# Federated Linear Contextual Bandits with Heterogeneous Clients

**Ethan Blaser**
University of Virginia

**Chuanhao Li**
University of Virginia

**Hongning Wang**
University of Virginia

## Abstract

The demand for collaborative and private bandit learning across multiple agents is surging due to the growing quantity of data generated from distributed systems. Federated bandit learning has emerged as a promising framework for private, efficient, and decentralized online learning. However, almost all previous works rely on strong assumptions of client homogeneity, i.e., all participating clients shall share the same bandit model; otherwise, they all would suffer linear regret. This greatly restricts the application of federated bandit learning in practice. In this work, we introduce a new approach for federated bandits for heterogeneous clients, which clusters clients for collaborative bandit learning under the federated learning setting. Our proposed algorithm achieves non-trivial sub-linear regret and communication cost for all clients, subject to the communication protocol under federated learning that at anytime only one model can be shared by the server.

## 1 INTRODUCTION

Bandit learning algorithms (Auer et al., 2002; Chapelle and Li, 2011; Li et al., 2010a; Abbasi-Yadkori et al., 2011) have become a reference solution to the problems of online decision optimization in a wide variety of applications, including recommender systems (Li et al., 2010a), clinical trials (Durand et al., 2018), and display advertising (Li et al., 2010b). Typically, these algorithms are operated by a centralized server; but due to the growing quantity of data generated from distributed systems, there is a surge in demand for private, efficient, and decentralized bandit learning across multiple clients. Federated bandit learning has emerged as a promising solution framework, where multiple clients collaborate to minimize their cumulative regret under the coordination of a single central server (Wang et al., 2020; Dubey and Pentland, 2020b; Shi and Shen, 2021; Li and Wang, 2022; He et al., 2022). The server's role is limited to facilitating joint model estimation across clients, without having access to any clients' arm pulling or reward history.

Although federated bandit learning has gained increasing interest from the research community, most existing approaches necessitate that all clients share the same underlying bandit model in order to achieve near-optimal sub-linear regret for a population of clients. This strong homogeneity assumption distills federated bandit learning to a joint estimation of a single global model across clients, subject to the federated learning communication protocol (Bonawitz et al., 2019; Kairouz et al., 2021). However, in reality, clients can have diverse objectives, resulting in different optimal policies. Imposing a single global model on a heterogeneous client population can easily cost every client linear regret (Hossain et al., 2021). Consequently, rational clients should choose not to participate in such a federated learning system, as they cannot determine if other participating clients share the same bandit model with them beforehand, and they can already achieve sub-linear regret independently (albeit inferior to the regret obtained when all clients genuinely share the same bandit model). This seriously impedes the practical application of existing federated bandit learning solutions.

In a parallel line of bandit research, studies in collaborative bandits aim to improve bandit learning in heterogeneous environments by facilitating collective model estimation among different clients. For example, clustered bandit algorithms group similar clients and use a shared bandit model for clients within the same group (Gentile et al., 2014; Li et al., 2016; Gentile et al., 2017; Cesa-Bianchi et al., 2013; Wu et al., 2016). When the relatedness among clients are provided, such as through an affinity graph, joint policy learning can be performed analytically (Cesa-Bianchi et al., 2013; Wu et al., 2016). However, most of the existing collaborative bandit learning algorithms operate under a centralized setting, in which data from all

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

clients is assumed to be directly accessible by a central server. As a result, these methods cannot address the demand for privacy and communication efficiency in online learning for distributed systems. Significant efforts are required to adapt these algorithms to distributed settings Mahadik et al. (2020).

In this paper, we introduce a novel approach for federated bandit learning among heterogeneous clients, extending collaborative bandit learning to the standard federated learning setting. The goal is to ensure that every participating client achieves regret reduction compared to their independent learning, thereby motivating all clients to participate. As the first work of this kind, we focus on estimating a linear contextual bandit model (Li et al., 2010a; Abbasi-Yadkori et al., 2011) for each client, which is also the most commonly employed model in federated bandits. Not surprisingly, regret reduction in a population of heterogeneous clients can be realized by clustering the clients, where collective model estimation is only performed within each cluster. But the key challenges lie in the communication protocol in federated learning. First, the server lacks real-time access to each client's data, resulting in delayed inferences of client clusters. Second, the server can only estimate and broadcast one global model at a time (He et al., 2020; Foley et al., 2022). This can cause communication congestion and delay the model updates. Both of them cost regret.

To address these challenges we develop a two-stage federated clustered bandit algorithm. In the first stage, all clients perform pure exploration to prepare a non-parametric clustering of clients based on the statistical homogeneity test (Li et al., 2021). Then in the second stage, a first-in-first-out queue is maintained on the server side to facilitate event-triggered communication (Wang et al., 2020) at the cluster level. We rigorously establish the upper bounds of cumulative regret and communication cost for this algorithm. Then, we empirically enhance the algorithm by allowing dynamic re-clustering of clients in the second stage and employ a priority queue to improve regret. We conduct comprehensive empirical comparisons of the newly proposed federated bandit algorithm against a set of representative baselines to demonstrate the effectiveness of our proposed framework.

## 2 RELATED WORK

Our work is closely related to studies in federated bandit learning and collaborative bandits. In this section, we discuss the most representative solutions in each area and highlight the relationships between them and our work.

**Federated Linear Contextual Bandits:** There have been several works that study the federated linear contextual bandit setting, where multiple clients work collaboratively to minimize their cumulative regret with the coordination of a single central server (Wang et al., 2020; Li and Wang, 2022; Huang et al., 2021). Wang et al. (2020) introduced DisLinUCB, where a set of homogeneous clients, each with the same linear bandit parameter, conduct joint model estimation through sharing sufficient statistics with a central server. Li and Wang (2022) and He et al. (2022) extended this setting by introducing an event-triggered asynchronous communication framework to achieve sub-linear communication cost as well as sub-linear regret in a homogeneous environment. Additionally, Dubey and Pentland (2020a) considers deferentially private federated contextual bandits in peer-to-peer communication networks. Fed-PE, proposed in (Huang et al., 2021), is a federated phase-based elimination algorithm for linear contextual bandits that handles both homogeneous and heterogeneous settings. However, in their setting, the client is trying to learn the fixed context vectors associated with each arm as opposed to the linear reward parameter (which is known in their setting). With the exception of Fed-PE, which utilizes a different bandit formulation altogether, all of these prior work rely on strong assumptions of client homogeneity, while our work seeks to extend federated linear contextual bandit learning to a heterogeneous environment.

**Collaborative Bandits:** Collaborative bandits seek to leverage similarities between heterogeneous clients to improve bandit learning. Clustered bandit algorithms are one example, where similar clients are grouped together, and a shared bandit model is used for all clients in the same group(Gentile et al., 2014; Li et al., 2016; Gentile et al., 2017; Cesa-Bianchi et al., 2013; Wu et al., 2016). Gentile et al. (2014) assumed that observations from different clients in the same cluster are associated with the same underlying bandit parameter. Gentile et al. (2017) further studied context-dependent clustering of clients, grouping clients based on their similarity along their bandit parameter's projection onto each context vector. Li et al. (2021) unified non-stationary and clustered bandit by allowing for a time varying bandit parameter for each client, which requires online estimation of the dynamic cluster structure at each time. Other works leverage explicit inter-client and inter-arm relational structures, such as social networks (Buccapatnam et al., 2013; Cesa-Bianchi et al., 2013; Wu et al., 2016; Hong et al., 2021; Caron et al., 2012; Mannor and Shamir, 2011) to facilitate collaboration. However, most existing collaborative bandit solutions are designed under a centralized setting, where all clients' observation data is readily available at a central server. Liu et al. (2022) and Korda et al. (2016) consider online cluster estimation in a distributed setting. However, their federated learning architectures do not align with the standard federated learning architecture and real world implementations where a single central server broadcasts a single global model at each timestep (McMahan et al., 2016; He et al., 2020; Foley et al., 2022). Specifically, Liu et al. (2022) utilizes a hierarchical server configuration that is distinct from the standard single-server FL setup. On the other hand, Korda et al. (2016) is based on a peer-to-peer (P2P) communication network, which stands in contrast to the centralized communication model and also overlooks the potential communication costs associated with such a decentralized approach.

# 3 METHODOLOGY

In this section, we begin by outlining the problem setting investigated in this work. Then we present our two-stage federated clustered bandit algorithm designed to serve a population of heterogeneous clients under the standard communication setup in federated learning. We provide theoretical analysis of the upper regret bound for our developed solution. Lastly, we introduce a set of improvements to our proposed algorithm, including dynamic re-clustering of clients using an adaptive clustering criterion, and the implementation of a priority queue to enhance online performance, both of which were found empirically effective.

## 3.1 Problem Setting

A federated bandit learning system consists of two components: 1) $N$ clients, which take actions and get reward feedback from their environment (e.g., edge devices in a recommendation system interacting with end users) and 2) a central server coordinating client communication for collaborative model estimation. In each time step $t = 1, 2, ...T$, each client $i \in N$ chooses an action $x_{t,i}$ from its action set $\mathcal{A}_{t,i} = \{x_{t,1}, x_{t,2}..., x_{t,K}\}$, where $x \in \mathbb{R}^d$. Adhering to the standard linear reward assumption from (Li et al., 2010b), the corresponding reward received by client $i$ is $y_{t,i} = \langle \theta_i^*, x_{t,i} \rangle + \eta_t$, where noise $\eta_t$ comes from a $\sigma^2$ sub-Gaussian distribution, and $\theta_i^*$ is the true linear reward parameter for client $i$. Without loss of generality, we assume $\|x\|_2 \leq 1$ and $\|\theta_i^*\| \leq 1$.

The learning system interacts with the environment for $T$ rounds, aiming to minimize the cumulative pseudo-regret $R_T = \sum_{t=0}^{T} \sum_{i=0}^{N} \max_{x \in \mathcal{A}_{t,i}} \langle \theta_i^*, x \rangle - \langle \theta_i^*, x_{t,i} \rangle$.

Following the federated learning setting, we assume a star-shaped communication network, where the clients cannot directly communicate among themselves. Instead, they must share the learning algorithm's parameters (e.g., gradients, model weights, or sufficient statistics) through the central server. To preserve data-privacy, raw observations collected by each client $(x_{t,i}, y_{t,i})$ are stored locally and will not be shared with the server. At every timestep $t = 1, ...T$, the central server is capable of using the shared learning algorithm to update and broadcast one model to the selected clients. The communication cost is defined as the amount of sufficient statistics communicated across the learning system over the entire time-horizon.

Unlike existing federated bandit works (Wang et al., 2020; Li and Wang, 2022; He et al., 2022) which assume homogeneous clients, we adopt the standard clustered bandit setting to model a heterogeneous learning environment. Without an underlying cluster structure in the environment, collaboration between clients would be infeasible. Therefore, we assume that clients sharing similar reward models form clusters, collectively represented as $\mathcal{C} = \{C_1, C_2, ..., C_M\}$. The composition and quantity of these clusters, are unknown to the system, necessitating on-the-fly inference. Consistent with prevalent clustered bandit practices (Gentile et al., 2014, 2017; Liu et al., 2022), we use unknown environmental parameters $\epsilon$ and $\gamma$ to delineate the ground-truth cluster structures:

**Assumption 1** (Proximity within clusters)**.** *For any two clients $i, j$ within a particular cluster $C_k \in \mathcal{C}$, $\|\theta_i^* - \theta_j^*\| \leq \epsilon$ where $\epsilon = 1/(N\sqrt{T})$.*

**Assumption 2** (Separateness among clusters)**.** *For any two clusters $C_k, C_l \in \mathcal{C}$, $\forall i \in C_k$, $j \in C_l$, $\|\theta_i^* - \theta_j^*\| \geq \gamma \geq 0$ (Gentile et al., 2014, 2017; Li et al., 2021; Liu et al., 2022).*

Contrary to previous clustered bandit assumptions of identical reward models within a cluster, our Assumption 1 offers more flexibility. It enables similar clients (represented by $\epsilon$) to collaborate, amplifying the system's collaborative benefit. We also adopt a standard context regularity assumption found in clustered bandits.

**Assumption 3** (Context regularity)**.** *At each time $t$, $\forall i \in \{N\}$ arm set $\mathcal{A}_{t,i}$ is generated i.i.d. from a sub-Gaussian random vector $x_{t,i} \in \mathbb{R}^d$, such that $\mathbb{E}[x_{t,i} x_{t,i}^\top]$ is full-rank with minimum eigenvalue $\lambda_c > 0$ (Gentile et al., 2014, 2017; Li et al., 2019).*

Notably, our context regularity Assumption 3 is weaker than those in (Gentile et al., 2014, 2017; Li et al., 2019). Ours only requires the lower bound on the minimum eigenvalue of $\mathbb{E}[x_{t,i} x_{t,i}^\top]$, while others require the imposition of a variance condition on the stochastic process generating $x_{t,i}$.

To facilitate our later discussions, we use $\mathcal{H}_{t,i} = \{(x_{\tau,i}, y_{\tau,i})\}_{\tau=1}^{t}$ to represent the set of $t$ observations from client $i$. $(\mathbf{X}_i, \mathbf{y}_i)$ denote design matrices and feedback vectors of $\mathcal{H}_{t,i}$ where each row of $\mathbf{X}$ is the context vector of an arm and the corresponding element in $\mathbf{y}$ is the observed reward for this arm. Note that $\mathbf{X}_j$ only contains the observations made by client $j$ and does not include aggregated observations from other clients in the cluster. We also define the weighted norm of a vector $x \in \mathbb{R}^d$ as $\|x\|_A = \sqrt{x^\top A x}$, where $A \in \mathbb{R}^{d \times d}$ is a positive definite matrix.

### 3.2 Algorithm: HetoFedBandit

In this section, we present our two-stage federated clustered bandit algorithm. As discussed in Section 1, there are two primary challenges associated with extending clustered bandit learning to the federated learning setting. The first challenge is to identify the subsets of heterogeneous clients that can benefit from collaboration among themselves. To achieve this, in the first stage of our algorithm, all clients conduct random exploration ahead of a non-parametric clustering of clients based on the statistical homogeneity test (Li et al., 2021). The second challenge arises from the communication network setting in federated learning framework, which allows only one model to be broadcast at each time step (He et al., 2020; Foley et al., 2022). To accommodate this constraint, a first-in-first-out queue is utilized on the server side, enabling event-triggered collaboration (Wang et al., 2020) at the cluster level during our algorithm's second phase. We provide an overview of the key components of our algorithm, with the full details available in Algorithm 2.

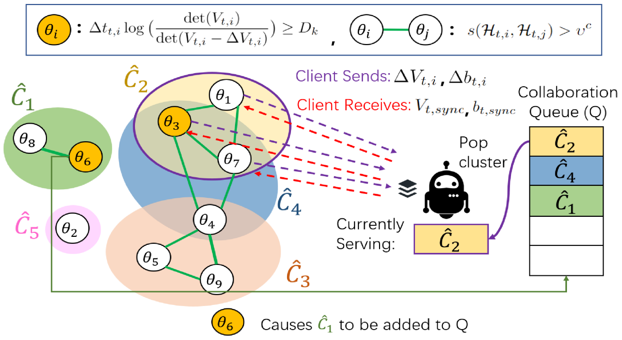

$\theta_i$ : $\Delta t_{t,i} \log(\frac{\det(V_{t,i})}{\det(V_{t,i} - \Delta V_{t,i})}) \geq D_k$ ,
$\theta_i$ — $\theta_j$ : $s(\mathcal{H}_{t,i}, \mathcal{H}_{t,j}) > \upsilon^c$
$\hat{C}_1$
$\hat{C}_2$
$\hat{C}_3$
$\hat{C}_4$
$\hat{C}_5$
$\theta_1$
$\theta_2$
$\theta_3$
$\theta_4$
$\theta_5$
$\theta_6$
$\theta_7$
$\theta_8$
$\theta_9$
Client Sends: $\Delta V_{t,i}$, $\Delta b_{t,i}$
Client Receives: $V_{t,sync}$, $b_{t,sync}$
Collaboration Queue (Q)
$\hat{C}_2$
$\hat{C}_4$
$\hat{C}_1$
Pop cluster
Currently Serving:
$\hat{C}_2$
$\theta_6$ Causes $\hat{C}_1$ to be added to Q

Figure 1: Execution of HetoFedBandit after pure exploration phase $T_0$. Each client $i \in N$ is represented by a node in the client graph $\mathcal{G}$ on the left hand side. Edges between clients indicate potential collaborators, as defined by the homogeneity test. The colored ellipsoids represent the estimated clusters $\hat{\mathcal{C}} = \{\hat{C}_1, ..., \hat{C}_5\}$, which are the maximal cliques of $\mathcal{G}$. Clients exceeding their communication threshold are highlighted in orange. Currently, client $\theta_6$ has exceeded the communication threshold $D_1$ for cluster $\hat{C}_1$, which causes cluster $\hat{C}_1$ to be added to the queue. The server pops cluster $\hat{C}_2$ from the queue and facilitates collaboration among $\{\theta_1, \theta_3, \theta_7\}$. In the next timestep, the server will serve cluster $\hat{C}_4$, queued for removal.

**Pure Exploration Phase** Under our relaxed context regularity assumption, we execute a short exploration phase of length $T_0$ to guarantee the accuracy of our homogeneity test. Our discussion on the choice of $T_0$ is deferred to Section 3.3. Although our derived theoretical value for $T_0$ depends on an unknown environmental parameter $\gamma$, in practice, $T_0$ can be tuned as a hyperparameter.

During this exploration stage, for each $t \in \{0...T_0\}$, every client $i \in [N]$ selects an action $x_{t,i}$ by uniformly sampling from $\mathcal{A}_{t,i}$ in parallel. After receiving reward $y_{t,i}$, each client updates their local sufficient statistics $V_{t,i} = V_{t-1,i} + x_{t,i}x_{t,i}^\top$ and $b_{t,i} = b_{t-1,i} + x_{t,i}y_{t,i}$. Upon completion of $T_0$ rounds of pure exploration, each client then shares its sufficient statistics $(V_{T_0,i}, b_{T_0,i})$ to the central server. We present the complete exploration algorithm in Appendix A.

**Cluster Estimation** The key challenge in online clustering of bandits is to measure the similarity between different bandit models. Previous works identify whether a set of clients share exactly the same underlying reward model; while we cluster similar clients (as defined by $\epsilon$) to widen the radius of beneficial collaboration. We realize this by testing whether $\|\theta_1^* - \theta_2^*\| \leq \epsilon$ via the homogeneity test introduced in (Li et al., 2021).

Specifically, we utilize a $\chi^2$ test of homogeneity, where the test statistic $s(\mathcal{H}_{t,1}, \mathcal{H}_{t,2})$ follows the non-central $\chi^2$-distribution (Chow, 1960; Cantrell et al., 1991). The test determines whether the parameters of linear regression models associated with two datasets are similar, assuming equal variance. Since $\theta_1^*$ and $\theta_2^*$ are unobservable, the test utilizes the maximum likelihood estimator (MLE) for $\theta$ on a dataset $\mathcal{H}$, which we denote $\vartheta = (\mathbf{X}^\top\mathbf{X})^-\mathbf{X}^\top\mathbf{y}$, where $(\cdot)^-$ denotes the generalized matrix inverse:

$$s(\mathcal{H}_{t,1}, \mathcal{H}_{t,2}) = \frac{||\mathbf{X}_1(\vartheta_1 - \vartheta_{1,2})||^2 + ||\mathbf{X}_2(\vartheta_2 - \vartheta_{1,2})||^2}{\sigma^2} \tag{1}$$

where $\vartheta_{1,2}$ denotes the estimator using data from $\mathcal{H}_{t,1}$ and $\mathcal{H}_{t,2}$.

When $s(\mathcal{H}_{t,1}, \mathcal{H}_{t,2})$ exceeds a chosen threshold $\upsilon^c$, it indicates a deviation between the combined estimator and the individual estimators on the two datasets. Thus, we conclude $\|\theta_1^* - \theta_2^*\| > \epsilon$; otherwise, we conclude $\|\theta_1^* - \theta_2^*\| \leq \epsilon$. Therefore, to determine the sets of clients that can collaborate, the central server performs this pairwise homogeneity test among each pair of clients. If two clients $i, j$ satisfy the homogeneity test $s(\mathcal{H}_{T_0,i}, \mathcal{H}_{T_0,j}) \geq \upsilon^c$, then we add an undirected edge between them in a client graph $\mathcal{G}$ indicating they benefit from mutual collaboration.

Next, our algorithm uses $\mathcal{G}$ to determine the clusters of clients that can benefit from collaboration. Because our algorithm allows collaboration between non-identical clients, we must ensure every client within a cluster is sufficiently similar to every other; otherwise, linear regret can be caused by the incompatible model sharing. For this purpose, we require each of our estimated clusters be a maximal clique of $\mathcal{G}$ (line 5 in Algorithm 1). We denote the set of resulting clusters as $\hat{\mathcal{C}} = \{\hat{C}_1, \hat{C}_2, ...\hat{C}_{\hat{M}}\}$ and the set of cluster indices in $\mathcal{C}$ that client $i$ belongs to as $\mathcal{K}_i$.

Using the maximal cliques of $\mathcal{G}$ as the cluster estimates introduces a unique challenge that affects our subsequent algorithmic design: the estimated clusters may not be disjoint. In Figure 1, we can see that client 3, represented by $\theta_3$, is a member of two clusters, $\hat{C}_2$ and $\hat{C}_4$. Therefore, $\theta_3$ will receive shared model updates from clients $\{\theta_1, \theta_7, \theta_4\}$. However, the absence of an edge between $\theta_1$ and $\theta_4$ implies that simultaneous collaboration between $\{\theta_3, \theta_1\}$ and $\{\theta_3, \theta_4\}$ is not allowed by our algorithm. As a result, when $\theta_3$ is collaborating with $\theta_1$, it should only share its local data, excluding what it has received from the server. Later we describe our queue-based sequential approach to resolve this.

**Algorithm 1** Cluster Estimation

```
1: for $(i, j) \in N$ do
2:     if $s(\mathcal{H}_{T_0,i}, \mathcal{H}_{T_0,j}) \le \upsilon^c$ then add edge $e(i, j)$ to $\mathcal{G}$
3: end for
4: $\hat{\mathcal{C}} = \{\hat{C}_1, \hat{C}_2, ...\hat{C}_{\hat{M}}\} = \text{maximal\_cliques}(\mathcal{G})$
5: Set $\mathcal{K}_i = \{k : i \in \hat{C}_k\}$ for each client $i$
```

**Optimistic Learning Phase** Upon identifying client clusters suitable for collaboration, we proceed to the optimistic learning phase of our algorithm. Here, clients optimistically choose arms, utilizing the collaboration with other similar clients to enhance their local model estimates. At each time step $t \in \{T_0...T\}$, each client $i \in [N]$ optimistically selects an arm $x_{t,i} \in \mathcal{A}_{t,i}$ using the UCB strategy based on its sufficient statistics $\{V_{t,i}, b_{t,i}\}$:

$$x_t = \arg\max_{x \in \mathcal{A}_{t,i}} x^\top \hat{\theta}_{t-1,i} + \text{CB}_{t-1,i}(x) \tag{2}$$

where $\hat{\theta}_{t-1,i} = \overline{V}_{t-1,i}^{-1} b_{t-1,i}$ is the ridge regression estimator with regularization parameter $\lambda$; $\overline{V}_{t-1,i} = V_{t-1,i} + \lambda I$; and the confidence bound of reward estimation for arm $x$ is $\text{CB}_{t-1,i}(x) = \alpha_{t-1,i} \|x\|_{\overline{V}_{t-1,i}^{-1}}$, where $\alpha_{t-1,i} = \sigma\sqrt{2\log\left(\frac{\det(\overline{V}_{t-1,i})^{1/2}}{\delta \det(\lambda I)^{1/2}}\right)} + \sqrt{\lambda}$. Note that $V_{t,i}$ is formulated using data locally collected by client $i$ in conjunction with data from the clients with whom client $i$ has previously collaborated. After client $i$ observes reward $y_{t,i}$, it updates its local sufficient statistics to improve the reward estimates in future rounds.

**Communication Protocol** Our algorithm integrates the event-triggered communication protocol from Wang et al. (2020) to efficiently balance communication and regret minimization within clusters. It uses delayed communication, where clients store observations and rewards in a local buffer $\Delta V_{t,i}$ and $\Delta b_{t,i}$. Clients request server collaboration when the informativeness of the stored updates surpass a certain threshold. Specifically, If $\Delta t_{t,i} \log(\det(V_{t,i}) / \det(V_{t,i} - \Delta V_{t,i})) \ge D_k$ for any $k \in \mathcal{K}_i$, with $D_k$ as the communication threshold for the estimated cluster $\hat{C}_k$, the client sends a collaboration request for $\hat{C}_k$.

Multiple clients across different clusters can trigger simultaneous communication requests, and single clients can request collaboration for multiple clusters because the estimated clusters are not disjoint. In such cases, the central server uses a first-in-first-out queue (FIFO) $Q$ to manage the clusters needing collaboration one-by-one. At each timestep $t \in \{T_0...T\}$, it serves one cluster from the queue, ensuring no inter-cluster data contamination by computing $V_{t,sync}$ and $b_{t,sync}$ using only the clients' upload buffers. Despite the single global-model restriction in federated learning, our algorithm still helps multiple groups of similar clients in a pseudo round-robin manner. As a result, a cluster of clients can resume engaging with the environment without being hindered by the server's processing time for unrelated clusters that don't offer collaborative advantage.

This queuing strategy enhances the system's efficiency, allowing clusters to re-engage with the environment without idling for the server's processing of all other clusters. However, before computing and sharing $\{V_{t,sync}, b_{t,sync}\}$ for collaboration, our algorithm mandates the complete upload of local buffers from every client in that cluster. This signifies that our algorithm employs asynchronous communication at the cluster level but still requires synchronous communication among clients within the same cluster. In practical distributed systems, clients often exhibit variable response times and occasional unavailability. Adapting our algorithm to support asynchronous communication at the individual client level, such that they can collaborate without awaiting updates from all other clients within the cluster, remains an important open research question.

### 3.3 Theoretical Results

As presented in Section 3.2, our algorithm first utilizes a homogeneity test to cluster similar clients in a heterogeneous environment. We prove that with our homogeneity test, Algorithm 1 correctly identifies the underlying clusters.

**Theorem 3.1 (Clustering Correctness).** *Under the condition that we set the homogeneity test threshold* $\upsilon^c \ge F^{-1}(1 - \frac{\delta}{N^2}, df, \psi^c)$, *with probability at least* $1 - \delta$, *we have* $\hat{\mathcal{C}} = \mathcal{C}$.

**Algorithm 2** HetoFedBandit

```
Input: $T$, $\delta \in (0,1)$, exploration length $T_0$, $\lambda > 0$, neighbor identification $\upsilon^c$
Initialization: Clients: $\forall i \in N$: $V_{0,i} = \mathbf{0}_{d \times d}, b_{0,i} = \mathbf{0}_d, \mathcal{H}_{0,i} = \emptyset, \Delta V_{0,i} = \mathbf{0}_{d \times d}, \Delta b_{0,i} = \mathbf{0}_d, \Delta t_{i,0} = 0$, $\mathcal{K}_i = \emptyset$ ; Server: Client graph $\mathcal{G}$ with $N$ nodes, FIFO queue $Q$;
Pure Exploration Phase (Algorithm 3)
Cluster Estimation (Algorithm 1)
Cluster communication thresholds $\mathcal{D} = [D_1, ..., D_{\hat{M}}]$ where $D_k = (T \log |\hat{C}_k| T)/(d|\hat{C}_k|)$
for $t = T_0 + 1, ..., T$ do
    for Client $i \in N$ do
        Choose arm $x_{t,i} \in \mathcal{A}_{t,i}$ by Eq. 2 observe reward $y_{t,i}$
        Update client $i$: $\mathcal{H}_{t,i} = \mathcal{H}_{t-1,i} \cup (x_{t,i}, y_{t,i})$, $V_{t,i} \mathrel{+}= x_{t,i} x_{t,i}^\top$, $b_{t,i} \mathrel{+}= x_{t,i} y_{t,i}$,
        $\Delta V_{t,i} \mathrel{+}= x_{t,i} x_{t,i}^\top$, $\Delta b_{t,i} \mathrel{+}= x_{t,i} y_{t,i}, \Delta t_{t,i} \mathrel{+}= 1$
        for $k \in \mathcal{K}_i$ do
            if $\Delta t_{t,i} \log(\det(V_{t,i}) / \det(V_{t,i} - \Delta V_{t,i})) \geq D_k$ then
                Collaboration Request: Server adds $\hat{C}_k$ to $Q$
            end if
        end for
    end for
    if $Q$ is non-empty then
        Server pops $\hat{C}_k$ from $Q$
        Every client $i \in \hat{C}_k$ sends $\Delta V_{t,j}$, $\Delta b_{t,j}$ to server
        Each client in $\hat{C}_k$ receives $V_{t,sync} = \sum_{j \in \hat{C}_k} \Delta V_{t,j}$, $b_{t,sync} = \sum_{j \in \hat{C}_k} \Delta b_{t,j}$ from the server
        Local client updates: $V_{t,i} \mathrel{+}= V_{t,sync} - \Delta V_{t,i}$, $b_{t,i} \mathrel{+}= b_{t,sync} - \Delta b_{t,i}$, $\Delta V_{t,i} = 0$, $\Delta b_{t,i} = 0, \Delta t_{t,i} = 0$
    end if
end for
```

$F^{-1}(\cdot)$ is the inverse of the CDF of the non-central $\chi^2$ distribution, and $\psi^c \doteq \frac{1}{\sigma^2}$. We provide the complete proof of Theorem 3.1 in Appendix C. Moreover, Algorithm 2 adopts a UCB-based arm selection, which requires the construction of a confidence ellipsoid.

**Lemma 3.2** (**Confidence Ellipsoids**). *Suppose client $i$ is a member of cluster $\hat{C}_k \in \hat{\mathcal{C}}$, and is therefore collaborating with clients $j \in \hat{C}_k$. For any $\delta > 0$, with probability at least $1 - \delta$, for all $t \geq 0$ and all clients $i \in N$, $\theta_i^*$ lies in the set:*

$$\beta_{t,i} = \Bigg\{ \theta \in \mathbb{R}^d : \|\hat{\theta}_{t,i} - \theta\|_{\overline{V}_{t,i}} \leq \sigma \sqrt{2 \log\left( \frac{\det(\overline{V}_{t,i})^{1/2}}{\det(\lambda I)^{1/2} \delta} \right)} + \sqrt{\lambda} + \Big\| \sum_{j \in \hat{C}_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Big\|_{\overline{V}_{t,i}^{-1}} \Bigg\}$$

We provide the complete proof of Lemma 3.2 in Appendix D.

Our algorithm enables collaboration among heterogeneous clients, which introduces extra biases represented by the term $H = \| \sum_{j \in \hat{C}_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j - \theta_i) \|_{\overline{V}_{t,i}^{-1}}$. With improved analysis compared with Li et al. (2021), we utilize our cluster estimation procedure Algorithm 1 to control the magnitude of the bias term $H$ by judiciously picking the threshold $\upsilon^c$.

Then, based on the constructed confidence ellipsoid, we prove Theorem 3.3, which provides upper bounds of the cumulative regret $R_T$ and communication cost $C_T$ incurred by HetoFedBandit. While the good/bad epoch decomposition used in our analysis is first introduced by Wang et al. (2020), additional care needs to be taken when bounding the extra regret introduced by the delays in serving clusters before they can be removed from the queue. We present the complete proof of Theorem 3.3 in Appendix E.

**Theorem 3.3** (**Regret and Communication Cost**). *With an exploration phase length of $T_0 = \frac{16\psi^d \sigma^2}{\lambda_c \gamma^2}$, with probability $1 - \delta$ our protocol achieves a cumulative regret of*

$$R_T = O\bigg( \frac{N\psi^d \sigma^2}{\lambda_c \gamma^2} + \sum_{k=1}^{M} d\sqrt{|C_k| T} \log^2(|C_k| T) + d|C_k|^2 M \log(|C_k|\, T) \bigg) \tag{3}$$

*where $\psi^d = F^{-1}\big(\frac{\delta}{N^2(M-1)}; d, \upsilon^c\big)$, with communication cost*

$$C_T = O(Nd^2) + \sum_{k=1}^{M} O(|C_k|^{1.5} \cdot d^3) \tag{4}$$

**Remark 1.** *The regret upper-bound has three components. The first term is our version of the "problem hardness" (Li et al., 2021; Gentile et al., 2014) which is independent of T. This "hardness" factor is determined by the cluster separation parameter $\gamma$ from Assumption 2. The second term is the standard regret upper bound from centralized clustered bandit algorithms (Li et al., 2021; Gentile et al., 2014). The third term arises from the potential waiting time clusters may experience in the queue before the server serves them. Our communication cost matches that of an idealized algorithm executing DisLinUCB(Wang et al., 2020) within each ground-truth cluster.*

We compare our regret and communication upper-bound under three cases. **Case 1 - Single cluster:** Setting $M = 1$ reduces the problem to a nearly homogeneous setting, where every client is within $\epsilon$ of everyone else. Under this setting, our regret becomes $\tilde{O}(d\sqrt{NT})$ where logarithmic factors and factors that do not depend on $T$ (since it is assumed that $T \gg N$) are omitted in $\tilde{O}$. Additionally communication cost becomes

$O(N^{1.5}d^3)$. Our algorithm matches the regret and communication cost of (Wang et al., 2020), which is designed for homogeneous clients. **Case 2 - N clusters:** Setting $M = N$ reduces the problem to a completely heterogeneous setting, where no client can benefit from collaboration as each client is at least $\epsilon$ away from others. Algorithm 2 has regret $\tilde{O}(dN\sqrt{T})$, which recovers the regret of running LinUCB (Abbasi-Yadkori et al., 2011) independently on each client. In this setting our communication becomes $O(Nd^3)$. **Case 3 - Equal Size Clusters:** Setting $|C_k| = N/M, \forall k$ gives us $M$ clusters of equal size. In this setting our regret becomes $\tilde{O}(d\sqrt{MNT})$, where the first term recovers the results presented in Gentile et al. (2014); Li et al. (2021). Our communication becomes $O(d^3N^{1.5}/\sqrt{M})$.

### 3.4 Empirical Enhancements: HetoFedBandit-E

In this section, we describe the details of our proposed empirical enhancements to our HetoFedBandit algorithm, where we perform re-clustering to improve the quality of estimated clusters of clients and replace the first-in-first-out queue with a priority queue to help clusters where a shared model update can most rapidly reduce regret for clients in that cluster. The detailed description of our enhanced algorithm HetoFedBandit-E can be found in Algorithm 5 in Appendix F.

#### 3.4.1 Data-Dependent Clustering

We propose a data-dependent clustering procedure to enhance collaboration among clients with similar observational histories. Our homogeneity test for cluster formation ensures an upper bound on the bias term $H$, as outlined in Lemma 3.2. This term depends on the differences in underlying parameters ($\theta_i^*$ vs., $\theta_j^*$) and each client's observation history. For instance, if client $j$'s observations are in the null space of $(\theta_j^* - \theta_i^*)$, collaborating with client $j$ will not introduce excessive bias to client $i$. But without further assumptions about the context vector sequence, we must conservatively assume in our original design that every client $j$'s entire observation history aligns with $(\theta_j^* - \theta_i^*)$.

Previously, we used a homogeneity test with threshold $\epsilon = \frac{1}{N\sqrt{T}}$ to verify clients' collaboration across all timesteps. Now, we can relax the homogeneity test threshold to check if two clients can collaborate at a specific timestep $t$ by examining if $\|\theta_i - \theta_j\| \leq \epsilon = 1/\left(N\sqrt{\lambda_{\max}(\mathbf{X}_j^\top \mathbf{X}_j)}\right)$.To achieve this, we modify our algorithm to forgo single round cluster estimation. Instead, every time a client requests collaboration, we re-cluster the clients using the data-dependent thresholds for our pairwise homogeneity tests. By making these thresholds data-dependent, each client can collaborate with more neighbors earlier, boosting overall collaborative benefits in our learning system.

#### 3.4.2 Priority Queue

The second enhancement to HetoFedBandit involves utilizing a priority queue instead of a FIFO queue to determine the order in which to serve clusters requesting collaboration. As is demonstrated in (Wang et al., 2020; Li and Wang, 2022), the cumulative regret incurred by a federated bandit algorithm is determined by the determinant ratios of the clients within the system: $\Delta t_{t,i} \log\big(\frac{\det(V_{t,i})}{\det(V_{t,i}-\Delta V_{t,i})}\big)$. Since the central server cannot assist all clusters at once, determinant ratios of awaiting clients can increase as they linger in the queue. In the original HetoFedBandit, clusters are attended based on their request order. However, an earlier-joining cluster might have a slower regret accumulation compared to a later one with a larger and faster growing determinant ratio. By utilizing a priority queue that serves the clusters based on: $\arg\max_{\hat{C}_k \in \hat{\mathcal{C}}} \sum_{i \in \hat{C}_k} \Delta t_{t,i} \log(\frac{\det(V_{t,i})}{\det(V_{t,i}-\Delta V_{t,i})})$ the server ensures clusters are addressed in an order that minimizes the system-wide cumulative regret.

## 4 EXPERIMENTS

In this section, we investigate the empirical performance of HetoFedBandit and HetoFedBandit-E, by comparing them against several baseline models on both simulated and real-world datasets.

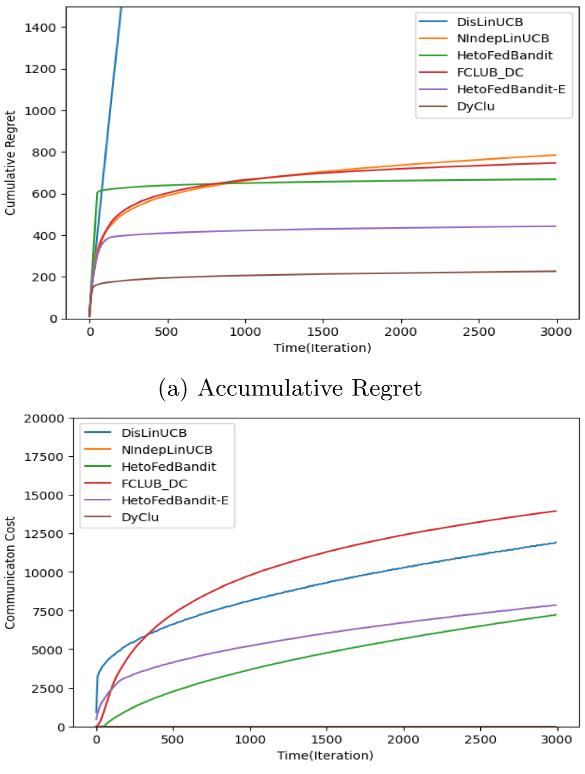
### 4.1 Baselines

In our evaluation, we compare our proposed HetoFedBandit algorithm with several representative algorithms from both the clustered and federated bandit learning domains. We compare against, LinUCB algorithm from (Abbasi-Yadkori et al., 2011), DisLinUCB (Wang et al., 2020), FCLUB_DC (Liu et al., 2022), and DyClu (Li et al., 2021). To ensure compatibility with our setting, we set the number of local servers in FCLUB_DC to be equal to the number of clients.

### 4.2 Synthetic Dataset

We first present the results of our empirical analysis of HetoFedBandit and HetoFedBandit-E on a synthetic dataset.

**Synthetic Dataset Generation** In this section, we describe the pre-processing procedure for the synthetic dataset used in Section 4.2. We first create an action pool $\{x_k\}_{k=1}^K$ where $x$ is sampled from $N(0_d, I_d)$. To create a set of $N$ clients in accordance with our environment assumptions, we first sample $M$ cluster centers $\{\theta_m\}_{m=1}^M$ from $N(0_d, I_d)$ that are $\gamma + 2\epsilon$ away from



(a) Accumulative Regret

(b) Communication Cost

Figure 2: Experimental Results on Simulated Dataset

each other (enforced via rejection sampling). Then, we randomly assign each client index $i \in N$ to one of the $M$ clusters. To generate each $\theta_i$, we first sample a vector on the unit d-sphere, then we scale it by a value uniformly sampled from $[0, \epsilon]$ with $\epsilon = 1/N\sqrt{T}$ and add it to the cluster center $\theta_k$ corresponding to the cluster client $i$ was assigned to. At each time step $t = 1, 2, ...T$ for each client in $[N]$ is presented a subset of 25 arms are sampled from $\{x_k\}_{k=1}^{K}$, and shared with the client. The reward of the selected arm is generated by the linear function governed by the corresponding bandit parameter and context. In our experiments, we chose $d = 25$, $K = 1000$, $N = 50$, $M = 5$, and $T = 3000$. Since we conducted our experiment in a synthetic environment, we utilized the known values of $\gamma = 0.85$, $\sigma = 0.1$, and chose $\lambda = 0.1$, $\delta = 0.1$ for our algorithm's hyper-parameters. Note that while we utilized the known values for $\sigma$ and $\gamma$ in our experiment, these hyper-parameters can be tuned in practice using the "doubling-trick", where the algorithm is repeatedly run in the same environment with increasing horizons (Auer et al., 1995; Besson and Kaufmann, 2018). We present additional sensitivity analysis of the environmental parameters in Appendix H due to the space limit.

**Results** In Figure 2a, we compare the accumulated regret of the different bandit algorithms on the simulated dataset. HetoFedBandit and HetoFedBandit-E outperform the other decentralized bandit baselines, with HetoFedBandit-E achieving a regret that is closest to the state-of-the-art centralized clustering bandit algorithm, DyClu. We observe that DisLinUCB experiences linear regret in our heterogeneous environment. While N-Independent LinUCB achieves sublinear regret, its cumulative regret is higher than our HetoFedBandit due to the absence of collaboration among similar clients. HetoFedBandit outperforms FCLUB_DC, underscoring the strength of our federated clustered bandit approach in a heterogeneous setting. FCLUB_DC's fixed clustering schedule results in delayed cluster identification, diminishing the quality of client collaboration. Notably, while FCLUB_DC presumes the central server aids all local client clusters concurrently at each step, HetoFedBandit still excels despite adhering to a single model assumption in federated learning. We further present an ablation study on our empirical enhancements in Appendix G.

In Figure 2b, we observe that our algorithms exhibit the lowest communication cost among baselines while achieving encouraging regret. This demonstrates the communication efficiency of our approach, which is a critical factor in distributed systems. Notably, HetoFedBandit-E has a higher communication cost compared to HetoFedBandit, because the dynamic re-clustering requires the additional sharing of sufficient statistics from clients outside the cluster that requests collaboration.

### 4.3 LastFM Dataset

In this section, we present the results of our empirical analysis of HetoFedBandit and HetoFedBandit-E on the LastFM dataset, demonstrating their effectiveness in distributed recommender systems.

**LastFM Dataset** The dataset used in this experiment is extracted from the LastFM-2k dataset, which originally contains 1892 clients (users) and 16632 items (artists) (Cantador et al., 2011). Each "client" can be considered as an edge device serving a particular user in a distributed recommender system. The "listened artists" of each client are treated as positive feedback. To adapt this dataset for our experiments, we kept clients with over 350 observations, resulting in a dataset with $N = 75$ clients and $T = 41284$ interactions. The dataset was pre-processed following the procedure in (Cesa-Bianchi et al., 2013) to accommodate the linear bandit setting (with $d = 25$ and the action set $K = 25$). Since the environmental parameters $\sigma, \gamma$ are unknown for this real-world dataset, we directly tuned the values of our test threshold $\upsilon^c = 0.01$, $T_0 = 5000$, and $\alpha_{t,i} = \alpha = 0.3$ $\forall i \in [N]$, $\forall t \in T$ using a grid search.

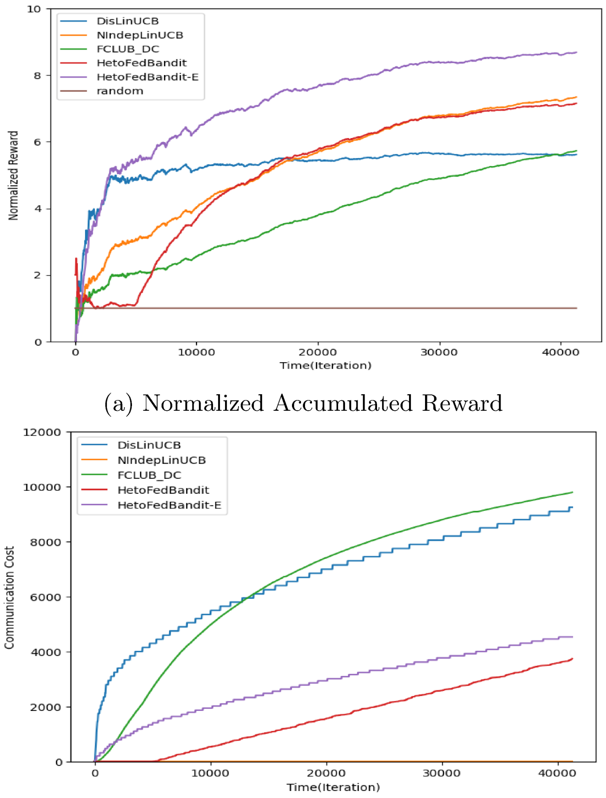
**Results** We show that our models group users with similar musical preferences for collaborative model learning, enhancing recommendation quality compared to other distributed bandit learning methods. In Figure 3, we present the normalized cumulative rewards and communication costs of the federated bandit algorithms on the LastFM dataset. We observe that HetoFedBandit-E outperforms the other decentralized bandit baselines on the real-world dataset, achieving the highest average normalized reward. In line with observations made using synthetic datasets, DisLinUCB's performance is suboptimal in environments with heterogeneous clients, demonstrated by NIndepLinUCB outperforming DisLinUCB. Moreover, while the normalized cumulative reward of FCLUB_DC shows an improving trend over time, its prefixed-clustering schedule delays the identification of the underlying cluster structure compared to HetoFedBandit and HetoFedBandit-E.

Notably, our basic algorithm design, HetoFedBandit, falls short of NIndepLinUCB on real-world data due to its single-timestep cluster estimation. In a simulated environment, where context vectors adhere closely to Assumption 3, one-time cluster estimation post-exploration is usually sufficient. However, in real-world datasets, the distribution of context vectors may evolve over time, leading to potential inaccuracies in the clusters initially estimated after the exploration phase. This underlines the importance of our empirical enhancements, which incorporates dynamic data-dependent re-clustering, demonstrating its ability to adapt to shifts in the observed context distribution.

In Figure 3b, we observe that our algorithms once again exhibit the lowest communication cost among the compared baselines. Similar to our observations in Section 4.2, HetoFedBandit-E has a higher communication cost compared to HetoFedBandit.

## 5 CONCLUSION

In this work, we address the challenge of heterogeneous clients in federated bandit learning by introducing HetoFedBandit. Our approach combines the strengths of federated learning and collaborative bandit learning, enabling efficient communication and learning among clients with diverse objectives. We demonstrate through rigorous theoretical analysis that participating clients achieve regret reduction compared to their independent learning across various environmental settings, thereby motivating all clients to participate in such a federated learning system. We also empirically demonstrate that our algorithm achieves encouraging performance compared to existing federated bandit learning solutions on both simulated and real-world datasets.



(a) Normalized Accumulated Reward

(b) Communication Cost

Figure 3: Experimental Results on LastFM Dataset

Our work not only addresses the limitations of existing federated bandit learning solutions, but also opens up new possibilities for practical applications in distributed systems. Our current approach requires that each client within the federated learning system trusts the central server to only facilitate collaboration for social good. However, in the real world, this "trust" is not something that should be naively assumed. To realize a truly federated model of bandit learning, the power to decide on collaboration should be transferred to the clients. In this regard, future research should consider viewing federated learning through the lens of mechanism design, so that each client perceives participating in the federated learning system as their best course of action.

### Acknowledgments

We thank the anonymous reviewers for their insightful suggestions and comments. This material is based upon work supported by the NSF Graduate Research Fellowship under Grant No. 1842490 and NSF Award IIS-2213700 and IIS-2128019.

## References

Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 322–331, 1995. doi: 10.1109/SFCS.1995.492488.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

L. Besson and E. Kaufmann. What doubling tricks can and can't do for multi-armed bandits, 2018.

K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, B. McMahan, et al. Towards federated learning at scale: System design. *Proceedings of machine learning and systems*, 1:374–388, 2019.

S. Buccapatnam, A. Eryilmaz, and N. B. Shroff. Multi-armed bandits in the presence of side observations in social networks. In *52nd IEEE Conference on Decision and Control*, pages 7309–7314. IEEE, 2013.

I. Cantador, P. Brusilovsky, and T. Kuflik. 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011). In *Proceedings of the 5th ACM conference on Recommender systems*, RecSys 2011, New York, NY, USA, 2011. ACM.

R. S. Cantrell, P. M. Burrows, and Q. H. Vuong. Interpretation and use of generalized chow tests. *International Economic Review*, pages 725–741, 1991.

S. Caron, B. Kveton, M. Lelarge, and S. Bhagat. Leveraging side observations in stochastic bandits. *CoRR*, abs/1210.4839, 2012. URL `http://arxiv.org/abs/1210.4839`.

N. Cesa-Bianchi, C. Gentile, and G. Zappella. A gang of bandits. In *Advances in Neural Information Processing Systems*, pages 737–745, 2013.

O. Chapelle and L. Li. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24, 2011.

G. C. Chow. Tests of equality between sets of coefficients in two linear regressions. *Econometrica: Journal of the Econometric Society*, pages 591–605, 1960.

A. Dubey and A. Pentland. Differentially-private federated linear bandits. *Advances in Neural Information Processing Systems*, 33:6003–6014, 2020a.

A. Dubey and A. t. S. Pentland. Differentially-private federated linear bandits. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6003–6014. Curran Associates, Inc., 2020b. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/4311359ed4969e8401880e3c1836fbe1-Paper.pdf`.

A. Durand, C. Achilleos, D. Iacovides, K. Strati, G. D. Mitsis, and J. Pineau. Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis. In *Machine learning for healthcare conference*, pages 67–82. PMLR, 2018.

P. Foley, M. J. Sheller, B. Edwards, S. Pati, W. Riviera, M. Sharma, P. N. Moorthy, S.-h. Wang, J. Martin, P. Mirhaji, P. Shah, and S. Bakas. Openfl: the open federated learning library. *Physics in Medicine & Biology*, 2022. doi: 10.1088/1361-6560/ac97d9. URL `http://iopscience.iop.org/article/10.1088/1361-6560/ac97d9`.

C. Gentile, S. Li, and G. Zappella. Online clustering of bandits. In *International Conference on Machine Learning*, pages 757–765, 2014.

C. Gentile, S. Li, P. Kar, A. Karatzoglou, G. Zappella, and E. Etrue. On context-dependent clustering of bandits. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1253–1262. JMLR. org, 2017.

C. He, S. Li, J. So, M. Zhang, H. Wang, X. Wang, P. Vepakomma, A. Singh, H. Qiu, L. Shen, P. Zhao, Y. Kang, Y. Liu, R. Raskar, Q. Yang, M. Annavaram, and S. Avestimehr. Fedml: A research library and benchmark for federated machine learning. *Advances in Neural Information Processing Systems, Best Paper Award at Federate Learning Workshop*, 2020.

J. He, T. Wang, Y. Min, and Q. Gu. A simple and provably efficient algorithm for asynchronous federated contextual linear bandits. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=Fx7oXUVEPW`.

J. Hong, B. Kveton, M. Zaheer, and M. Ghavamzadeh. Hierarchical bayesian bandits. *CoRR*, abs/2111.06929, 2021. URL `https://arxiv.org/abs/2111.06929`.

S. Hossain, E. Micha, and N. Shah. Fair algorithms for multi-agent multi-armed bandits. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL `https://openreview.net/forum?id=AlD5WD2ANIQ`.

R. Huang, W. Wu, J. Yang, and C. Shen. Federated linear contextual bandits, 2021. URL `https://arxiv.org/abs/2110.14177`.

P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

N. Korda, B. Szorenyi, and S. Li. Distributed clustering of linear bandits in peer to peer networks. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1301–1309, New York, New York, USA, 20–22 Jun 2016. PMLR. URL `https://proceedings.mlr.press/v48/korda16.html`.

C. Li and H. Wang. Asynchronous upper confidence bound algorithms for federated linear bandits. In G. Camps-Valls, F. J. R. Ruiz, and I. Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 6529–6553. PMLR, 28–30 Mar 2022. URL `https://proceedings.mlr.press/v151/li22e.html`.

C. Li, Q. Wu, and H. Wang. Unifying clustered and non-stationary bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 1063–1071. PMLR, 2021.

L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 661–670, New York, NY, USA, 2010a. Association for Computing Machinery. ISBN 9781605587998. doi: 10.1145/1772690.1772758. URL `https://doi.org/10.1145/1772690.1772758`.

S. Li, A. Karatzoglou, and C. Gentile. Collaborative filtering bandits. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 539–548. ACM, 2016.

S. Li, W. Chen, and K.-S. Leung. Improved algorithm on online clustering of bandits. *arXiv preprint arXiv:1902.09162*, 2019.

W. Li, X. Wang, R. Zhang, Y. Cui, J. Mao, and R. Jin. Exploitation and exploration in a performance based contextual advertising system. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 27–36, 2010b.

X. Liu, H. Zhao, T. Yu, S. Li, and J. C. S. Lui. Federated online clustering of bandits, 2022. URL `https://arxiv.org/abs/2208.14865`.

K. Mahadik, Q. Wu, S. Li, and A. Sabne. Fast distributed bandits for online recommendation systems. In *Proceedings of the 34th ACM international conference on supercomputing*, pages 1–13, 2020.

S. Mannor and O. Shamir. From bandits to experts: On the value of side-observations. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL `https://proceedings.neurips.cc/paper_files/paper/2011/file/e1e32e235eee1f970470a3a6658dfdd5-Paper.pdf`.

H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas. Federated learning of deep networks using model averaging. *CoRR*, abs/1602.05629, 2016. URL `http://arxiv.org/abs/1602.05629`.

C. Shi and C. Shen. Federated multi-armed bandits. *CoRR*, abs/2101.12204, 2021. URL `https://arxiv.org/abs/2101.12204`.

Y. Wang, J. Hu, X. Chen, and L. Wang. Distributed bandit learning: Near-optimal regret with efficient communication. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL `https://openreview.net/forum?id=SJxZnR4YvB`.

Q. Wu, H. Wang, Q. Gu, and H. Wang. Contextual bandits in a collaborative environment. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 529–538. ACM, 2016.

# A PURE EXPLORATION PHASE ALGORITHM

In this section, we provide the details of Pure Exploration Phase Algorithm introduced in Section 3.2. We present the description of this algorithm here due to the space limitations in the main paper.

**Algorithm 3** Pure Exploration Phase

```
for $t = 1, 2, ..., T_0$ do
    for Agent $i \in N$ do
        Choose arm $x_{t,i} \in \mathcal{A}_{i,t}$ uniformly at random and observe reward $y_{t,i}$
        Update agent $i$: $\mathcal{H}_{t,i} = \mathcal{H}_{t-1,i} \cup (x_{t,i}, y_{t,i})$, $V_{t,i} \mathrel{+}= x_{t,i}x_{t,i}^\top$, $b_{t,i} \mathrel{+}= x_{t,i}y_{t,i}$,
        $\Delta V_{t,i} \mathrel{+}= x_{t,i}x_{t,i}^\top$, $\Delta b_{t,i} \mathrel{+}= x_{t,i}y_{t,i}$, $\Delta t_{0,i} \mathrel{+}= 1$
    end for
end for
```

# B TECHNICAL LEMMAS

In this section, we introduce the technical lemmas utilized in the subsequent proofs in this paper.

**Lemma B.1** (Lemma 11 in Abbasi-Yadkori et al. (2011)). *Let $\{X_t\}_{t=1}^{\infty}$ be a sequence in $\mathbb{R}^d$, $V$ is a $d \times d$ positive definite matrix and define $\bar{V}_t = V + \sum_{s=1}^{t} X_s X_s^\top$, where $V = \lambda I$. Additionally we have that $\lambda_{\min}(V) \geq \max(1, L^2)$ and $\|X_t\|_2 \leq L$ for all $t$, then*

$$\log\left(\frac{\det(\bar{V_n})}{\det(V)}\right) \leq \sum_{t=1}^{T} \|X_t\|^2_{\bar{V}_{t-1}^{-1}} \leq 2\log\left(\frac{\det(\bar{V_n})}{\det(V)}\right) \tag{5}$$

**Lemma B.2** (Theorem 1 of Abbasi-Yadkori et al. (2011)). *Let $\{\mathcal{F}_t\}_{t=0}^{\infty}$ be a filtration. Let $\{\eta_t\}_{t=1}^{\infty}$ be a real-valued stochastic process such that $\eta_t$ is $\mathcal{F}_t$-measurable, and $\eta_t$ follows conditionally zero mean $R$-sub-Gaussian for some $R \geq 0$. Let $\{X_t\}_{t=1}^{\infty}$ be an $\mathbb{R}^d$-valued stochastic process such that $X_t$ is $\mathcal{F}_{t-1}$-measurable. Assume that $V$ is a $d \times d$ positive definite matrix. For any $t > 0$, define*

$$V_t = V + \sum_{\tau=1}^{t} X_\tau X_\tau^\top \quad \mathcal{S}_t = \sum_{\tau=1}^{t} \eta_\tau X_\tau$$

*Then for any $\delta > 0$, with probability at least $1 - \delta$,*

$$||\mathcal{S}_t||_{V_t^{-1}} \leq R\sqrt{2\log\frac{\det(V_t)^{1/2}}{\det(V)^{1/2}\delta}}, \quad \forall t \geq 0$$

**Lemma B.3** (Determinant-Trace Inequality). *Suppose $X_1, X_2, ..., X_t \in \mathbb{R}^d$ and for any $1 \leq \tau \leq t$, $\|X_\tau\|_2 \leq L$, Let $\overline{V_t} = \lambda I + \sum_{\tau=1}^{t} X_\tau X_\tau^\top$ for some $\lambda > 0$. Then,*

$$\det(\overline{V_t}) \leq (\lambda + tL^2/d)^d$$

**Lemma B.4** (Lemma 12 from (Li et al., 2021)). *When the underlying bandit parameters $\theta_i^*$ and $\theta_j^*$ of two observation sequence $\mathcal{H}_{t-1,i}$ and $\mathcal{H}_{t-1,j}$ from client $i$ and $j$ are not the same, the probability that the cluster identification phase clusters them together corresponds to the type-II error probability given in Lemma B.6, which can be upper bounded by:*

$$P\left(S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c \middle| \|\theta_i^* - \theta_j^*\| > \epsilon\right) \leq F(\upsilon^c; d, \psi^d)$$

*under the condition that both $\lambda_{\min}(\sum_{(\mathbf{x}_k, y_k) \in \mathcal{H}_{t-1,i}} \mathbf{x}_k \mathbf{x}_k^\top)$ and $\lambda_{\min}(\sum_{(\mathbf{x}_k, y_k) \in \mathcal{H}_{t-1,2}} \mathbf{x}_k \mathbf{x}_k^\top)$ are at least $\frac{2\psi^d \sigma^2}{\gamma^2}$.*

**Lemma B.5** (Lemma B1 from Li and Wang (2022)). *Denote the number of observations that have been used to update $\{V_{i,t}, b_{i,t}\}$ as $\tau_i$, i.e., $V_{i,t} = \lambda I + \sum_{s=1}^{\tau_i} \mathbf{x}_s \mathbf{x}_s^\top$. Then under Assumption 3, with probability at least $1 - \delta$, we have:*

$$\lambda_{\min}(V_{i,t}) \geq \lambda + \frac{\lambda_c \tau_i}{8}$$

*$\forall \tau_i \in \{\tau_{min}, \tau_{min} + 1, \ldots, T\}, i \in [N]$, where $\tau_{min} = \lceil \frac{64}{3\lambda_c} \log(\frac{2NTd}{\delta}) \rceil$.*

**Lemma B.6.** *Lemma 3 from Li et al. (2021) When $\mathbf{X}_1$ and $\mathbf{X}_2$ are rank-sufficient, the type-II error probability can be upper bounded by,*

$$P\Big(s(\mathcal{H}_{t-1,1},\mathcal{H}_{t-1,2}) \leq \upsilon \,|\, \|\theta_1-\theta_2\| > \epsilon\Big) \leq F\Big(\upsilon^c; d, \frac{||\theta_1^*-\theta_2^*||^2/\sigma^2}{1/\lambda_{min}(\mathbf{X}_1^\top\mathbf{X}_1)+1/\lambda_{min}(\mathbf{X}_2^\top\mathbf{X}_2)}\Big).$$

# C PROOF OF THEOREM 3.1

In this section, we provide the full proof of Theorem 3.1, which states that utilizing our homogeneity test with threshold $\upsilon^c \geq F^{-1}(1-\frac{\delta}{N^2}; df, \psi^c)$, after the exploration phase of length $T_0 = \frac{16\psi^d\sigma^2}{\lambda_c\gamma^2}$, the clusters $\hat{\mathcal{C}} = \{\hat{C}_1, \hat{C}_2, ..., \hat{C}_{\hat{M}}\}$ estimated by HetoFedBandit match the ground-truth clusters of the environment $\mathcal{C} = \{C_1, C_2, ..., C_M\}$.

The homogeneity test statistic $s(\mathcal{H}_{t-1,1},\mathcal{H}_{t-1,2})$ follows a non-central $\chi^2$ distribution $s(\mathcal{H}_{t-1,1},\mathcal{H}_{t-1,2}) \sim \chi^2(df,\psi)$, where the degree of freedom

$$df = rank(\mathbf{X}_1) + rank(\mathbf{X}_2) - rank\left(\left[\mathbf{X}_1\mathbf{X}_2\right]\right)$$

and the non-centrality parameter

$$\psi = \frac{1}{\sigma^2}\left[\mathbf{X}_1\theta_1\mathbf{X}_2\theta_2^*\right]^\top \left[\mathbf{I}_{t_1+t_2} - \left[\mathbf{X}_1\mathbf{X}_2\right]\left(\mathbf{X}_1^\top\mathbf{X}_1 + \mathbf{X}_2^\top\mathbf{X}_2\right)^- \left[\mathbf{X}_1^\top \quad \mathbf{X}_2^\top\right]\right]\left[\mathbf{X}_1\theta_1\mathbf{X}_2\theta_2^*\right]$$

(Li et al., 2021).

Based on the definition and properties of the test statistic, we next prove two corollaries. First we will prove that with high probability $\mathcal{C} \subseteq \hat{\mathcal{C}}$. Then we will prove that $\hat{\mathcal{C}} \subseteq \mathcal{C}$. As a result, the conjunction of these events holding simultaneously demonstrates that $\hat{\mathcal{C}} = \mathcal{C}$, proving that our estimated clusters are correct with a high probability.

## C.1 Lower Bounding $P(\mathcal{C} \subseteq \hat{\mathcal{C}})$

Recall that based on our cluster definition presented in Assumption 1, all clients that belong to the same cluster are within $\epsilon$ of each other. We denote the ground-truth client graph $G^*$ as the graph where $\exists e(i,j) \in G^* \ \forall i,j \in N$ where $\|\theta_i^* - \theta_j^*\| \leq \epsilon$. By Assumption 2, we know that clients that do not belong to the same cluster are separated by $\gamma$, so that the ground-truth clusters $\mathcal{C}$ are the maximal cliques of $G^*$. Thus, in order to prove that $P(\mathcal{C} \subseteq \hat{\mathcal{C}})$, we need to show that the set of edges in the ground-truth client graph $G^*$ is a subset of the edges in the estimated client graph $G$. To achieve this, we need to prove an upper-bound of the type-I error probability of the homogeneity test, which corresponds to the probability that our algorithm fails to cluster two clients together when the underlying bandit parameters $\|\theta_i^* - \theta_j^*\| \leq \epsilon$.

**Lemma C.1.** *The type-I error probability of the test can be upper bounded by:*

$$P\Big(s(\mathcal{H}_{t-1,1},\mathcal{H}_{t-1,2}) > \upsilon \,|\, \|\theta_1^*-\theta_2^*\| \leq \epsilon\Big) \leq 1 - F(\upsilon; df, \psi^c),$$

*where $F(\upsilon; df, \psi^c)$ denotes the cumulative density function (CDF) of distribution $\chi^2(df,\psi^c)$ evaluated at $\upsilon$, and $\psi^c := \frac{1}{\sigma^2}$ denotes its non-centrality parameter.*

*Proof.* Denote $\zeta = \theta_2^* - \theta_1^*$. Then $\theta_2^* = \theta_1^* + \zeta$. When $\|\zeta\| \leq \epsilon$, the non-centrality parameter $\psi$ becomes:

$$\begin{aligned}
\psi &= \frac{1}{\sigma^2}\begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2(\theta_1^*+\zeta)\end{bmatrix}^\top \left[\mathbf{I}_{t_1+t_2} - \begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\left(\mathbf{X}_1^\top\mathbf{X}_1 + \mathbf{X}_2^\top\mathbf{X}_2\right)^{-1}\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\right]\begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2(\theta_1^*+\zeta)\end{bmatrix}\\
\sigma^2\psi &= \begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2\theta_1^*\end{bmatrix}^\top \left[\mathbf{I}_{t_1+t_2} - \begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\left(\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\right)^{-1}\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\right]\begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2\theta_1^*\end{bmatrix}\\
&\quad + \begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2\theta_1^*\end{bmatrix}^\top \left[\mathbf{I}_{t_1+t_2} - \begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\left(\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\right)^{-1}\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\right]\begin{bmatrix}0\\ \mathbf{X}_2\zeta\end{bmatrix}\\
&\quad + \begin{bmatrix}0\\ \mathbf{X}_2\zeta\end{bmatrix}^\top \left[\mathbf{I}_{t_1+t_2} - \begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\left(\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\right)^{-1}\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\right]\begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2\theta_1^*\end{bmatrix}\\
&\quad + \begin{bmatrix}0\\ \mathbf{X}_2\zeta\end{bmatrix}^\top \left[\mathbf{I}_{t_1+t_2} - \begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\left(\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}\right)^{-1}\begin{bmatrix}\mathbf{X}_1^\top & \mathbf{X}_2^\top\end{bmatrix}\right]\begin{bmatrix}0\\ \mathbf{X}_2\zeta\end{bmatrix}
\end{aligned}$$

Since $\begin{bmatrix}\mathbf{X}_1\theta_1^*\\ \mathbf{X}_2\theta_1^*\end{bmatrix}$ is in the column space of $\begin{bmatrix}\mathbf{X}_1\\ \mathbf{X}_2\end{bmatrix}$, the first term in the above result is zero. The second and third terms can be shown equal to zero as well using the property that matrix product is distributive with respect to matrix addition, which leaves us only the last term. Therefore, by substituting $\zeta = \theta_2^* - \theta_1^*$ back, we obtain:

$$\begin{aligned}\psi &= \frac{1}{\sigma^2}(\theta_1^* - \theta_2^*)^\top \mathbf{X}_2^\top \mathbf{X}_2(\mathbf{X}_1^\top \mathbf{X}_1 + \mathbf{X}_2^\top \mathbf{X}_2)^{-1}\mathbf{X}_1^\top \mathbf{X}_1(\theta_1^* - \theta_2^*)\\ &\leq \frac{1}{\sigma^2}\|\theta_1^* - \theta_2^*\|^2 \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2(\mathbf{X}_1^\top \mathbf{X}_1 + \mathbf{X}_2^\top \mathbf{X}_2)^{-1}\mathbf{X}_1^\top \mathbf{X}_1)\\ &\leq \frac{\epsilon^2}{\sigma^2}\lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2(\mathbf{X}_1^\top \mathbf{X}_1 + \mathbf{X}_2^\top \mathbf{X}_2)^{-1}\mathbf{X}_1^\top \mathbf{X}_1)\end{aligned}$$

The first inequality uses the Rayleigh-Ritz theorem, and the second inequality is a result of Assumption 1. Furthermore, we can use the relation $\mathbf{Y}(\mathbf{X}+\mathbf{Y})^{-1}\mathbf{X} = (\mathbf{X}^{-1} + \mathbf{Y}^{-1})^{-1}$, where $\mathbf{X}$ and $\mathbf{Y}$ are both invertible matrices, to further simplify our upper bound for $\psi$. This relation can be derived by taking inverse on both sides of the equation $\mathbf{X}^{-1}(\mathbf{X}+\mathbf{Y})\mathbf{Y}^{-1} = \mathbf{X}^{-1}\mathbf{X}\mathbf{Y}^{-1} + \mathbf{X}^{-1}\mathbf{Y}\mathbf{Y}^{-1} = \mathbf{Y}^{-1} + \mathbf{X}^{-1}$. This gives us the following,

$$\begin{aligned}\psi &= \frac{\epsilon^2}{\sigma^2}\lambda_{\max}\left(\left((\mathbf{X}_1^\top \mathbf{X}_1)^{-1} + (\mathbf{X}_2^\top \mathbf{X}_2)^{-1}\right)^{-1}\right)\\ &\leq \frac{\epsilon^2}{\sigma^2}\frac{1}{\lambda_{\min}((\mathbf{X}_1^\top \mathbf{X}_1)^{-1} + (\mathbf{X}_2^\top \mathbf{X}_2)^{-1})}\\ &\leq \frac{\epsilon^2}{\sigma^2}\frac{1}{\lambda_{\min}((\mathbf{X}_1^\top \mathbf{X}_1)^{-1}) + \lambda_{\min}((\mathbf{X}_2^\top \mathbf{X}_2)^{-1})}\\ &\leq \frac{\epsilon^2}{\sigma^2}\frac{1}{\frac{1}{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1)} + \frac{1}{\lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)}}\\ &= \frac{\epsilon^2}{\sigma^2}\frac{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1) \times \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)}{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1) + \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)}\\ &\leq \frac{\epsilon^2}{\sigma^2}\max\{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1), \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)\}\end{aligned}$$

Where the last inequality holds because

$$\frac{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1)}{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1) + \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)} \leq 1 \text{ and } \frac{\lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)}{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1) + \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)} \leq 1$$

Denote the number of observations in $\mathbf{X}_i$ as $\tau_i$. Furthermore, since $\|x_{t,i}\| \leq 1$, we know that $\lambda_{\max}(\mathbf{X}_i^\top \mathbf{X}_i) \leq \tau_i$. Thus we can further upper bound

$$\begin{aligned}\psi &\leq \frac{\epsilon^2}{\sigma^2}\max\{\lambda_{\max}(\mathbf{X}_1^\top \mathbf{X}_1), \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2)\}\\ &\leq \frac{\epsilon^2}{\sigma^2}\max\{\tau_i, \tau_j\}\\ &\leq \frac{\epsilon^2}{\sigma^2}T\end{aligned}$$

Assumption 1 tells us that $\epsilon = \frac{1}{N\sqrt{T}}$ for $(i,j)$ in the same cluster $C_k$.

$$\psi \leq \frac{T}{\sigma^2 N^2 T} \leq \frac{1}{\sigma^2} := \psi^c$$

Therefore, when $\|\theta_1^* - \theta_2^*\| < \epsilon$, the test statistic $s(\mathcal{H}_{t-1,1}, \mathcal{H}_{t-1,2}) \sim \chi^2(df, 0, \psi^c)$. The type-I error probability can be upper bounded by $P\big(s(\mathcal{H}_{t-1,1}, \mathcal{H}_{t-1,2}) > \upsilon \,\big|\, \|\theta_1^* - \theta_2^*\|\big) \leq 1 - F(\upsilon; df, \psi^c)$, which concludes the proof of Lemma C.1. □

**Corollary C.1.1.** *Under the condition that we set the threshold* $\upsilon$ *to* $\upsilon^c \geq F^{-1}(1 - \frac{\delta}{N^2}, df, \psi^c)$*, we have* $P(\mathcal{C} \subseteq \hat{\mathcal{C}}) \geq 1 - \delta$*.*

*Proof.* In our setting (Assumption 1), all users who are within $\epsilon = \frac{1}{N\sqrt{T}}$ of each other belong to the same ground-truth cluster. Our algorithm uses the pairwise homogeneity test to assess whether each pair of clients is within $\epsilon$ of each other. As we showed in Lemma C.1, the type-I error probability of our pairwise neighbor identification is upper-bounded by $1 - F(\upsilon; df, \psi^c)$. Therefore, to achieve a type-I error probability of $\delta/N^2$ between two individual clients, we can solve for the required threshold $\upsilon^c$

$$\begin{aligned} \frac{\delta}{N^2} &\leq 1 - F(\upsilon; df, \psi^c) \\ \Rightarrow F(\upsilon; df, \psi^c) &\leq 1 - \frac{\delta}{N^2} \\ \Rightarrow F^{-1}(1 - \frac{\delta}{N^2}, df, \psi^c) &\leq \upsilon^c \end{aligned}$$

Taking the union bound over all $N^2$ pairwise tests proves the that the set of edges in the ground-truth client graph $G^*$ is a subset of the edges estimated client graph $G$. Therefore the corollary is proven. □

### C.2 Lower Bounding $P(\hat{\mathcal{C}} \subseteq \mathcal{C})$

In this section, we prove that with high probability $P(\hat{\mathcal{C}} \subseteq \mathcal{C})$. To achieve this, we demonstrate that the set of edges in the estimated client graph $G$ is a subset of the ground-truth edges in $G^*$. To achieve this, we utilize the type-II error probability upper-bound to ensure that with high probability clients with different underlying parameters are not clustered together. Using this type-II error probability, we follow similar steps in Lemma 13 of (Li et al., 2021) to prove:

**Lemma C.2.** *If the cluster identification module clusters observation history $\mathcal{H}_{t-1,i}$ and $\mathcal{H}_{t-1,j}$ together, the probability that they actually have the same underlying bandit parameters is denoted as $P\big(\|\theta_i^* - \theta_j^*\| \leq \epsilon | s(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big)$.*

$$P\big(\|\theta_i^* - \theta_j^*\| \leq \epsilon | s(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big) \geq F(\upsilon^c; df, \psi^c)$$

*under the condition that both $\lambda_{\min}\big(\sum_{(\mathbf{x}_k,y_k)\in\mathcal{H}_{t-1,i}} \mathbf{x}_k\mathbf{x}_k^\top\big)$ and $\lambda_{\min}\big(\sum_{(\mathbf{x}_k,y_k)\in\mathcal{H}_{t-1,j}} \mathbf{x}_k\mathbf{x}_k^\top\big)$ are at least $\frac{2\psi^d\sigma^2}{\gamma^2}$, where $\psi^d = F^{-1}\big(\frac{(1-F(\upsilon^c;d,\psi^c))}{M-1}; d, \upsilon^c\big)$.*

*Proof.* Compared with the type-I and type-II error probabilities given in Lemma C.1 and B.4, the probability $P(\|\theta_i^* - \theta_j^*\| \leq \epsilon | S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c)$ also depends on the population being tested on.

Denote the events $\big\{\|\theta_i^* - \theta_j^*\| > \epsilon\big\} \cap \big\{S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) > \upsilon^c\big\}$ as True Positive $(TP)$, $\big\{\|\theta_i^* - \theta_j^*\| \leq \epsilon\big\} \cap \big\{S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big\}$ as True Negative $(TN)$, $\big\{\|\theta_i^* - \theta_j^*\| \leq \epsilon\big\} \cap \big\{S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) > \upsilon^c\big\}$ as False Positive $(FP)$, and $\big\{\|\theta_i^* - \theta_j^*\| > \epsilon\big\} \cap \big\{S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big\}$ as False Negative $(FN)$ of cluster identification, respectively. We can rewrite the probabilities in Lemma C.1, B.4 and C.2 as:

$$\begin{aligned} P\big(S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) > \upsilon^c | \|\theta_i^* - \theta_j^*\| \leq \epsilon\big) &= \frac{P(FP)}{P(TN + FP)} \leq 1 - F(\upsilon^c; df, \psi^c) \\ P\big(s(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c | \|\theta_i^* - \theta_j^*\| > \epsilon\big) &= \frac{P(FN)}{P(FN + TP)} \leq F(\upsilon^c; df, \psi^d) \\ P\big(\|\theta_i^* - \theta_j^*\| \leq \epsilon | s(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big) &= \frac{P(TN)}{P(TN + FN)} > \frac{1}{1 + \frac{P(FN)}{P(TN)}} \end{aligned}$$

We can upper bound $\frac{P(FN)}{P(TN)}$ by:

$$\frac{P(FN)}{P(TN)} \leq \frac{P(TP + FN)}{P(TN + FP)} \cdot \frac{F(\upsilon^c; df, \psi^d)}{F(\upsilon^c; df, \psi^c)}$$

where $\frac{TP+FN}{TN+FP}$ denotes the ratio between the number of positive instances $(\|\theta_i^* - \theta_j^*\| > \epsilon)$ and negative instances $(\|\theta_i^* - \theta_j^*\| \leq \epsilon)$ in the population. We can upper bound this ratio for any pair $(i, j)$ uniformly sampled from $[N]$, since we need to run the test on all $N^2$ pairs. First we note that $\frac{P(TP+FN)}{P(TN+FP)} = \frac{P(\|\theta_i^* - \theta_j^*\| > \epsilon)}{P(\|\theta_i^* - \theta_j^*\| \leq \epsilon)}$. We upper-bound this ratio by giving a lower bound on the probability of two randomly sampled clients belonging to the same cluster as $P(\|\theta_i^* - \theta_j^*\| \leq \epsilon)$:

$$
\begin{aligned}
P(\|\theta_i^* - \theta_j^*\| \leq \epsilon) &= \sum_{k=1}^{M} \frac{|C_k|}{N} \times \frac{|C_k| - 1}{N-1} \\
&> \sum_{k=1}^{M} \big(\frac{|C_k| - 1}{N-1}\big)^2 \\
&> \sum_{k=1}^{M} \frac{1}{M^2} \\
&= \frac{1}{M}
\end{aligned}
$$

The second inequality is true because the probability that two uniformly sampled clients belonging to the same cluster is minimized when the clusters are all of equal sizes. Therefore we have

$$
\frac{P(\|\theta_i^* - \theta_j^*\| > \epsilon)}{P(\|\theta_i^* - \theta_j^*\| \leq \epsilon)} \leq \frac{1 - \frac{1}{M}}{\frac{1}{M}} = M - 1
$$

It is worth noting that in the event that $M = 1$, the ratio can trivially be upper bounded by 1. With this upper bound of $\frac{P(FN)}{P(TN)}$, we can now write:

$$
P\big(\|\theta_i^* - \theta_j^*\| \leq \epsilon | S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big) \geq 1 / \Big(1 + (M-1) \cdot \frac{F(\upsilon^c; df, \psi^d)}{F(\upsilon^c; df, \psi^c)}\Big)
$$

Then by setting $\psi^d = F^{-1}\big(\frac{(1 - F(\upsilon^c; df, \psi^c))}{(M-1)}; df, \upsilon^c\big)$, we have:

$$
P\big(\|\theta_i^* - \theta_j^*\| \leq \epsilon | S(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big) \geq 1 / \Big(1 + (M-1) \cdot \frac{F(\upsilon^c; df, \psi^d)}{F(\upsilon^c; df, \psi^c)}\Big) = F(\upsilon^c; df, \psi^c)
$$

and the lemma is proven. □

**Corollary C.2.1.** *Under the condition that we set the threshold* $\upsilon^c \geq F^{-1}(1 - \frac{\delta}{N^2}, df, \psi^c)$*, with an exploration phase length of* $T_0 = \min\{\frac{64}{3\lambda_c} \log(\frac{2Td}{\delta}), \frac{16\psi^d \sigma^2}{\lambda_c \gamma^2}\}$*, we have* $P(\hat{C} \subseteq C) \geq 1 - \delta$*.*

*Proof.* Under Assumption 3, and with exploration length $T_0 = \min\{\frac{64}{3\lambda_c} \log(2Td/\delta), \frac{16\psi^d \sigma^2}{\lambda_c \gamma^2}\}$, the application of Lemma B.5 from (Li and Wang, 2022) gives with probability $1 - \delta$ that

$$
\lambda_{\min}(\mathbf{X}_i^\top \mathbf{X}_i) \geq \frac{\lambda_c T_0}{8} = \frac{2\psi^d \sigma^2}{\gamma^2}
$$

As a result, we can apply Lemma C.2, which gives

$$
P\big(\|\theta_i^* - \theta_j^*\| \leq \epsilon | s(\mathcal{H}_{t-1,i}, \mathcal{H}_{t-1,j}) \leq \upsilon^c\big) \geq F(\upsilon^c; df, \psi^c)
$$

Using the same steps as shown in Corollary C.1.1, we can see our choice of test statistic threshold $\upsilon^c \geq F^{-1}(1 - \frac{\delta}{N^2}; df, \psi^c)$ results in this event occurring with probability $1 - \frac{\delta}{N^2}$. Because our algorithm conducts this pairwise homogeneity test across all pairs of clients, a union bound over all $N^2$ pairwise tests proves the corollary. □

The combination of Corollaries C.1.1 and C.2.1 prove that based on our choice of $\upsilon^c$ and $T_0$, $\hat{\mathcal{C}} = \mathcal{C}$ with probability $1 - \delta$.

# D PROOF OF LEMMA 3.2

In this section, we present the complete proof of the confidence ellipsoids, following similar steps to the proof of Theorem 2 in (Abbasi-Yadkori et al., 2011).

Before we begin the proof, we will introduce a couple of useful notations to prevent clutter. Recall from Section 3.1 that the design matrix of client $i$, denoted as $\mathbf{X}_i$, only contains the observations made by client $i$ through timestep $t$ and does not include aggregated observations from other clients. In this proof, we assume without loss of generality, that client $i$ is a member of ground-truth cluster $C_k$ and is therefore collaborating with clients $j \in C_k$. As a result, we can denote $\overline{V}_{t,i}^{-1} = \lambda I + \sum_{j \in C_k} \mathbf{X}_j^\top \mathbf{X}_j$ and $b_{t,i} = \sum_{j \in C_k} \mathbf{X}_j^\top (\mathbf{X}_j \theta_j^* + \eta_j)$ due to the sharing of sufficient statistics among clients in $C_k$ (line 21 in Alg. 2), where we denote $\eta_j = (\eta_{1,j}, \eta_{2,j}, ..., \eta_{t,j})^\top$. Note that in this proof, we only focus on the case where client $i$ is collaborating with members of its ground-truth cluster, because in Theorem 3.1, we already prove with high probability $\mathcal{C} = \hat{\mathcal{C}}$. In our subsequent regret analysis in Theorem 3.3, we demonstrate that the regret incurred when $\mathcal{C} \neq \hat{\mathcal{C}}$ is upper bounded by a constant.

*Proof.*

$$
\begin{aligned}
\hat{\theta}_{t,i} &= \overline{V}_{t,i}^{-1} b_{t,i} \\
&= \overline{V}_{t,i}^{-1} \sum_{j \in C_k} \mathbf{X}_j^\top (\mathbf{X}_j \theta_j^* + \eta_j) \\
&= \overline{V}_{t,i}^{-1} \Big[ \sum_{j \in C_k} \mathbf{X}_j^\top \mathbf{X}_j \theta_j^* + \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j \Big] \\
&= \overline{V}_{t,i}^{-1} \Big[ \sum_{j \in C_k} \mathbf{X}_j^\top \mathbf{X}_j \theta_i^* + \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) + \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j \Big] \\
&= \overline{V}_{t,i}^{-1} \Big[ (\lambda I + \sum_{j \in C_k} \mathbf{X}_j^\top \mathbf{X}_j) \theta_i^* - \lambda \theta_i^* + \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) + \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j \Big] \\
&= \overline{V}_{t,i}^{-1} \overline{V}_{t,i} \theta_i^* - \lambda \overline{V}_{t,i}^{-1} \theta_i^* + \overline{V}_{t,i}^{-1} \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) + \overline{V}_{t,i}^{-1} \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j
\end{aligned}
$$

As a result, we have,

$$
\hat{\theta}_{t,i} - \theta_i^* = \overline{V}_{t,i}^{-1} \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j - \lambda \overline{V}_{t,i}^{-1} \theta_i^* + \overline{V}_{t,i}^{-1} \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*).
$$

Applying the self-normalized bound gives:

$$
\begin{aligned}
\|\theta_i^* - \hat{\theta}_{t,i}\|_{\overline{V}_{t,i}} &\leq \| \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j \|_{\overline{V}_{t,i}^{-1}} + \sqrt{\lambda} \|\theta_i^*\|_{\overline{V}_{t,i}^{-1}} + \Big\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Big\|_{\overline{V}_{t,i}^{-1}} \\
&\leq \| \sum_{j \in C_k} \mathbf{X}_j^\top \eta_j \|_{\overline{V}_{t,i}^{-1}} + \sqrt{\lambda} \|\theta_i^*\|_2 + \Big\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Big\|_{\overline{V}_{t,i}^{-1}}
\end{aligned}
$$

where we used that $\|\theta_*\|^2_{\overline{V}_{t,i}^{-1}} \leq \frac{1}{\lambda_{\min}(\overline{V}_{t,i})} \|\theta_*\|^2 \leq \frac{1}{\lambda} \|\theta_*\|^2$.

The application of Lemma B.2 and using $\|\theta_i^*\|_2 \leq 1$ give:

$$
\begin{aligned}
&\|\theta_i^* - \hat{\theta}_{t,i}\|_{\overline{V}_{t,i}} \\
&\leq \sigma \sqrt{2 \log\Big( \frac{\det(\overline{V}_{t,i})^{1/2} \det(\lambda I)^{-1/2}}{\delta} \Big)} + \sqrt{\lambda} + \Big\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Big\|_{\overline{V}_{t,i}^{-1}} := \beta_{t,i}
\end{aligned}
$$

with probability at least $1 - \delta$. Then with a union bound over all $N$ clients applied to the inequality above, we prove that $\|\theta_i^* - \hat{\theta}_{t,i}\|_{\overline{V}_{t,i}} \leq \beta_{t,i}, \forall i, t$ with probability at least $1 - N\delta$. □

# E PROOF OF THEOREM 3.3

In this section we present the full proof of our algorithm's cumulative regret and communication upper bounds. Before proving the theorem, we will need to prove the following lemmas.

**Lemma E.1** (**Heterogeneity Term Bound**). *Under the condition that the homogeneity test threshold $\upsilon^c$ is set to be greater than $F^{-1}(1-\frac{\delta}{N^2}, df, \psi^c)$, and with an exploration phase length of $T_0 = \min\{\frac{64}{3\lambda_c}\log(\frac{2Td}{\delta}), \frac{16\psi^d\sigma^2}{\lambda_c\gamma^2}\}$ we have with probability $1-\delta$:*

$$\Big\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Big\|_{\overline{V}_{t,i}^{-1}} \leq 1$$

*Proof.*

$$\begin{aligned}
\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \|_{\overline{V}_{t,i}^{-1}} &\leq \sum_{j \in C_k \setminus \{i\}} \| \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \|_{\overline{V}_{t,i}^{-1}} \\
&= \sum_{j \in C_k \setminus \{i\}} \sqrt{(\theta_j^* - \theta_i^*)^\top \mathbf{X}_j^\top \mathbf{X}_j (\lambda I + \sum_{i \in C_k} \mathbf{X}_i^\top \mathbf{X}_i)^{-1} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*)} \\
&\leq \sum_{j \in C_k \setminus \{i\}} \sqrt{(\theta_j^* - \theta_i^*)^\top \mathbf{X}_j^\top \mathbf{X}_j (\mathbf{X}_j^\top \mathbf{X}_j)^{-1} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*)} \\
&\leq \sum_{j \in C_k \setminus \{i\}} \|\theta_j^* - \theta_i^*\| \sqrt{\lambda_{\max}(\mathbf{X}_j^\top \mathbf{X}_j)} \\
&\leq \sum_{j \in C_k \setminus \{i\}} \|\theta_j^* - \theta_i^*\| \sqrt{t} \\
&\leq \sum_{j \in C_k \setminus \{i\}} \|\theta_j^* - \theta_i^*\| \sqrt{T}
\end{aligned}$$

where the first inequality is given by the triangle inequality. The second inequality holds because the sum over all clients $\overline{V}_{t,i} = \sum_{i \in C_k} \mathbf{X}_i^\top \mathbf{X}_i$ necessarily includes $\mathbf{X}_j^\top \mathbf{X}_j$, hence $\overline{V}_{t,i} \geq \mathbf{X}_j^\top \mathbf{X}_j$. Additionally, $\mathbf{X}_j^\top \mathbf{X}_j$ is positive semi-definite so that $\overline{V}_{t,i}^{-1} \leq (\mathbf{X}_j^\top \mathbf{X}_j)^{-1}$. The third inequality is given by the Rayleigh-Ritz Theorem. We have the last inequality because we know that since $\|x_{t,i}\| \leq 1$, $\lambda_{\max}(\mathbf{X}_i^\top \mathbf{X}_i) \leq \tau_i \leq T$.

Theorem 3.1 shows that by setting $\upsilon_c \geq F^{-1}(\frac{\delta}{N^2}, df, \psi^c)$ and $T_0 = \min\{\frac{64}{3\lambda_c}\log(\frac{2Td}{\delta}), \frac{16\psi^d\sigma^2}{\lambda_c\gamma^2}\}$, with probability $1-\delta$ we have $\hat{\mathcal{C}} = \mathcal{C}$. Therefore, since $i, j$ belong to the same ground-truth cluster $C_k$, we have by Assumption 1, $\|\theta_j^* - \theta_i^*\| \leq \epsilon_t = \frac{1}{N\sqrt{t}}$. As a result, we can further upper bound the heterogeneity term by

$$\Big\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Big\|_{\overline{V}_{t,i}^{-1}} \leq \sum_{j \in C_k \setminus \{i\}} \frac{\sqrt{T}}{N\sqrt{T}} \leq 1$$

□

**Lemma E.2.** *We define the single step pseudo regret $r_{t,i} = \langle \theta_i^*, x_{t,i}^* - x_{t,i} \rangle$ where $x_{t,i}^* = \arg\max_{x \in \mathcal{A}_{t,i}} \langle x, \theta_{t,i}^* \rangle$. With probability $1 - N\delta$, $r_{t,i}$ is bounded by*

$$r_{t,i} \leq 2\bigg(\sigma\sqrt{2\log\bigg(\frac{\det(\overline{V}_{t,i})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\bigg)} + \sqrt{\lambda}S + O(1)\bigg)\|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} = O\bigg(\sigma\sqrt{d\log\frac{T}{\delta}}\bigg)\|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} \qquad (6)$$

*Proof.* Assume without loss of generality $\theta_i^* \in C_k$,

$$\begin{aligned}
r_{t,i} &= \langle \theta_i^*, x_{t,i}^* \rangle - \langle \theta_i^*, x_{t,i} \rangle \\
&\leq \langle \tilde{\theta}_{t,i}, x_{t,i} \rangle - \langle \theta_i^*, x_{t,i} \rangle \\
&= \langle \tilde{\theta}_{t,i} - \theta_i^*, x_{t,i} \rangle \\
&= \langle \tilde{\theta}_{t,i} - \hat{\theta}_{t,i}, x_{t,i} \rangle + \langle \hat{\theta}_{t,i} - \theta_i^*, x_{t,i} \rangle \\
&\leq \|\tilde{\theta}_{t,i} - \hat{\theta}_{t,i}\|_{\overline{V}_{t,i}} \|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} + \|\hat{\theta}_{t,i} - \theta_i^*\|_{\overline{V}_{t,i}} \|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} \\
&\leq 2\Bigg(\sigma\sqrt{2\log\Bigg(\frac{\det(\overline{V}_{t,i})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\Bigg)} + \sqrt{\lambda}S + \Bigg\| \sum_{j \in C_k \setminus \{i\}} \mathbf{X}_j^\top \mathbf{X}_j (\theta_j^* - \theta_i^*) \Bigg\|_{\overline{V}_{t,i}^{-1}}\Bigg) \|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} \qquad (7)
\end{aligned}$$

The first inequality is because $\langle \tilde{\theta}_{t,i}, x_{t,i} \rangle$ is optimistic. Applying Lemma E.1 to upper bound the heterogeneity term gives

$$\begin{aligned}
\text{RHS of Eq.(7)} &\leq 2\Bigg(\sigma\sqrt{2\log\Bigg(\frac{\det(\overline{V}_{t,i})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\Bigg)} + \sqrt{\lambda}S + O(1)\Bigg)\|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} \\
&= O\Bigg(\sigma\sqrt{d\log\frac{T}{\delta}}\Bigg)\|x_{t,i}\|_{\overline{V}_{t,i}^{-1}}
\end{aligned}$$

□

Now we are equipped to prove Theorem 3.3.

*Proof.* The cumulative regret of our system can be decomposed into three components. The first component is the regret accumulated under our exploration stage. During these timesteps we can trivially upper bound the instantaneous regret by 2. The second component considers the regret during timesteps in which our estimated clusters are correct. The third component considers the regret accumulated during the timesteps in which our estimated clusters are incorrect, which we can also upper bound the instantaneous regret by 2.

$$R_T \leq \sum_{t=0}^{T_0}\sum_{i=1}^{N} 2 + \sum_{t=T_0+1}^{T}\sum_{k=1}^{\hat{M}}\sum_{i \in \hat{C}_k} r_{t,i} \cdot \mathbb{1}\{\hat{\mathcal{C}} = \mathcal{C}\} + \sum_{t=T_0+1}^{T}\sum_{k=1}^{\hat{M}}\sum_{i \in \hat{C}_k} 2 \cdot \mathbb{1}\{\hat{\mathcal{C}} \neq \mathcal{C}\}$$

Note that because our cluster estimation is non-parametric the number of estimated clusters $\hat{M}$ is not a hyper-parameter to our clustering algorithm.

According to Theorem 3.1, if we select $\upsilon^c \geq F^{-1}(1 - \frac{\delta}{N^2}, df, \psi^c)$ and $T_0 = \frac{16\psi^d\sigma^2}{\lambda_c\gamma^2}$, the probability that $\hat{\mathcal{C}} = \mathcal{C}$ is $1 - \delta$. Therefore, by setting $\delta = \frac{1}{N^2T}$, the regret contributed by the rightmost term is of the order $O(1)$. As a result, our high-probability regret bound is given by:

$$R_T \leq \frac{32N\psi^d\sigma^2}{\lambda_c\gamma^2} + \sum_{t=T_0+1}^{T}\sum_{k=1}^{\hat{M}}\sum_{i \in \hat{C}_k} r_{t,i} \cdot \mathbb{1}\{\hat{\mathcal{C}} = \mathcal{C}\} + O(1)$$

In the subsequent steps, we will focus on the regret accumulated when $\hat{\mathcal{C}} = \mathcal{C}$. This means we only need to examine the instances when the estimated number of clusters and their compositions exactly match the actual clusters. Consequently, in subsequent discussions $\hat{M} = M$ and $\hat{C}_k = C_k$ for all $k \in [M]$.

Now we prove Theorem 3.3 following the steps in the proof of Theorem 4 in Wang et al. (2020). We consider the case where Eq.(6) holds because with the same choice of $\delta = \frac{1}{N^2T}$, the expected instantaneous regret resulting during timesteps when Eq.(6) does not hold is $O(1)$.

In our communication protocol, for each cluster $C_k$, there will be a number of epochs separated by communication rounds. We denote $|C_k|$ denotes the number of clients in cluster $C_k$. If there are $P_k$ epochs within cluster $C_k$, then $V_{P_k}$ will be the matrix with all samples from $C_k$ included. Similarly we denote the last globally shared $V$ to the clients in $C_k$ in epoch $p$ as $V_p$.

From Lemma B.3, we have $\det(V_0) = \lambda^d$. $\det(V_{P,k}) \leq \left(\frac{tr(V_p)}{d}\right)^d \leq \left(\lambda + \frac{|C_k|\,T}{d}\right)^d$. Therefore by the pigeonhole principle:

$$\log \frac{\det(V_p)}{\det(V_0)} \leq d \log\left(1 + \frac{|C_k|\,T}{\lambda d}\right)$$

It follows that for all but $R := d\log\left(1 + \frac{|C_k|T}{\lambda d}\right)$ epochs:

$$1 \leq \frac{\det(V_j)}{\det(V_{j-1})} \leq 2 \tag{8}$$

In these "good epochs" where Eq (8) is satisfied, we can follow Theorem 4 from Wang et al. (2020) and treat all of the $|C_k|T$ observations from cluster $k$ as observations from an imaginary single agent in a round-robin manner. We similarly use $\tilde{V}_{t,i} = \lambda I + \sum_{\{(p,q):(p<t)\vee(p=t\wedge q<i)\}} x_{p,q}x_{p,q}^\top$ to denote the $\overline{V}_{t,i}$ this agent calculates before seeing $x_{t,i}$. If $x_{t,i}$ is in a good epoch, then:

$$1 \leq \frac{\det(\tilde{V}_{t,i})}{\det(\overline{V}_{t,i})} \leq \frac{\det(V_j)}{\det(V_{j-1})} \leq 2$$

We similarly denote $\mathcal{B}_{p,k}$ as the set of $(t,i)$ pairs that belong to epoch $p$ and $P_{good,k}$ as the set of good epochs in cluster $k$. In that event we can use the regret bound for a single agent:

$$\begin{aligned}
R_{good} &= \sum_{t=T_0+1}^{T} \sum_{k=1}^{M} \sum_{i\in C_k} r_{t,i} \\
&\leq \sum_{k=1}^{M} \sqrt{|C_k|T \sum_{p\in P_{good,k}} \sum_{(t,i)\in\mathcal{B}_{p,k}} r_{t,i}^2} \\
&\leq \sum_{k=1}^{M} O\left(\sqrt{d|C_k|T\log(T/\delta) \sum_{p\in P_{good,k}} \sum_{(t,i)\in\mathcal{B}_{p,k}} \min(\|x_{t,i}\|^2_{\tilde{V}_{t,i}^{-1}}, 1)}\right) \\
&\leq \sum_{k=1}^{M} O\left(\sqrt{d|C_k|T\log(T/\delta) \sum_{p\in P_{good,k}} \log\left(\frac{\det(V_p)}{\det(V_{p-1})}\right)}\right) \\
&\leq \sum_{k=1}^{M} O\left(\sqrt{d|C_k|T\log(T/\delta)\log\left(\frac{\det(V_p)}{\det(V_0)}\right)}\right) \\
&\leq \sum_{k=1}^{M} O\left(d\sqrt{|C_k|\,T}\log(|C_k|\,T)\right)
\end{aligned}$$

Now we must analyze the regret caused by the bad epochs, of which there are $R = O(d\log(|C_k|\,T))$ within each cluster $C_k \in \mathcal{C}$. This part of the analysis differs from the proof in Theorem 4 of (Wang et al., 2020) due to the fact that in our protocol, clusters that have requested collaboration may have to wait in the queue until they are served in the event that multiple clusters have requested collaboration at the same timestep.

Consider the regret for a particular cluster $C_k \in \mathcal{C}$ during this bad epoch. Suppose that the bad epoch starts at time $t_0$ and lasts $n$ timesteps. We denote the time $t_q$ when the cluster $k$ is added to the queue awaiting collaboration. We can decompose the regret of this cluster during the bad epoch into two parts, corresponding to the timesteps before and after $C_k$ has been added to the queue:

$$REG_{bad}(k) = \sum_{t=t_0}^{t_q-1} r_{t,i} + \sum_{t=t_q}^{n} r_{t,i}$$

Based on our algorithm design, we can see in line 13 of Algorithm 2 that a cluster is only added to the queue when at least one client in that cluster has exceeded its communication threshold $D_k$. Therefore we know that before $t_q$, we can upper bound the regret of the cluster $k$ from $t_0$ to $t_q - 1$ as:

$$\begin{aligned}\sum_{t=t_0}^{t_q-1} r_{t,i} &\leq O\bigg(\sqrt{d \log \frac{T}{\delta}}\bigg) \sum_{i \in C_k} \sum_{t=t_0}^{t_q-1} \|x_{t,i}\|_{\overline{V}_{t,i}^{-1}} \\ &\leq O\bigg(\sqrt{d \log \frac{T}{\delta}}\bigg) \sum_{i \in C_k} \sqrt{(t_q - 1 - t_0) \log \frac{\det(V_{t_q-1,i})}{\det(V_{t_q-1,i} - \Delta V_{t_q-1,i})}} \\ &\leq O\bigg(\sqrt{d \log \frac{T}{\delta}}\bigg) |C_k| \sqrt{D_k}\end{aligned}$$

Once cluster $k$ is added to the queue at timestep $t_q$, it may have to wait to be served by the central server based on how many clusters have requested collaboration before it. Recall that our queue is a FIFO queue (line 19 in Algorithm 2), and we have $M$ total clusters. Therefore the maximum time cluster $C_k$ could have to wait in the queue is $M$ timesteps. Each timestep the cluster is waiting in the queue, a client in this cluster will miss $|C_k|$ observations. For each of these missed observations, we can upper bound the regret incurred by 2, giving

$$\sum_{t=t_q}^{n} r_{t,i} \leq 2(M+1)|C_k|^2$$

Combining our results, we have the following bound on the regret of cluster $C_k$ during a bad epoch:

$$REG_{bad}(k) \leq O\bigg(\sqrt{d \log \frac{T}{\delta}}\bigg) |C_k| \sqrt{D_k} + 2(M+1)|C_k|^2$$

As we know we have at most $R = O(d \log(|C_k|\, T))$ bad epochs, we can further bound it by

$$REG_{bad}(k) \leq O\bigg(\sqrt{D_k} |C_k| d^{1.5} \log^{1.5}(|C_k|T) + 2d|C_k|^2(M+1) \log(|C_k|\, T)\bigg)$$

with the choice of $D_k = \frac{T \log |C_k| T}{d|C_k|}$, our regret becomes:

$$REG_{bad}(k) \leq O\bigg(d\sqrt{|C_k|T} \log^2(|C_k|T) + 2d|C_k|^2 M \log(|C_k|\, T)\bigg)$$

The summation over all $M$ clusters gives the regret for all clusters in all of the bad epochs:

$$REG_{bad} \leq \sum_{k=1}^{M} O\bigg(d\sqrt{|C_k|T} \log^2(|C_k|T) + 2d|C_k|^2 M \log(|C_k|\, T)\bigg)$$

Combining the regret from the exploration phase, good epochs, and bad epochs gives a final cumulative regret upper bound of:

$$\begin{aligned}R_T \leq \frac{32 N \psi^d \sigma^2}{\lambda_c \gamma^2} &+ \sum_{k=1}^{M} O\bigg(d\sqrt{|C_k|\, T} \log(|C_k|\, T)\bigg) \\ &+ \sum_{k=1}^{M} O\bigg(d\sqrt{|C_k|T} \log^2(|C_k|T) + 2d|C_k|^2 M \log(|C_k|\, T)\bigg) + O(1)\end{aligned}$$

This can be further simplified into the following,

$$R_T \leq O\bigg(\frac{N\psi^d\sigma^2}{\lambda_c\gamma^2} + \sum_{k=1}^{M} d\sqrt{|C_k|T}\log^2(|C_k|T) + 2d|C_k|^2 M\log(|C_k|\,T)\bigg)$$

### E.1 Communication cost

The cumulative communication cost $C_T$ of our algorithm can be divided into two parts. The first is the communication cost associated with the pure exploration and cluster estimation phase. During the pure exploration phase, no clients communicate with the central server, so that the communication cost associated with that phase is trivially 0. At the end of the exploration phase, all $i \in [N]$ clients share with server their sufficient statistics $V_{T_0,i}$ and $b_{T_0,i}$, each of which are $d \times d$ and $d \times 1$ respectively. Therefore, the communication cost of the cluster estimation is $C_{cluster\_est} = N(d^2 + d) = O(Nd^2)$.

Next, we characterize the communication cost of the second phase, the federated clustered bandit phase. In our communication protocol, for each cluster $C_k$, there will be a number of epochs separated by communication rounds. Denote the length of an epoch as $\alpha$, so that there can be at most $\lceil \frac{T}{\alpha} \rceil$ epochs with length longer than $\alpha$. For an epoch with less than $\alpha$ time steps, similarly, we denote the first time step of this epoch as $t_s$ and the last as $t_e$, i.e., $t_e - t_s < \alpha$. Therefore, $\log \frac{\det(V_{t_e})}{\det(V_{t_s})} > \frac{D_k}{\alpha}$. Following the same argument as in the regret proof, the number of epochs with less than $\alpha$ time steps is at most $\lceil \frac{R\alpha}{D_k} \rceil$. Then $C_{fed\_cluster}(k) = |C_k| \cdot (\lceil \frac{T}{\alpha} \rceil + \lceil \frac{R\alpha}{D_k} \rceil)$, because at the end of each epoch, the synchronization round incurs $2|C_k|$ communication cost. We minimize $C_{fed\_cluster}(k)$ by choosing $\alpha = \sqrt{\frac{D_k T}{R}}$, so that $C_{fed\_cluster}(k) = O(|C_k| \cdot \sqrt{\frac{TR}{D}})$. With our choice of $D_k = \frac{T\log|C_k|T}{d|C_k|}$, we have

$$\begin{aligned} C_{fed\_cluster}(k) &= O(|C_k| \cdot \sqrt{\frac{TR}{\frac{T\log|C_k|T}{d|C_k|}}}) \\ &= O(|C_k| \cdot d\sqrt{|C_k|}) \end{aligned}$$

Combining our communication cost from our two phases together gives:

$$C_T = O(Nd^2) + \sum_{k=1}^{M} O(d^3|C_k|^{1.5})$$

□

## F EMPIRICALLY ENHANCED ALGORITHM

In this section, we present the details of our proposed empirical enhancements to our HetoFedBandit algorithm, where we perform re-clustering to improve the quality of estimated clusters of clients and replace the first-in-first-out queue with a priority queue to help clusters where a shared model update can most rapidly reduce regret for clients in that cluster. We describe the enhancements in 3.4, but present the complete enhanced algorithm in Algorithm 4 and 5.

## G ADDITIONAL EMPIRICAL ENHANCEMENT EVALUATION

In this work, we demonstrate the effectiveness of our empirical enhancements on two synthetic datasets. In Section 4.2, we analyzed the performance of both HetoFedBandit and HetoFedBandit-E on a balanced synthetic dataset that was generated following the procedure described in Section 4.2. In this section, we evaluate our models on an imbalanced synthetic dataset to emphasize the distinct contributions of our priority queue and data-dependent re-clustering enhancements.

**Dataset** In this imbalanced dataset, we deliberately vary the distribution of clients and the sizes of clusters. We establish $N = 50$ clients and $M = 13$ ground-truth clusters. Instead of randomly assigning clients to clusters like we did in Section 4.2, we manually assigned 26 clients to cluster $C_1$, and the remaining 24 clients were assigned in pairs to the remaining 12 cluster centers. After being assigned to a cluster center, we follow the same procedure from Section 4.2 to generate the client parameters within $\epsilon$ of the cluster centers. For the other environment settings, we used $d = 25$, $K = 1000$, $\gamma = 0.85$ and $T = 2500$.

**Algorithm 4** Data-Dependent Clustering

```
Re-initialize client graph $\mathcal{G}$ with no edges
for $(i, j) \in N$ do
    Server Computes $\upsilon^c = F^{-1}(1 - \frac{\delta}{N^2}, df, \psi^c)$ where
    $\psi^c = \frac{\epsilon^2}{\sigma^2} \lambda_{\max}(\mathbf{X}_2^\top \mathbf{X}_2 (\mathbf{X}_1^\top \mathbf{X}_1 + \mathbf{X}_2^\top \mathbf{X}_2)^{-1} \mathbf{X}_1^\top \mathbf{X}_1)$
    if $s(\mathcal{H}_{t,i}, \mathcal{H}_{t,j}) \leq \upsilon^c$ then
        Add edge $e(i, j)$ to $\mathcal{G}$
    end if
end for
$\hat{\mathcal{C}} = \{\hat{C}_1, \hat{C}_2, ...\hat{C}_{\hat{M}}\} = \text{maximal\_cliques}(\mathcal{G})$
Set $\mathcal{K}_i = \{k : i \in \hat{C}_k\}$ for each client $i$
Cluster communication thresholds $\mathcal{D} = [D_1, ..., D_{\hat{M}}]$ where $D_k = (T \log |\hat{C}_k| T)/(d|\hat{C}_k|)$
```

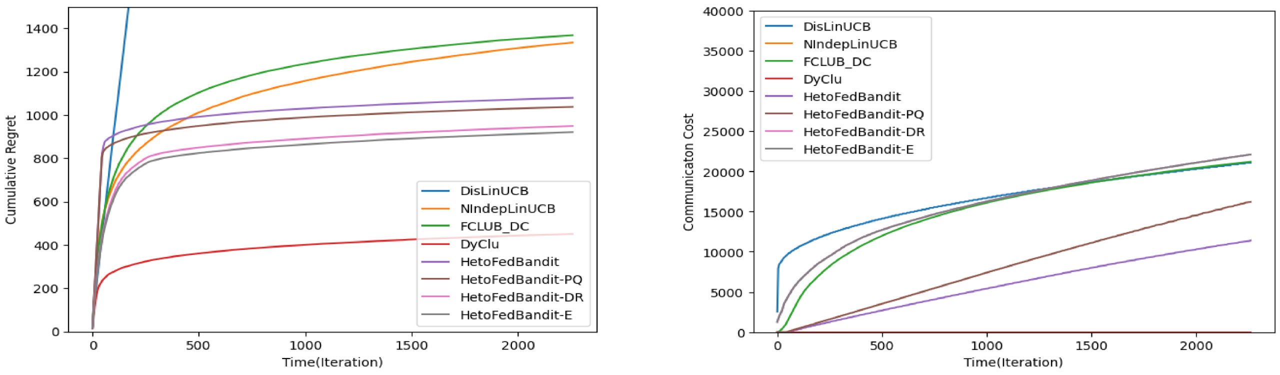
**Algorithm 5** HetoFedBandit-E

```
Input: $T$, $\delta \in (0, 1)$, regularization parameter $\lambda > 0$
Initialize Clients: $\forall i \in N$: $V_{0,i} = \mathbf{0}_{d\times d}, b_{0,i} = \mathbf{0}_d, \mathcal{H}_{0,i} = \emptyset, \Delta V_{0,i} = \mathbf{0}_{d\times d}, \Delta b_{0,i} = \mathbf{0}_d, \Delta t_{i,0} = 0, \mathcal{K}_i = \emptyset$
Initialize Server: Client graph $\mathcal{G}$ with $N$ nodes;
Initialize empty Priority Queue $Q$;
for $t = T_0 + 1, ..., T$ do
    for Client $i \in N$ do
        $\overline{V}_{t-1,i} = V_{t-1,i} + \lambda I$, $\hat{\theta}_{t-1,i} = \overline{V}_{t-1,i}^{-1} b_{t-1,i}$
        Choose arm $x_{t,i} \in \mathcal{A}_{t,i}$ by Equation 2 observe reward $y_{t,i}$
        Update agent $i$: $\mathcal{H}_{t,i} = \mathcal{H}_{t-1,i} \cup (x_{t,i}, y_{t,i})$, $V_{t,i}$ += $x_{t,i} x_{t,i}^\top$, $b_{t,i}$ += $x_{t,i} y_{t,i}$,
        $\Delta V_{t,i}$ += $x_{t,i} x_{t,i}^\top$, $\Delta b_{t,i}$ += $x_{t,i} y_{t,i}, \Delta t_{t,i}$ += $1$
        if $\Delta t_{t,i} \log(\det(V_{t,i}) / \det(V_{t,i} - \Delta V_{t,i})) \geq D_k$ then
            Empty Priority Queue $Q$;
            Every client $i \in N$ sends $\Delta V_{t,j}$ and $\Delta b_{t,j}$ to server
            Data Dependent Cluster Estimation (Algorithm 4)
            Send collaboration request to server, which then adds $\hat{C}_k \forall k \in \mathcal{K}_i$ to $Q$
        end if
    end for
    if Q is non-empty then
        Server pops cluster $\hat{C}_k = \arg\max_{\hat{C}_k \in \hat{\mathcal{C}}} \sum_{i \in \hat{C}_k} \Delta t_{t,i} \log(\frac{\det(V_{t,i})}{\det(V_{t,i} - \Delta V_{t,i})})$ from Q
        Server Computes: $V_{t,sync} = \sum_{j \in \hat{C}_k} \Delta V_{t,j}$, $b_{t,sync} = \sum_{j \in \hat{C}_k} \Delta b_{t,j}$
        Each client in $\hat{C}_k$ receives $V_{t,sync}$ and $b_{t,sync}$ from the server and updates their local model
        $V_{t,i}$ += $V_{t,sync} - \Delta V_{t,i}$, $b_{t,i}$ += $b_{t,sync} - \Delta b_{t,i}$, $\Delta V_{t,i} = 0$, $\Delta b_{t,i} = 0, \Delta t_{t,i} = 0$
    end if
end for
```

**Models** In order to evaluate the contributions of each enhancement proposed in Section 3.4, we implemented two additional enhanced algorithms of HetoFedBandit. In HetoFedBandit-PQ, we replace the server's FIFO queue with a priority queue that selects a cluster to collaborate with based on their determinant ratios. HetoFedBandit-DR performs data-dependent clustering at each collaboration round. HetoFedBandit-E, as described in Algorithm 5, is our fully enhanced algorithm, where both a priority queue and data-dependent clustering are employed.

**Results** In Figure 4a, we conducted an empirical evaluation of the individual enhancements proposed in Section 3.4. A comparison between HetoFedBandit-DR and HetoFedBandit demonstrates that the use of data-dependent clustering significantly improved performance on our imbalanced synthetic dataset. By employing a data-dependent clustering threshold, our algorithm facilitated greater collaboration among clients with similar observation histories during the early rounds. Although this enhancement incurred additional communication cost, the cost remained sub-linear and comparable to that of DisLinUCB.

Comparing HetoFedBandit with HetoFedBandit-PQ, our observations suggest that the utilization of a priority



(a) Accumulative Regret

(b) Communication Cost

Figure 4: Experimental Results on Imbalanced Synthetic Dataset

queue yielded modest improvements in cumulative regret, particularly in the initial rounds when multiple clients simultaneously requested collaboration, leading to queue congestion. In this imbalanced environment, we observed significant delays for the larger cluster $C_1$ when using a FIFO queue. The larger size of cluster $C_1$ resulted in a higher value of the cluster determinant ratio, indicating its potential for greater regret reduction in the federated learning system. However, due to the FIFO queue, several smaller clusters that that benefited less from collaboration were served ahead of $C_1$. Nevertheless, as the algorithm progressed, the frequency of communication among clients decreased, resulting in reduced queue congestion. As a result, HetoFedBandit and HetoFedBandit-PQ exhibited similar performance in the later rounds.

## H SENSITIVITY ANALYSIS

According to our regret analysis, the performance of HetoFedBandit depends on three key environment parameters: the number of ground-truth clusters $M$, and the number of clients $N$, and the cluster separation parameter $\gamma$. In this experiment, we analyze their influence on HetoFedBandit and baselines by varying these parameters while keeping the others fixed. The accumulated regret under different settings are reported in Table 1. As suggested by our theoretical analysis, a larger client to cluster ratio $\frac{N}{M}$ leads to higher regret of both HetoFedBandit (HFB) and HetoFedBandit-E (HFB-E) as shown in setting 1, 2 and 3, since observations are split into more clusters with smaller size each. Lastly, as shown in settings 4 and 5, 6, decreasing the environment separation introduces a higher regret of HetoFedBandit since a longer exploration period is required to discern which clients are safe for collaboration. Additionally, the decreased cluster separation in setting 5 leads to an increase in regret for HetoFedBandit-E, as well as DyClu due to the increased likelihood of a clustering error when the clusters are closer together.

Table 1: Comparison of accumulated regret under different environment settings.

| | $N$ | $M$ | $\gamma$ | $T$ | NIndep-LinUCB | DisLinUCB | FCLUB_DC | DyClu | HFB | HFB-E |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 30 | 1 | 0.85 | 3000 | 772.03 | 59.20 | 648.22 | 48.77 | 576.31 | 173.89 |
| 2 | 30 | 4 | 0.85 | 3000 | 784.80 | 23776.10 | 784.01 | 227.18 | 669.17 | 443.89 |
| 3 | 30 | 30 | 0.85 | 3000 | 781.35 | 25124.46 | 791.19 | 776.86 | 883.51 | 822.24 |
| 4 | 30 | 4 | 0.65 | 3000 | 777.73 | 20129.05 | 788.12 | 231.57 | 699.89 | 461.54 |
| 5 | 30 | 4 | 0.05 | 3000 | 787.79 | 23823.61 | 771.04 | 269.45 | 916.73 | 582.21 |