Personalized Federated $\mathcal{X}$-armed Bandit

Wenjie Li
Purdue University [1]

Jean Honorio
The University of Melbourne

Qifan Song
Purdue University

Abstract

In this work, we study the personalized federated $\mathcal{X}$-armed bandit problem, where the heterogeneous local objectives of the clients are optimized simultaneously in the federated learning paradigm. We propose the `PF-PNE` algorithm with a unique double elimination strategy, which safely eliminates the non-optimal regions while encouraging federated collaboration through biased but effective evaluations of the local objectives. The proposed `PF-PNE` algorithm is able to optimize local objectives with arbitrary levels of heterogeneity, and its limited communications protects the confidentiality of the client-wise reward data. Our theoretical analysis shows the benefit of the proposed algorithm over single-client algorithms. Experimentally, `PF-PNE` outperforms multiple baselines on both synthetic and real life datasets.

1 INTRODUCTION

Federated bandit is a novel research area that combines sequential decision-making with federated learning, addressing data heterogeneity and privacy protection concerns for trustworthy machine learning [McMahan et al., 2017, Shi and Shen, 2021a, Zhu et al., 2021]. Unlike traditional bandit models that focus solely on the exploration-exploitation tradeoff, federated bandit also considers the implications of modern data privacy concerns. Federated learning involves data from non-i.i.d. distributions, making collaborations between clients essential for accurate inferences for the global model. However, due to the concerns of communication cost and user privacy, these collaborations must be limited, and direct local data transmission is avoided. To make accurate decisions, clients must coordinate their exploration and exploitation, utilizing minimal communication among them.

Most existing federated bandit research has mainly focused on finite arms (i.e., multi-armed bandit) or linear contextual bandits, where the expected reward is a linear function of the chosen contextual vector [Shi and Shen, 2021a, Shi et al., 2021b, Huang et al., 2021, Dubey and Pentland, 2020]. Some recent works on neural bandits have extended the results to nonlinear reward functions Zhang et al. [2020], Dai et al. [2023]. However, more complicated problems such as dynamic pricing and hyper-parameter optimization require solutions for domains with infinite or even uncountable cardinality, posing challenges to the current federated bandit algorithms' applicability in real-world scenarios. For example, when deploying base stations for different locations, several hyper-parameters need to be tuned for the best performance of the base stations. The hyper-parameters are often chosen from a fixed domain, e.g., a hypercube in $\mathbb{R}^d$. The best set of hyper-parameters for different locations could be different, but the performance of a fixed set of hyper-parameters should be similar for locations that are close to each other, thus encouraging federated learning.

Several kernelized bandit algorithms are proposed to address such problems with nonlinear rewards and infinite arm domains [Chowdhury and Gopalan, 2017, Li et al., 2022a]. However, these works are based on very different assumptions from ours and have relatively high computational costs. The only work closely related to our research is Li et al. [2022b]. However, they only consider optimizing the cumulative regret on the *global objective*, which refers to the average of all the client-wise local objectives and thus the best point "on average". In our paper, we aim to optimize all the local objectives at the same time so that each client locates its own optimum. This is much more challenging but beneficial to real applications. We compare our work with some of the existing works in Table 1.

We highlight our major contributions as follows.

- **Personalized federated $\mathcal{X}$-armed bandit.** We propose the personalized federated $\mathcal{X}$-armed bandit problem, where different clients optimize their

Proceedings of the 27[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

Table 1: Comparison of the average regret upper bounds, the communication cost for sufficiently large T and the other properties. **Columns**: "Commun. rounds" refer to the number of communication rounds. "Personalized" refers to whether the local objectives or the global objectives are optimized. **Rows:** `Centralized` results are adapted from the single-client $\mathcal{X}$-armed bandit algorithms such as `HOO` [Bubeck et al., 2011] and `HCT` [Azar et al., 2014] by assuming that the server makes all the decisions with access to all client-wise information. `Fed-PNE` is a federated $\mathcal{X}$-armed bandit algorithm that optimizes the global regret and is thus not personalized. Therefore, the comparisons with these two algorithms are not completely fair. **Notations:** M denotes the total number of clients; T denotes the time horizon; for simplicity of comparison, we assume that all objectives $f_1, f_2, \cdots, f_M, \overline{f}$ share the same the near-optimality dimension denoted by d (in Definition 1).; $d_{\text{new}} \leq d$ is the optimality-difference dimension (in Definition 2).

Bandit algorithms	Average Regret	Commun. rounds	Personalized
`HOO` (Bubeck et al. [2011])	$\widetilde{\mathcal{O}}\left(T^{\frac{d+1}{d+2}}\right)$	N.A.	✓
`BLiN` (Feng et al. [2021])	$\widetilde{\mathcal{O}}\left(T^{\frac{d+1}{d+2}}\right)$	N.A.	✓
`Centralized`*	$\widetilde{\mathcal{O}}\left(M^{-\frac{1}{d+2}}T^{\frac{d+1}{d+2}}\right)$	$\mathcal{O}(MT)$	✗
`Fed-PNE`* (Li et al. [2022b])	$\widetilde{\mathcal{O}}\left(M^{-\frac{1}{d+2}}T^{\frac{d+1}{d+2}}\right)$	$\widetilde{\mathcal{O}}(M\log T)$	✗
`PF-PNE` (this work)	$\widetilde{\mathcal{O}}\left(M^{-\frac{1}{d+2}}T^{\frac{d+1}{d+2}}+T^{\frac{d_{\text{mew}}+1}{d_{\text{new}}+2}}\right)$	$\widetilde{\mathcal{O}}(M\log T)$	✓

own local objectives defined on a domain $\mathcal{X}$. The new problem is much more challenging than prior research due to the heterogeneity of the local objectives and the limited communication.

- `PF-PNE` **algorithm and double elimination.** We propose the first algorithm, `PF-PNE` to solve the personalized federated $\mathcal{X}$-armed bandit problem. The algorithm incorporates a novel double elimination strategy which guarantees that the non-optimal regions are only eliminated after thorough checks. The first round of elimination removes potential non-optimal regions and the second round of elimination uses biased evaluations, obtained from sever-client communications, to avoid redundant evaluations and unnecessary costs.
- **Theoretical analysis and empirical evidence.** Theoretically, we prove that the proposed algorithm enjoys a $\widetilde{\mathcal{O}}\left(M^{\frac{d_{\min}+1}{d_{\min}+2}}T^{\frac{d_{\min}+1}{d_{\min}+2}}+MT^{\frac{d_{\text{new}}+1}{d_{\text{new}}+2}}\right)$ regret bound (where $d_{\min}$ and d_{new} are defined in Definitions 1 and 2). When the local objectives are similar, d_{new} is very small, yielding smaller average regret than single client algorithms. Moreover, the algorithm only requires limited communications in total, which greatly protects user data confidentiality. Empirically, we provide evidence to support our theoretical claims on both synthetic objectives and real-life datasets. `PF-PNE` outperforms existing centralized and federated bandit algorithm baselines.

2 PRELIMINARIES

In this section, we discuss the concepts, notations and assumptions used in this paper, most of which follow those used in Li et al. [2022b]. For an integer $N \in \mathbb{N}$, $[N]$ is used to represent the set of positive integers no larger than N, i.e., $\{1, 2, \cdots, N\}$. For a set $\mathcal{A}$, $|\mathcal{A}|$ denotes the number of elements in $\mathcal{A}$. For a real number $a \in \mathbb{R}$, we use $\lceil a \rceil$ and $\lfloor a \rfloor$ to represent the smallest integer larger than a, and the largest integer smaller than a respectively. Throughout this paper, we use the subscript notation to represent the client (local) side definitions, e.g., the local objective f_m and the local near-optimality dimension d_m. We use the overline notation to represent the server (global) side definitions, e.g., the global objective $\overline{f}$ and the global near-optimality dimension $\overline{d}$. In big-$\mathcal{O}$ notations, we use $\widetilde{\mathcal{O}}(\cdot)$ to hide the logarithmic terms, i.e., for two functions $a(n), b(n)$, $a(n) = \widetilde{\mathcal{O}}(b(n))$ represents that $a(n)/b(n) \leq \log^k(n), \forall n > 0$ for some $k > 0$.

2.1 Problem Setting

We denote the available measurable space of arms as $\mathcal{X}$. In accordance with the practical applications, we formulate the problem setting as follows: we assume that in total $M \in \mathbb{N}$ clients want to collaboratively solve their problems, and thus M *local objectives* are available, denoted by $\{f_m\}_{m=1}^M$, all defined on the same space $\mathcal{X}$ and bounded by $[0, 1]$. These local objectives could be non-convex, non-differentiable and even non-continuous. With a limited budget T, each client can only evaluate its own local objective once per round by choosing an arm $x_{m,t} \in \mathcal{X}$ at each round $t \in [T]$

and then observes a noisy feedback $r_{m,t} \in [0,1]$ defined as $r_{m,t} := f_m(x_{m,t}) + \epsilon_{m,t}$, where $\epsilon_{m,t}$ is a zero-mean and bounded random noise independent from previous evaluations and other clients' evaluations.

Similar to the prior federated bandit works such as Shi and Shen [2021a], Huang et al. [2021], Li et al. [2022b], we assume that a central server exists and it is able to communicate with the clients in every round. To protect user privacy and confidentiality, the central server can only share summary statistics of the rewards (e.g., the empirical mean and variance) from different clients. The original rewards of each evaluation should be kept confidential. The clients are not allowed to communicate with each other and we assume that the server and all the clients are completely synchronized [McMahan et al., 2017, Shi and Shen, 2021a]. Note that the number of clients M could be very large and thus incurring very high communication costs when the clients choose to communicate with the server. Therefore, we need to take into consideration such costs in the algorithm design and the analysis. This work aims to design an algorithm, that adapts to the heterogeneity among local objectives, such that collaborative search helps when the local objectives are similar.

2.2 Performance Measure

In the setting of Li et al. [2022b], the clients are required to jointly solve for the *global maximizer*, i.e., the objective is to find the point x that maximizes the *global objective*, $\overline{f}(x) := \frac{1}{M}\sum_{m=1}^{M} f_m(x)$. However, as we have mentioned in the examples in Section 1, very often the global maximizer is not the best option for every client and the clients would want to maximize their own benefit by finding the maximizer of their local objectives. Therefore, instead of the *global regret* defined in Li et al. [2022b] where the performance of the clients is measured on the global objective, we want to minimize the expectation of the *local cumulative regret*, defined as follows

$$R(T) = \sum_{t=1}^{T}\sum_{m=1}^{M} f_m^* - \sum_{t=1}^{T}\sum_{m=1}^{M} f_m(x_{m,t})$$

where f_m^* denotes the optimal value of f_m on $\mathcal{X}$ and similar notation is used for $\overline{f}$. In order to find their own optimum, the clients can only utilize the noisy evaluations $r_{m,t}$ of their own local objective functions f_m, and the information communicated with the central server. Moreover, it is expected that some assumptions on the similarity between the local objectives and the global objective are necessary so that the communications are useful. Otherwise, the local objectives could be completely different and collaboration among the clients would be meaningless. We will discuss our assumption in Section 2.4.

2.3 Hierarchical Partitioning

Similar to the existing works on $\mathcal{X}$-armed bandit [e.g., Azar et al., 2014, Shang et al., 2019, Bartlett et al., 2019, Li et al., 2022b], our algorithms rely on the recursively-defined hierarchical partitioning $\mathcal{P} := \{\mathcal{P}_{h,i}\}_{h,i}$ of the parameter space $\mathcal{X}$. The hierarchical partition discretizes the space $\mathcal{X}$ into several nodes on each layer by the following relationship:

$$\mathcal{P}_{0,1} := \mathcal{X}, \qquad \mathcal{P}_{h,i} := \bigcup_{j=0}^{k-1} \mathcal{P}_{h+1,ki-j},$$

where for every node $\mathcal{P}_{h,i}$ inside the partition, h and i represent its depth and index respectively. For each $h \geq 0, i > 0, \{\mathcal{P}_{h+1,ki-j}\}_{j=0}^{k-1}$ are disjoint children nodes of the node $\mathcal{P}_{h,i}$ and k is the number of children for one node. The union of all the nodes on each depth h equals the parameter set $\mathcal{X}$. The partition is settled and shared with all the clients and the central server before the federated learning process because the partition is deterministic and contains no information of the reward evaluations.

2.4 Assumptions

We first present the assumptions that are also observed in prior $\mathcal{X}$-armed bandit works [Bubeck et al., 2011, Azar et al., 2014, Grill et al., 2015, Li et al., 2022b].

Assumption 1. (Dissimilarity Function) *The space $\mathcal{X}$ is equipped with a dissimilarity function $\ell : \mathcal{X}^2 \mapsto \mathbb{R}$ such that $\ell(x,x') \geq 0, \forall (x,x') \in \mathcal{X}^2$ and $\ell(x,x) = 0$*

We will assume that Assumption 1 is satisfied throughout this work. Given the dissimilarity function ℓ, the diameter of a set $\mathcal{A} \subset \mathcal{X}$ is defined as $\operatorname{diam}(\mathcal{A}) = \sup_{x,y\in\mathcal{A}} \ell(x,y)$. The open ball of radius r and with center c is then defined as $\mathcal{B}(c,r) = \{x \in \mathcal{X} : \ell(x,c) \leq r\}$. We now introduce the local smoothness assumptions.

Assumption 2. (Local Smoothness) *We assume that there exist constants $\nu_1, \nu_2 > 0$, and $0 < \rho < 1$ such that for all nodes $\mathcal{P}_{h,i}, \mathcal{P}_{h,j} \in \mathcal{P}$ on depth h,*

- $\operatorname{diam}(\mathcal{P}_{h,i}) \leq \nu_1 \rho^h$
- $\exists x^\circ_{h,i} \in \mathcal{P}_{h,i}$ *s.t.* $\mathcal{B}_{h,i} := \mathcal{B}\left(x^\circ_{h,i}, \nu_2\rho^h\right) \subset \mathcal{P}_{h,i}$
- $\mathcal{B}_{h,i} \cap \mathcal{B}_{h,j} = \emptyset$ *for all* $1 \leq i < j \leq k^h$.
- *For any objective $f \in \{f_1, f_2, \cdots, f_M\} \bigcup \{\overline{f}\}$, it satisfies that for all $x, y \in \mathcal{X}$, we have*

$$f^* - f(y) \leq f^* - f(x) + \max\{f^* - f(x), \ell(x,y)\}$$

Remark 2.1. In other words, we assume that all the local objectives as well as the global objective satisfy the smoothness property. Similar to the existing works on the $\mathcal{X}$-armed bandit problem, our proposed algorithm `PF-PNE` does not need the dissimilarity function ℓ as an explicit input. Only the smoothness constants ν_1, ρ are used in the objective [Bubeck et al., 2011, Azar et al., 2014]. As mentioned by Bubeck et al. [2011], Grill et al. [2015], Li et al. [2022b], most regular functions satisfy Assumption 2 on the standard equal-sized partition with accessible ν_1 and ρ.

Next, we present the additional assumption(s) on the similarity among the local objectives for the benefit of federated learning.

Assumption 3. (Difference in Optimal Values). *The global optimal value and the local optimal values are bounded by some (known) constant Δ, i.e., $\forall m \in [M]$, the following property is satisfied*

$$|\overline{f}^* - f_m^*| \leq \Delta$$

Remark 2.2. Assumption 3 is very weak since we only want an upper bound on the difference in the optimal values of $\overline{f}^*$ and all the f_m^*. Note that all the objectives are bounded, therefore we always have $\Delta \leq 1$. However, setting a very large Δ would basically mean that we have no prior knowledge of the similarity between the local objectives and they can be very different.

Assumption 4. (Near-optimal Similarity). *Let Δ be the upper bound on the difference in local and global optimum in Assumption 3. At any ϵ-near-optimal point of the global objective $\overline{f}$, the local objective is at least $(\omega\epsilon + \Delta)$-near-optimal for some $\omega \geq 1$, i.e., if $x \in \mathcal{X}, \epsilon > 0$ satisfies $\overline{f}^* - \overline{f}(x) \leq \epsilon$, then*

$$f_m^* - f_m(x) \leq \Delta + \omega\epsilon, \forall m \in [M]$$

Remark 2.3. Note that Assumption 4 is also mild because we only require near-optimal points in the global objective to be near-optimal on local objectives, with the optimality difference thresholded by the number Δ and the factor ω. Such an assumption also makes sense in real life. For example, when we tune hyper-parameters on machine learning models, if one set of hyper-parameters achieves 0.8 reward (e.g., accuracy) globally on average, then we should expect that the reward for the same set of hyper-parameters on the local objectives is not too bad, say, less than 0.7. Compared with assumptions that require everywhere similarity, e.g., $|\overline{f}(x) - f_m(x)| \leq \epsilon$ for every $x \in \mathcal{X}$ and some small $\epsilon > 0$, Assumption 4 is obviously much weaker.

3 ALGORITHM AND ANALYSIS

Challenges. The personalized federated $\mathcal{X}$-armed bandit problem encounters several new challenges. First of all, instead of optimizing the *global objective* as in Shi and Shen [2021a], Li et al. [2022b], the algorithms need to optimize all the local objectives of the clients at the same time, which is obviously much harder. Besides, we have no assumptions on the difference between the local objectives except for Assumption 4. Therefore, the algorithm needs to adapt to the heterogeneity in the local objectives, i.e., more collaborations should be encouraged when they are similar, and reckless collaborations should be prevented when they are different. Last but not least, the federated learning setting implies that only limited information, e.g., summary statistics such as the empirical average of the rewards, can be shared across the clients within limited communication rounds.

We first introduce a naive but insightful idea to solve the personalized federated $\mathcal{X}$-armed bandit problem, which inspires our final algorithm. The approach is quite straightforward: we can ask the clients to collaboratively eliminate some regions of the domain $\mathcal{X}$, and narrow the search region of the local optimums down to a smaller subdomain. The clients can then continue the learning process by restricting their domain to the small subdomain instead of the original $\mathcal{X}$. For example, if the original parameter domain is $\mathcal{X} = [0, 1]$, the clients could first collaboratively learn the "near-optimal" subdomain that is good for every client, say $\mathcal{X}' = [0.5, 0.6]$, i.e., $f_m(x) - f_m^*$ is small for every m on $\mathcal{X}'$. Then the clients can individually finetune the subdomain to find their local optimums. However, such an approach suffers from the problem of which region could be safely eliminated, since the local objectives could be very different on the non-near-optimal regions of $\overline{f}$ (and thus not breaking Assumption 4). On one hand, if the optimum of any client is eliminated in the collaborative learning process, then linear regret will be induced. On the other hand, if we can only eliminate a small part from $\mathcal{X}$, then the collaboration between the clients will simply be ineffective.

3.1 The `PF-PNE` Algorithm

Algorithm Details and Double Elimination. Based on the above preliminary idea and its potential issues, we propose the new Personalized-Federated-Phased-Node-Elimination (`PF-PNE`) algorithm that has a unique *double-elimination strategy* so that collaboration among the clients are encouraged while no nodes would be readily removed without careful checks. The algorithm details are shown in Algo-

Algorithm 1 PF-PNE: server

Input: k-nary partition $\mathcal{P}$, smoothness parameters ν_1, ρ, transition layer H_0
Initialize $\mathcal{K}^1 = \{(0,1)\}, h = 0$
while not reaching the time horizon T **do**
  Update $h = h + 1$
  if $h < H_0$ **then**
    Receive local estimates $\{\widehat{\mu}_{m,h,i}\}_{m\in[M],(h,i)\in\mathcal{K}^p}$ from all the clients
    for every $(h,i) \in \mathcal{K}^h$ **do**
      Calculate the global mean estimate $\overline{\mu}_{h,i} = \frac{1}{M}\sum_{m=1}^{M} \widehat{\mu}_{m,h,i}$
    end for
    Compute $(h, i^p) = \arg\max_{(h,i)\in\mathcal{K}^h} \overline{\mu}_{h,i}$
    Compute $\mathcal{E}^h = \big\{(h,i) \in \mathcal{K}^h \mid \overline{\mu}_{h,i} + b_{h,i} + \nu_1\rho^h < \overline{\mu}_{h,i^p} - b_{h,i^p}\big\}$
    Update $\mathcal{K}^h = \mathcal{K}^h \backslash \mathcal{E}^h$
    Broadcast the new set $\mathcal{K}^h$ and the statistics $\{\overline{\mu}_{h,i}, b_{h,i}\}_{(h,i)\in\mathcal{K}^h}$ to every client m.
    Compute $\mathcal{K}^{h+1} = \{(h+1, ki-j) \mid (h,i) \in (\mathcal{K}^h), j \in [0, k-1] \cap \mathbb{N}\}$
  end if
end while

Algorithm 2 PF-PNE: m-th client

Input: k-nary partition $\mathcal{P}$, smoothness parameters ν_1, ρ, transition layer H_0
Initialize $h = 0$, $H_0 := \operatorname{argmin}_{h\in\mathbb{N}} \big(\nu_1\rho^h \le \Delta\big)$, $\mathcal{K}_m^0$ =the first broadcast $\mathcal{K}^h$ from the server.
if $h < H_0$ **then**
  while not reaching the time horizon T **do**
    for each $(h,i) \in \mathcal{K}^h$ sequentially **do**
      Pull the node $\lceil \frac{\tau_h}{M} \rceil$ times and receive rewards $\{r_{m,h,i,t}\}$
    end for
    Calculate $\widehat{\mu}_{m,h,i} = \frac{1}{T_{m,h,i}} \sum_t r_{m,h,i,t}$ for every $(h,i) \in \mathcal{K}^h$
    Send the local estimates $\{\widehat{\mu}_{m,h,i}\}_{(h,i)\in\mathcal{K}^h}$ to the server
    Receive new $\mathcal{K}^h$ and $\{\overline{\mu}_{h,i}, b_{h,i}\}_{(h,i)\in\mathcal{K}^h}$
    Update $\widehat{\mu}_{m,h,i} = \overline{\mu}_{h,i}, b_{m,h,i} = b_{h,i}$ for every node $(h,i) \in \mathcal{K}^h$
    Compute $\mathcal{K}^{h+1} = \{(h+1, ki-j) \mid (h,i) \in \mathcal{K}^h, j \in [0, k-1] \cap \mathbb{N}\}$
    Update $h = h + 1$
  end while
else
  Set $\mathcal{K}^h = \emptyset$ for all $h > H_0$
  Run PE$(\mathcal{P}, \nu_1, \rho)$
end if

rithms 2, 3, and 1. In these algorithms, we have indexed the nodes using their depths and indices (h,i). $T_{m,h,i}$ denotes the numbers of pulled samples on the m-th local objective in h,ith node and $b_{m,h,i} = c\sqrt{\frac{\log(c_1 T/\delta)}{T_{m,h,i}}}$ is the corresponding confidence bound. $T_{h,i} = \sum_{m=1}^{M} T_{m,h,i}$ and $b_{h,i}$ is the confidence bound defined similarly with respect to the global objective. The details of all notations can be found in Appendix A.

Each client maintains two sets of "active" nodes at each depth, $\mathcal{K}^h$ and $\mathcal{K}_m^h$, i.e., they are the sets of nodes that we believe potentially contains the global optimum and the local optimum respectively. $\mathcal{K}^h$ is the set of global active nodes and controlled by the server, the algorithm tries to explore the global object $\overline{f}$ over the region $\mathcal{K}^h$ collaboratively by all clients. $\mathcal{K}_m^h$ is the set of local active nodes and controlled by the client, the algorithm tries to explore the local objective f_m over the region $\mathcal{K}_m^h \backslash \mathcal{K}^h$ locally by the m-th client. At each depth h, $\mathcal{K}^h \subseteq \mathcal{K}_m^h$. Similarly, the server and the clients also maintain two sets of nodes $\mathcal{E}^h$, $\mathcal{E}_m^h$ to be eliminated/removed from $\mathcal{K}^h$ and $\mathcal{K}_m^h$ respectively. Each node is pulled/evaluated either globally or locally for at least $\tau_h := \lceil c^2\nu_1^{-2}\log(c_1 T/\delta)\rho^{-2h}\rceil$ times for an accurate estimation of the reward, where c is an absolute constant and δ is the confidence parameter.

The algorithm consists of two stages, while each stage has several phases. The depth H_0 for transitioning between the two stages is $H_0 := \operatorname{argmin}_{h\in\mathbb{N}} \big(\nu\rho^h \le \Delta\big)$, or equivalently $H_0 = \lceil \log_\rho(\Delta/\nu)\rceil$. This implies that when we have control over the optimality difference Δ, we can optimize the local objectives jointly, and when the control is lost, the clients should find their local optimums separately.

- In the first stage, collaboration among the clients is encouraged to accelerate the learning process. At each depth h starting from the root $\mathcal{K}^0 = (0,1)$, the server sends the new $\mathcal{K}^h$ to the clients. The clients evaluate the nodes and send their average rewards $\widehat{\mu}_{m,h,i}$ back to the server. The server will then compute the global average $\overline{\mu}_{h,i}$, select the best node $\mathcal{P}_{h,i^p}$ (in terms of the $\overline{\mu}_{\cdot,\cdot}$ value), and determine the set of nodes $\mathcal{E}^h$ to be eliminated by using the elimination criterion $\overline{\mu}_{h,i} + b_{h,i} + \nu_1\rho^h < \overline{\mu}_{h,i^p} - b_{h,i^p}$. The updated set $\mathcal{K}^h$ and the global average rewards of the nodes inside the set are then communicated to the clients. For nodes still inside $\mathcal{K}^h$, their local statistics $\widehat{\mu}_{m,h,i}, b_{m,h,i}$ are replaced by the global ones $\overline{\mu}_{h,i}, b_{h,i}$
- In the second stage, the server terminates the collaboration and the clients initiate the Personalized Elimination (PE) algorithm. Starting again from the root, they evaluate the nodes that are in the set $\mathcal{K}_m^h \setminus \mathcal{K}^h$ until at least τ_h local samples are ob-

tained, including the nodes that are eliminated in the first stage ($\mathcal{E}^h$). They will then find the best node in $\mathcal{K}_m^h$ and determine the set of bad nodes $\mathcal{E}_m^h$ with a similar elimination criterion as the first stage. For all the nodes inside $\mathcal{E}_m^h$, they can now be safely removed because they have been eliminated twice, ergo double elimination, once from $\mathcal{K}^h$ and once from $\mathcal{K}_m^h$. At the same time, $\mathcal{K}^h$ will be protected from elimination and no further exploration is needed for the nodes in this set.

Algorithm 3 `PE`: m-th client

Input: partition $\mathcal{P}$, smoothness parameters ν_1, ρ
Initialize $h = 0$, $\mathcal{K}_m^0 = \{(0,1)\}$.
while not reaching the time horizon T **do**
  while $T_{m,h,i} < \tau_h$ for any $(h,i) \in \mathcal{K}_m^h \backslash \mathcal{K}^h$ **do**
    Pull the node and receive reward $r_{m,h,i,t}$
  end while
  Calculate $\widehat{\mu}_{m,h,i} = \frac{1}{T_{m,h,i}} \sum_t r_{m,h,i,t}$ for every $(h,i) \in \mathcal{K}_m^h \backslash \mathcal{K}^h$
  Compute $(h, i_m^p) = \arg\max_{(h,i)\in\mathcal{K}_m^h} \widehat{\mu}_{m,h,i}$
  Compute $\mathcal{E}_m^h = \big\{(h,i) \in \mathcal{K}_m^h \backslash \mathcal{K}^h \mid \widehat{\mu}_{m,h,i} + b_{m,h,i} + \nu_1 \rho^h < \widehat{\mu}_{h,i_m^p} - b_{h,i_m^p}\big\}$
  Compute $\mathcal{K}_m^{h+1} = \big\{(h+1, ki - j) \mid (h,i) \in (\mathcal{K}_m^h \setminus \mathcal{E}_m^h), j \in [0, k-1] \cap \mathbb{N}\big\}$
  Update $h = h + 1$
end while

Remark 3.1. In Algorithm 2 and 3 for client m, "pulling a node $\mathcal{P}_{h,i}$" refers to evaluating the local objective f_m at a particular point $x \in \mathcal{P}_{h,i}$ in order to obtain the reward. Note that we assume local smoothness on all the objectives (Assumption 2), therefore whether we randomly choose the evaluation point inside the node for each evaluation, or use one pre-determined point for all nodes does not affect the final regret bound. For simplicity, we choose the latter design in our analysis and our experiments. Similar results are observed in [Bubeck et al., 2011, Azar et al., 2014, Li et al., 2022b].

Algorithm Uniqueness. The uniqueness of the design in `PF-PNE` is four-fold:

- **(Collaboration)**. At each depth h, the number of samples needed for the nodes in $\mathcal{K}^h$ is reduced by the collaboration between the clients, and thus making the per-client cumulative regret smaller than single-client algorithms on these nodes.
- **(Double Elimination)**. The double-elimination strategy guarantees that the nodes are safely eliminated so that the optimum of any local objective will not be directly removed, and thus protecting the cumulative regret from being linear (See Theorem 3.1 and Remark 3.2).
- **(Biased Evaluation Helps)**. In the second stage, the clients will utilize the global average reward and confidence bound information on $\mathcal{K}^h$ in the first stage to perform the second elimination process. No further exploration or eliminations will be performed on those nodes. On one hand, this strategy essentially reduces the sampling cost for the nodes in $\mathcal{K}^h$ for every client. On the other hand, despite that the global average $\overline{\mu}_{h,i}$ of the rewards is a biased evaluation of the local objective f_m at node $\mathcal{P}_{h,i}$, we could still use it to substitute the local average $\widehat{\mu}_{m,h,i}$. As we show in the analysis, the size of the bias is under control and the biased evaluations are still helpful.
- **(Limited Communications)** Based on our design and the choice of the stage transitioning criterion, the communication cost is always limited, both in terms of rounds and information, which makes sure that no frequent communications between the server and the clients are needed.

3.2 Theoretical Analysis

In order to analyze the cumulative regret of the proposed algorithm, we introduce the definition of the near-optimality dimension, which is a common notation in the existing literature that measures the number of near-optimal regions and thus the difficulty of the problem [Bubeck et al., 2011, Azar et al., 2014, Shang et al., 2019, Li et al., 2022a].

Definition 1. (Near-optimality Dimension) *Let $\epsilon_h > 0$ and $\epsilon_h' > 0$ be two functions of h, for any subset of ϵ_h-optimal nodes for the function f, $\mathcal{X}_{f,\epsilon_h} = \{x \in \mathcal{X} : f^* - f(x) \le \epsilon_h\}$, there exists a constant C such that $\mathcal{N}_f(\epsilon_h, \epsilon_h') \le C(\epsilon_h')^{-d}, \forall h \ge 0$, where $d := d_f(\epsilon_h, \epsilon_h')$ is the near-optimality dimension of the function f and $\mathcal{N}_f(\epsilon_h, \epsilon_h')$ is the ϵ_h'-cover number of the set $\mathcal{X}_{f,\epsilon_h}$ w.r.t. the dissimilarity ℓ.*

Using the above near-optimality dimension definition, we denote $d_m = d_{f_m}(12\nu\rho^h, \rho^h)$ for every $m \in [M]$ and $\overline{d} = d_{\overline{f}}(6\nu\rho^h, \rho^h)$. Define $d_{\max} = \max\{d_1, d_2, \cdots, d_M\}$ and $d_{\min} = \min\{d_1, d_2, \cdots, d_M\}$. Now we provide the general upper bound using near-optimality dimension, on the cumulative regret of the proposed `PF-PNE` algorithm as follows.

Theorem 3.1. *Suppose that all the local objectives $f_1, f_2 \cdots, f_M$ and the global objective $\overline{f}$ all satisfy Assumptions 2, 3. Setting $\delta = 1/M$ in Algorithm 2, 3, and 1, the expected cumulative regret of the `PF-PNE` algorithm satisfies*

$$\mathbb{E}[R(T)] = \widetilde{\mathcal{O}}\left(M^{\frac{\overline{d}+1}{\overline{d}+2}} T^{\frac{\overline{d}+1}{\overline{d}+2}} + MT^{\frac{d_{\max}+1}{d_{\max}+2}}\right)$$

The number of communication rounds of `PF-PNE` scales

as $\min\{C_1 M \log \frac{1}{\Delta}, C_2(M \log MT)\}$*, where* C_1*,* C_2 *are two absolute constants.*

Remark 3.2. We relegate the proof of the above theorem to Appendix B. We emphasize that since we only use Assumption 3 without any requirements on the size of Δ, the regret upper bound in Theorem 3.1 displays a preliminary and natural result.

The regret bound consists of two terms

- The first term $\widetilde{\mathcal{O}}\left(M^{\frac{\overline{d}+1}{\overline{d}+2}} T^{\frac{\overline{d}+1}{\overline{d}+2}}\right)$ comes from the first stage of elimination in PF-PNE, which might continue for a large number of rounds if Δ is very small. In that case, the federated learning process could be viewed as M clients optimizing $\overline{f}$ jointly, and thus the regret is related to the near-optimality dimension $\overline{d}$ of $\overline{f}$.
- The second term $\widetilde{\mathcal{O}}\left(MT^{\frac{d_{\max}+1}{d_{\max}+2}}\right)$ comes from the second round of elimination in PF-PNE. When Δ is large, e.g., $\Delta = 1$, it implies that we have almost no prior beliefs on the difference between the local and global optimum values. In that case, the local objectives could be very different and we can only bound the cumulative regret asymptotically by the objective with the largest near-optimality dimension, or equivalently, the hardest local objective.

Note that both two terms are sublinear with respect to T, it means that PF-PNE is always capable of finding the optimums of the local objective, regardless of whether the prior knowledge of Δ is small or large. Therefore, we claim that PF-PNE "always works".

In order to analyze the regret more tightly, we introduce the following new notation d_{new} to measure the difference in local and global optimal nodes, and thus the size of $\mathcal{K}_m^h \setminus \mathcal{K}^h$. This set $\mathcal{K}_m^h \setminus \mathcal{K}^h$ is, in the worst case, $\Omega(\rho^{-d_m h})$ at each depth h, such as when we terminate the first stage early with large Δ and $\mathcal{K}^h$ is simply empty, but it should be much smaller when the objectives are similar. Moreover, we need to assume a reasonably small Δ for PF-PNE to outperform single-client algorithms.

Definition 2. (Optimality-Difference Dimension) *Using the same notations as in Definition 1, the optimality-difference dimension is defined to be the smallest number* $d_{new} \geq 0$ *such that* $\mathcal{N}_{f_m}(12\nu\rho^h, \nu\rho^h) \setminus \mathcal{N}_{\overline{f}}(6\nu\rho^h, \rho^h) \leq C_0 \rho^{-d_{new} h}, \forall m \in [M]$.

Remark 3.3. First of all, the above definition could be defined with respect to each client, but asymptotically we would care about the largest one across all the clients. Based on the definition of optimality-difference dimension, we know that $d_{\text{new}} \leq d_{\max}$ and it is a tighter measure of the number of local near-optimal nodes that are non-optimal globally. Using the definition, we provide the following corollary as a tighter upper bound on the cumulative regret.

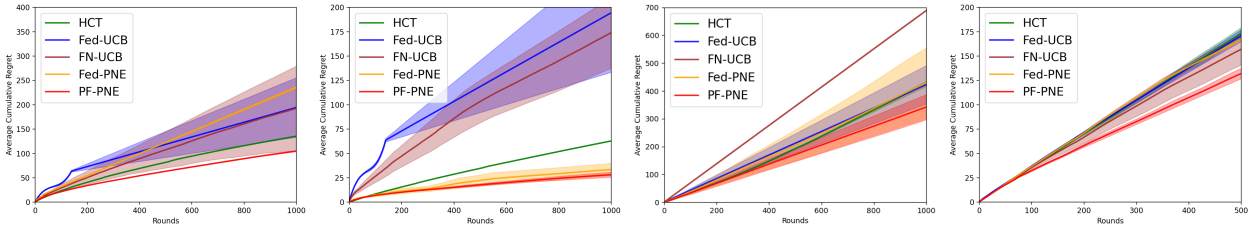
Corollary 3.1. *Suppose that all the assumptions in Theorem 3.1 and Assumption 4 are satisfied, and* d_{new} *is the optimality-difference dimension as in Definition 2. Assume that* $\Delta \leq C_3(\log MT/T)^{\frac{1}{\max\{d_{\min}, d_{new}\}+2}}$*, where* C_3 *is an absolute constant, the expected cumulative regret of the* PF-PNE *algorithm satisfies*

$$\mathbb{E}[R(T)] = \widetilde{\mathcal{O}}\left(M^{\frac{d_{\min}+1}{d_{\min}+2}} T^{\frac{d_{\min}+1}{d_{\min}+2}} + MT^{\frac{d_{new}+1}{d_{new}+2}}\right)$$

Remark 3.4. When the local objectives are similar and most local near-optimal nodes are also globally near-optimal, then $d_{\text{new}} \ll d_{\min}$ and the above regret bound will be dominated by the first term. It means that on average, the client-wise regret is of order $\widetilde{\mathcal{O}}\left(M^{-\frac{1}{d_{\min}+2}} T^{\frac{d_{\min}+1}{d_{\min}+2}}\right)$, then the cumulative regret will be smaller than running the $\mathcal{X}$-armed bandit algorithms (e.g., Azar et al. [2014], Li et al. [2023b]) separately on the clients. Similar to the arguments in Remark 3.2, when the local objectives are different, d_{new} will be almost the same as $d_{\max}$ and running the PF-PNE algorithm will be asymptotically the same as running the $\mathcal{X}$-armed bandit algorithms. Therefore, the proposed PF-PNE algorithm adapts to the heterogeneity of the local objectives.

Remark 3.5. (Communication Cost) The number of communication rounds in PF-PNE is always bounded by the minimum between a constant that depends on Δ in Assumption 3 and a term that depends on T logarithmically.

- When $\Delta > 0$ is slightly large (e.g., 0.1) and T is sufficiently large, the communication cost is always bounded by a constant $C_1 M \log \frac{1}{\Delta}$. The bound makes sense intuitively because as long as the local objectives are different, we should ask the clients to find their local optimums by themselves and further communications become futile at some point in the learning process. Therefore, both the number of communication **rounds** and the amount of **information** communicated will be bounded.
- If Δ is a very small number, or even the extreme case 0, and the term $C_2(M \log MT)$ dominates the communication cost, it means that the PF-PNE algorithm will degenerate to almost the same as Fed-PNE in Li et al. [2022b], because the second stage will not be activated and the second round of elimination will never be executed. In this case, the communication cost would be the same as Fed-PNE. Li et al. [2022b] have proved that the amount of **information** transferred would be of order $\widetilde{\mathcal{O}}(M \log T \vee MT^{\frac{d}{d+2}})$, which is still sublin-



(a) Garland (1D) (b) Himmelblau (2D) (c) Rastrigin (10D) (d) Landmine

Figure 1: Cumulative regret of different algorithms on the synthetic functions and the real-life datasets. Unlimited communications are allowed for centralized algorithms. `FNUCB` runs much slower compared with the other algorithms due to the training of neural nets. The other four algorithms take similar time to finish.

ear. Such a communication cost is proven to be unavoidable [Li et al., 2022b].

The extreme case $\Delta = 0$ could happen only if the local objectives have the same optimum at the same point. In all our experiments, we observe that the communication cost is always bounded.

4 EXPERIMENTS

In this section, we provide the empirical evaluations of the proposed `PF-PNE` algorithm on both synthetic and real-world objectives. We compare `PF-PNE` with `HCT` [Azar et al., 2014], `Fed1-UCB` [Shi and Shen, 2021a], `FN-UCB` [Dai et al., 2023] and `Fed-PNE` [Li et al., 2022b]. The curves in the figures are averaged over 10 independent runs of each algorithm with the shaded regions representing 1 standard deviation error bar. Additional experimental details and algorithm implementations can be found in Appendix C.

Remark 4.1. For `HCT`, we run the algorithm on the M local objectives and then plot the average local regret across all the objectives with no communications. For the other federated algorithms, we also plot the average cumulative regret across the clients. Such a comparison is fair since we essentially compare the single-client algorithms with the federated algorithms on their average performance across multiple clients.

Synthetic Objectives. We first conduct experiments on three synthetic objectives, Garland, Himmelblau, and Rastrigin, with the parameter domains $\mathcal{X}$ to be $[0, 1]$, $[-5, 5]^2$, and $[-1, 1]^{10}$ respectively. We apply random shifts to the original synthetic functions to create the local objectives. The shifts are zero-mean normal random variables and are applied to every dimension of the objective. The average cumulative regrets of different algorithms are provided in Figure 1(a), 1(b), and 1(c). As can be observed in the figures, `PF-PNE` has the smallest averaged cumulative regret. The performance of `Fed-PNE` largely depends on the similarity between the local objectives and the global objective because it is designed to optimize the global objective. When they are very different, e.g., on Garland and Rastrigin, the performance of `Fed-PNE`, although better than its competitors, is far from satisfactory.

Landmine Detection. The landmine dataset [Liu et al., 2007] consists of multiple landmine fields with different locations of the landmine extracted from radar images. For each client, we randomly assign one of the landmine fields to the client. We federatedly tune the hyper-parameters of support vector machines with the RBF kernel parameter chosen from $[0.01, 10]$ and the L_2 regularization parameter chosen from $[10^{-4}, 10]$. The local objectives are the AUC-ROC scores of the support vector machine evaluated on the local landmine fields. We provide the average cumulative regret of different algorithms in Figure 1(d). As shown in the figure, our algorithm achieves the smallest regret.

5 DISCUSSIONS AND CONCLUSIONS

In this work, we study the personalized federated $\mathcal{X}$-armed bandit problem and propose the first algorithm for such problems. The proposed `PF-PNE` algorithm utilizes the hierarchical partition and the idea of double elimination to help the clients locate their own optimums. `PF-PNE` is unique in its adaptivity to the heterogeneity of the local objectives and its little communication cost for the federated learning process. Several interesting future directions are also inspired by our work. For example, is our similarity measure Δ the best assumption to quantify the difference between the local objectives, or is there an even weaker/ more useful assumption for personalized federated $\mathcal{X}$-armed bandit? Besides, `PF-PNE` still needs the smoothness parameters and the prior knowledge on the bound Δ as part of the input, and it would be interesting to explore parameter-free algorithms in our setting.

6 ACKNOWLEDGEMENTS

This work was done before Wenjie joined Amazon. Jean Honorio gratefully ack nowledges the support of the National Science Foundation (DMS-2134209).

References

Mohammad Gheshlaghi Azar, Alessandro Lazaric, and Emma Brunskill. Online stochastic optimization under correlated bandit feedback. In *International Conference on Machine Learning*, pages 1557–1565. PMLR, 2014.

Peter L. Bartlett, Victor Gabillon, and Michal Valko. A simple parameter-free and adaptive approach to optimization under a minimal local smoothness assumption. In *30th International Conference on Algorithmic Learning Theory*, 2019.

Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. χ-armed bandits. *Journal of Machine Learning Research*, 12(46):1655–1695, 2011.

Sayak Ray Chowdhury and Aditya Gopalan. On kernelized multi-armed bandits. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 844–853. PMLR, 06–11 Aug 2017.

Zhongxiang Dai, Bryan Kian Hsiang Low, and Patrick Jaillet. Federated bayesian optimization via thompson sampling. In *Advances in Neural Information Processing Systems*, volume 33, pages 9687–9699. Curran Associates, Inc., 2020.

Zhongxiang Dai, Yao Shu, Arun Verma, Flint Xiaofeng Fan, Bryan Kian Hsiang Low, and Patrick Jaillet. Federated neural bandits. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=38m4h8HcNRL`.

Abhimanyu Dubey and Alex Sandy Pentland. Differentially-private federated linear bandits. In *Advances in Neural Information Processing Systems*, volume 33, pages 6003–6014. Curran Associates, Inc., 2020.

Yasong Feng, Zengfeng Huang, and Tianyu Wang. Lipschitz bandits with batched feedback, 2021.

Jean-Bastien Grill, Michal Valko, Remi Munos, and Remi Munos. Black-box optimization of noisy functions with unknown smoothness. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2015.

Ruiquan Huang, Weiqiang Wu, Jing Yang, and Cong Shen. Federated linear contextual bandits. In *Advances in Neural Information Processing Systems*, 2021.

Chuanhao Li, Huazheng Wang, Mengdi Wang, and Hongning Wang. Communication efficient distributed learning for kernelized contextual bandits. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022a. URL `https://openreview.net/forum?id=6rVXMHImDzv`.

Wenjie Li, Qifan Song, Jean Honorio, and Guang Lin. Federated x-armed bandit, 2022b. URL `https://arxiv.org/abs/2205.15268`.

Wenjie Li, Haoze Li, Jean Honorio, and Qifan Song. Pyxab – a python library for $\mathcal{X}$-armed bandit and online blackbox optimization algorithms, 2023a. URL `https://arxiv.org/abs/2303.04030`.

Wenjie Li, Chi-Hua Wang, Guang Cheng, and Qifan Song. Optimum-statistical collaboration towards general and efficient black-box optimization. *Transactions on Machine Learning Research*, 2023b. ISSN 2835-8856. URL `https://openreview.net/forum?id=ClIcmwdlxn`.

Qiuhua Liu, Xuejun Liao, and Lawrence Carin. Semi-supervised multitask learning. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

Xuedong Shang, Emilie Kaufmann, and Michal Valko. General parallel optimization a without metric. In *Algorithmic Learning Theory*, pages 762–788, 2019.

Chengshuai Shi and Cong Shen. Federated multi-armed bandits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(11):9603–9611, May 2021a.

Chengshuai Shi, Cong Shen, and Jing Yang. Federated multi-armed bandits with personalization. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 2917–2925. PMLR, 13–15 Apr 2021b.

Weitong Zhang, Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural thompson sampling. *arXiv preprint arXiv:2010.00827*, 2020.

Zhaowei Zhu, Jingxuan Zhu, Ji Liu, and Yang Liu. Federated bandit: A gossiping approach. In *Proceedings of the 2021 ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, pages 3–4, 2021.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes] See Section 2
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes] See Section 3
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Not Applicable]
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes] See Section 2 and 3
   (b) Complete proofs of all theoretical results. [Yes] See Appendix.
   (c) Clear explanations of any assumptions. [Yes] See Section 3
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes] See Section 4 and Appendix C
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes] See Section 4 and Appendix C
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes] See Section 4 and Appendix C
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes] See Section 4 and Appendix C
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Appendix to "Personalized Federated $\mathcal{X}$-armed Bandit"

A NOTATIONS AND USEFUL LEMMAS

A.1 Notations

Here we list all the notations used in the proof of our cumulative regret bound:

- $\mathcal{L}_t$ denotes all the nodes in the exploration tree at time t
- $\mathcal{T}_m^h$: The time steps of client m spent on layer h.
- $\mathcal{K}^h$: The set of pre-eliminated nodes in the server on layer h.
- $\mathcal{E}^h$: The set of nodes to be eliminated in the server on layer h.
- $\mathcal{K}_m^h$: The set of pre-eliminated nodes in the client m on layer h.
- $\mathcal{E}_m^h$: The set of nodes to be eliminated in the client m on layer h.
- $\overline{\mathcal{K}}^h$: The set of post-eliminated nodes from the server at depth h, i.e., $\overline{\mathcal{K}}^h = \mathcal{K}^h \setminus \mathcal{E}^h$.
- $\overline{\mathcal{K}}_m^h$: The set of post-eliminated nodes from the client m at depth h, i.e., $\overline{\mathcal{K}}_m^h = \left(\mathcal{K}_m^h \setminus \overline{\mathcal{K}}^h\right) \setminus \mathcal{E}_m^h$.
- (h, i^p): the depth and the index of the node $\mathcal{P}_{h,i^p}$ chosen by the server on layer h from $\mathcal{K}^h$.
- (h, i^*): the depth and the index of the node $\mathcal{P}_{h,i^*}$ that contains (one of) the maximizer $\overline{x}^*$ of the global objective $\overline{f}$ on depth h.
- (h, i_m^*): the depth and the index of the node $\mathcal{P}_{h,i_m^*}$ that contains (one of) the maximizer x_m^* of the local objective f_m on depth h.
- $T_{h,i}$: the number of times the node $\mathcal{P}_{h,i}$ is sampled globally, i.e., $T_{h,i} = \sum_{m=1}^M T_{m,h,i}$.
- $T_{m,h,i}$: the number of times the node $\mathcal{P}_{h,i}$ is sampled from client m.
- $b_{h,i} = c\sqrt{\frac{\log(c_1 T/\delta)}{T_{h,i}}}$: confidence bound for the node (h,i) on the global objective
- $b_{m,h,i} = c\sqrt{\frac{\log(c_1 T/\delta)}{T_{m,h,i}}}$: confidence bound for the node (h,i) on the m-th lobal objective
- H_t: the maximum depth reached by the algorithms at time t
- τ_h: the minimum required number of samples needed for a node on depth h, defined below.

The threshold for every depth. The number of times τ_h needed for the statistical error (the UCB term) of every node on depth h to be better than the optimization error is the solution to

$$\nu_1 \rho^h \approx c\sqrt{\frac{\log(c_1 T/\delta)}{\tau_h}}, \tag{1}$$

which is equivalent as the following choice of the threshold

$$\frac{c^2}{\nu_1^2}\rho^{-2h} \le \tau_h = \left\lceil \frac{c^2 \log(c_1 T/\delta)}{\nu_1^2}\rho^{-2h} \right\rceil \le 2\frac{c^2 \log(c_1 T/\delta)}{\nu_1^2}\rho^{-2h}. \tag{2}$$

Notably, this choice of the threshold is the same as the threshold value in the `HCT` algorithm [Azar et al., 2014]. In other words, we design our algorithm so that the samples are from different clients uniformly and thus the

estimators are unbiased, and at the same time we minimize the unspent budget due to such distribution. There is still some (manageable) unspent budget due to the floor operation in the computation of $t_{m,h,i}$. However because of the expansion criterion (line 5-6) in `Fed-PNE`, we are able to travel to very deep layers inside the partition very fast when there are a lot of clients, and thus `Fed-PNE` is faster than single-client $\mathcal{X}$-armed bandit algorithms.

A.2 Supporting Lemmas

Lemma A.1. (Hoeffding's Inequality) *Let $X_1, \dots, X_n$ be independent random variables such that $a_i \leq X_i \leq b_i$ almost surely. Consider the sum of these random variables, $S_n = X_1 + \cdots + X_n$. Then for all $t > 0$, we have*

$$\mathbb{P}\left(|S_n - \mathbb{E}\left[S_n\right]| \geq t\right) \leq 2\exp\left(-\frac{2t^2}{\sum_{i=1}^{n}\left(b_i - a_i\right)^2}\right).$$

Here $\mathbb{E}\left[S_n\right]$ is the expected value of S_n.

Lemma A.2. (High Probability Event) *At each time t, define the "good" events E_t^1, E_t^2 as*

$$\begin{aligned} E_t^1 &= \left\{\forall h' \leq H_t, \forall (h,i) \in \mathcal{K}^{h'}, \forall T_{h,i} \in [MT], |\overline{f}(x_{h,i}) - \overline{\mu}_{h,i}| \leq c\sqrt{\frac{\log(c_1 T/\delta)}{T_{h,i}}}\right\} \\ E_t^2 &= \left\{\forall h' \leq H_t, \forall m \in [M], \forall (h,i) \in \mathcal{K}_m^{h'} \setminus \mathcal{K}^{h'}, \forall T_{m,h,i} \in [MT], |f_m(x_{h,i}) - \widehat{\mu}_{m,h,i}| \leq c\sqrt{\frac{\log(c_1 T/\delta)}{T_{m,h,i}}}\right\} \end{aligned} \tag{3}$$

where the right hand sides in thee two events are the confidence bound $b_{h,i}$ and $b_{m,h,i}$ respectively for the node $\mathcal{P}_{h,i}$ and $c \geq 2, c_1 \geq (2M^2)^{1/8}$ are two constants. Define the event $E_t = E_t^1 \bigcap E_t^2$, then for any fixed round t, we have $\mathbb{P}(E_t) \geq 1 - 2\delta/T^6$

Proof. For the first event E_t^1, by utilizing the results of Lemma B.2 in Li et al. [2022b] and the Hoeffding's Inequality, we know that

$$\mathbb{P}\left(\left|\sum_{m\in[M]}\sum_{t\in[t_{m,h,i}]} r_{m,h,i,t} - \sum_{m\in[M]} t_{m,h,i} f_m(x_{h,i})\right| \geq x\right) \leq 2\exp\left(-\frac{2x^2}{T_{h,i}}\right). \tag{4}$$

Therefore by the union bound, the probability of the compliment event E_t^{1c} can be bounded as

$$\begin{aligned} \mathbb{P}\left(E_t^{1c}\right) &\leq \sum_{h'\in H_t}\sum_{(h,i)\in\mathcal{K}^{h'}}\sum_{T_{h,i}=1}^{MT}\mathbb{P}\Big(|\overline{f}(x_{h,i}) - \overline{\mu}_{h,i}| > b_{h,i}\Big) \leq \sum_{h'\in H_t}\sum_{(h,i)\in\mathcal{K}^{h'}} 2MT\exp\Big(-2T_{h,i}b_{h,i}^2\Big) \\ &= 2MT\exp\Big(-2c^2\log(c_1T/\delta)\Big)\left(\sum_{h'\in H_t}|\mathcal{K}^{h'}|\right) \leq 2MT^2\left(\frac{\delta}{c_1 T}\right)^{2c^2} \leq \frac{\delta}{T^6}. \end{aligned} \tag{5}$$

For the second event E_t^2, similarly we have the probability of the compliment event bounded as

$$\begin{aligned} \mathbb{P}\left(E_t^{2c}\right) &\leq \sum_{m=1}^{M}\sum_{h'\in H_t}\sum_{(h,i)\in\mathcal{K}_m^{h'}}\sum_{T_{m,h,i}=1}^{MT}\mathbb{P}\Big(|f_m(x_{h,i}) - \widehat{\mu}_{m,h,i}| > b_{m,h,i}\Big) \\ &\leq \sum_{m=1}^{M}\sum_{h'\in H_t}\sum_{(h,i)\in\mathcal{K}_m^{h'}} 2MT\exp\Big(-2T_{m,h,i}b_{m,sh,i}^2\Big) \\ &= 2M^2T\exp\Big(-2c^2\log(c_1T/\delta)\Big)\left(\sum_{h'\in H_t}|\mathcal{K}_m^{h'}|\right) \leq 2M^2T^2\left(\frac{\delta}{c_1 T}\right)^{2c^2} \leq \frac{\delta}{T^6}. \end{aligned} \tag{6}$$

Finally by the union bound, we know that

$$\mathbb{P}\left(E_t\right) = 1 - \mathbb{P}\left(E_t^c\right) = 1 - \mathbb{P}\left(E_t^{1c}\bigcup E_t^{2c}\right) \geq 1 - \mathbb{P}\left(E_t^{1c}\right) - \mathbb{P}\left(E_t^{2c}\right) \geq 1 - 2\delta/T^6 \tag{7}$$

□

Lemma A.3. (Optimality in Global Objective, Lemma A.4 in Li et al. [2022b]). *For any client m, under the high probability event E_t at time $t \in \mathcal{T}_m^{h+1}$, the representative point $x_{h,i}$ of every un-eliminated node $\mathcal{P}_{h,i}$ at the previous depth, i.e., $(h,i) \in \overline{\mathcal{K}}^h$, is at least $6\nu_1\rho^h$-optimal, that is*

$$\overline{f}^* - \overline{f}(x_{h,i}) \leq 6\nu_1\rho^h, \forall (h,i) \in \overline{\mathcal{K}}^h. \tag{8}$$

Proof. The proof is provided for completeness. Under the high probability event E_t, we have the following inequality for every node $\mathcal{P}_{h,i}$ such that $(h,i) \in \overline{\mathcal{K}}^h$

$$|\overline{f}(x_{h,i}) - \overline{\mu}_{h,i}| \leq b_{h,i} = c\sqrt{\frac{\log(c_1 T/\delta)}{T_{h,i}}}. \tag{9}$$

Therefore the following set of inequalities hold

$$\begin{aligned} \overline{f}(x_{h,i}) + \nu_1\rho^h + 2b_{h,i} &\geq \overline{\mu}_{h,i} + \nu_1\rho^h + b_{h,i} \geq \overline{\mu}_{h,i^p} - b_{h,i^p} \geq \overline{\mu}_{h,i^*} - b_{h,i^p} \\ &\geq \overline{f}(x_{h,i^*}) - b_{h,i^*} - b_{h,i^p} \geq \overline{f}^* - \nu_1\rho^h - b_{h,i^*} - b_{h,i^p}, \end{aligned} \tag{10}$$

where the second inequality holds because $\mathcal{P}_{h,i}$ is not eliminated. The third inequality holds because of the elimination criterion in Algorithm 1, and the last one follows from the weak lipchitzness assumption (Assumption 2). In conclusion, we have the following upper bound on the regret

$$\overline{f}^* - \overline{f}(x_{h,i}) \leq 2\nu_1\rho^h + 2b_{h,i} + b_{h,i^*} + b_{h,i^p} \leq 6\nu_1\rho^h \tag{11}$$

where the last inequality holds because we sample each node enough number of times ($T_{h,i}$ larger than the threshold τ_h) so that $b_{h,i} \leq \nu_1\rho^h$ and thus $b_{h,i}, b_{h,i^*}, b_{h,i^p}$ are all smaller than $\nu_1\rho^h$. □

Lemma A.4. (Optimality in Local Objective) *For any client m, under the high probability event E_t at time $t \in \mathcal{T}_m^{h+1}$, the representative point $x_{h,i}$ of every un-eliminated node $\mathcal{P}_{h,i}$ at the previous depth h, i.e., $(h,i) \in \overline{\mathcal{K}}_m^h \backslash \overline{\mathcal{K}}^h$, is at least $(11\nu_1\rho^h + \Delta)$-optimal, that is*

$$f_m^* - f_m(x_{h,i}) \leq 11\nu_1\rho^h + \Delta, \forall (h,i) \in \overline{\mathcal{K}}_m^h \backslash \overline{\mathcal{K}}^h \tag{12}$$

Proof. If $(h, i_m^*) \in \overline{\mathcal{K}}_m^h \backslash \overline{\mathcal{K}}^h$, i.e., the node that contains the local optimum at depth h, is in the set $\overline{\mathcal{K}}_m^h \backslash \overline{\mathcal{K}}^h$, then we have the following inequalities

$$\begin{aligned} f_m(x_{h,i}) + \nu_1\rho^h + 2b_{m,h,i} &\geq \widehat{\mu}_{m,h,i} + \nu_1\rho^h + b_{m,h,i} \geq \widehat{\mu}_{m,h,i^p_m} - b_{m,h,i^p_m} \geq \widehat{\mu}_{m,h,i^*_m} - b_{m,h,i^p_m} \\ &\geq f_m(x_{h,i^*_m}) - b_{m,h,i^*_m} - b_{m,h,i^p_m} \geq f_m^* - \nu_1\rho^h - b_{m,h,i^*_m} - b_{m,h,i^p_m}, \end{aligned} \tag{13}$$

Therefore we know that the following bound holds

$$f_m^* - f_m(x_{h,i}) \leq 2\nu_1\rho^h + b_{m,h,i^*_m} + b_{m,h,i^p_m} + 2b_{m,h,i} \leq 6\nu_1\rho^h \tag{14}$$

where the last inequality is because $b_{m,h,i^*_m}, b_{m,h,i^p_m}, b_{m,h,i}$ are all smaller than $\nu_1\rho^h$. On the other hand, if $(h, i_m^*) \in \overline{\mathcal{K}}^h$, i.e., the node that contains the local optimum at depth h, is inside $\overline{\mathcal{K}}^h$, then we have the following inequalities

$$\begin{aligned} f_m(x_{h,i}) + \nu_1\rho^h + 2b_{m,h,i} &\geq \widehat{\mu}_{m,h,i} + \nu_1\rho^h + b_{m,h,i} \geq \widehat{\mu}_{m,h,i^p_m} - b_{m,h,i^p_m} \geq \widehat{\mu}_{m,h,i^*_m} - b_{m,h,i^p_m} \\ &= \overline{\mu}_{h,i^*_m} - b_{m,h,i^p_m} \geq \overline{f}(x_{h,i^*_m}) - b_{h,i^*_m} - b_{m,h,i^p_m} \geq \overline{f}^* - 6\nu_1\rho^h - b_{h,i^*_m} - b_{m,h,i^p_m} \end{aligned} \tag{15}$$

where the equality is because we have $\widehat{\mu}_{m,h,i^*_m} = \overline{\mu}_{h,i^*_m}$ and $b_{m,h,i^*_m} = b_{h,i^*_m}$ in Algorithm 2. The last inequality is from Lemma A.3, because $(h, i_m^*) \in \overline{\mathcal{K}}^h$ and thus it is uneliminated. Now we know that

$$\begin{aligned} f_m^* - f_m(x_{h,i}) &\leq f_m^* - \overline{f}^* + \overline{f}^* - f_m(x_{h,i}) \leq \Delta + \overline{f}^* - f_m(x_{h,i}) \\ &\leq \Delta + 7\nu_1\rho^h + b_{m,h,i^*_m} + b_{m,h,i^p_m} + 2b_{m,h,i} \leq 11\nu_1\rho^h + \Delta \end{aligned} \tag{16}$$

□

Lemma A.5. (Lemma 3 in Bubeck et al. [2011]) *For a node $\mathcal{P}_{h,i}$, define $f_{h,i}^* = \sup_{x \in \mathcal{P}_{h,i}} f(x)$ to be the maximum of the function on that region. Suppose that $f^* - f_{h,i}^* \leq c\nu_1\rho^h$ for some $c \geq 0$, then all x in $\mathcal{P}_{h,i}$ are $\max\{2c, c+1\}\nu_1\rho^h$-optimal.*

B MAIN PROOFS

In this section, we provide the proofs of the main theorem (Theorem 3.1) in this paper.

Proof. Let E_t be the high probability event in Lemma A.2. Let $\mathbb{I}_{E_t}$ denote whether the event E_t is true, i.e., $\mathbb{I}_{E_t} = 1$ if E_t is true and 0 otherwise. We first decompose the regret into two terms

$$\begin{aligned} R(T) &= \sum_{m=1}^{M}\sum_{t=1}^{T}(f_m^* - f(x_{m,t})) = \sum_{m=1}^{M}\sum_{t=1}^{T}(f_m^* - f(x_{m,t}))\,\mathbb{I}_{E_t} + \sum_{m=1}^{M}\sum_{t=1}^{T}(f_m^* - f(x_{m,t}))\,\mathbb{I}_{E_t^c} \\ &= R(T)^E + R(T)^{E^c}. \end{aligned} \tag{17}$$

For the second term, note that we can bound its expectation as follows

$$\mathbb{E}\left[R(T)^{E^c}\right] = \mathbb{E}\left[\sum_{m=1}^{M}\sum_{t=1}^{T}(f_m^* - f(x_{m,t}))\,\mathbb{I}_{E_t^c}\right] \le \sum_{m=1}^{M}\sum_{t=1}^{T}\mathbb{P}\left(E_t^c\right) \le \sum_{m=1}^{M}\sum_{t=1}^{T}(2\delta/T^6) = \frac{2M\delta}{T^5}. \tag{18}$$

where the second inequality follows from Lemma A.2. Now we bound the first term $R(T)^E$ in the decomposition under the event E_t. Let H be a constant depth to be decided later, we know that the term $R(T)^E$ can be written into the following form

$$\begin{aligned} R(T)^E &= \sum_{m=1}^{M}\sum_{t=1}^{T}(f_m^* - f_m(x_{m,t}))\,\mathbb{I}_{E_t} \\ &\le \underbrace{\sum_{m=1}^{M}\sum_{h=1}^{H}\sum_{(h,i)\in\mathcal{K}^h}(f_m^* - f_m(x_{h,i}))\left\lceil\frac{\tau_h}{M}\right\rceil}_{(a)} + \underbrace{\sum_{m=1}^{M}\sum_{h=1}^{H}\sum_{(h,i)\in\mathcal{K}_m^h\setminus\mathcal{K}^h}(f_m^* - f_m(x_{h,i}))\,\tau_h}_{(b)} \\ &\quad + \underbrace{\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{h_t>H}(f_m^* - f_m(x_{h_t,i_t}))}_{(c)} \end{aligned} \tag{19}$$

At every depth $h > 0$, for the globally un-eliminated nodes at the previous depth, i.e., for any $\mathcal{P}_{h-1,j}$ such that $(h-1,j) \in \overline{\mathcal{K}}^{h-1}$, by Lemma A.3, we have

$$\overline{f}^* - \overline{f}(x_{h-1,j}) \le 6\nu_1\rho^{h-1}. \tag{20}$$

By setting $\Delta = \nu_1\rho^{H-1}$ (to be explicitly defined later), for the locally un-eliminated nodes at the previous depth, i.e., for any $\mathcal{P}_{h-1,j}$ such that $(h-1,j) \in \overline{\mathcal{K}}_m^{h-1}\setminus\overline{\mathcal{K}}^{h-1}$, by Lemma A.4, we have the following inequality

$$f_m^* - f_m(x_{h-1,j}) \le (11\nu_1\rho^{h-1} + \Delta) \tag{21}$$

By Lemma A.5 and Assumption 4, since the set $\mathcal{K}^h$ is created by expanding $\overline{\mathcal{K}}^{h-1}$, for the representative point $x_{h,i}$ of the node $\mathcal{P}_{h,i}$ such that $(h,i) \in \mathcal{K}^h$, we have the following upper bound on the suboptimality gap at the point $x_{h,i}$ when $h \le H_0$.

$$f_m^* - \overline{f}(x_{h,i}) \le \overline{f}^* - \overline{f}(x_{h,i}) + \Delta \le (12\nu_1\rho^{h-1} + \Delta) \le 13\nu_1\rho^{h-1} \tag{22}$$

Similarly by Lemma A.5, since the set $\mathcal{K}_m^h$ is created by expanding $\overline{\mathcal{K}}_m^{h-1}$, therefore for the representative point $x_{h,i}$ of the node $\mathcal{P}_{h,i}$ such that $(h,i) \in \mathcal{K}_m^h\setminus\mathcal{K}^h$, we have the following upper bound on the suboptimality gap at the point $x_{h,i}$ when $h \le H_0$.

$$f_m^* - f_m(x_{h,i}) \le 24\nu_1\rho^{h-1}. \tag{23}$$

- In the case when $H \leq H_0$, we know that for term (a) in Eqn. (19), we have

$$\begin{aligned}
(a) &\leq \sum_{h=1}^{H} \left\lceil \frac{\tau_h}{M} \right\rceil \sum_{(h,i)\in\mathcal{K}^h} \sum_{m=1}^{M} (f_m^* - f_m(x_{h,i})) \leq \sum_{h=1}^{H} \left\lceil \frac{\tau_h}{M} \right\rceil \sum_{(h,i)\in\mathcal{K}^h} \sum_{m=1}^{M} \left(f_m^* - \overline{f}(x_{h,i})\right) \\
&\leq \sum_{h>0}^{H} 13M\nu_1\rho^{h-1} \max\left\{1, \frac{4c^2\log(c_1T/\delta)}{M\nu_1^2}\rho^{-2h}\right\} k|\overline{\mathcal{K}}^{h-1}| \\
&\leq \sum_{0<h\leq h_0} 13kCM\nu_1\rho^{h-1}|\overline{\mathcal{K}}^{h-1}| + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2} \sum_{h=1}^{H} \rho^{-2h}|\overline{\mathcal{K}}^{h-1}| \\
&\leq \sum_{0<h\leq h_0} 13kCM\nu_1(\rho^{h-1})^{-(\overline{d}-1)} + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2} \sum_{h=1}^{H} (\rho^{h-1})^{-(\overline{d}+1)} \\
&\leq \sum_{0<h\leq h_0} 13kCM\nu_1(\rho^{h-1})^{-(\overline{d}-1)} + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\rho^{-H(\overline{d}+1)}
\end{aligned} \tag{24}$$

where $h_0 = \lfloor \frac{1}{2}\log_{\rho^{-1}} \frac{M\nu_1^2}{4c^2} \rfloor$. For the term (b), we have the following inequality

$$\begin{aligned}
(b) &\leq \sum_{m=1}^{M}\sum_{h=1}^{H} \sum_{(h,i)\in\mathcal{K}_m^h\setminus\mathcal{K}^h} (f_m^* - f_m(x_{h,i}))\,\tau_h \leq \sum_{m=1}^{M}\sum_{h=1}^{H} \sum_{(h,i)\in\mathcal{K}_m^h\setminus\mathcal{K}^h} \frac{48c^2\log(c_1T/\delta)}{\nu_1\rho^2}\rho^{-(h-1)} \\
&\leq \sum_{m=1}^{M}\sum_{h=1}^{H} \frac{48c^2\log(c_1T/\delta)}{\nu_1\rho^2}\rho^{-(h-1)}k|\overline{\mathcal{K}}_m^{h-1}| \leq \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2}\sum_{h=0}^{H-1} M\rho^{-h(d_{\max}+1)} \\
&= \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)}
\end{aligned} \tag{25}$$

For term (c), it could be bounded by

$$(c) \leq \sum_{t=1}^{T} 24M\nu_1\rho^H \leq 24M\nu_1\rho^H T \tag{26}$$

Therefore if we combine the bounds on the three terms (a), (b), and (c), in Eqns. (24), (25), (26), we have the following inequality

$$\begin{aligned}
&R(T)^E \leq (a) + (b) + (c) \\
&\leq C_0 + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\rho^{-H(\overline{d}+1)} + \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)} + 24\nu_1\rho^H MT \\
&\leq C_0 + 2\max\left\{\frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\rho^{-H(\overline{d}+1)}, \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)}\right\} + 24\nu_1\rho^H MT \\
&\leq C_0 + C_1\max\left\{M^{\frac{\overline{d}+1}{\overline{d}+2}}T^{\frac{\overline{d}+1}{\overline{d}+2}}(\log(MT))^{\frac{\overline{d}+1}{\overline{d}+2}}, MT^{\frac{d_{\max}+1}{d_{\max}+2}}(\log(MT))^{\frac{d_{\max}+1}{d_{\max}+2}}\right\}
\end{aligned} \tag{27}$$

where C_0 is a constant, $C_1 = (2+2\log c_1)\max\left\{\left(\frac{52kc^2C(24\nu_1)^{(\overline{d}+1)}}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\right)^{\frac{1}{\overline{d}+2}}, \left(\frac{48kc^2C(24\nu_1)^{(d_{\max}+1)}}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)}\right)^{\frac{1}{d_{\max}+2}}\right\}$ and the last inequality is by balancing the size of the dominating terms using H. Now combining all the above bounds on $R(T)^E$ and $R(T)^{E^c}$, we know that the regret is of order $\widetilde{\mathcal{O}}\left(M^{\frac{\overline{d}+1}{\overline{d}+2}}T^{\frac{\overline{d}+1}{\overline{d}+2}} + MT^{\frac{d_{\max}+1}{d_{\max}+2}}\right)$.

- In the case when $H \geq H_0$, it means that the federated learning process terminated before even reaching H, then the clients optimize their local objectives separately. Therefore, we know that for term (a) in Eqn.

(19), we have

$$
\begin{aligned}
(a) &\leq \sum_{h=1}^{H} \left\lceil \frac{\tau_h}{M} \right\rceil \sum_{(h,i)\in\mathcal{K}^h} \sum_{m=1}^{M} (f_m^* - f_m(x_{h,i})) \leq \sum_{h=1}^{H} \left\lceil \frac{\tau_h}{M} \right\rceil \sum_{(h,i)\in\mathcal{K}^h} \sum_{m=1}^{M} \left(f_m^* - \overline{f}(x_{h,i})\right) \\
&\leq \sum_{h>0}^{H_0} 13M\nu_1\rho^{h-1} \max\left\{1, \frac{4c^2\log(c_1T/\delta)}{M\nu_1^2}\rho^{-2h}\right\} k|\overline{\mathcal{K}}^{h-1}| \\
&\leq \sum_{0<h\leq h_0} 13kCM\nu_1\rho^{h-1}|\overline{\mathcal{K}}^{h-1}| + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2} \sum_{h=1}^{H_0} \rho^{-2h}|\overline{\mathcal{K}}^{h-1}| \\
&\leq \sum_{0<h\leq h_0} 13kCM\nu_1(\rho^{h-1})^{-(\overline{d}-1)} + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2} \sum_{h=1}^{H_0} (\rho^{h-1})^{-(\overline{d}+1)} \\
&\leq \sum_{0<h\leq h_0} 13kCM\nu_1(\rho^{h-1})^{-(\overline{d}-1)} + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)} \rho^{-H_0(\overline{d}+1)}
\end{aligned} \tag{28}
$$

where $h_0 = \lfloor \frac{1}{2}\log_{\rho^{-1}} \frac{M\nu_1^2}{4c^2} \rfloor$. For the term (b), we have the following inequality

$$
\begin{aligned}
(b) &\leq \sum_{m=1}^{M}\sum_{h=1}^{H} \sum_{(h,i)\in\mathcal{K}_m^h\setminus\mathcal{K}^h} (f_m^* - f_m(x_{h,i}))\,\tau_h \leq \sum_{m=1}^{M}\sum_{h=1}^{H} \sum_{(h,i)\in\mathcal{K}_m^h\setminus\mathcal{K}^h} \frac{48c^2\log(c_1T/\delta)}{\nu_1\rho^2}\rho^{-(h-1)} \\
&\leq \sum_{m=1}^{M}\sum_{h=1}^{H} \frac{48c^2\log(c_1T/\delta)}{\nu_1\rho^2}\rho^{-(h-1)}k|\overline{\mathcal{K}}_m^{h-1}| \leq \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2} \sum_{h=0}^{H-1} M\rho^{-h(d_{\max}+1)} \\
&= \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)}
\end{aligned} \tag{29}
$$

where the second inequality is because when $h \in [1, H_0]$, we have $f_m^* - f_m(x_{h,i}) \leq 24\nu_1\rho^{h-1}$. When $h > H_0$, it means that the clients start learning separately and thus we have $f_m^* - f_m(x_{h,i}) \leq 12\nu_1\rho^{h-1}$ by Lemma A.4. Therefore in the worst case, $f_m^* - f_m(x_{h,i}) \leq 24\nu_1\rho^{h-1}$. For term (c), it could be bounded by

$$
(c) \leq \sum_{t=1}^{T} 12M\nu_1\rho^H \leq 12M\nu_1\rho^H T \tag{30}
$$

Therefore if we combine the bounds on the three terms (a), (b), and (c), we have the following inequality

$$
\begin{aligned}
&R(T)^E \leq (a) + (b) + (c) \\
&\leq C_0 + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\rho^{-H_0(\overline{d}+1)} + \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)} + 12\nu_1\rho^H MT \\
&\leq C_0 + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\rho^{-H(\overline{d}+1)} + \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)} + 12\nu_1\rho^H MT \\
&\leq C_0 + 2\max\left\{\frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\rho^{-H(\overline{d}+1)}, \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)} M\rho^{-H(d_{\max}+1)}\right\} + 12\nu_1\rho^H MT \\
&\leq C_0 + C_1\max\left\{M^{\frac{\overline{d}+1}{\overline{d}+2}}T^{\frac{\overline{d}+1}{\overline{d}+2}}(\log(MT))^{\frac{\overline{d}+1}{\overline{d}+2}}, MT^{\frac{d_{\max}+1}{d_{\max}+2}}(\log(MT))^{\frac{d_{\max}+1}{d_{\max}+2}}\right\}
\end{aligned} \tag{31}
$$

where C_0 is a constant, $C_1 = (2 + 2\log c_1)\max\left\{\left(\frac{52kc^2C(12\nu_1)^{(\overline{d}+1)}}{\nu_1\rho^2(\rho^{-(\overline{d}+1)}-1)}\right)^{\frac{1}{\overline{d}+2}}, \left(\frac{48kc^2C(12\nu_1)^{(d_{\max}+1)}}{\nu_1\rho^2(\rho^{-(d_{\max}+1)}-1)}\right)^{\frac{1}{d_{\max}+2}}\right\}$ and the last inequality is by balancing the size of the dominating terms using H. Now combining all the above bounds on $R(T)^E$ and $R(T)^{E^c}$, we know that the regret is of order $\widetilde{\mathcal{O}}\left(M^{\frac{\overline{d}+1}{\overline{d}+2}}T^{\frac{\overline{d}+1}{\overline{d}+2}} + MT^{\frac{d_{\max}+1}{d_{\max}+2}}\right)$.

□

B.1 Proof of Corollary 3.1

When we assume Assumption 4, we basically assume that near-optimal nodes in $\overline{f}$ are also near-optimal in the local objectives. Without loss of generality, we assume that $\omega = 1$ in Assumption 4. If $\omega \neq 1$, we only have to change a few constants in the proof.

For term (a) in Eqn. (19), note that every $6\nu_1\rho^h$-near-optimal node for $\overline{f}$ is $\Delta + 6\nu_1\rho^h \leq 12\nu_1\rho^h$-near-optimal in every f_m, that means $\overline{\mathcal{K}}^{h-1} \subseteq \overline{\mathcal{K}}_m^{h-1}, \forall m \in [M], \forall h \leq H$. Therefore, we could bound term (a) as

$$(a) \leq \sum_{0<h\leq h_0} 13kCM\nu_1(\rho^{h-1})^{-(d_{\min}-1)} + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\min}+1)}-1)}\rho^{-H_0(d_{\min}+1)} \tag{32}$$

whereas for term (b) in Eqn. (19), since $\mathcal{N}_{f_m}(12\nu\rho^h, \nu\rho^h) \setminus \mathcal{N}_{\overline{f}}(6\nu\rho^h, \rho^h) \leq C_0\rho^{-d_{\text{new}}h}, \forall m \in [M]$, we have the following bound

$$(b) \leq \frac{48kc^2C_0\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\text{new}}+1)}-1)}M\rho^{-H(d_{\text{new}}+1)} \tag{33}$$

If we combine the bounds on the three terms (a), (b), and (c), we have the following inequality

$$\begin{aligned}
R(T)^E &\leq (a) + (b) + (c) \\
&\leq C_0' + \frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\min}+1)}-1)}\rho^{-H_0(d_{\min}+1)} + \frac{48kc^2C_0\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\text{new}}+1)}-1)}M\rho^{-H(d_{\text{new}}+1)} + 24\nu_1\rho^H MT \\
&\leq C_0' + 2\max\left\{\frac{52kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\min}+1)}-1)}\rho^{-H(d_{\min}+1)}, \frac{48kc^2C\log(c_1T/\delta)}{\nu_1\rho^2(\rho^{-(d_{\text{new}}+1)}-1)}M\rho^{-H(d_{\text{new}}+1)}\right\} + 24\nu_1\rho^H MT \\
&\leq C_0' + C_1'\max\left\{M^{\frac{d_{\min}+1}{d_{\min}+2}}T^{\frac{d_{\min}+1}{d_{\min}+2}}(\log(MT))^{\frac{d_{\min}+1}{d_{\min}+2}}, MT^{\frac{d_{\text{new}}+1}{d_{\text{new}}+2}}(\log(MT))^{\frac{d_{\text{new}}+1}{d_{\text{new}}+2}}\right\}
\end{aligned} \tag{34}$$

where $C_0' > 0, C_1' > 0$ is another set of constants. Therefore the final regret for the `PF-PNE` algorithm is bounded by $\widetilde{\mathcal{O}}\left(M^{\frac{d_{\min}+1}{d_{\min}+2}}T^{\frac{d_{\min}+1}{d_{\min}+2}} + MT^{\frac{d_{\text{new}}+1}{d_{\text{new}}+2}}\right)$.

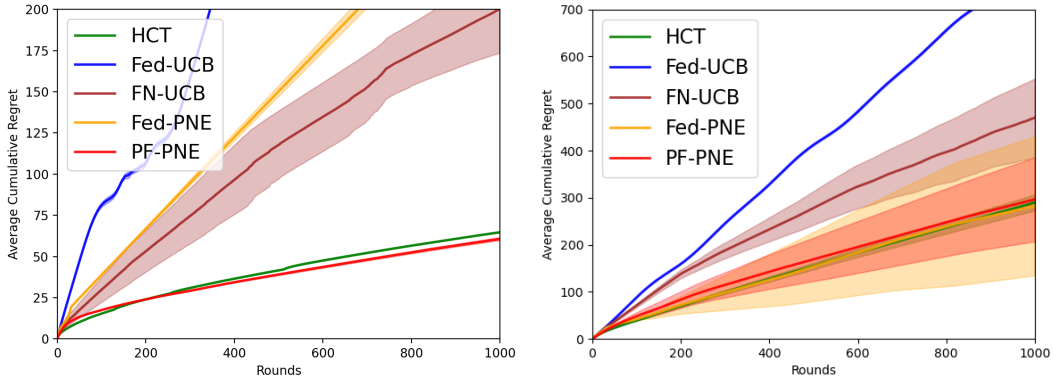
C EXPERIMENTAL DETAILS

In this section, we provide all the details related to the algorithms, datasets, and hyper-parameters in Section 4. We also provide more federated $\mathcal{X}$-armed bandit experiments.

C.1 Algorithms and Hyper-parameters

For the implementation of hierarchical partitioning and centralized $\mathcal{X}$-armed bandit algorithms, we have used the publicly available open-source package PyXAB by Li et al. [2023a]. We list the algorithms used in our experiments and the hyper-parameter settings of these algorithms.

- `HCT`. The `HCT` algorithm is a (single-client) $\mathcal{X}$-armed bandit algorithm proposed by Azar et al. [2014]. We have used the publicly-available implementation by Li et al. [2023a] at the link `https://github.com/WilliamLwj/PyXAB`
- `Fed1-UCB`. The `Fed1-UCB` algorithm is a multi-armed bandit algorithm proposed by Shi and Shen [2021a]. We have followed Li et al. [2022b] and generate 20 arms on each dimension randomly for each trial of the algorithm for 1-D and 2-D objective functions. For other high-dimensional functions, we have randomly generated 1000 arms for Fed1-UCB. The hyper-parameters are set to be the same as the original paper and their codebase.
- `FN-UCB`. The `FN-UCB` algorithm is a neural bandit algorithm proposed by Dai et al. [2023]. We have used the public implementation Dai et al. [2023] at the link `https://github.com/daizhongxiang/Federated-Neural-Bandits` with the default hyperparamter choices. Similar to `FN-UCB`, we have generated 20 arms on each dimension randomly for each trial of the algorithm for 1-D and 2-D objective functions. For other high-dimensional functions, we have randomly generated 1000 arms.



(a) Doublesine (1D) (b) Ackley (2D)

Figure 2: Cumulative regret of different algorithms on the synthetic functions. Unlimited communications are allowed for centralized algorithms.

- `Fed-PNE`. We have followed Li et al. [2022b] and used their parameter settings for the `Fed-PNE` algorithm.
  - The smoothness parameters ν_1 and ρ are set to be $\nu_1 = 1$ and $\rho = 0.5$.
  - The confidence parameters c and c_1 are set to be $c = 0.1$ and $c_1 = 1$.

  Notably, the performance of `Fed-PNE` is originally measured by the global regret, i.e., the regret on the average of all local objective. However, in this paper we measure the performance on the local objectives. We believe this is the main reason why `Fed-PNE` performs non-ideally in our experiments.

- `PF-PNE`. Since `PF-PNE` can be viewed as an "upgraded" version of `Fed-PNE`, we have used the same hyper-parameter setting as the `Fed-PNE`, i.e., $\nu_1 = 1, \rho = 0.5, c = 0.1$ and $c_1 = 1$. For the additional hyper-parameter Δ, we have set it to be $\Delta = 0.01$ in all the experiments. Tuning these hyper-parameters will not affect the final result too much.

C.2 Objective Functions and Dataset

Synthetic Functions. Garland, DoubleSine, Himmelblau, and Rastrigin are synthetic functions that are used very frequently in the experiments of $\mathcal{X}$-armed bandit algorithms because of their large number of local optimums and their extreme unsmoothness, which appeared in works such as Azar et al. [2014], Grill et al. [2015], Shang et al. [2019], Bartlett et al. [2019], Li et al. [2023b]. Garland and DoubleSine are defined on the domain $[0, 1]$, Himmelblau is defined on $[-5, 5]$, while Rastrigin can be defined on $[-1, 1]^k$ where k is an arbitrarily large integer. We have normalized these functions so that their values are between [0, 1] to fulfill the requirements in the analysis. The local objectives are the shifted versions of the original objectives, with a random shift on each dimension. Random noise is added to the function evaluations.

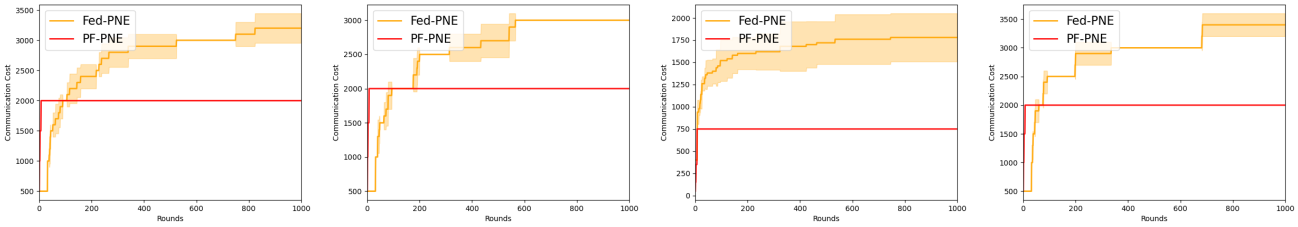
Landmine Dataset. The landmine dataset contains multiple landmine fields with features from radar images. We have followed Dai et al. [2020] and split the dataset into equal-sized training set and testing set. Each client randomly chooses one landmine field and optimize one SVM machine to detect the landmines in the particular field. The local objectives are the AUC-ROC scores on one landmine objective. The original dataset can be downloaded from `http://www.ee.duke.edu/~lcarin/LandmineData.zip`

C.3 Additional Experiments

We have conducted additional experiments on two more synthetic objectives Doublesine (1D) and Ackley (2D). Similarly, we add random shifts to the each dimension of the original objectives to produce the local objectives of each client. The experimental results are similar to what we present in the main paper. `PF-PNE` performs slightly better than `HCT` and much better than `Fed-PNE` on Doublesine. On the other hand, `PF-PNE` performs similarly as `HCT` and `Fed-PNE` on Ackley. Both results are aligned with our theoretical analysis.

C.4 Communication Cost

We provide the communication cost comparison between `Fed-PNE` and `PF-PNE` on the synthetic objectives, as shown in Figure 3. As can be observed, the communication cost of `Fed-PNE` keeps increasing. However, the communication cost of `PF-PNE` stops to increase after a certain point in the learning process, proving the correctness of our theory.



(a) Garland (1D) (b) Himmelblau (2D) (c) Rastrigin (10D) (d) Ackley (2D)

Figure 3: Communication cost comparison between `Fed-PNE` and `PF-PNE`