_k}^{-1} R}{\phi(t)} - 2\beta_k D_k(t). \tag{43}
$$

In addition, it can be derived that $\Delta_k(T-1) = 0$ because $\Delta_k(T-1)$ decides round T's data increment for client k. The newly sampled data points $D_k(T)$ will not be involved in the training rounds which ranges from 0 to $T-1$. Therefore

$D_k(T)$ cannot bring any benefit for client k under high collection cost. So it's feasible for client k to set $\Delta_k(T-1)=0$ and stop data collection.

Based on this, the boundary condition can be expressed as

$$\lambda_k(T-1) = \frac{\partial\left(\frac{\overline{\delta_k}^{-1}D(T-1)}{\phi(T-1)}R - \beta_k D_k(T-1)^2\right)}{\partial D_k(T-1)} \tag{44}$$

$$= \frac{\overline{\delta_k}^{-1}R}{\phi(T-1)} - 2\beta_k D_k(T-1). \tag{45}$$

According to (34) and (35), we have

$$\begin{aligned}
\lambda_k(t) &= \theta\lambda_k(t+1) + \frac{\overline{\delta_k}^{-1}R}{\phi(t)} - 2\beta_k D_k(t) \\
&= \theta^2\lambda_k(t+2) + \theta\left(\frac{\overline{\delta_k}^{-1}R}{\phi(t+1)} - 2\beta_k D_k(t+1)\right) + \frac{\overline{\delta_k}^{-1}R}{\phi(t)} - 2\beta_k D_k(t) \\
&= \cdots \\
&= \theta^{T-1-t}\lambda_k(T-1) + \sum_{\tau=0}^{T-2-t}\theta^\tau\left(\frac{\overline{\delta_k}^{-1}R}{\phi(t+\tau)} - 2\beta_k D_k(t+\tau)\right) \\
&= \theta^{T-1-t}\left(\frac{\overline{\delta_k}^{-1}R}{\phi(T-1)} - 2\beta_k D_k(T-1)\right) + \sum_{\tau=0}^{T-2-t}\theta^\tau\left(\frac{\overline{\delta_k}^{-1}R}{\phi(t+\tau)} - 2\beta_k D_k(t+\tau)\right) \\
&= \sum_{\tau=0}^{T-1-t}\theta^\tau\left(\frac{\overline{\delta_k}^{-1}R}{\phi(t+\tau)} - 2\beta_k D(t+\tau)\right) \\
&= \sum_{\tau=t}^{T-1}\theta^{\tau-t}\left(\frac{\overline{\delta_k}^{-1}R}{\phi(\tau)} - 2\beta_k D_k(\tau)\right).
\end{aligned} \tag{46}$$

Then,

$$\lambda_k(t+1) = \sum_{\tau=t+1}^{T-1}\theta^{\tau-t-1}\left(\frac{\overline{\delta_k}^{-1}R}{\phi(\tau)} - 2\beta_k D_k(\tau)\right). \tag{47}$$

Substituting (29) into (22), we obtain

$$\Delta_k(t) = \frac{1}{2\alpha_k}\sum_{\tau=t+1}^{T-1}\theta^{\tau-t-1}\left(\frac{\overline{\delta_k}^{-1}R}{\phi(\tau)} - 2\beta_k D_k(\tau)\right). \tag{48}$$

Substituting (30) into (9), we obtain

$$\begin{aligned}
D_k(t+1) &= \theta D_k(t) + \Delta_k(t) \\
&= \theta^2 D_k(t-1) + \theta\Delta_k(t-1) + \Delta_k(t) \\
&= \cdots \\
&= \theta^{t+1}D_k(0) + \sum_{\tau=0}^{t}\theta^{t-\tau}\Delta_k(\tau).
\end{aligned} \tag{49}$$

with $t \in [0, T-2]$ and $\Delta_k(T-1)=0$. □

A.4. Proof of Proposition 5.3

Proof. According to the Brouwer's fixed point theorem, we need to prove that Ψ is a continuous mapping from a closed set to itself.

First, we try to prove that Ψ is a mapping from a close set to itself. We bound $D_k(t)$ as [0, U], where $D_k(t) \geq 0$ means data size must be non-negative, and $D_k(t) \leq U$ means the maximum of data size cannot exceed U. It's especially feasible when a client considers its memory limit, resource budget, training expenditure and so on. Then, the domain of Ψ can be bounded as

$$\Pi = [0, U] \times [0, U] \times \cdots \times [0, U] \tag{50}$$

Therefore, Ψ is a mapping from a close set Π to itself.

Then, we try to prove that Ψ is a continuous mapping in Π. It's obvious that $\Psi_k(t)$ is continuous because $D_k(t)$ is continuous. Ψ is a linear combination of $\Psi_k(t)$, which refers that it's continuous.
Proof ended. □

You can have as much text here as you want. The main body must be at most 8 pages long. For the final version, one more page can be added. If you want, you can use an appendix like this one.

The `\onecolumn` command above can be kept in place if you prefer a one-column appendix, or can be removed if you prefer a two-column appendix. Apart from this possible change, the style (font size, spacing, margins, page numbering, etc.) should be kept the same as the main body.