Adaptive Data Admission and Retention for Streaming Federated Learning

Zhuoyi Zhao
University of Toronto
Toronto, ON, Canada
zhuoyijoeyzhao@gmail.com

Ben Liang
University of Toronto
Toronto, ON, Canada
liang@ece.utoronto.ca

arXiv:2607.23987v1 [cs.LG] 27 Jul 2026

Abstract

We study streaming federated learning with limited client memory, where newly generated training data incur time-varying sampling costs and must be selectively admitted and retained over time. We consider a joint server-side admission and client-side memory-management framework with the objective of minimizing the cumulative excess population risk under a sampling-cost budget and buffer constraints. We first derive a learning-error bound that explicitly captures the effects of instantaneous training sample size, distinct-sample growth, and reuse imbalance through a characterization of the effective sample size. Through a surrogate penalty obtained from this bound, we develop an Active-Constraint Drift-Plus-Penalty (ACDPP) policy that combines a structured client-side K-step retention rule with a server-side online admission rule and a time-varying rectangular admission region. We further present a sequence of comparison arguments, via an auxiliary constant-admission policy, that connects the ACDPP learning bound to a costless oracle benchmark. This yields explicit guarantees in terms of sublinear regret and sampling-cost violation, while the buffer-occupancy violation is controlled through offline selection of the retention horizon. Experiments on multiple datasets demonstrate that the proposed policy remains close to the oracle benchmark while satisfying the sampling-cost and buffer constraints.

CCS Concepts

• **Computing methodologies** → **Machine learning**; *Distributed computing methodologies*; • **Networks** → Network dynamics.

Keywords

streaming federated learning, online sampling, stochastic optimization

ACM Reference Format:
Zhuoyi Zhao and Ben Liang. 2026. Adaptive Data Admission and Retention for Streaming Federated Learning. In . ACM, New York, NY, USA, 15 pages.

1 Introduction

Federated learning (FL) enables multiple devices to collaboratively train a shared model without directly transferring their raw data to a central server [11, 17]. By keeping data local, FL has emerged as a promising paradigm for privacy-aware and large-scale model training in mobile and edge systems [1, 25]. The performance of FL, however, depends fundamentally on the quality, quantity, and diversity of the data available across participating clients. In practice, client data are often highly heterogeneous due to differences in user behavior, sensing environments, and local data-generation processes and generalization [12, 22, 23]. Moreover, in many real-world applications, local datasets evolve over time as new samples are continuously generated, collected, or labeled [7, 10, 20, 21, 26]. As a result, beyond model aggregation itself, an important question in FL is how to manage the training data pipeline under statistical heterogeneity and temporal data evolution.

In particular, client data may be generated and collected in an online fashion rather than provided as a fixed offline dataset. Examples include edge sensing, mobile crowdsourcing, and human-in-the-loop labeling systems, where new samples arrive sequentially and may incur non-negligible acquisition, annotation, or transmission costs before they can be used for training [9, 28]. Such costs are often time-varying due to changes in sensing conditions, network status, device availability, or labeling effort. At the same time, client devices are typically memory-limited and cannot retain all previously observed samples indefinitely. As a result, the learner must continuously decide whether to admit newly arrived samples and which historical samples to retain for future reuse. This creates a fundamental trade-off: admitting more fresh samples may improve data diversity and generalization, while reusing retained samples can reduce immediate acquisition cost and increase the effective training set size. Therefore, beyond classical federated optimization, streaming FL calls for a principled framework that manages sample admission and buffer occupancy under dynamic cost and memory constraints.

While prior works have studied resource management [9, 28], and sample selection [7, 16, 20, 21] in streaming FL, they do not provide a unified framework that directly optimizes learning performance under joint admission and memory constraints. In particular, existing resource-centric approaches mainly focus on communication, computation, or energy efficiency [9, 28], whereas learning-centric sample selection methods typically do not explicitly account for long-term sampling cost, buffer occupancy, or the coupled effect of sample admission and retention [7, 20, 21].

Our work bridges this gap by proposing a learning-centric modeling framework for streaming FL with limited client memory, which jointly captures server-side sample admission and client-side buffer management. Specifically, we formulate the problem as a constrained stochastic optimization problem whose objective is to minimize the cumulative excess population risk under long-term sampling-cost and buffer-occupancy constraints. This formulation enables a principled treatment of the trade-off among sample freshness, sample reuse, memory limitation, and admission cost, and lays the foundation for both the benchmark design and the online policy developed in this paper.

Conference'17, Washington, DC, USA
2026.

The main contributions of this paper are summarized as follows.

- We establish a learning-error bound for streaming FL with memory-based training, which explicitly characterizes the roles of the instantaneous training sample size, the cumulative number of distinct samples, and the reuse imbalance caused by repeated use of stored samples. In particular, we introduce an effective sample size representation that makes these effects transparent and amenable to control design.
- We propose an ACDPP policy that combines a structured client-side K-step retention rule with a server-side online admission rule based on a modified form of Lyapunov drift-plus-penalty optimization. In particular, to strengthen the control design, we impose a time-varying rectangular constraint on the feasible admission region, which gradually drives the admission process toward a target operating point.
- We prove explicit guarantees for the proposed policy, including a sublinear regret bound relative to a costless oracle benchmark and a sublinear sampling-cost violation bound derived via Lyapunov analysis. Our analysis proceeds through a sequence of finite-horizon comparisons that connect the ACDPP-induced learning bound to the oracle through its surrogate penalty and an auxiliary constant-admission policy. In addition, we provide an offline method for choosing the time-invariant retention horizon K to ensure uniformly bounded cumulative buffer-occupancy violation.
- We conduct experiments on multiple canonical datasets to validate the proposed policy and demonstrate its effectiveness under streaming sample admission, buffer constraints, and time-varying sampling costs.

2 Related Work

2.1 Centralized Learning with Streaming Data

In centralized streaming learning, samples arrive sequentially rather than as a fixed dataset. As a result, the learner typically trains on only a partially observed dataset at any point in time [5, 6]. This is often due to limited storage or the use of memory-efficient stream summaries in high-volume data streams [2, 3, 24]. Despite these connections, centralized streaming learning does not capture the network-edge setting considered here. It overlooks the distributed nature of data generation and storage across edge devices, along with the resulting heterogeneity in local data distributions, memory capacities, and resource constraints. It also generally does not account for the time-varying operating conditions that are common in edge environments.

2.2 Streaming Federated Learning

Several works have studied online resource management for streaming FL using Lyapunov optimization. CEFL [28] jointly optimizes admission control, load balancing, and accuracy tuning to minimize operational costs. Hu et al. [9] further incorporate device scheduling and bandwidth allocation under long-term energy constraints. While these approaches enable online decision-making with constraint guarantees, they primarily focus on system-level costs rather than learning performance, and assume unlimited on-device storage.

Another line of research develops data valuation metrics for sample selection under limited on-device storage. Gong et al. [7] propose gradient-norm based selection to accelerate convergence, Shi et al. [20] employ contrastive learning for importance scoring in unlabeled settings, and Sun et al. [21] learn query policies via multi-agent reinforcement learning. Notably, while these score-based methods implicitly assume that high-gradient-norm or high-loss samples are more valuable for learning, they do not explicitly optimize for the underlying population risk. In contrast, our framework directly minimizes the cumulative excess risk with respect to the target distribution, providing theoretical regret guarantees. Furthermore, these scoring mechanisms focus on *which* sample to select while our admission control determines *how many* samples to retain under cost constraints.

Marfoq et al. [16] provide a foundational analysis for streaming FL, characterizing the bias-optimization tradeoff when mixing historical and fresh clients. They establish that the optimal strategy depends on the ratio between gradient variability and effective sample size, with the extreme cases being uniform weighting when gradient variability dominates and historical-only weighting when sample size effects dominate. However, their analysis does not address online admission decisions, storage management, or explicit optimization of the mixing strategy under resource constraints.

Recent studies also address non-stationary data streams where distributions evolve over time. Zhang et al. [26] propose Fed-HIST to leverage historical model knowledge. Huynh et al. [10] analyze convergence under Markovian data streams. These works focus on adapting to distribution drift, whereas our work targets stationary environments where the optimal model remains fixed and the challenge lies in optimal sample management under resource constraints.

3 Preliminaries

3.1 Network Model

We consider a streaming FL system consisting of a parameter server (PS) and M clients, operating over discrete communication rounds indexed by $t \in \{1, 2, \ldots, T\}$. Each client is indexed by $m \in \{1, 2, \ldots, M\}$. Training samples are generated in a streaming manner: at the beginning of each round t, the PS determines the admission decisions, and the admitted samples are stored locally at the clients for future training.[1] We assume that client m is associated with a local data distribution $\mathcal{P}_m$, and that each sample admitted at client m is drawn independently and identically distributed (i.i.d.) from $\mathcal{P}_m$.

Due to privacy and security constraints, sample admission and local memory management are decoupled across the server and the clients. In particular, the PS never accesses raw samples; instead, it only determines the admission amount at each client in each round. Once admitted, a sample is stored locally in the buffer of the corresponding client. Buffer management is then carried out independently by each client through local dropping actions based solely on client-side information. The admitted and retained samples constitute the local training memory used for subsequent local model updates.

[1] An alternative is for the clients to locally determine admission. In that case, due to the positive sampling costs, it is necessary to consider fairness and incentives among the clients. Such game theoretic analysis is outside the scope of this work. Instead, here we focus on overall system cost and performance, so it suffices to consider server-side admission control.

Server-side admission actions and sampling cost. At each round t, the PS decides how many new samples to admit at each client. For each client m, samples are indexed by their generation order, and (m, i) denotes the i-th sample generated at client m. To represent the sample-level admission status for subsequent buffer and age evolution, we define the binary *admission* indicator

$$a_{m,i}(t) \in \{0, 1\}, \tag{1}$$

where $a_{m,i}(t) = 1$ means that sample (m, i) is admitted to the local buffer of client m in round t. Accordingly, we define the admission amount at client m in round t as

$$\lambda_m(t) \triangleq \sum_i a_{m,i}(t), \quad \forall m, \tag{2}$$

which is the number of samples admitted at client m in round t.

Admitting one new sample in round t incurs a time-varying per-sample *sampling cost* $c(t) \geq 0$, revealed to the PS at the beginning of round t. Let $\bar{c}_T \triangleq \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[c(t)]$ denote the average per-sample sampling cost over the horizon T. We impose the average sampling-cost constraint

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[c(t)\sum_{m=1}^{M} \lambda_m(t)\right] \leq C, \tag{3}$$

where $C > 0$ is the prescribed per-round sampling-cost budget.

Client memory state and buffer dropping. Each client maintains a local buffer that stores admitted samples for future training. Let $q^t_{m,i} \in \{0, 1\}$ indicate whether sample (m, i) is present in the local buffer of client m at the beginning of the local training phase of round t, i.e., after the admission and dropping actions in round t have been executed. The resulting buffer occupancy is defined as

$$n_m(t) \triangleq \sum_i q^t_{m,i}, \tag{4}$$

which is the number of samples stored in the local buffer of client m in round t. Accordingly, let $\mathcal{Q}_m(t)$ denote the set of samples stored in the local buffer of client m in round t, so that $n_m(t) = |\mathcal{Q}_m(t)|$. To capture limited memory, we impose an average buffer-occupancy constraint on each client:

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[n_m(t)] \leq B_m, \qquad m = 1, \ldots, M, \tag{5}$$

where $B_m > 0$ is the prescribed average memory budget of client m.

Client m independently manages its local buffer through a binary *drop* action $d_{m,i}(t) \in \{0, 1\}$, where $d_{m,i}(t) = 1$ means that sample (m, i) is removed from the local buffer of client m in round t. Dropping actions are determined locally based solely on client-side information.

Sample age and its evolution. For each sample (m, i), let $A^t_{m,i} \in \mathbb{N}$ denote its *age* at the beginning of the local training phase of round t. The age serves as a residence-time counter and is updated only when the sample is present in the local buffer at the beginning of the next local training phase. Specifically,

$$A^{t+1}_{m,i} = \begin{cases} A^t_{m,i} + 1, & \text{if } q^t_{m,i} = 1 \text{ and } q^{t+1}_{m,i} = 1, \\ 1, & \text{if } q^t_{m,i} = 0 \text{ and } q^{t+1}_{m,i} = 1, \\ A^t_{m,i}, & \text{if } q^{t+1}_{m,i} = 0, \end{cases} \tag{6}$$

with initialization $A^0_{m,i} = 0$ for all m, i. Thus, the age of a sample is incremented while it remains in the buffer, initialized to 1 when it newly enters the buffer, and frozen once it is no longer stored.

3.2 Distributed Learning Model

Let $\theta^{(t)} \in \Theta$ denote the global model at the beginning of the local training phase of round t. We assume that all clients participate in every communication round. Moreover, due to the limited buffer size at each client, we consider a *full-batch local update* regime, in which each client directly uses all samples currently stored in its local buffer to perform local training.

Let $\ell(\theta; z)$ denote the sample-wise loss of model parameter θ on data sample z. The empirical risk over the local buffer of client m in round t is defined as

$$\widehat{F}_{m,t}(\theta) \triangleq \frac{1}{n_m(t)} \sum_{z \in \mathcal{Q}_m(t)} \ell(\theta; z). \tag{7}$$

Accordingly, the corresponding local gradient is

$$\nabla \widehat{F}_{m,t}(\theta) = \frac{1}{n_m(t)} \sum_{z \in \mathcal{Q}_m(t)} \nabla_\theta \ell(\theta; z). \tag{8}$$

Starting from $\theta^{(t,0)}_m = \theta^{(t)}$, each client performs $E \geq 1$ steps of gradient descent with stepsize $\eta > 0$ on the empirical risk $\widehat{F}_{m,t}$:

$$\theta^{(t,e+1)}_m = \theta^{(t,e)}_m - \eta\, \nabla \widehat{F}_{m,t}\left(\theta^{(t,e)}_m\right), \qquad e = 0, \ldots, E-1. \tag{9}$$

After the local updates, client m uploads the model difference

$$\Delta^{(t)}_m \triangleq \theta^{(t,E)}_m - \theta^{(t)}. \tag{10}$$

Let $\alpha_m \geq 0$ denote the prescribed importance weight of client m, with $\sum_{m=1}^{M} \alpha_m = 1$. The server then aggregates the client updates according to a FedAvg-style rule:

$$\theta^{(t+1)} = \theta^{(t)} + \sum_{m=1}^{M} w_{m,t}\, \Delta^{(t)}_m, \qquad w_{m,t} \geq 0, \quad \sum_{m=1}^{M} w_{m,t} = 1, \tag{11}$$

where a natural choice of w_m, t, for example, is memory-proportional weighting, i.e.

$$w_{m,t} = \frac{\alpha_m n_m(t)}{\sum_{j=1}^{M} \alpha_j n_j(t)}. \tag{12}$$

3.3 Problem Formulation

Learning objective. We define the *target population* distribution as the weighted mixture

$$\mathcal{P} \triangleq \sum_{m=1}^{M} \alpha_m \mathcal{P}_m. \tag{13}$$

Our goal is to learn a model that minimizes the population risk

$$F_{\mathcal{P}}(\theta) \triangleq \mathbb{E}_{z\sim\mathcal{P}}[\ell(\theta; z)], \qquad \theta \in \Theta, \tag{14}$$

and let $\theta^\star \in \arg\min_{\theta\in\Theta} F_{\mathcal{P}}(\theta)$.

Server and client control variables. The system is controlled through a decomposed architecture. On the server side, the PS determines the admission amounts $\{\lambda_m(t)\}_{m=1}^{M}$ at each round t, which can be equivalently represented by the sample-level admission indicators $\{a_{m,i}(t)\}_{m,i}$ via (2). On the client side, each client m independently determines the sample-level dropping actions $\{d_{m,i}(t)\}_i$ to manage its local buffer. Together, the admission and

dropping actions govern the evolution of the client buffers and must jointly satisfy the sampling-cost constraint (3) and the average buffer-occupancy constraint (5).

Let π^{S} denote the server-side policy that maps server-observable information (e.g., the history of communicated updates and the current cost $c(t)$) to admission actions. For each client m, let π_m^{C} denote the client-side policy that maps local buffer states (including sample ages) to dropping actions. We denote the resulting joint policy by

$$\pi \triangleq \left(\pi^{\mathrm{S}}, \{\pi_m^{\mathrm{C}}\}_{m=1}^{M}\right). \tag{15}$$

Optimization objective and constraint violation. Under the learning dynamics in Section 3.2, the global model sequence $\{\theta^{(t)}\}_{t=1}^{T}$ is induced by the joint policy π through its effect on sample admissions, buffer evolution, and the resulting local updates. Although performance is evaluated with respect to the target population distribution $\mathcal{P}$, the model at each round is trained only on the samples currently stored in the client buffers. These stored samples are time-varying and are jointly determined by the admission and dropping actions over time. Hence, policy design in our setting amounts to controlling the evolving buffer contents so that training on the stored samples leads to low expected loss under the target population distribution $\mathcal{P}$.

To evaluate the learning performance induced by a given joint policy π, we define the *cumulative excess population risk* up to horizon T as

$$\mathcal{E}_T(\pi) \triangleq \sum_{t=1}^{T} \mathbb{E}\left[F_{\mathcal{P}}(\theta^{(t)}) - F_{\mathcal{P}}(\theta^{\star})\right]. \tag{16}$$

This quantity measures the cumulative suboptimality of the model sequence induced by policy π relative to the population-risk minimizer $\theta^{\star}$.

Our goal is to develop joint policies π that minimize the cumulative excess population risk $\mathcal{E}_T(\pi)$ subject to the sampling-cost constraint (3) and the average buffer-occupancy constraint (5).

4 Streaming FL Convergence Analysis

In this section, we analyze the learning performance of the streaming FL under server-side sample admission and client-side memory management. The key challenge is that the model is trained on time-varying buffer contents induced by the joint policy, rather than on direct samples from the target population distribution. Our focus is therefore on understanding how the evolving stored samples and their repeated reuse affect the convergence behavior of the learning process. In particular, we derive a bound on the cumulative excess population risk in (16).

4.1 Assumptions

We impose the following standard regularity assumptions.

Assumption 1 (Bounded domain)**.** The parameter set Θ has diameter D, i.e.,

$$\|\theta - \theta'\|_2 \le D, \qquad \forall \theta, \theta' \in \Theta.$$

Assumption 2 (Smoothness and bounded gradients)**.** The population risk $F_{\mathcal{P}}$ is L-smooth on Θ, i.e.,

$$\|\nabla F_{\mathcal{P}}(\theta) - \nabla F_{\mathcal{P}}(\theta')\|_2 \le L\|\theta - \theta'\|_2, \qquad \forall \theta, \theta' \in \Theta.$$

Moreover, the loss $\ell(\theta; z)$ is differentiable in θ, and its per-sample gradients are uniformly bounded:

$$\|\nabla_\theta \ell(\theta; z)\|_2 \le G, \qquad \forall \theta \in \Theta,\ z.$$

Hence, any empirical gradient formed as an average over stored samples is also bounded in norm by G.

Assumption 3 (Bounded loss and finite pseudo-dimension)**.** The loss is bounded as $\ell(\theta; z) \in [0, B]$ for all $\theta \in \Theta$ and z. In addition, the loss-composed hypothesis class $\ell \circ \mathcal{H}$ has finite pseudo-dimension, denoted by $\mathrm{Pdim}(\ell \circ \mathcal{H})$.

For sharper control of the stochastic-gradient term, we further impose the following assumption.

Assumption 4 (Bounded gradient variance)**.** The per-sample gradient has uniformly bounded population variance:

$$\mathbb{E}_{z\sim\mathcal{P}}\left[\|\nabla_\theta \ell(\theta; z) - \nabla F_{\mathcal{P}}(\theta)\|_2^2\right] \le \sigma^2, \qquad \forall \theta \in \Theta. \tag{17}$$

4.2 Learning Error Bound

We next derive an upper bound on the cumulative excess population risk $\mathcal{E}_T(\pi)$ under a given joint policy π. The bound depends on the policy-induced evolution of the training samples, in particular on the instantaneous training sample size, the cumulative number of distinct samples, and the loss of diversity caused by repeated reuse of stored samples.

Under the full-batch local training regime, let $n^{\pi}(t)$ denote the total number of stored samples used for training at round t under policy π. Further, let $N^{\pi}(t)$ denote the number of *distinct* samples that have been used in local training up to round t. Since retained samples may be reused multiple times across rounds, the quantity $N^{\pi}(t)$ alone does not fully reflect the amount of statistical information available for learning. To capture the loss of diversity caused by uneven sample reuse, we introduce an *effective sample size*.

Specifically, for each distinct sample $i \in \{1, \ldots, N^{\pi}(t)\}$, let $A_i^{\pi}(t)$ denote the cumulative number of times that sample i has been used in local training up to round t. We define

$$\widetilde{\mathcal{V}}^{\pi}(t) \triangleq \frac{\bar{A}^{\pi}(t)^2}{\bar{A}^{\pi}(t)^2 + \mathrm{Var}(A^{\pi}(t))}, \tag{18}$$

and

$$N_{\mathrm{eff}}^{\pi}(t) \triangleq N^{\pi}(t)\,\widetilde{\mathcal{V}}^{\pi}(t), \tag{19}$$

where $\bar{A}^{\pi}(t)$ and $\mathrm{Var}(A^{\pi}(t))$ are the empirical mean and variance of $\{A_i^{\pi}(t)\}_{i=1}^{N^{\pi}(t)}$, respectively. The factor $\widetilde{\mathcal{V}}^{\pi}(t) \in (0, 1]$ measures the uniformity of sample reuse: it is close to 1 when the reuse counts are nearly uniform, and becomes smaller when the reuse pattern is highly uneven. Accordingly, $N_{\mathrm{eff}}^{\pi}(t) = N^{\pi}(t)\widetilde{\mathcal{V}}^{\pi}(t)$ quantifies the effective amount of statistical diversity retained in the stored samples.

We now state the main excess-risk bound. The proof combines (i) a smoothness-based optimization recursion, (ii) variance control for stochastic gradients under sample reuse, and (iii) a uniform generalization bound controlled by the pseudo-dimension; detailed arguments are deferred to the appendix.

The bound is closely related in spirit to the general streaming-learning analysis of Theorem 4.4 in [16], but is specialized here to the present full-batch memory-based federated setting. Rather than giving an abstract order-level characterization as in [16], we express

the bound explicitly in terms of the policy-induced quantities $n^\pi(t)$, $N^\pi(t)$, and $\widetilde{\mathcal{V}}^\pi(t)$ through $N^\pi_{\text{eff}}(t) = N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)$. This explicit decomposition makes the separate effects of training sample size, distinct-sample growth, and reuse imbalance transparent, and will be instrumental for the subsequent online control design.

Theorem 1. *Under Assumptions 1–4, let $\{\theta_\pi^{(t)}\}_{t=1}^T$ be the model sequence generated by the learning dynamics under a given joint policy π. Then the cumulative excess population risk satisfies*

$$\mathcal{E}_T(\pi) \le \bar{\mathcal{E}}_T(\pi), \tag{20}$$

where

$$\bar{\mathcal{E}}_T(\pi) = \sum_{t=1}^{T}\Bigg[D\sigma\sqrt{\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}} + \eta\sigma^2\left(\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}\right) + 10B\sqrt{\frac{\mathrm{Pdim}(\ell\circ\mathcal{H})}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}}\sqrt{1+\log\left(\frac{N^\pi(t)}{\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)} + C_2\Bigg]. \tag{21}$$

Here, C_2 collects initialization and lower-order constant terms, including $\frac{D^2}{2\eta}$ and other algorithm-dependent constants.

Proof. See Appendix A in [27]. □

5 Active-Constraint DPP Policy

Utilizing the excess-risk bound in (20), we develop a joint admission-retention design for the streaming FL system. Our goal is to construct a structured online control policy that satisfies the long-term sampling-cost constraint and minimizes a tractable surrogate of the cumulative excess population risk.

While Theorem 1 provides an explicit policy-dependent upper bound, it remains difficult to optimize directly under a general buffer-management policy, because the reuse-uniformity factor $\widetilde{\mathcal{V}}^\pi(t)$ depends on the entire sample-retention pattern. To address this difficulty, we first impose a structured client-side retention rule, namely the K-step retention policy, which regularizes sample reuse and yields a tractable surrogate for online control. Based on this client-side structure, we then construct a DPP-based server-side admission policy using a cost-debt virtual queue together with a learning-oriented penalty term. To further strengthen the control design, we impose a *time-varying rectangular constraint* on the admission action space. This construction extends the classical DPP framework in [18] by allowing the feasible admission interval to shrink gradually over time toward the target operating point.

Algorithm 1 summarizes the overall Active-Constraint DPP policy, including the client-side K-step retention rule, the server-side online admission update, and the resulting distributed learning procedure. We next formalize each component of this policy.

Algorithm 1: Active-Constraint DPP Algorithm

Inputs: $\theta^{(1)}$, $Z(1)=0$, $\{q^0_{m,i}\}$, $\{A^0_{m,i}\}$, $N(0)$, $n(0)$, C, $\{B_m\}_{m=1}^M$, K, V, $\bar{\Lambda}$, ρ
Output: $\{\theta^{(t)}\}_{t=1}^{T+1}$

```
for t = 1, 2, ..., T do
    The server does:
        Observe 𝕆(t) and c(t)
        Solve (34) to obtain λ*(t)
        Update Z(t+1) and {A^{t+1}_{m,i}} using (26) and (6)
        Update N^π(t) and n^π(t)
    Each client m in parallel does:
        Admit λ_m(t) new samples using λ*(t)
        Perform local training and compute Δ_m^{(t)} using (8), (9), and (10)
        Update {q^{t+1}_{m,i}} using (22)
    The server does:
        Aggregate and update θ^{(t+1)} using (11)
```

5.1 K-Step Retention Policy

We first introduce the K-step retention policy, which imposes a simple deterministic client-side retention rule and makes the reuse pattern of stored samples explicitly characterizable. The key idea is that, under K-step retention, every admitted sample remains in the local buffer for a fixed number of rounds, so that the reuse-uniformity factor $\widetilde{\mathcal{V}}^\pi(t)$ can be expressed in closed form.

Definition 1 (K-step retention policy)**.** Fix an integer retention horizon $K \ge 1$. A client-side policy is called a *K-step retention policy* if, for every client $m \in \{1,\dots,M\}$, every admitted sample (m,i), and every round $t \in \{1,\dots,T\}$, the dropping action satisfies

$$d_{m,i}(t) = \mathbf{1}\{A^t_{m,i} = K\}, \qquad \forall m,i,t. \tag{22}$$

That is, each sample is retained in the local buffer for exactly K rounds after admission and is removed when its age reaches K.

We next characterize the reuse-uniformity factor under the K-step retention policy. Under this policy, every sample admitted no later than round $t-K+1$ has already contributed exactly K training uses by round t, while only the samples admitted in the most recent $K-1$ rounds incur a reuse deficit. Define

$$D^{(1)}_\partial(t) \triangleq \sum_{j=t-K+2}^{t}\big(K-(t-j+1)\big)\sum_{m=1}^{M}\lambda_m(j), \tag{23}$$

and

$$D^{(2)}_\partial(t) \triangleq \sum_{j=t-K+2}^{t}\big(K^2-(t-j+1)^2\big)\sum_{m=1}^{M}\lambda_m(j). \tag{24}$$

Then the reuse-uniformity factor under the K-step retention policy admits the exact representation

$$\widetilde{\mathcal{V}}^\pi(t) = \frac{\big(KN^\pi(t) - D^{(1)}_\partial(t)\big)^2}{N^\pi(t)\big(K^2N^\pi(t) - D^{(2)}_\partial(t)\big)}. \tag{25}$$

Hence, the finite-horizon deviation of $\widetilde{\mathcal{V}}^\pi(t)$ from one is determined entirely by the boundary deficits induced by the samples admitted in the most recent $K-1$ rounds.

5.2 Online Admission Control with Time-Varying Rectangular Constraints

We next formalize the server-side online admission control, including the virtual queue, the Lyapunov function, the penalty term, and a new DPP admission rule with time-varying rectangular constraints.

Cost-Debt Queue. Let $Z(t)$ denote the cost-debt queue at the beginning of round t. It evolves as

$$Z(t+1) = \big[Z(t) + C_t - C\big]^+, \tag{26}$$

where $C > 0$ is the prescribed per-round sampling-cost budget and

$$C_t \triangleq c(t)\Lambda(t), \qquad \Lambda(t) \triangleq \sum_{m=1}^{M} \lambda_m(t), \tag{27}$$

is the aggregate sampling cost incurred in round t under the DPP policy, with initialization $Z(0) = 0$.

The queue $Z(t)$ measures the accumulated violation of the budget constraint: when $C_t > C$, the system incurs positive debt and $Z(t)$ grows; otherwise the debt is reduced. Hence, a large value of $Z(t)$ indicates that the system has been operating above the desired average cost level.

Lyapunov Function and Penalty Function. We adopt the quadratic Lyapunov function

$$L(t) = \frac{1}{2} Z(t)^2, \tag{28}$$

which captures the evolution of the cost-debt dynamics.

Motivated by the learning-error bound in Theorem 1 together with the K-step retention structure developed in Section 5.1, we introduce the following per-round penalty, whose cumulative sum will serve as a surrogate objective for online control:

$$\begin{aligned} p(t) \triangleq\ & D\sigma\sqrt{\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)}} + \tilde{\eta}\sigma^2\left(\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)}\right) \\ & + 10B\sqrt{\frac{\mathrm{Pdim}(\ell \circ \mathcal{H})}{N^\pi(t)}}\sqrt{1 + \log\left(\frac{N^\pi(t)}{\mathrm{Pdim}(\ell \circ \mathcal{H})}\right)}. \end{aligned} \tag{29}$$

Accordingly, the cumulative quantity $\sum_{t=1}^{T} p(t)$ serves as a surrogate objective in place of the learning upper bound $\bar{\mathcal{E}}_T(\pi)$.

One-Slot Drift-Plus-Penalty. Let

$$\mathbb{O}(t) := \{Z(t), q_{m,i}^t, A_{m,i}^t\}_{m,i}$$

denote the system state at the beginning of the local training phase of round t.[2] We define the one-slot Lyapunov drift as

$$\Delta(\mathbb{O}(t)) \triangleq \mathbb{E}[L(t+1) - L(t) \mid \mathbb{O}(t)]\,. \tag{30}$$

The DPP framework minimizes a weighted sum of the one-slot drift and the per-round penalty. Accordingly, we consider the drift-plus-penalty expression

$$\Delta(\mathbb{O}(t)) + V\mathbb{E}[p(t) \mid \mathbb{O}(t)], \tag{31}$$

where $V > 0$ is a control parameter that trades off learning performance and budget satisfaction.

Substituting the queue evolution of $Z(t)$ into (30) and using standard quadratic-drift bounds yields

$$\Delta(\mathbb{O}(t)) + V\mathbb{E}[p(t) \mid \mathbb{O}(t)] \le B + \mathbb{E}[J_t(\boldsymbol{\lambda}) \mid \mathbb{O}(t)]\,, \tag{32}$$

where $B < \infty$ is a uniform constant independent of the control action, and

$$J_t(\boldsymbol{\lambda}) \triangleq Z(t)\big(c(t)\Lambda(t) - C\big) + Vp(t), \tag{33}$$

with

$$\boldsymbol{\lambda}(t) \triangleq (\lambda_1(t), \ldots, \lambda_M(t)).$$

Acitve-Constraint DPP Admission Rule. Motivated by the upper bound in (32), the ACDPP policy selects, at each round t, an admission action vector $\boldsymbol{\lambda}^\star(t)$ that minimizes the right-hand side of (33), namely,

$$\boldsymbol{\lambda}^\star(t) \in \arg\min_{\boldsymbol{\lambda}}\ J_t(\boldsymbol{\lambda}) \quad \text{s.t.} \quad \sum_{m=1}^{M} \lambda_m(t) \in [\Lambda_{\min}(t), \Lambda_{\max}(t)]. \tag{34}$$

Here, the bounds $\Lambda_{\min}$ and $\Lambda_{\max}$ impose a rectangular constraint on the aggregate admission rate.

Unlike the standard DPP policy, we set time-varying constraints as follows:

$$\Lambda_{\min}(t) = \bar{\Lambda}(1 - \rho^t), \qquad \Lambda_{\max}(t) = \frac{\bar{\Lambda}}{1 - \rho^t}, \tag{35}$$

for some constant $\rho \in (0, 1)$. As $t \to \infty$, the admissible interval shrinks to the singleton $\{\bar{\Lambda}\}$. The parameter ρ controls the contraction speed: smaller ρ leads to faster concentration, while larger ρ preserves more flexibility for a longer period. We will discuss how to choose ρ, which depends on T, after we derive the regret and sampling-cost violation bounds in Section 6.

This design is motivated by the structure of the learning-error surrogate. For sufficiently large t, the dominant terms depend primarily on the instantaneous training sample size $n^\pi(t)$. Under the buffer constraints, these terms are minimized when $n^\pi(t)$ is stable over time and close to its time average. This follows from the Cauchy–Schwarz inequality: for a fixed time-average sample size, temporal fluctuations in $n^\pi(t)$ increase the cumulative contribution of terms involving $1/n^\pi(t)$, whereas a constant sample size yields the smallest value. Accordingly, the time-varying rectangular constraint allows the admission rate to vary more freely in the early stage so as to adapt to the time-varying sampling cost, while gradually shrinking the feasible interval to promote a more stable long-term batch-size profile.

Although (34) is written in terms of the admission vector $\boldsymbol{\lambda}(t)$, the objective depends on the control only through the aggregate admission rate $\Lambda(t) = \sum_{m=1}^{M} \lambda_m(t)$. Therefore, the optimization reduces to a one-dimensional per-round problem, which yields low-complexity online implementation.

6 Performance Guarantees

This section analyzes the performance of the proposed ACDPP policy through a multi-step comparison framework. Our goal is to derive a regret bound for ACDPP relative to a costless oracle benchmark, together with explicit constraint-violation guarantees. We first establish a T-slot upper bound on the surrogate penalty induced by ACDPP. We then construct a comparison chain from the ACDPP-induced learning upper bound to the costless oracle benchmark and characterize the order of each correction term along this chain. These ingredients are finally combined to obtain the regret bound, the sampling-cost violation bound, and an offline method for selecting the retention horizon K to control the buffer-occupancy violation.

6.1 Preliminaries

To analyze the performance of the proposed ACDPP policy, we first introduce the ingredients used in the subsequent regret analysis. We

[2] Under K-step retention, $q_{m,i}^t$ and $A_{m,i}^t$ are induced by the historical admission process and are therefore inferable from server-side actions. Their use in the drift analysis does not require access to raw client data.

begin with a costless oracle benchmark, together with the regret and constraint-violation metrics used to evaluate the proposed policy over a finite horizon. We then introduce an auxiliary constant-admission policy under the K-step retention rule, which serves as a tractable intermediate comparison object for both parameter selection and regret analysis.

6.1.1 Costless Oracle Benchmark, Regret, and Constraint Violations. We first introduce an idealized *costless oracle* benchmark together with the regret and constraint-violation metrics.

Consider the system without the long-term sampling-cost constraint. In this costless regime, the upper bound in (20) is minimized when training samples are always fresh and no sample reuse occurs. Under the per-client buffer constraint B_m, this idealized behavior is achieved by the following *costless oracle policy*: at every round t, client m admits exactly B_m new samples, uses each sample once for local training, and immediately discards it. This policy satisfies the buffer constraints and maximizes the number of distinct samples used for learning.

Let $\bar{\mathcal{E}}_T^{\mathrm{ACDPP}}$ denote the upper bound on the cumulative excess population risk under the proposed ACDPP policy up to round T, and let $\bar{\mathcal{E}}_T^{\mathrm{oracle}}$ denote the corresponding quantity under the costless oracle policy. We define the regret as

$$\mathrm{Reg}(T) \triangleq \bar{\mathcal{E}}_T^{\mathrm{ACDPP}} - \bar{\mathcal{E}}_T^{\mathrm{oracle}}. \tag{36}$$

The regret in (36) will be used to quantify the finite-horizon performance gap between the proposed online policy and the costless oracle benchmark.

To measure the extent to which a policy violates the sampling-cost and buffer constraints, we define the cumulative sampling-cost violation and cumulative buffer-occupancy violation up to horizon T as

$$\mathrm{Vio}^{\mathrm{S}}(T) \triangleq \left[\sum_{t=1}^{T} \mathbb{E}\left[c(t)\sum_{m=1}^{M}\lambda_m(t) - C\right]\right]^+, \tag{37}$$

and

$$\mathrm{Vio}^{\mathrm{B}}(T) \triangleq \sum_{m=1}^{M}\left[\sum_{t=1}^{T}\mathbb{E}[n_m(t) - B_m]\right]^+. \tag{38}$$

These quantities will be used together with the regret to characterize the finite-horizon performance of the proposed policy.

6.1.2 Auxiliary Constant-Admission Policy. We next introduce an auxiliary comparison policy under the K-step retention rule.

Definition 2 (Auxiliary constant-admission policy)**.** Fix nonnegative admission rates $\{\lambda_m\}_{m=1}^M$. A server-side policy is called a *constant-admission policy* if, for every round t and client m,

$$\sum_i a_{m,i}(t) = \lambda_m, \qquad \forall m, t. \tag{39}$$

Combined with the K-step retention policy in Definition 1, this yields the joint policy π^{CK}.

Under π^{CK}, each client admits a constant number of new samples per round, while each admitted sample is retained for exactly K rounds. Consequently, for all sufficiently large t, the buffer occupancy at client m satisfies

$$n_m(t) = K\lambda_m, \tag{40}$$

and the total training sample size is

$$n^{\mathrm{CK}}(t) = K\bar{\Lambda}, \qquad \bar{\Lambda} \triangleq \sum_{m=1}^{M}\lambda_m. \tag{41}$$

Moreover, the cumulative number of distinct samples used in local training up to round t grows linearly as

$$N^{\mathrm{CK}}(t) = \bar{\Lambda}t. \tag{42}$$

Substituting (40)–(42) into Theorem 1, together with the exact finite-horizon reuse-uniformity factor induced by π^{CK}, yields the following specialization of the cumulative excess-risk bound.

Corollary 1. *For the auxiliary constant-admission policy π^{CK}, the cumulative excess population risk satisfies*

$$\mathcal{E}_T(\pi^{\mathrm{CK}}) \le \bar{\mathcal{E}}_T(\pi^{\mathrm{CK}}), \tag{43}$$

where

$$\bar{\mathcal{E}}_T(\pi^{\mathrm{CK}}) = \sum_{t=1}^{T}\left[D\sigma\sqrt{\frac{1}{\bar{\Lambda}K} - \frac{1}{\bar{\Lambda}t\,\widetilde{\mathcal{V}}^{\mathrm{CK}}(t)}} + \eta\sigma^2\left(\frac{1}{\bar{\Lambda}K} - \frac{1}{\bar{\Lambda}t\,\widetilde{\mathcal{V}}^{\mathrm{CK}}(t)}\right) + 10B\sqrt{\frac{\mathrm{Pdim}(\ell\circ\mathcal{H})}{\bar{\Lambda}t\,\widetilde{\mathcal{V}}^{\mathrm{CK}}(t)}}\sqrt{1+\log\left(\frac{\bar{\Lambda}t}{\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)} + C_2\right], \tag{44}$$

and

$$\widetilde{\mathcal{V}}^{\mathrm{CK}}(t) = \frac{\left(Kt - \frac{K(K-1)}{2}\right)^2}{t\left(K^2t - \frac{K(K-1)(2K-1)}{6}\right)}. \tag{45}$$

The specialized bound in Corollary 1 depends on the aggregate admission level $\bar{\Lambda}$, the retention horizon K, and the finite-horizon reuse-uniformity factor $\widetilde{\mathcal{V}}^{\mathrm{CK}}(t)$. We therefore choose these parameters so as to optimize the benchmark bound under the sampling-cost and buffer constraints.

Under the auxiliary constant-admission policy π^{CK}, the average sampling cost over horizon T must satisfy

$$\bar{c}_T\,\bar{\Lambda} \le C, \tag{46}$$

and the per-client buffer constraint requires

$$K\lambda_m \le B_m, \qquad \forall m. \tag{47}$$

Accordingly, the benchmark parameter design is formulated as

$$\min_{\{\lambda_m\}_{m=1}^M, K\in\mathbb{N}}\left\{\bar{\mathcal{E}}_T(\pi^{\mathrm{CK}}) : K\lambda_m \le B_m, \forall m,\ \bar{c}_T\sum_{m=1}^{M}\lambda_m \le C, \lambda_m \ge 0, \forall m\right\}. \tag{48}$$

Because $\bar{\mathcal{E}}_T(\pi^{\mathrm{CK}})$ depends on the admission vector $\{\lambda_m\}_{m=1}^M$ only through the induced aggregate level $\bar{\Lambda} = \sum_{m=1}^M \lambda_m$, the problem is low-dimensional. In particular, for any fixed horizon T, the benchmark parameters can be obtained by searching over feasible values of the integer variable K together with the corresponding admissible aggregate admission levels.

The following corollary gives the resulting closed-form benchmark choice in the infinite-horizon regime.

Corollary 2. *In the infinite-horizon regime, an optimal benchmark solution is obtained by saturating the per-client buffer constraints, i.e.,*

$$\lambda_m^\star = \frac{B_m}{K^\star}, \qquad \forall m, \tag{49}$$

where $K^\star$ is chosen as the better of the two nearest feasible integers, namely

$$K^\star = \left\lceil \frac{\bar{c}\sum_{m=1}^{M} B_m}{C} \right\rceil \quad or \quad K^\star = \max\left\{ \left\lfloor \frac{\bar{c}\sum_{m=1}^{M} B_m}{C} \right\rfloor, 1 \right\}. \quad (50)$$

In the sequel, we adopt the benchmark design in Corollary 2, then the per-client buffer constraint is saturated in steady state. In particular, for all $t > K$, the buffer occupancy satisfies

$$n_m(t) = B_m, \quad \forall m.$$

6.2 T-Slot ACDPP Upper Bound

We next establish a standard T-slot upper bound for the surrogate penalty induced by the proposed ACDPP policy. This Lyapunov-based result provides the key control relation used in the subsequent regret analysis.

Lemma 1. *The cumulative surrogate penalty under the proposed ACDPP policy satisfies*

$$\mathbb{E}[L(T+1) - L(1)] + V\sum_{t=1}^{T}\mathbb{E}[p(t)] \le BT + V\sum_{t=1}^{T}\bar{p}^{CK}(t), \quad (51)$$

where

$$\begin{aligned}\bar{p}^{CK}(t) = D\sigma\sqrt{\frac{1}{(K-1)\Lambda_{\min}(t-K)+\bar{\Lambda}} - \frac{1}{\sum_{\tau=1}^{t-1}\Lambda_{\min}(\tau)+\bar{\Lambda}}} \\ + \eta\sigma^2\left(\frac{1}{(K-1)\Lambda_{\min}(t-K)+\bar{\Lambda}} - \frac{1}{\sum_{\tau=1}^{t-1}\Lambda_{\min}(\tau)+\bar{\Lambda}}\right) \\ + 10B\sqrt{\frac{\mathrm{Pdim}(\ell\circ\mathcal{H})}{\sum_{\tau=1}^{t-1}\Lambda_{\min}(\tau)+\bar{\Lambda}}}\sqrt{1+\log\left(\frac{\sum_{\tau=1}^{t-1}\Lambda_{\max}(\tau)+\bar{\Lambda}}{\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)}.\end{aligned} \quad (52)$$

Proof. See Appendix B in [27]. □

6.3 Regret Decomposition

To derive the regret bound, we decompose the gap between the ACDPP-induced learning upper bound and the costless oracle benchmark into a sequence of finite-horizon correction terms. The key idea is to connect $\bar{\mathcal{E}}_T^{\mathrm{ACDPP}}$ to $\bar{\mathcal{E}}_T^{\mathrm{oracle}}$ through several intermediate quantities. In particular, we introduce the auxiliary comparison target

$$\begin{aligned}\bar{p}_\infty^{CK}(t) \triangleq D\sigma\sqrt{\frac{1}{\bar{\Lambda}K} - \frac{1}{\bar{\Lambda}t}} + \eta\sigma^2\left(\frac{1}{\bar{\Lambda}K} - \frac{1}{\bar{\Lambda}t}\right) \\ + 10B\sqrt{\frac{\mathrm{Pdim}(\ell\circ\mathcal{H})}{\bar{\Lambda}t}}\sqrt{1+\log\left(\frac{\bar{\Lambda}t}{\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)}.\end{aligned} \quad (53)$$

which corresponds to the surrogate penalty of the constant-admission policy in the infinite-horizon. The overall comparison chain is

$$\bar{\mathcal{E}}_T^{\mathrm{ACDPP}} \;\to\; \sum_{t=1}^{T}\mathbb{E}[p(t)] \;\to\; \sum_{t=1}^{T}\bar{p}^{CK}(t) \;\to\; \sum_{t=1}^{T}\bar{p}_\infty^{CK}(t) \;\to\; \bar{\mathcal{E}}_T^{\mathrm{oracle}},$$

where each arrow represents a finite-horizon comparison step. Intuitively, the first step relates the actual ACDPP-induced learning upper bound to its surrogate penalty; the second step uses the T-slot ACDPP upper bound to compare this surrogate with the finite-horizon comparison form induced by the auxiliary constant-admission policy; the third step removes the transient effects caused by the time-varying rectangular action space; and the final step connects the resulting infinite-horizon comparison target to the costless oracle benchmark.

More precisely, define

$$\Omega_1(T) \triangleq \bar{\mathcal{E}}_T^{\mathrm{ACDPP}} - \sum_{t=1}^{T}\mathbb{E}[p(t)], \quad (54)$$

$$\Omega_2(T) \triangleq \sum_{t=1}^{T}\bar{p}^{CK}(t) - \sum_{t=1}^{T}\bar{p}_\infty^{CK}(t), \quad (55)$$

and

$$\Omega_3(T) \triangleq \sum_{t=1}^{T}\bar{p}_\infty^{CK}(t) - \bar{\mathcal{E}}_T^{\mathrm{oracle}}. \quad (56)$$

In addition, the T-slot ACDPP upper bound in Lemma 1 yields

$$\sum_{t=1}^{T}\mathbb{E}[p(t)] \le \frac{B}{V}T + \sum_{t=1}^{T}\bar{p}^{CK}(t) + \frac{\mathbb{E}[L(1)]}{V}. \quad (57)$$

Combining (54)–(57), we obtain

$$\bar{\mathcal{E}}_T^{\mathrm{ACDPP}} - \bar{\mathcal{E}}_T^{\mathrm{oracle}} \le -\Omega_1(T) + \Omega_2(T) + \Omega_3(T) + O\left(\frac{T}{V}\right). \quad (58)$$

It therefore remains to characterize the orders of the three correction terms. The technical proofs are deferred to the appendix.

Lemma 2. *Under the K-step retention policy,*

$$\Omega_1(T) = O(1). \quad (59)$$

Proof. See Appendix C in [27]. □

Lemma 3. *Under the time-varying rectangular constraint* (35)*,*

$$\Omega_2(T) = O\left(\frac{1}{1-\rho}\right). \quad (60)$$

Proof. See Appendix D in [27]. □

Lemma 4. *For any fixed comparison operating point $(\bar{\Lambda}, K)$ satisfying $\bar{\Lambda}K = \bar{B}$,*

$$\Omega_3(T) = O(\log T). \quad (61)$$

Proof. See Appendix E in [27]. □

6.4 Regret and Constraint Guarantees

Combining the T-slot ACDPP upper bound in Lemma 1 with the regret decomposition in Section 6.3 yields the main guarantees of the proposed policy: a regret bound relative to the costless oracle benchmark and a sampling-cost violation bound. We also provide an offline method for selecting the retention horizon K such that the cumulative buffer-occupancy violation remains bounded as $O(1)$.

Theorem 2. *For any fixed ACDPP parameter $V > 0$ and any fixed $0 < \rho < 1$, the regret as defined in* (36) *satisfies*

$$\mathrm{Reg}(T) = O\left(\frac{T}{V}\right) + O\left(\frac{1}{1-\rho}\right) + O(\log T). \quad (62)$$

Proof. Substituting Lemmas 2–4 into (58) and absorbing the $O(1)$ term into the constants proves (62). □

Theorem 3. *For any fixed ACDPP parameter $V > 0$ and any fixed $0 < \rho < 1$, the cumulative sampling-cost constraint violation satisfies*

$$\mathrm{Vio}^{\mathrm{S}}(T) = O\left(\sqrt{VT + \frac{V}{1-\rho}}\right). \quad (63)$$

Proof. See Appendix F in [27]. □

Remark 1. Let $V = T^{\gamma}$, and $\rho = 1 - T^{-\phi}$. Then the cumulative regret and cumulative sampling-cost violation satisfy

$$\mathrm{Reg}(T) = O\big(T^{1-\gamma}\big) + O\Big(T^{\phi}\Big) + O(\log T), \tag{64}$$

and

$$\mathrm{Vio}^{\mathrm{S}}(T) = O\Big(T^{\frac{1+\gamma}{2}} + T^{\frac{\gamma+\phi}{2}}\Big). \tag{65}$$

Therefore, for any $\gamma \in (0,1)$ and $\phi \in (0,1)$, both $\mathrm{Reg}(T)$ and $\mathrm{Vio}^{\mathrm{S}}(T)$ are sublinear in T.

Remark 2. The cumulative buffer-occupancy violation is controlled through the offline selection of the retention horizon K. Since K is a time-invariant policy parameter rather than an online control variable, it must be chosen prior to execution of the proposed policy. Accordingly, one may evaluate the buffer-occupancy violation induced under candidate values of K and select a feasible retention horizon such that the resulting violation remains uniformly bounded over the horizon of interest. For such a choice of K, the cumulative buffer-occupancy violation satisfies

$$\mathrm{Vio}^{\mathrm{B}}(T) = O(1). \tag{66}$$

Thus, unlike the sampling-cost violation, which is regulated online through the virtual queue, the buffer-occupancy violation is absorbed into the offline parameter-selection stage through the choice of K.

7 Numerical Evaluation

In addition to the regret analysis in Section 6, we further evaluate the ACDPP policy on three image classification datasets: MNIST [14], CIFAR-10 [13], and ImageNette [4].[3] We compare the proposed ACDPP policy against three baselines: the costless oracle benchmark, the constant-admission policy, and a hybrid policy inspired by [16] that combines fresh-sample admission with stale-sample reuse under the same cost budget. Methods in [7, 9, 20, 21, 28] cannot be included as baselines, since they do not explicitly model the admission-quantity decision as explained in Section 2.2.

7.1 Experimental Setup

We consider a federated learning system with $M = 10$ clients. For MNIST and CIFAR-10, we adopt a non-i.i.d. partition in which each client contains samples from only two out of the ten classes. For ImageNette, the training data are partitioned i.i.d. and evenly across clients. We adopt LeNet [15] as the prediction model for MNIST, while for CIFAR-10 and ImageNette we use ResNet-9 [8]. For MNIST, the client buffer capacities are set to $\{8, 9, 10, 11, 12, 8, 9, 10, 11, 12\}$, and the long-term sampling-cost budget is $C = 55$. For CIFAR-10 and ImageNette, due to the increased task difficulty, both the client buffer capacities and the sampling-cost budget are set to be, respectively, 2× and 4× those of MNIST. The per-sample acquisition cost $c(t)$ is independently drawn from $\mathrm{Unif}[1, 10]$. In each communication round, every client performs $E = 10$ local epochs for all three datasets. The learning rates are set to $\eta = 0.5$ for MNIST, $\eta = 5 \times 10^{-3}$ for CIFAR-10, and $\eta = 5 \times 10^{-4}$ for ImageNette. For the hybrid baseline, five clients train only on retained stale samples, while the other five admit new samples according to the constant-admission rule under the same cost budget.

[3] Code is available at https://github.com/zhuoyijoeyzhao/AdaSamplingSFL.

7.2 Main Results

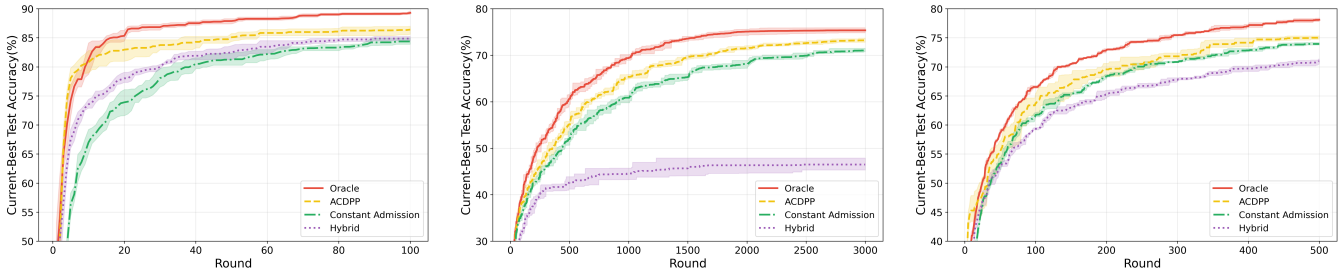
We report the current-best test accuracy on all three datasets in Figure 1. Each curve is averaged over 10 independent runs, and the shaded region indicates the 80% confidence interval. For brevity, we present the buffer occupancy and per-round sampling cost evolution on ImageNette only in Figure 2, since the same qualitative behavior is observed on the other datasets.

As shown in Figure 1, the proposed ACDPP policy consistently outperforms both the hybrid baseline and the constant-admission baseline, while remaining close to the oracle policy. On MNIST in Figure 1(a), ACDPP improves the average test accuracy by approximately 1.9 and 3.5 percentage points over the hybrid and constant-admission baselines, respectively. On CIFAR-10 in Figure 1(b), the corresponding improvements are 26.7 and 3.5 percentage points. On ImageNette in Figure 1(c), ACDPP further improves the average test accuracy by 4.1 and 1.8 percentage points over the hybrid and constant-admission baselines, respectively. This advantage is also reflected in the convergence speed visible in Figure 1. To quantify this effect, we use the first communication round at which each policy reaches a target test accuracy, with target levels set to 85% for MNIST, 70% for CIFAR-10, and 70% for ImageNette. Under this criterion, ACDPP converges about 2.3× and 2.2× faster than the hybrid baseline on MNIST and ImageNette, respectively, while on CIFAR-10 the hybrid baseline fails to reach the target accuracy within the experiment horizon, as seen in Figure 1(b). Relative to the constant-admission baseline, the corresponding speedups are 2.6×, 1.7×, and 1.3× on MNIST, CIFAR-10, and ImageNette, respectively. An additional observation is that the hybrid baseline behaves very differently across tasks. Similar to the interpretation in [16], it can converge faster than constant-admission on a simple task such as MNIST. However, on more difficult tasks, especially under distribution heterogeneity, its weakness becomes much more pronounced. As shown in Figure 1(b), on non-i.i.d. CIFAR-10 the hybrid baseline performs substantially worse than both ACDPP and constant-admission, even though its fresh-sample component is scheduled according to the optimal constant-admission constant-admission rule under the same cost budget.

Figure 2 further verifies that the proposed ACDPP policy satisfies both the buffer and sampling-cost constraints. In particular, the buffer occupancy in Figure 2(a) remains within the prescribed memory budget, while the stability of the cost-debt queue in Figure 2(b) is consistent with the long-term sampling-cost guarantee established by the Lyapunov analysis. This reflects the two-level design of ACDPP: sampling cost is controlled online, while buffer feasibility is enforced offline through the choice of the retention horizon K. Consequently, ACDPP remains resource-feasible while still achieving performance close to the costless oracle benchmark.

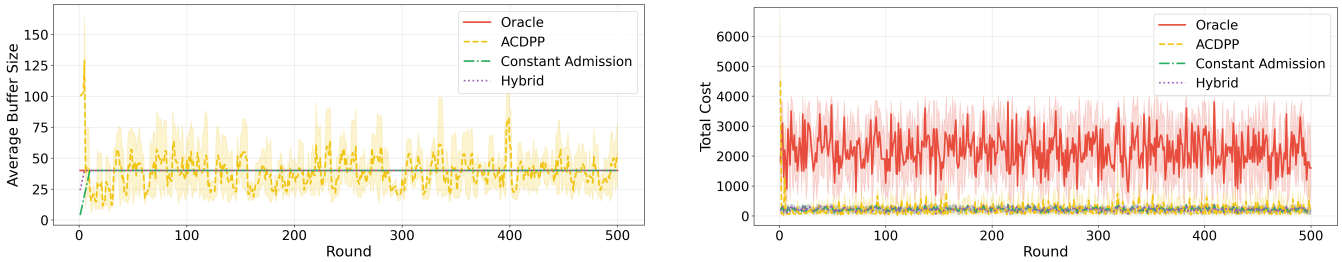
8 Conclusion

In this work, we have studied streaming FL with limited client memory and time-varying sampling costs. We consider a framework of joint server-side sample-admission and client-side memory-management with the objective of minimizing the cumulative excess population risk under sampling-cost and buffer constraints. To tackle this problem, we first derive a learning-error bound that explicitly captures the roles of instantaneous training sample size, distinct-sample growth, and reuse imbalance through an effective



(a) Test accuracy on MNIST (non-i.i.d.) (b) Test 2 on CIFAR-10 (non-i.i.d.) (c) Test accuracy on ImageNette (i.i.d.)

Figure 1: Current best test accuracy over communication rounds on MNIST, CIFAR-10, and ImageNette.



(a) Buffer occupancy over communication rounds. (b) Sampling cost evolution over communication rounds.

Figure 2: Buffer occupancy and cost evolution over communication rounds.

sample size representation. The proposed ACDPP policy leverages a novel learning-error bound that captures the impacts of sample reuse and effective sample size, as well as DPP minimization with a surrogate penalty and time-varying rectangular admission constraints. Through a multi-step comparison analysis together with Lyapunov drift arguments, we have established explicit guarantees in terms of sublinear regret and sampling-cost violation, while showing that the buffer-occupancy violation can be controlled through offline selection of the retention horizon. Experimental results on multiple datasets further demonstrate that the proposed policy achieves strong learning performance in comparison with common baselines and the oracle benchmark. An interesting direction for future work is to extend the present framework to settings with concept drift and to integrate online sample admission with labeling decisions.

References

[1] Keith Bonawitz et al. 2019. Towards federated learning at scale: System design. *Proc. Mach. Learn. Syst.* 1 (2019), 374–388.

[2] Graham Cormode and Shan Muthukrishnan. 2005. An improved data stream summary: the count-min sketch and its applications. *J. Algorithms* 55, 1 (2005).

[3] Matthias De Lange et al. 2021. A continual learning survey: Defying forgetting in classification tasks. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 7 (2021), 3366–3385.

[4] Jia Deng et al. 2009. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*. 248–255.

[5] Joao Gama et al. 2013. On evaluating stream learning algorithms. *Mach. Learn.* 90, 3 (2013), 317–346.

[6] Heitor Murilo Gomes et al. 2019. Machine learning for streaming data: state of the art, challenges, and opportunities. *ACM SIGKDD Explor. Newsl.* 21, 2 (2019).

[7] Chen Gong et al. 2024. ODE: An online data selection framework for federated learning with limited storage. *IEEE/ACM Trans. Netw.* 32, 4 (2024), 2794–2809.

[8] Kaiming He et al. 2016. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*. 770–778.

[9] Chung-Hsuan Hu et al. 2024. Energy-efficient federated edge learning with streaming data: A lyapunov optimization approach. *IEEE Trans. Commun.* (2024).

[10] Tan-Khiem Huynh et al. 2025. Streaming federated learning with Markovian data. *arXiv preprint arXiv:2503.18807* (2025).

[11] Peter Kairouz and H Brendan McMahan. 2021. Advances and open problems in federated learning. *Found. Trends Mach. Learn.* 14, 1-2 (2021), 1–210.

[12] Sai Praneeth Karimireddy et al. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *Int. Conf. Mach. Learn. (ICML)*.

[13] Alex Krizhevsky et al. 2009. Learning multiple layers of features from tiny images. (2009).

[14] Yann. LeCun. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/* (1998).

[15] Yann LeCun et al. 2002. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (2002).

[16] Othmane Marfoq et al. 2023. Federated learning for data streams. In *Int. Conf. Artif. Intell. Stat. (AISTATS)*.

[17] Brendan McMahan et al. 2017. Communication-efficient learning of deep networks from decentralized data. In *Int. Conf. Artif. Intell. Stat. (AISTATS)*.

[18] Michael Neely. 2022. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Springer Nature.

[19] Michael J Neely. 2010. Stability and capacity regions or discrete time queueing networks. *arXiv preprint arXiv:1003.3396* (2010).

[20] Jiahe Shi et al. 2023. Self-supervised on-device federated learning from unlabeled streams. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* 42, 12 (2023).

[21] Yuchang Sun et al. 2025. Learn how to query from unlabeled data streams in federated learning. In *AAAI Conf. Artif. Intell. (AAAI)*.

[22] Alysa Ziying Tan et al. 2022. Towards personalized federated learning. *IEEE Trans. Neural Netw. Learn. Syst.* 34, 12 (2022).

[23] Jianyu Wang et al. 2020. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*.

[24] Liyuan Wang et al. 2024. A comprehensive survey of continual learning: Theory, method and application. *IEEE Trans. Pattern Anal. Mach. Intell.* 46, 8 (2024).

[25] Shiqiang Wang et al. 2019. Adaptive federated learning in resource constrained edge computing systems. *IEEE J. Sel. Areas Commun.* (2019).

[26] Ruirui Zhang et al. 2024. Federating from History in Streaming Federated Learning. In *Int. Symp. Theory Algorithmic Found. Protocol Des. Mobile Netw. Mobile Comput. (MobiHoc)*. 151–160.

[27] Zhuoyi Zhao and Ben Liang. 2026. Adaptive Data Admission and Retention for Streaming Federated Learning. https://tinyurl.com/AdaDataSamplingSFL.

[28] Zhi Zhou et al. 2020. CEFL: Online Admission Control, Data Scheduling, and Accuracy Tuning for Cost-Efficient Federated Learning Across Edge Nodes. *IEEE Internet Things J.* 7, 10 (2020), 9341–9356.

Supplementary Material for the paper "Adaptive Data Admission and Retention for Streaming Federated Learning"

Zhuoyi Zhao and Ben Liang

A Proof of Theorem 1

We prove Theorem 1 by combining an optimization recursion (which yields an optimization-regret bound) with (i) a variance/bias control term expressed via the instantaneous mini-batch size n_t and the effective sample size N_{eff}^t, and (ii) a uniform generalization bound controlled by the pseudo-dimension.

A.1 Optimization regret bound from recursion

We first isolate the optimization component of the cumulative excess-risk bound. The derivation builds on the recursion-based proof framework in Appendix B.5 of [16], adapted to the present full-batch memory-based streaming federated learning model. In particular, we follow the same empirical-optimization viewpoint as [16]: the optimization recursion is first established with respect to a policy-induced empirical objective, and the gap between this empirical objective and the population objective is then controlled by a separate uniform generalization bound.

Under policy π, let

$$\omega_i^\pi(t) \triangleq \frac{A_i^\pi(t)}{\sum_{j=1}^{N^\pi(t)} A_j^\pi(t)}, \qquad i = 1, \dots, N^\pi(t), \tag{67}$$

and define the cumulative multiplicity-weighted empirical objective

$$\widehat{F}_t^{\text{all}}(\theta) \triangleq \sum_{i=1}^{N^\pi(t)} \omega_i^\pi(t)\ell(\theta; z_i). \tag{68}$$

This objective is the analogue of the global weighted empirical objective $L_S^{(\lambda)}$ in [16]. It aggregates all distinct samples used up to round t, with weights proportional to their cumulative reuse counts.

In contrast, the actual local update at round t is performed using only the samples currently stored in the clients' memories. We denote the corresponding memory-induced empirical objective by

$$\widehat{F}_t^{\text{mem}}(\theta) \triangleq \frac{1}{n^\pi(t)} \sum_{z \in Q(t)} \ell(\theta; z). \tag{69}$$

The cumulative empirical optimization error is defined as

$$\mathcal{E}_T^{\text{opt}}(\pi) \triangleq \sum_{t=1}^{T} \mathbb{E}\left[\widehat{F}_t^{\text{all}}(\theta^{(t)}) - \widehat{F}_t^{\text{all},\star}\right]. \tag{70}$$

where $\widehat{F}_t^{\text{all},\star} \triangleq \min_{\theta\in\Theta} \widehat{F}_t^{\text{all}}(\theta)$. The key round-dependent quantity is therefore the empirical gradient mismatch between the cumulative multiplicity-weighted objective and the current memory-induced objective:

$$\bar{\sigma}^2(t) \triangleq \mathbb{E}\left[\sup_{\theta\in\Theta} \left\|\nabla\widehat{F}_t^{\text{all}}(\theta) - \nabla\widehat{F}_t^{\text{mem}}(\theta)\right\|_2^2\right]. \tag{71}$$

Thus, $\bar{\sigma}^2(t)$ measures how far the gradient used by the round-t memory-based update deviates from the gradient of the policy-induced cumulative empirical objective.

The following lemma rewrites the one-step optimization recursion in Appendix B.5 of [16] under our notation and derives the corresponding cumulative empirical optimization-error bound.

Lemma 5. *Under Assumptions 1–4, the iterates $\{\theta^{(t)}\}_{t=1}^T$ generated under policy π satisfy the following one-step optimization recursion:*

$$\mathbb{E}\left[\|\theta^{(t+1)} - \theta^\star\|_2^2\right] \le \mathbb{E}\left[\|\theta^{(t)} - \theta^\star\|_2^2\right] - 2\eta\,\mathbb{E}\left[\widehat{F}_t^{\text{all}}(\theta^{(t)}) - \widehat{F}_t^{\text{all},\star}\right] + 2\eta D\,\bar{\sigma}(t) + \eta^2\left(2\bar{\sigma}^2(t) + C_1\right) + \eta^4 C_2, \tag{72}$$

where

$$C_1 \triangleq G\left(5G + 2LD\sqrt{2(1 - E^{-1})}\right), \qquad C_2 \triangleq 4L^2G^2(1 - E^{-1}), \tag{73}$$

and

$$\mathcal{E}_T^{\text{opt}}(\pi) \le \frac{D^2}{2\eta} + D\sum_{t=1}^{T}\bar{\sigma}(t) + \frac{\eta}{2}\sum_{t=1}^{T}\left(2\bar{\sigma}^2(t) + C_1\right) + \frac{\eta^3}{2}C_2T. \tag{74}$$

Proof. The one-step recursion in (72) follows directly from the derivation in Appendix B.5 of [16], after rewriting the empirical objective and the empirical gradient mismatch under the notation introduced above. The higher-order terms are regrouped into the constants C_1 and C_2 in (73).

Rearranging (72) gives

$$2\eta\mathbb{E}\left[\widehat{F}_t^{\text{all}}(\theta^{(t)}) - \widehat{F}_t^{\text{all},\star}\right] \le \mathbb{E}\left[\|\theta^{(t)} - \theta^\star\|_2^2\right] - \mathbb{E}\left[\|\theta^{(t+1)} - \theta^\star\|_2^2\right] + 2\eta D\bar{\sigma}(t) + \eta^2\left(2\bar{\sigma}^2(t) + C_1\right) + \eta^4 C_2. \tag{75}$$

Summing (75) over $t = 1, \dots, T$ telescopes the squared-distance terms:

$$\sum_{t=1}^{T}\left(\mathbb{E}\|\theta^{(t)} - \theta^\star\|_2^2 - \mathbb{E}\|\theta^{(t+1)} - \theta^\star\|_2^2\right) = \mathbb{E}\|\theta^{(1)} - \theta^\star\|_2^2 - \mathbb{E}\|\theta^{(T+1)} - \theta^\star\|_2^2 \le D^2,$$

where the last inequality follows from the boundedness of Θ and the nonnegativity of squared norms. Dividing both sides by 2η yields (74). □

A.2 Variance control via effective sample size

We next refine the empirical gradient mismatch $\bar{\sigma}^2(t)$ by explicitly separating the effects of distinct-sample growth and sample-reuse imbalance. Recall from Assumption 4 that the single-sample population-gradient variance satisfies

$$\mathbb{E}_{z\sim\mathcal{P}}\left[\|\nabla_\theta\ell(\theta; z) - \nabla F_{\mathcal{P}}(\theta)\|_2^2\right] \le \sigma^2, \qquad \forall\theta \in \Theta. \tag{76}$$

Under the effective-sample-size variance control induced by repeated sample reuse, the empirical gradient mismatch at round t

satisfies

$$\bar{\sigma}^2(t) \le \sigma^2\left(\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}\right). \tag{77}$$

Substituting (77) into Lemma 5 yields

$$\mathcal{E}_T^{\mathrm{opt}}(\pi) \le \frac{D^2}{2\eta} + D\sigma\sum_{t=1}^{T}\sqrt{\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}} + \eta\sigma^2\sum_{t=1}^{T}\left(\frac{1}{n^\pi(t)} - \frac{1}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}\right) + \frac{\eta C_1}{2}T + \frac{\eta^3 C_2}{2}T. \tag{78}$$

A.3 Generalization term controlled by pseudo-dimension

We next bound the generalization gap between the population objective $F_{\mathcal{P}}$ and the cumulative multiplicity-weighted empirical objective $\widehat{F}_t^{\mathrm{all}}$. It is essentially an application of Theorem 4.1 in [16] to our setting, with the effective sample size decomposition. As a result, the generalization bound is expressed explicitly in terms of cumulative distinct-sample growth and reuse imbalance. The key point is that, under repeated sample reuse, the effective statistical complexity is governed not only by the number of distinct samples, but also by the uniformity of their reuse.

Lemma 6 (Uniform generalization bound with effective sample size)**.** *Assume $\ell(\theta; z) \in [0, B]$ for all $\theta \in \Theta$ and z. Then for each round t, the expected generalization gap is bounded by*

$$\mathbb{E}\left[\sup_{\theta\in\Theta}\left|F_{\mathcal{P}}(\theta) - \widehat{F}_t^{\mathrm{all}}(\theta)\right|\right] \le 10B\sqrt{\frac{\mathrm{Pdim}(\ell\circ\mathcal{H})}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}}\sqrt{1 + \log\left(\frac{N^\pi(t)}{\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)}. \tag{79}$$

Proof. Recall the policy-induced weights $\omega_i^\pi(t)$ and the cumulative multiplicity-weighted empirical objective $\widehat{F}_t^{\mathrm{all}}$ defined in (67)–(68).

Consider the loss-composed hypothesis class

$$\mathcal{F} \triangleq \{z \mapsto \ell(\theta; z) : \theta \in \Theta\}, \tag{80}$$

which is uniformly bounded in $[0, B]$ by assumption. By a standard weighted symmetrization argument, the expected uniform generalization gap is bounded by twice the weighted empirical Rademacher complexity of $\mathcal{F}$. Moreover, the effective normalization is determined by the ℓ_2 norm of the weight vector:

$$\sum_{i=1}^{N^\pi(t)}(\omega_i^\pi(t))^2 = \frac{\sum_{i=1}^{N^\pi(t)}(A_i^\pi(t))^2}{\left(\sum_{i=1}^{N^\pi(t)}A_i^\pi(t)\right)^2} = \frac{1}{N_{\mathrm{eff}}^\pi(t)} = \frac{1}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}, \tag{81}$$

Applying the standard pseudo-dimension bound for bounded real-valued function classes yields

$$\mathbb{E}\left[\sup_{\theta\in\Theta}\left|F_{\mathcal{P}}(\theta) - \widehat{F}_t^{\mathrm{all}}(\theta)\right|\right] \le 10B\sqrt{\frac{\mathrm{Pdim}(\ell\circ\mathcal{H})}{N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)}}\sqrt{1 + \log\left(\frac{N^\pi(t)}{\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)}, \tag{82}$$

which proves (79). □

A.4 Proof of Theorem 1

Proof of Theorem 1. We combine the optimization-error bound in (78) with the generalization bound in Lemma 6. Specifically, (78) controls the cumulative optimization component $\mathcal{E}_T^{\mathrm{opt}}(\pi)$ with respect to the cumulative multiplicity-weighted empirical objective $\widehat{F}_t^{\mathrm{all}}$ in terms of the policy-induced training sample size $n^\pi(t)$ and effective sample size $N^\pi(t)\widetilde{\mathcal{V}}^\pi(t)$, while Lemma 6 bounds the generalization gap between the population objective $F_{\mathcal{P}}$ and $\widehat{F}_t^{\mathrm{all}}$ by a term depending on the pseudo-dimension and the effective sample size. Combining the optimization bound in (78) with the generalization bound in Lemma 6 yields the desired cumulative excess-risk upper bound. Finally, absorbing the initialization term $\frac{D^2}{2\eta}$ and other lower-order constants into the constant term C_2 gives the statement of Theorem 1. □

B Proof of Lemma 1

Consider the feasible constant-admission comparison policy associated with the benchmark operating point $(\bar{\Lambda}, K)$. For an arbitrary round t, let $\mathbb{O}(t)$ denote the realized system state induced by the proposed ACDPP policy over rounds $1, \ldots, t-1$, and suppose that at round t the server switches to the comparison admission vector $\boldsymbol{\lambda}^{CK}(t)$.

Under the K-step retention policy, each admitted sample remains in memory for exactly K rounds. Hence the instantaneous batch size and cumulative distinct sample count under the comparison policy satisfy

$$n^C(t) = n(t) - \Lambda(t-K) + \bar{\Lambda}, \qquad N^C(t) = N(t-1) + \bar{\Lambda}. \tag{83}$$

Substituting (83) into the penalty definition (29) gives the one-slot drift-plus-penalty objective

$$J_t(\boldsymbol{\lambda}^{CK}(t)) = Z(t)\big(c(t)\bar{\Lambda} - C\big) + V\,p^{CK}(t), \tag{84}$$

where $p^{CK}(t)$ is the per-round penalty induced by the comparison policy.

Next, since the admissible action space satisfies

$$\Lambda_{\min}(t) \le \Lambda(t) \le \Lambda_{\max}(t), \tag{85}$$

we have the lower bounds

$$n^C(t) \ge (K-1)\Lambda_{\min}(t-K) + \bar{\Lambda}, \tag{86}$$

and

$$N^C(t) \ge \sum_{\tau=1}^{t-1}\Lambda_{\min}(\tau) + \bar{\Lambda}. \tag{87}$$

Since the penalty function in (29) is decreasing in both the instantaneous batch size and the cumulative distinct-sample count, it follows that

$$p^{CK}(t) \le \bar{p}^{CK}(t), \tag{88}$$

where $\bar{p}^{CK}(t)$ is given by (52). Therefore,

$$J_t(\boldsymbol{\lambda}^{CK}(t)) \le Z(t)\big(c(t)\bar{\Lambda} - C\big) + V\bar{p}^{CK}(t). \tag{89}$$

Recall that the one-slot drift-plus-penalty upper bound holds for any feasible admission vector $\boldsymbol{\lambda}$:

$$\Delta(\mathbb{O}(t)) + V\mathbb{E}[p(t) \mid \mathbb{O}(t)] \le B + \mathbb{E}[J_t(\boldsymbol{\lambda}) \mid \mathbb{O}(t)]. \tag{90}$$

By definition, the ACDPP policy chooses

$$\boldsymbol{\lambda}^{\star}(t) \in \arg\min_{\boldsymbol{\lambda}} J_t(\boldsymbol{\lambda}), \tag{91}$$

and hence

$$J_t(\boldsymbol{\lambda}^{\star}(t)) \le J_t(\boldsymbol{\lambda}^{CK}(t)). \tag{92}$$

Substituting (92) into (90) yields

$$\Delta(\mathbb{O}(t)) + V\mathbb{E}[p(t) \mid \mathbb{O}(t)] \le B + \mathbb{E}[J_t(\boldsymbol{\lambda}^{CK}(t)) \mid \mathbb{O}(t)]. \tag{93}$$

Using (89), we obtain

$$\Delta(\mathbb{O}(t))+V\mathbb{E}[p(t)\,|\,\mathbb{O}(t)] \le B+\mathbb{E}\big[Z(t)\big(c(t)\bar{\Lambda}-C\big)\,|\,\mathbb{O}(t)\big]+V\bar{p}^{CK}(t). \tag{94}$$

Now take total expectation. Since the sampling-cost process $\{c(t)\}$ is independent of the queue state $Z(t)$ under the comparison policy, we have

$$\mathbb{E}\big[Z(t)\big(c(t)\bar{\Lambda}-C\big)\big] = \mathbb{E}[Z(t)]\,\mathbb{E}\big[c(t)\bar{\Lambda}-C\big]. \tag{95}$$

Because the benchmark comparison policy is feasible, it satisfies the long-term sampling-cost constraint, and therefore

$$\mathbb{E}\big[c(t)\bar{\Lambda}-C\big] \le 0. \tag{96}$$

Hence (94) implies

$$\mathbb{E}[L(t+1)-L(t)] + V\mathbb{E}[p(t)] \le B + V\bar{p}^{CK}(t). \tag{97}$$

Summing (97) over $t = 1, \dots, T$ gives

$$\mathbb{E}[L(T+1)-L(1)] + V\sum_{t=1}^{T}\mathbb{E}[p(t)] \le BT + V\sum_{t=1}^{T}\bar{p}^{CK}(t), \tag{98}$$

which proves (51).

C Proof of Lemma 2

The only difference between the finite-horizon learning upper bound $\bar{\mathcal{E}}_T^{\mathrm{ACDPP}}$ and the cumulative surrogate penalty $\sum_{t=1}^{T}\mathbb{E}[p(t)]$ comes from the reuse-uniformity factor $\widetilde{\mathcal{V}}^{\pi}(t)$, since the surrogate penalty replaces $\frac{1}{N^{\pi}(t)\widetilde{\mathcal{V}}^{\pi}(t)}$ by $\frac{1}{N^{\pi}(t)}$.

Here and below, $O_t(\cdot)$ and $\Theta_t(\cdot)$ denote big-O and big-Θ with respect to the per-round index t. Under the K-step retention policy, only the samples admitted in the most recent $K-1$ rounds fail to complete their full K reuse cycles by round t. Therefore, the first- and second-moment deficits of the reuse counts are both $O_t(1)$, while the cumulative number of distinct samples satisfies $N^{\pi}(t) = \Theta_t(t)$. It follows that

$$1 - \widetilde{\mathcal{V}}^{\pi}(t) = O_t(t^{-1}), \tag{99}$$

and hence

$$\frac{1}{N^{\pi}(t)\widetilde{\mathcal{V}}^{\pi}(t)} - \frac{1}{N^{\pi}(t)} = \frac{1-\widetilde{\mathcal{V}}^{\pi}(t)}{N^{\pi}(t)\widetilde{\mathcal{V}}^{\pi}(t)} = O_t(t^{-2}). \tag{100}$$

Now compare the terms in the learning upper bound with their surrogate counterparts. Since $n^{\pi}(t) = \Theta_t(1)$ under K-step retention and $N^{\pi}(t) = \Theta_t(t)$, the quantity $\frac{1}{n^{\pi}(t)} - \frac{1}{N^{\pi}(t)\widetilde{\mathcal{V}}^{\pi}(t)}$ remains bounded away from zero for all sufficiently large t. Therefore, the square-root function is locally Lipschitz in a neighborhood of this argument, and the square-root term differs from its surrogate counterpart by at most $O_t(t^{-2})$. The same order follows immediately for the linear term.

For the pseudo-dimension term, define

$$x_t \triangleq \frac{1}{N^{\pi}(t)\widetilde{\mathcal{V}}^{\pi}(t)}, \qquad y_t \triangleq \frac{1}{N^{\pi}(t)}.$$

By (100), we have $x_t - y_t = O_t(t^{-2})$. Since the map

$$x \mapsto 10B\sqrt{\mathrm{Pdim}(\ell\circ\mathcal{H})\,x}\sqrt{1+\log\left(\frac{1}{x\,\mathrm{Pdim}(\ell\circ\mathcal{H})}\right)}$$

is locally Lipschitz for sufficiently small positive x, the pseudo-dimension term also differs from its surrogate counterpart by at most $O_t(t^{-2})$.

Combining the square-root, linear, and pseudo-dimension terms yields a per-round discrepancy of order $O_t(t^{-2})$. Summing over $t = 1, \dots, T$ gives

$$\bar{\mathcal{E}}_T^{\mathrm{ACDPP}} - \sum_{t=1}^{T}\mathbb{E}[p(t)] = O(1), \tag{101}$$

which proves Lemma 2.

D Proof of Lemma 3

Under the time-varying rectangular constraint (35), the lower bounds on the instantaneous training sample size and the cumulative number of distinct samples are

$$n_{\mathrm{lb}}(t) = K\bar{\Lambda} - (K-1)\bar{\Lambda}\,\rho^{\,t-K}, \qquad N_{\mathrm{lb}}(t) = \bar{\Lambda}t - \bar{\Lambda}\sum_{\tau=1}^{t-1}\rho^{\tau}. \tag{102}$$

It follows from (102) that, for all $t \ge K$,

$$\begin{aligned}\frac{1}{n_{\mathrm{lb}}(t)} - \frac{1}{\bar{\Lambda}K} &= \frac{1}{\bar{\Lambda}}\left(\frac{1}{K-(K-1)\rho^{t-K}} - \frac{1}{K}\right)\\ &= \frac{(K-1)\rho^{t-K}}{\bar{\Lambda}K\big(K-(K-1)\rho^{t-K}\big)} \le \kappa_1\rho^{t-K},\end{aligned} \tag{103}$$

where the inequality follows from $K-(K-1)\rho^{t-K} \ge 1$, and $\kappa_1 = \frac{K-1}{\bar{\Lambda}K}$.

Similarly, because

$$\sum_{\tau=1}^{t-1}\rho^{\tau} = \frac{\rho(1-\rho^{t-1})}{1-\rho} \le \frac{1}{1-\rho},$$

we have

$$N_{\mathrm{lb}}(t) = \bar{\Lambda}t - \delta_t, \qquad 0 \le \delta_t \le \frac{\bar{\Lambda}}{1-\rho}.$$

Therefore,

$$\frac{1}{N_{\mathrm{lb}}(t)} - \frac{1}{\bar{\Lambda}t} = \frac{\delta_t}{(\bar{\Lambda}t)(\bar{\Lambda}t-\delta_t)}.$$

Using the bound on δ_t, we obtain

$$\frac{1}{N_{\mathrm{lb}}(t)} - \frac{1}{\bar{\Lambda}t} \le \frac{\bar{\Lambda}/(1-\rho)}{(\bar{\Lambda}t)\,\big(\bar{\Lambda}t-\bar{\Lambda}/(1-\rho)\big)} = \frac{1}{\bar{\Lambda}\big((1-\rho)t^2-t\big)}. \tag{104}$$

We now compare $\bar{p}^{CK}(t)$ with $\bar{p}_{\infty}^{CK}(t)$ term by term. For the square-root term, define

$$u_t \triangleq \frac{1}{n_{\mathrm{lb}}(t)} - \frac{1}{N_{\mathrm{lb}}(t)}, \qquad v_t \triangleq \frac{1}{\bar{\Lambda}K} - \frac{1}{\bar{\Lambda}t}.$$

By (??) and (104),

$$u_t - v_t \le \kappa_1\rho^{\,t-K} + \frac{1}{\bar{\Lambda}\big((1-\rho)t^2-t\big)}.$$

Since v_t is bounded away from zero for all sufficiently large t, the map $x \mapsto \sqrt{x}$ is locally Lipschitz around v_t. Therefore, there exists a constant $\kappa_2 > 0$ such that the square-root term in $\bar{p}^{CK}(t)$ differs from its counterpart in $\bar{p}_\infty^{CK}(t)$ by at most

$$\kappa_2 \rho^{\,t-K} + \frac{\kappa_2}{(1-\rho)t^2 - t}.$$

The same bound also applies to the linear term.

For the pseudo-dimension term, the only dependence is through the reciprocal sample-size factor and the logarithmic factor. Using (104), together with the local Lipschitz continuity of the map

$$x \mapsto 10B\sqrt{\mathrm{Pdim}(\ell \circ \mathcal{H})\, x}\sqrt{1 + \log\left(\frac{1}{x\,\mathrm{Pdim}(\ell \circ \mathcal{H})}\right)},$$

there exists a constant $\kappa_3 > 0$ such that the pseudo-dimension term differs by at most

$$\kappa_3 \rho^{\,t-K} + \frac{\kappa_3}{(1-\rho)t^2 - t}.$$

Combining the three terms, there exists a constant $\kappa_4 > 0$ such that

$$\left|\bar{p}^{CK}(t) - \bar{p}_\infty^{CK}(t)\right| \le \kappa_4 \rho^{\,t-K} + \frac{\kappa_4}{(1-\rho)t^2 - t}, \qquad t \ge K. \quad (105)$$

Summing (105) over $t = 1, \ldots, T$ gives

$$\begin{aligned}\sum_{t=1}^{T}\left|\bar{p}^{CK}(t) - \bar{p}_\infty^{CK}(t)\right| \le \sum_{t=1}^{K-1}\left|\bar{p}^{CK}(t) - \bar{p}_\infty^{CK}(t)\right| + \kappa_4 \sum_{t=K}^{T} \rho^{\,t-K} \\ + \kappa_4 \sum_{t=1}^{T} \frac{1}{(1-\rho)t^2 - t}.\end{aligned} \quad (106)$$

The first term is finite because it contains only finitely many indices. The second term is bounded by

$$\kappa_4 \sum_{j=0}^{\infty} \rho^j = \frac{\kappa_4}{1-\rho}.$$

For the third term, let

$$t_0 \triangleq \left\lceil \frac{2}{1-\rho} \right\rceil.$$

We split the sum as

$$\sum_{t=1}^{T} \frac{\kappa_4}{(1-\rho)t^2 - t} = \sum_{t=1}^{t_0-1} \frac{\kappa_4}{(1-\rho)t^2 - t} + \sum_{t=t_0}^{T} \frac{\kappa_4}{(1-\rho)t^2 - t}.$$

Since $t_0 = O((1-\rho)^{-1})$ and the per-round discrepancy is uniformly bounded, the first sum contributes $O((1-\rho)^{-1})$. For the second sum, when $t \ge t_0$, we have

$$t \ge \frac{2}{1-\rho},$$

which implies

$$\frac{1}{2}(1-\rho)t^2 \ge t,$$

or equivalently,

$$(1-\rho)t^2 - t \ge \frac{1}{2}(1-\rho)t^2.$$

Hence, when $t >= t_0$,

$$\frac{\kappa_4}{(1-\rho)t^2 - t} \le \frac{2\kappa_4}{(1-\rho)t^2}.$$

Therefore,

$$\sum_{t=t_0}^{T} \frac{\kappa_4}{(1-\rho)t^2 - t} \le \frac{2\kappa_4}{1-\rho} \sum_{t=t_0}^{\infty} \frac{1}{t^2} = O\left(\frac{1}{1-\rho}\right).$$

Combining the two parts yields

$$\sum_{t=1}^{T} \frac{1}{(1-\rho)t^2 - t} = O\left(\frac{1}{1-\rho}\right).$$

Hence

$$\sum_{t=1}^{T}\left|\bar{p}^{CK}(t) - \bar{p}_\infty^{CK}(t)\right| = O\left(\frac{1}{1-\rho}\right), \quad (107)$$

which proves Lemma 3.

E Proof of Lemma 4

Recall that, under the costless oracle policy, the cumulative learning upper bound takes the form

$$\begin{aligned}\bar{\mathcal{E}}_T^{\mathrm{oracle}} = \sum_{t=1}^{T}\Bigg[D\sigma\sqrt{\frac{1}{\bar{B}} - \frac{1}{\bar{B}t}} + \eta\sigma^2\left(\frac{1}{\bar{B}} - \frac{1}{\bar{B}t}\right) \\ + 10B\sqrt{\frac{\mathrm{Pdim}(\ell \circ \mathcal{H})}{\bar{B}t}}\sqrt{1 + \log\left(\frac{\bar{B}t}{\mathrm{Pdim}(\ell \circ \mathcal{H})}\right)}\Bigg],\end{aligned} \quad (108)$$

where $\bar{B} \triangleq \sum_{m=1}^{M} B_m$.

Compare the per-round term in $\bar{p}_\infty^{CK}(t)$ with the corresponding per-round term in (108). Under the condition $\bar{\Lambda}K = \bar{B}$, the steady-state terms coincide exactly:

$$\frac{1}{\bar{\Lambda}K} = \frac{1}{\bar{B}}.$$

Thus the only mismatch comes from the distinct-sample-growth terms

$$\frac{1}{\bar{\Lambda}t} \qquad \text{and} \qquad \frac{1}{\bar{B}t}.$$

Their difference is

$$\frac{1}{\bar{\Lambda}t} - \frac{1}{\bar{B}t} = \frac{1}{\bar{\Lambda}t} - \frac{1}{\bar{\Lambda}K\,t} = \frac{K-1}{\bar{B}\,t} = O_t(t^{-1}). \quad (109)$$

We again compare the three terms separately. For the square-root term, since the underlying argument is bounded away from zero for sufficiently large t, the square-root map is locally Lipschitz, and the discrepancy is therefore $O_t(t^{-1})$. The same order follows immediately for the linear term.

For the pseudo-dimension term, note that

$$10B\sqrt{\frac{\mathrm{Pdim}(\ell \circ \mathcal{H})}{\bar{\Lambda}t}}\sqrt{1 + \log\left(\frac{\bar{\Lambda}t}{\mathrm{Pdim}(\ell \circ \mathcal{H})}\right)}$$

and its oracle counterpart differ only through the factor $1/(\bar{\Lambda}t)$ versus $1/(\bar{B}t)$. Since the map

$$x \mapsto 10B\sqrt{\mathrm{Pdim}(\ell \circ \mathcal{H})\, x}\sqrt{1 + \log\left(\frac{1}{x\,\mathrm{Pdim}(\ell \circ \mathcal{H})}\right)}$$

is locally Lipschitz for sufficiently small positive x, the resulting discrepancy is of lower order than t^{-1} and can be absorbed into the same bound.

Hence the per-round difference between $\bar{p}_{\infty}^{CK}(t)$ and the oracle term is $O_t(t^{-1})$. Summing over $t = 1, \ldots, T$ yields

$$\sum_{t=1}^{T} \bar{p}_{\infty}^{CK}(t) - \bar{\mathcal{E}}_T^{\text{oracle}} = O(\log T), \tag{110}$$

which proves Lemma 4.

F Proof of Theorem 3

From Lemma 1 and the nonnegativity of the penalty term,

$$\mathbb{E}[L(T+1) - L(1)] \le BT + V \sum_{t=1}^{T} \bar{p}^{CK}(t). \tag{111}$$

Using Lemma 3, we have

$$\sum_{t=1}^{T} \bar{p}^{CK}(t) = \sum_{t=1}^{T} \bar{p}_{\infty}^{CK}(t) + O\left(\frac{1}{1-\rho}\right).$$

Since each term $\bar{p}_{\infty}^{CK}(t)$ is uniformly bounded in t, it follows that

$$\sum_{t=1}^{T} \bar{p}^{CK}(t) = O(T) + O\left(\frac{1}{1-\rho}\right). \tag{112}$$

Substituting (112) into (111) yields

$$\mathbb{E}[L(T+1)] = O(VT) + O\left(\frac{V}{1-\rho}\right). \tag{113}$$

Since $L(t) = \frac{1}{2} Z(t)^2$, it follows that

$$\mathbb{E}[Z(T+1)^2] = O(VT) + O\left(\frac{V}{1-\rho}\right), \tag{114}$$

and therefore, by Jensen's inequality,

$$\mathbb{E}[Z(T+1)] = O\left(\sqrt{VT + \frac{V}{1-\rho}}\right). \tag{115}$$

By the standard virtual-queue argument in Lyapunov optimization (e.g., [19, Theorem 2.8]), the cumulative constraint violation is upper bounded by $\mathbb{E}[Z(T+1)]$. Therefore,

$$\text{Vio}^{\text{S}}(T) \le O\left(\sqrt{VT + \frac{V}{1-\rho}}\right), \tag{116}$$

which proves (63).