
Federated Multi-armed Bandits with Efficient Bit-Level Communications

Haoran Zhang

Nanjing University
602023710004@smail.nju.edu.cn

Yang Xu

Nanjing University
yang.xu@smail.nju.edu.cn

Xuchuang Wang

University of Massachusetts Amherst
xuchuangw@gmail.com

Hao-Xu Chen

Nanjing University
2718522637@qq.com

Hao Qiu

Università degli Studi di Milano
hao.qiu@unimi.it

Lin Yang*

Nanjing University
linyang@nju.edu.cn

Yang Gao

Nanjing University
gaoy@nju.edu.cn

Abstract

In this work, we study the federated multi-armed bandit (FMAB) problem, where a set of agents collaboratively aim to minimize cumulative regret. Unlike traditional centralized bandit models, agents in FMAB settings are connected via a communication graph and cannot share data freely due to bandwidth limitations or privacy constraints. This raises a fundamental challenge: how to achieve optimal learning performance under stringent communication budgets. We propose a novel communication-efficient algorithm containing two points: one for eliminating suboptimal arms through early and frequent communication of key decisions, and the other for refining global estimates using incremental epoch, quantized, and differentially transmitted statistics. *Incremental Epoch-based Successive Elimination Algorithm* (EpoInc-SE) is presented by carefully balancing communication frequency and precision of global estimates. Theoretically, we derive tight upper bounds on both individual cumulative regret and group regret, and prove that our method asymptotically matches the lower bound of regret in federated settings. Experimental results on synthetic data validate the effectiveness of EpoInc-SE in various settings and under heterogeneous feedback.

1 Introduction

The stochastic multi-armed bandit (MAB) problem is a classic and influential research topic in the field of sequential decision-making [Bandits, Lattimore and Szepesvári, 2020]. In this model, a decision-maker must select an arm from a set of actions at each time step to pull and receive a random reward. The core objective is to maximize cumulative rewards, or equivalently, minimize cumulative regret, which is defined as the reward lost compared to always selecting the optimal arm. Due to its powerful modeling capabilities and wide applicability, the MAB framework has been successfully applied in numerous domains, such as clinical trials [Wang, 1991], online advertising recommendations [Tang et al., 2015] and resource allocation [Lattimore et al., 2015].

In recent years, with the development of distributed systems and increasing awareness of data privacy, traditional single-agent MAB models have gradually extended to multi-agent collaborative scenarios,

*Corresponding author

Table 1: Comparison of existing FMAB algorithms with our proposed method EpoInc-SE.

Algorithm	Central Server	Hete-/Homo-	Individual Regret	Group Regret	Bit-Com
Gossip_UCB [Zhu et al., 2021]	✗	Hete-	$O(\sum_{i:\Delta_i>0} N \Delta_i^{-1} \log T)$	$O(\sum_{i:\Delta_i>0} N^2 \Delta_i^{-1} \log T)$	$O(T \log T)$
Dis_UCB [Zhu and Liu, 2023]	✗	Hete-	$O(\sum_{i:\Delta_i>0} N_{\min}^{-1} \Delta_i^{-1} \log T)^1$	$O(\sum_{i:\Delta_i>0} N N_{\min}^{-1} \Delta_i^{-1} \log T)$	$O(T \log T)$
UCB-TCOM [Wang and Yang, 2023]	✗	Homo-	$O(\sum_{i:\Delta_i>0} N^{-1} \Delta_i^{-1} \log T)$	$O(\sum_{i:\Delta_i>0} \Delta_i^{-1} \log T)$	$\tilde{O}(\log T)$
Fed2-UCB [Shi and Shen, 2021]	✓	Homo-	$O(\sum_{i:\Delta_i>0} N^{-1} \Delta_i^{-1} \log T)$	$O(\sum_{i:\Delta_i>0} \Delta_i^{-1} \log T)$	$O(T \log T)$
EpoInc-SE (our work)	✗	Hete-	$O(\sum_{i:\Delta_i>0} N^{-1} \Delta_i^{-1} \log T)$	$O(\sum_{i:\Delta_i>0} \Delta_i^{-1} \log T)$	$O(\sqrt{\log T})$
Regret lower bound	✗	Hete-	$\Omega(\sum_{i:\Delta_i>0} N^{-1} \Delta_i^{-1} \log T)$	$\Omega(\sum_{i:\Delta_i>0} \Delta_i^{-1} \log T)$	—

¹ $N_{\min}$ denotes the smallest number of neighbors for any agent, including the agent itself.

where the Federated Multi-Armed Bandit (FMAB) problem has garnered increasing attention [Li et al., 2024, Boursier and Perchet, 2024]. In large-scale sensor networks, environmental constraints and cost considerations often prevent full physical connectivity or the deployment of a central server, naturally leading to the need for distributed information processing and computation [Xiao et al., 2005, Golovin et al., 2010]. Moreover, online service platforms with many servers often store large volumes of data locally. To protect user privacy, centralizing this data for computation and training is impractical, where federated learning is a useful and crucial approach to complete the two tasks Ellison and Fudenberg [1995], Sankararaman et al. [2019].

In FMAB problems, the primary objective is to minimize the total regret accumulated across all agents, referred to as *group regret* in this paper. This metric has been widely studied in prior works [Kalathil et al., 2014, Shi and Shen, 2021, Xu and Klabjan, 2024]. However, in many practical scenarios, the performance of individual agents also plays a critical role, as it can become the bottleneck that limits overall system effectiveness. In network optimization literature [Srikant and Ying, 2014], the max-min fairness metric—maximize the minimal individual reward—is widely used to measure a system’s fairness. Another important metric in FMAB problems is *communication bit* incurred by all agents. In real-world scenarios, communication is often expensive and establishing peer-to-peer (P2P) connectivity may be impractical. To evaluate this cost, the number of *communication rounds* is a commonly used metric [Cho et al., 2020, Li and Song, 2022]. However, a single round may involve transmitting multiple messages, which is misleading for evaluating an algorithm. Therefore, a more precise and informative measure is the total number of *communication bits*, which has been adopted in recent studies [Wang et al., 2020, Agarwal et al., 2022].

Related Works. The most relevant prior works on FMAB have been summarized in Table 1. Among them, Gossip_UCB [Zhu et al., 2021] and Dis_UCB [Zhu and Liu, 2023] are the two works most related to ours, which are all heterogeneous FMAB without a central server. However, they could not obtain the optimal results compared with EpoInc-SE (our work). Fed2-UCB [Shi and Shen, 2021] and UCB-TCOM [Wang and Yang, 2023] are two typical Homogeneous FMAB algorithms that use a central server and fully distributed communication, respectively. Meanwhile, they all achieve the optimal result and UCB-TCOM [Wang and Yang, 2023] also optimizes the communication rounds. The two algorithms provide classic frameworks for bandit learning. Hence, we apply a consensus estimator on them and take them as the baseline in Section 5. Several works have investigated the regret lower bound in federated bandit settings [Xu and Klabjan, 2023, Wang et al., 2020]. Specifically, they establish group regret lower bounds for heterogeneous and homogeneous FMAB problems, respectively. In addition, Xu and Klabjan [2023] shows that, in the absence of information exchange with neighbors, an agent in a federated setting can suffer linear regret $O(T)$. For bit communication techniques, there are not too many works on heterogeneous FMAB. But there are also some interesting works about bit communication in other directions [McMahan et al., 2017, Casteigts et al., 2019, Wang et al., 2019, Boursier and Perchet, 2019, Shi et al., 2021]. Wang et al. [2019] and Shi et al. [2021] reduced the communication rounds of the algorithm under a homogeneous setting. These works [McMahan et al., 2017, Boursier and Perchet, 2019] considered communication bits for the first time in bandit problems. The truncation of messages is proposed in Casteigts et al. [2019], Shi et al. [2021]. Our communication optimization method combines the advantages of works above.

Contribution. Addressing the aforementioned challenges, this paper proposes a novel federated multi-armed bandit algorithm designed for fully distributed communication networks. It enables

agents to effectively solve the global MAB problem and simultaneously optimize individual and group regrets by communicating solely with their neighbors. Distinct from previous work, our main contributions and innovations include:

(1) Optimization of Communication Efficiency: EpoInc-SE is the first work considering the communication cost in heterogeneous federated bandit learning. By employing epoch-based exploration to reduce communication rounds and designing an adaptive difference communication mechanism, we significantly reduce the communication rounds and the number of effective information bits per communication round. Additionally, we introduce a two-phase communication strategy to differentiate the transmission priority of different types of information, thereby enhancing communication efficiency while preserving regret performance.

(2) Near-Optimal Regret Performance: We propose the first lower bound of heterogeneous federated bandit $\Omega(\sum_i^N \Delta_i^{-1} N^{-1} \log T)$ with the dependence of the number of agent N , which matches the upper bound of EpoInc-SE in terms of agent number N , reward gap Δ_i and time horizon T , thereby proving the optimality of our algorithm design and analysis.

(3) Efficient Online Consensus Estimator: We design and introduce an epoch-based consensus estimation subroutine EBCES that allows each agent to unbiasedly estimate the global mean of arms under the condition of fully distributed communication and buffering broadcast. This is crucial for resolving heterogeneity and communication optimization.

The remainder of this paper is organized as follows: Section 2 details the problem model, including the communication model, the federated multi-armed bandit model, and performance metrics. Section 3 introduces our proposed communication-efficient algorithm design, covering its core ideas, epoch-based exploration strategy, arm elimination mechanism, adaptive difference communication scheme, and consensus estimation policy. Section 4 provides the theoretical analysis of the algorithm, including proofs for regret upper and lower bounds. Section 5 provides simulations to validate the theoretical analysis. Finally, Section 6 concludes the paper.

2 Problem Formulation

Communication Model. Consider a fully distributed network, which works only via finite paths among agents, i.e., without a central server. The data exchanged between agents, called a message, is a string of binary numbers. The network is a (fully) connected graph, defined as $\mathcal{G} = (\mathcal{N}, \mathcal{E}, \mathcal{A})$ [Huang et al., 2022]. $\mathcal{N}$ is an agent set, where agents are labeled by $\{1, \dots, N\}$. The $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ denotes the set of edges in the network. $\mathcal{A} = [a_{a,b}]_{(a,b) \in \mathcal{N} \times \mathcal{N}}$ is an adjacency matrix, used to determine whether two agents are connected. For agent j , define its neighborhood as $\mathcal{N}_j$. To meet computing needs, an additional matrix $\mathbf{W} = [\omega_{a,b}]_{(a,b) \in \mathcal{N} \times \mathcal{N}}$ describes the weight of agents in their neighborhoods. The definition of $\mathbf{W}$ is $\mathbf{W} = \mathbf{I} - \beta \mathcal{L}$, where $\mathcal{L}$ is the Laplacian matrix and $\beta \in (0, 1/N]$ is the coefficient that reflects whether this agent believes its neighbors. If $\beta = 0$, we have $\mathbf{W} = \mathbf{I}$, which means that the agent itself occupies all the weight. If $\beta = 1/N$, it indicates that the agent assigns equal trust to all of its neighbors.

System Model. There are N agents repeatedly making decisions from a fixed arm set $\mathcal{K} = \{1, \dots, K\}$ over a time horizon T . At each time step t , agent j selects an arm $A_j(t)$ and receives a reward $X_{i,j}(t)$ if $A_j(t) = i$. The reward $X_{i,j}(t)$ is drawn from a $\frac{1}{2}$ -sub-Gaussian distribution with an unknown (local) mean $\mu_{i,j} \in [0, 1]$. These rewards serve as real-time feedback to guide agents in learning the expected rewards of the arms. In federated settings, the reward $X_{i,j}$ is a local observation that agent j samples from arm i . For arm $A_j(t) = i$, there also exists a global reward $X_i(t)$ that is contributed by all agents' observations, i.e., $X_{A_j(t)}(t) = X_i(t) := \sum_{j=1}^N X_{i,j}(t)$. The mean of X_i is denoted by the global mean $\mu_i := \frac{1}{N} \sum_{j=1}^N \mu_{i,j} \in [0, 1]$. Without loss of generality, let $i^* := \arg \max_i \mu_i$ denote the unique optimal arm with the highest global mean among all arms. Define the suboptimality gap for each arm i as $\Delta_i := \mu_{i^*} - \mu_i$. In this work, we consider a heterogeneous setting where the local reward mean of arm i may differ across agents, i.e., $\mu_{i,j_1} \neq \mu_{i,j_2}$, for $j_1 \neq j_2$. There is no collision in this setting, implying that all agents can obtain rewards without degradation, even if they pull the same arm.

Individual regret. The individual regret of agent j is defined as the cumulative regret incurred if all agents are to follow agent j 's decision. This metric is crucial in federated learning, where the

Symbol/Term	Definition
$\mathcal{K}$	The arm set
$\mathcal{N}$	The agent set
$A_j(t)$	The decision of agent j at time slot t , $A_j(t) \in \mathcal{K}$
$X_{i,j}(t)$	The local reward of arm i sampled by agent j at time slot t , $X_{i,j}(t) \in [0, 1]$
$\mu_{i,j}$	The local mean of arm i for agent j , $\mu_{i,j} \in [0, 1]$
$X_i(t)$	The global reward of arm i , $X_i(t) = \sum_{j=1}^N X_{i,j}(t)$
μ_i	The global mean of arm i , $\mu_i = \sum_{j=1}^N \mu_{i,j}$
i^*	The optimal arm with the largest global mean, $i^* \in \arg \max_i \mu_i$
Δ_i	The gap between arm i and the optimal arm, $\Delta_i = \mu_{i^*} - \mu_i$

Table 2: Symbols of the multi-agent bandit problem

objective is to identify the arm with the highest global mean reward. Since relying solely on local rewards can be misleading, individual regret provides a more meaningful measure of an agent's contribution to the global goal. The formal definition is given as follows,

$$\mathbb{E}[R_j(T)] := T\mu_{i^*} - \sum_{t=1}^T \mathbb{E}[X_{A_j(t)}(t)]. \quad (1)$$

Group regret. The group regret is defined as the cumulative regret caused by all agents, which is also seen as a main performance metric of the algorithm. The optimal performance of an algorithm is that all agents pull the optimal arm during the time horizon T . Hence, for a distributed algorithm, the expected group regret of the entire system is defined as

$$\mathbb{E}[R(T)] := \sum_{j=1}^N \mathbb{E}[R_j(T)]. \quad (2)$$

Communication cost. Besides two regrets, the communication cost is also an important performance metric to evaluate a federated bandit algorithm. Two types of communication metrics are commonly used in previous works: communication rounds and communication bits. While the communication costs are primarily measured by the number of communication rounds, e.g., Tao et al. [2019], Yang et al. [2024], the number of communication bits is a more practical and realistic metric that accurately reflects the real-world cost of data transmission. In this paper, we define the communication cost in bit level, denoted $C(T)$ as the total number of bits consumed from communicating all messages among all agents. In this work, we assume that any message (a number of bits) can be broadcast within a single time slot. For any messages, we do not consider the identifiers, and there is no limitation on the message size.

3 EpoInc-SE: A Communication-Efficient Algorithm Design

3.1 Key Ideas and Algorithm Structure

To achieve optimal regret bounds under limited communication, the core challenge lies in ensuring that the exchanged information is just sufficient to support accurate consensus estimation. Due to the heterogeneity in rewards, these local estimates often deviate from the global mean, making communication with neighbors necessary. This creates a fundamental trade-off: sharing highly precise and real-time information provides limited improvement to the convergence of consensus estimation but results in excessive communication overhead. In Boursier and Perchet [2019], Wang et al. [2019], information quantization and epoch-based exploration techniques were proposed to control the size of the message adaptively. However, these approaches are typically developed for homogeneous bandits with centralized structures, such as having a leader agent directly connected to all followers, and thus do not apply to settings with heterogeneity and fully distributed graphs.

In this work, we extend the classical successive elimination algorithm to a fully distributed setting, achieving an improved regret bound of $O(\sum_{i:\Delta_i>0} N^{-1} \Delta_i^{-1} \log T)$. Building on this, we further optimize the communication cost at the bit level, which comprises two key components: the number

of communication rounds (optimized in Section 3.2) and the message size (optimized in Section 3.3). The first component leverages a batched sampling strategy to reduce communication rounds. In distributed bandit settings, each agent must sample arms sufficiently to ensure accurate local estimates. Rather than communicating after every pull, agents accumulate multiple samples before sharing with neighbors. This reduces communication frequency and ensures that shared information is more reliable, particularly as early-stage observations tend to be noisy. Epoch-based exploration thus offers a principled trade-off between statistical efficiency and communication cost, making the approach scalable in large, decentralized systems. The second component for optimizing communication complexity reduces the bit-width of the transmitted messages. Three kinds of statistics need to be optimized, including the indices of arms, the sample count of each arm, and the global estimate. The indexes of arms consume only $O(\log K)$ bits, which has no room for optimization. The sample count is implicit information considering the algorithm's synchronization. Hence, we improve the message size of global estimates from $O(\log T)$ to $O(1)$ bits. To realize this task, we propose a differential encoding approach where agents share only the quantized change between consecutive estimates instead of the full values.

To further reduce communication costs while maintaining regret performance, we categorize shared information into two types based on their impact on the algorithm's efficiency. The first type includes indexes of eliminated arms, which are broadcast immediately to accelerate arm elimination and improve regret. The second type of shared information includes global estimates of arm means, which are tightly coupled with the sampling process. Sharing them too frequently does not directly contribute to faster convergence, but can significantly increase communication overhead. Therefore, this type of information should be buffered and communicated less frequently. Based on this insight, our algorithm divides communication into two distinct phases. **Communication Phase I** is responsible for sharing the indexes of eliminated arms. This phase operates at a high frequency to ensure the timely elimination of suboptimal arms across agents. In contrast, **Communication Phase II** handles the exchange of global estimates. The frequency of this phase is intentionally kept low to limit communication costs. This design is especially effective when updates are frequent but minor, avoiding redundant communication.

Below, we introduce the details of the algorithm design. We start from the epoch-based exploration paradigm in Section 3.2, and then introduce three key modules of the algorithm design: an adaptive difference communication in Section 3.3, a consensus estimation policy in Section 3.4 and an arm elimination policy in Section 3.5.

3.2 The Epoch-based Exploration

In this section, we primarily introduce the epoch-based exploration strategy. To reduce the number of communication rounds overhead in federated learning, a buffering scheme that compresses communication rounds by delaying broadcasts is proposed. The main challenge is that excessive compression may cause newly collected local data to dominate the consensus estimation, which affects the contribution of agents' local observations in the global estimate (introduced in Section 3.4). Therefore, we carefully make a trade-off between minimizing communication rounds and maintaining the accuracy of the consensus estimation.

Next, we present an incremental epoch-based exploration where later epochs contain increasingly more time slots. Without loss of generality, we concentrate the discussion on a specific agent j . The time horizon T is partitioned into a series of subprocesses, labeled by $r_j = 1, 2, \dots$. In epoch r_j , agents maintain a candidate set $\mathcal{S}_j^{r_j}$ to decide which arm to pull. The initial set is $\mathcal{S}_j^0 = \mathcal{K}$. In epoch r_j , all arms in the candidate set $\mathcal{S}_j^{r_j}$ will be sampled $p^{r_j} = a(r_j + 1)$ times uniformly, where a is a hyper-parameter that determines the size of each epoch. Therefore, the number of epochs is at most $O(\sqrt{T})$ for the time horizon T . The time slot at the beginning of the epoch r_j is denoted by t_{r_j} , and the local empirical estimate on the arm i during epoch r_j is defined as follows

$$\bar{X}_{i,j}^r := \frac{\sum_{t=t_{r_j}}^{t_{r_j+1}-1} X_{i,j}(t) \mathbb{I}\{A_j(t) = i\}}{a(r_j + 1)}. \quad (3)$$

Algorithm 1: Epoch-based Successive Elimination Algorithm (EpoInc-SE)(for agent j)

Input: Weight matrix of graph $\mathbf{W}$, time horizon T , arm set $\mathcal{K}$, diameter D and integer a

Output: $t = 0, r_j = 0, U_{i,j}^r = 1, \mathcal{S}_j = \mathcal{K}, \bar{X}_{i,j} = 0, \mathcal{B}_j = \emptyset$ for all arms i

```
1 Let agent  $j$  pull each arm  $a$  times and receive a sequence of rewards  $\{X_{i,j}(t)\}_{t=1}^{t=a}$ ;  
2  $\tilde{\mu}_{i,j}^r \leftarrow \text{average}(X_{i,j}), \tilde{\mu}_{i,j}^r \leftarrow \bar{X}_{i,j}^r$ ;  
3 for  $r_j = 1, 2, \dots$  do  
4    $i^{\max} \leftarrow \arg \max_{i \in \mathcal{S}_j} \tilde{\mu}_{i,j}^r$ ;  
5   for  $p^{r_j} \in \{1, \dots, a(r_j + 1)\}$  do  
6     for  $i \in \mathcal{S}_j$  do  
7       Pull arm  $i$  and obtain the reward  $X_{i,j}$ ;  
8        $\bar{X}_{i,j} \leftarrow (\bar{X}_{i,j}(p^{r_j} - 1) + X_{i,j})/p^{r_j}$ ;  
9       // Communication phase I  
10      Receive  $\mathcal{B}_{j'}$  and  $\hat{r}_i$  from agent  $j$ 's neighborhood  $\mathcal{N}_j$ ;  
11      if  $\mathcal{B}_j \geq 1$  then  
12        Send  $\mathcal{B}_j, \hat{r}_i$  to all agent  $j$ 's neighbors;  
13      Update the new candidate set  $\mathcal{B}_j \leftarrow \mathcal{B}_j \cup \left( \bigcup_{j' \in \mathcal{N}_j} \mathcal{B}_{j'} \right)$ ;  
14      // Communication phase II  
15      Update the global estimates via the adaptive difference communication (Algorithm 2);  
16      Update  $U_{i,j}^r$  via equation (6);  
17      for  $i \in \mathcal{S}_j$  do  
18        if  $\tilde{\mu}_{i,j}^{r-1} \leq \tilde{\mu}_{i^{\max},j}^{r-1} - 2U_{i,j}^{r-1}$  then  
19           $\hat{r}_i \leftarrow r_j + \left\lceil \frac{|S_j|D}{a(r_j+1)} \right\rceil, \mathcal{B}_j \leftarrow \mathcal{B}_j \cup \{i\}$ ;  
20      for each arm  $i$  in  $\mathcal{B}_j$  whose  $r_j \geq \hat{r}_i$  do  
21        if  $|S_j| > 1$  then  $\mathcal{S}_j \leftarrow \mathcal{S}_j \setminus \{i\}, \mathcal{B}_j \leftarrow \mathcal{B}_j \setminus \{i\}$ ; else  $\mathcal{S}_j \leftarrow \mathcal{S}_j$ 
```

3.3 Adaptive Difference Communication

In this section, we optimize the communication cost by minimizing the message size. As outlined in Section 3.1, global estimates occupy the main channel resources and they consume $O(\log T)$ bits per message. Hence, we concentrate on optimizing the global estimate $\tilde{\mu}_{i,j}^r$ (introduced in Section 3.4).

A native intuition is to send brief information instead of precise global estimates. In the early stages of consensus estimation, high precision is unnecessary—initial rounds only require estimates accurate to a single decimal place. This motivates the use of information quantization, where the quantized global estimate at epoch r_j is denoted as $\bar{\mu}_{i,j}^r$. The quantization is adaptive: larger messages yield higher accuracy. Specifically, the estimate in epoch r_j is represented using $\lceil 1 + \log_2 r_j \rceil$ bits.

To further reduce the message size, another approach is based on sharing differential information rather than full local estimates. Instead of transmitting quantized statistics at each round, agents communicate only the differences between their current and previous values, i.e., quantization error $\delta_{i,j}^r := \bar{\mu}_{i,j}^r - \bar{\mu}_{i,j}^{r-1}$. This technique leverages the temporal smoothness of learning processes, significantly reducing the communication overhead while maintaining estimation accuracy.

Due to that $\bar{\mu}_{i,j}^r$ has $\lceil 1 + \log_2 r_j \rceil$ bits and $\bar{\mu}_{i,j}^r$ is a quantization version of $\tilde{\mu}_{i,j}^r$, the error between them is bounded by

$$|\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r| \leq \frac{1}{2^{\lceil 1 + \log_2 r_j \rceil}} \leq \frac{1}{2r_j} \stackrel{r_j \geq 2}{\leq} \frac{1}{r_j + 2} \quad (4)$$

Remark 1. Wang et al. [2019] has proved that $O(\log T)$ bits are sufficient to distinguish the optimal arm from the arm set $\mathcal{K}$, considering that the gap Δ_i may be small enough. The global estimates contribute a large number of bits to the communication cost. By exploiting the quantized variable $\bar{\mu}_{i,j}^r$, we reduce the message size to $O(\sqrt{\log T})$. By using differential communication, we can only

Algorithm 2: Epoch-based consensus estimation subroutine (EBCES) (for agent j)

Input: The local reward $X_{i,j}$, the candidate set $\mathcal{S}_{j'}$, the epoch r_j and the weight matrix

$$W = [\omega_{j,j'}]_{N \times N}$$

Output: The latest estimate $\tilde{\mu}_{i,j}$ and elimination arm set $\mathcal{B}_j$

```
1 if  $|\mathcal{S}_j| > 1$  then
2   | Communicate the global error  $\delta_{i,j}^{r-1}$  with its neighbors ;           // Communication phase II
3   | Receive the global estimate  $\delta_{i,j'}^{r-1}, j' \in \mathcal{N}_j$  from agent  $j$ 's neighbors;
4 for  $j' \in \mathcal{N}_j$  do
5   |  $\bar{\mu}_{i,j'}^{r-1} \leftarrow \bar{\mu}_{i,j'}^{r-2} + \delta_{i,j'}^{r-1}$ ;
6 Update the global estimate  $\mu_{i,j}^r$  via equation (5);
7  $\bar{\mu}_{i,j}^r \leftarrow \text{ceil}(\tilde{\mu}_{i,j}^r)$ ;
8  $\delta_{i,j}^r \leftarrow \bar{\mu}_{i,j}^r - \bar{\mu}_{i,j}^{r-1}$ ;
```

use $O(1)$ bits per message. It is also the first work that obtains optimal message size in heterogeneous federated bandit learning.

3.4 Consensus Estimation Policy

The heterogeneous reward in this problem motivates agents to learn the global means of arms and cooperate with their neighborhoods. Therefore, an epoch-based consensus estimation subroutine (EBCES), exploiting epoch-based exploration (Section 3.2) and differential information quantization (Section 3.3), is proposed to be integrated into EpoInc-SE.

The core idea of EBCES is to synchronize information exchange among neighbors and effectively utilize their shared data to achieve an unbiased global estimate. As discussed in Section 3.3, agents exchange $\delta_{i,j}^r$ instead of $\tilde{\mu}_{i,j}^r$ and reduce the information bits to $O(1)$ order. In the consensus estimation process, agents combine differential information $\delta_{i,j}^r$ with the historical quantized variable $\bar{\mu}_{i,j}^{r-1}$ to compute the latest global estimates from neighbors (Line 5, Algorithm 2). On the basis of the information exchange above, a fair estimation mechanism is presented in which samples from all agents are used equally to estimate the global mean.

To obtain global estimates, agent j combines its neighbors' historical data and its own average empirical reward $\bar{X}_{i,j}^r$. Up to the end of epoch r_j , agent j collects $\bar{\mu}_{i,j'}^{r-1}, j' \in \mathcal{N}_j$ and updates its global estimate as follows:

$$\tilde{\mu}_{i,j}^r = (1 - \sigma_i(r_j)) \sum_{j' \in \mathcal{N}_j} \omega_{j,j'} \bar{\mu}_{i,j'}^{r-1} + \sigma_i(r_j) \bar{X}_{i,j}^r, \quad (5)$$

where $\sigma_i(\cdot)$ is the weight that adjusts the contribution of the information in the global estimate $\tilde{\mu}_{i,j}^r$.

3.5 Arm Elimination Policy

In this section, we introduce an elimination arm set $\mathcal{B}_j^r$ that stores the indexes of arms identified as suboptimal. At the beginning of each epoch, each agent compares the global estimates of all arms and selects the one with the highest estimated reward as the baseline (Line 4, Algorithm 1). Then, each arm will be compared with arm $i^{\max}$, which is considered sub-optimal and added to the elimination set $\mathcal{B}_j$ if it falls below the baseline by a margin that is related to the confidence radius.

Denote the sample count of agent j on arm i until epoch r_j by $\tau_{i,j}^r = a(r_j + 1)(r_j + 2)/2$. The total sample count on arm i is $\tau_i^r = N\tau_{i,j}^r$ because of synchronous sampling. Define $\tilde{\mu}_{i,j}^r$ as the global estimate of agent j on arm i . According to Hoeffding's inequality, there exists a radius of confidence interval $U_{i,j}^r$ such that $\tilde{\mu}_{i,j}^r \in [\mu_i - U_{i,j}^r, \mu_i + U_{i,j}^r]$, with

$$U_{i,j}^r(\delta) := \underbrace{\sqrt{\frac{\log \delta^{-1}}{aN(r_j + 1)(r_j + 2)}}}_{(a)} + \underbrace{\frac{2C}{(r_j + 2)(1 - \lambda_2)}}_{(b)} + \underbrace{\frac{1}{r_j + 2}}_{(c)}, \quad (6)$$

where δ specifies the violation probability. Terms (a) and (b) are from Lemmas 2 and 3. Term (c) is caused by information quantization (introduced in Section 3.3). The detailed proof of the confidence interval (6) is proposed in Appendix D.1. Arm i is identified as sub-optimal if it satisfies

$$\tilde{\mu}_{i,j}^r + U_{i,j}^r \leq \tilde{\mu}_{i_{\max},j}^r - U_{i,j}^r. \quad (7)$$

To help agent j determine whether the information in $\mathcal{B}_j^r$ has been acknowledged by all agents, we introduce epoch label $\hat{r}_i$, which denotes the latest epoch by which all agents are guaranteed to know that arm i is suboptimal. Then, all agents agree to eliminate it from the candidate set at epoch $\hat{r}_i$,

$$\hat{r}_i \leftarrow r_j + \left\lceil \frac{D}{a(r_j + 1)} \right\rceil, \quad (8)$$

where D is the diameter of the graph $\mathcal{G}$. The elimination arm set $\mathcal{B}_j^r$ is constructed as

$$\mathcal{B}_j^r = \{i, i \in \mathcal{S}_j^r : \exists i' \in \mathcal{S}_j^r \setminus \{i\} \text{ such that } \tilde{\mu}_{i,j}^r \leq \tilde{\mu}_{i',j}^r - 2U_{i,j}^r\}. \quad (9)$$

The candidate set $\mathcal{S}_j$ is updated based on the information in $\mathcal{B}_j^r$ until only one arm remains, which is identified as the optimal arm. Once an arm is eliminated from $\mathcal{S}_j$, it is also removed from $\mathcal{B}_j$ to avoid redundant communication (Line 19, Algorithm 1).

4 Analysis

In this section, we summarize the theoretical results concerning both regret and communication cost. We begin by introducing Lemma 1, which provides a concentration bound for the global mean. Unlike traditional concentration results based on real-time data, our approach leverages batched information to perform consensus estimation, which obtains a similar performance. This key insight allows us to establish upper bounds on both communication rounds and regret, which are detailed in Section 4.1. Finally, Section 4.2 presents the corresponding lower bounds.

Lemma 1. *Let $i \in \mathcal{K}$ be any arm, $j \in \mathcal{N}$ be any agent, and $r \in \{1, 2, \dots\}$ be any epoch. Assume reward $X_{i,j}(t)$ is an i.i.d. process with unknown mean $\mu_{i,j}$. Set $\sigma_i(r_j) = \frac{2}{r_j+2}$. Then, with probability at least $1 - 2\delta$, the gap between the global estimate $\tilde{\mu}_{i,j}^r$ and the global mean μ_i satisfies*

$$|\tilde{\mu}_{i,j}^r - \mu_i| \leq \sqrt{\frac{\log \delta^{-1}}{aN(r_j + 1)(r_j + 2)}} + \frac{2C}{(1 - \lambda_2)(r_j + 2)} + \frac{1}{r_j + 2}, \quad (10)$$

where λ_2 is the second largest eigenvalue of the weight matrix $\mathbf{W}$ (characterizing network connectivity), $\sigma_i(r_j)$ is the variance proxy of local means across agents, and C is a positive constant determined by $\mathcal{G}$, i.e., $C = 1$ when $\mathcal{G}$ is balanced; otherwise, $C = \sqrt{N}$.

Proof Sketch of Lemma 1. We decompose the estimation error into three parts: quantization error $|\tilde{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r|$, consensus error $|\tilde{\mu}_{i,j}^r - \hat{\mu}_i^r|$, and concentration error $|\hat{\mu}_i^r - \mu_i|$. The quantization error arises from limited communication precision and is controlled by the quantization scheme. The consensus error results from averaging over a network with imperfect information propagation and is bounded using spectral properties of the communication matrix. Finally, the concentration error comes from estimating the true mean using sub-Gaussian samples and is handled via standard probabilistic bounds. Combining these yields a confidence interval for the global estimate.

4.1 Upper Bound

Theorem 1 (Upper regret bound). *Let $U_{i,j}^r$ in (6) with $\delta = T^{-2}$ be the radius of the confidence interval of a random $[0, 1]$ -valued i.i.d. process. Given any constant $\gamma > 0$ and $\sigma_i(r_j) = 2/(r_i + 2)$, Algorithm 1 could achieve the following results,*

(i) *For each agent j , individual regret:*

$$\mathbb{E}[R_j(T)] \leq O\left(\sum_{i:\Delta_i > 0} N^{-1} \Delta_i^{-1} \log T\right),$$

(ii) *Group regret:*

$$\mathbb{E}[R(T)] \leq O\left(\sum_{i:\Delta_i > 0} \Delta_i^{-1} \log T\right),$$

Proof Sketch of Theorem 1. To bound both types of regret, we first establish an upper bound on the sample counts of sub-optimal arms. From equation (7), we derive an instance-dependent bound, then account for the additional samples required to synchronize agents under the communication delay. This adjustment depends on the graph’s diameter and the theoretical result in equation (7). Finally, combining the gap terms with these sample counts yields the overall regret bounds. Detailed proofs are provided in Appendix D.2.

Theorem 2 (Upper communication cost). *When all agents execute EpoInc-SE (Algorithm 1), the total communication cost over the entire time horizon is at most $O(\sum_{i:\Delta_i>0} \sqrt{N}\Delta_i^{-1}\sqrt{\log T})$ bits, in order to achieve the regret bounds stated in Theorem 1.*

Proof Sketch of Theorem 2. To optimize communication cost, the key idea is to minimize both the size of global estimates and the number of communication rounds. The message size is reduced to $O(1)$ through differential communication, while the epoch-based exploration policy decreases the communication frequency to the order of $O(\sqrt{\log T})$. By combining these two strategies, we derive the upper bound of the overall communication cost.

Remark 2. *The lower bound of the communication cost is still an important problem for heterogeneous bandit problems. The previous work Wang et al. [2019] provides a constant level lower bound for the homogeneous bandit. However, the heterogeneous bandit setting poses a more challenging property, i.e., heterogeneous feedback, which implies that more essential communication is needed for obtaining a near-optimal result. It is an interesting work that is worth learning.*

Remark 3. *In the worst case, the message in the communication phase I will consume at most $O(NK)$ bits. In the communication phase II, the message will consume at most $K \log K + K \log \log T$ bits.*

4.2 Lower regret bound

Besides the upper bounds of regrets, we also present lower bounds for this problem. We investigate the lower bounds of both individual and group regrets, where the elements in the upper bounds match those in the lower bounds.

Theorem 3 (Regret lower bound). *For FMAB problems with any number of agents, arms, and stochastic rewards satisfying a 1-Gaussian distribution, if the graph $\mathcal{G}$ is connected, any federated bandit algorithm must incur regrets at least:*

(i) *For each agent j , individual regret:*

$$\liminf_{T \rightarrow \infty} \frac{\mathbb{E}[R_j(T)]}{\log T} \geq \sum_{i:\Delta_i>0} \frac{2}{N\Delta_i}.$$

(ii) *Group regret:*

$$\liminf_{T \rightarrow \infty} \frac{\mathbb{E}[R(T)]}{\log T} \geq \sum_{i:\Delta_i>0} \frac{2}{\Delta_i}.$$

Proof Sketch of Theorem 3. To derive the lower bound for federated heterogeneous bandit problems, the main challenge lies in handling the randomness of decision-making across multiple agents. To address this, we introduce an auxiliary “replica” subsystem that allows agent j to access other agents’ observations on the same arm. This additional information does not tighten the lower bound. We then show that it does not affect the consensus estimation, after which the regret lower bound follows directly from the proof techniques in [Lattimore and Szepesvári, 2020].

Remark 4. *The lower bound of the heterogeneous federated bandit problem is also discussed in Theorem 3 in Xu and Klabjan [2023]. Different from the previous work, we prove the lower bound by using a relaxed model. In this model, each agent obtains additional information (Details are given in Section 3), and the result is obtained directly by using Theorem 1 in Garivier et al. [2019].*

5 Experiments

In this section, we conduct a series of experiments on different algorithms. All experiments are repeated for 50 trials, with means plotted as lines and standard deviations as shaded regions.

Setups and Baselines. Unless otherwise stated, the experiment scenario involves a network of $N = 8$ agents and $K = 10$ arms, parameters $C = \sqrt{N}$, $a = 5$ and $T = 10^6$. To ensure a fair comparison, we use a ring graph and all agents occupy the same weight in their neighborhoods, which is a common undirected connected graph with the second largest eigenvalue $\lambda_2 = 0.5713$. We consider four baselines: Gossip_UCB [Zhu et al., 2021], Dis-UCB [Zhu and Liu, 2023], Fed2-UCB [Shi and Shen, 2021] and UCB-TCOM [Wang and Yang, 2023]. The theoretical results of them are discussed in Table 1. The individual regret is evaluated by the maximum individual regret among all agents. For Fed2-UCB and UCB-TCOM, a consensus estimation module is attached to them to help them against heterogeneity. The two algorithms communicate the historical reward at each time slot. Hence, the consensus estimator is $\hat{\mu}_i(t) = \frac{1}{Nt} \sum_{j=1}^N \sum_{k=1}^t X_{i,j}(k)$.

Observations. From Figures 1a and 1b, both group and individual regrets of EpoInc-SE are only slightly worse than that of Fed2-UCB, primarily because Fed2-UCB leverages a central server, which enables faster convergence compared to a fully distributed graph. Figure 1c shows that EpoInc-SE results in the lowest bit-level communication overhead among all compared methods, highlighting the efficiency of our algorithm in minimizing communication costs. Figures 1d-1f indicate that EpoInc-SE still has good performance when N, K, δ vary. We also make some experiments under homogeneous settings in Section E and some experiments in large-scale multi-agent systems in Section F.

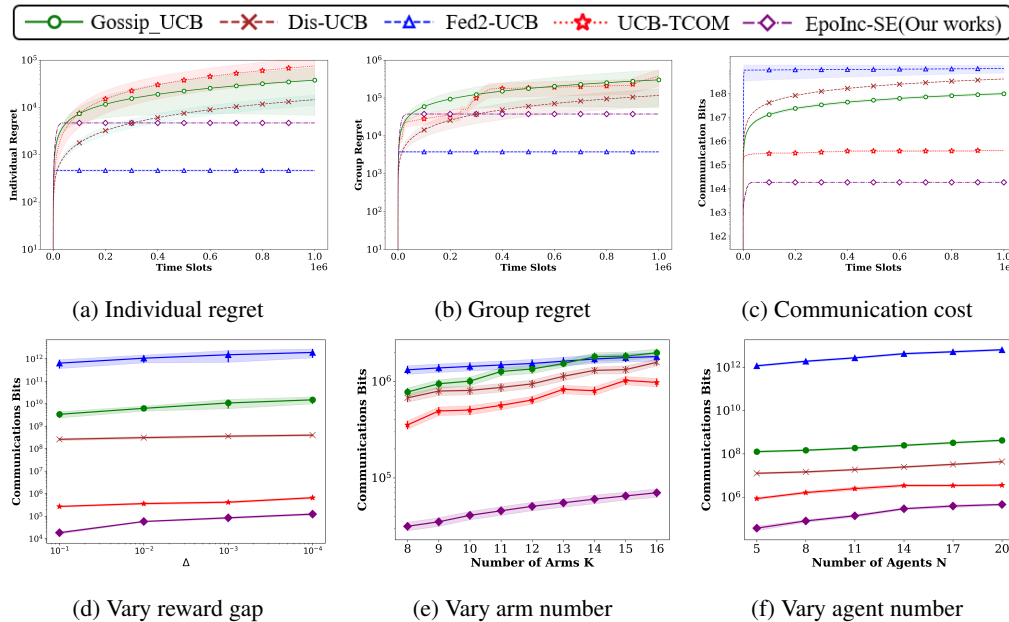


Figure 1: Performance comparison with different algorithms.

6 Conclusion

This paper proposed a communication-saving solution to address the challenges faced by FMAB problems in environments with heterogeneous rewards and fully distributed communication networks. We successfully tackled the shortcomings of existing research in areas such as information fusion, regret performance, and communication efficiency, particularly in scenarios where there is no central coordinator and agents can only communicate with their neighbors. The main contribution of this paper lies in the design and introduction of an efficient federated learning algorithm EpoInc-SE, which greatly reduces the communication cost and obtains $O(1)$ bits. Besides, we first propose a lower regret bound in this setting and prove that EpoInc-SE is near-optimal under limited communication. Future work could explore applying this framework to more complex network dynamics or considering more advanced privacy-preserving mechanisms.

7 Acknowledgment

This work was supported by NSFC (no. 62306138), JiangsuNSF (no. BK20230784), and the Innovation Program of State Key Laboratory for Novel Software Technology at Nanjing University (no. ZZKT2025B25). HQ acknowledges the financial support from the EU Horizon CL4-2022-HUMAN-02 research and innovation action under grant agreement 101120237, project ELIAS (European Lighthouse of AI for Sustainability) and from the One Health Action Hub, University Task Force for the resilience of territorial ecosystems, funded by Università degli Studi di Milano (PSR 2021-GSA-Linea 6).

References

- Mridul Agarwal, Vaneet Aggarwal, and Kamyar Azizzadenesheli. Multi-agent multi-armed bandits with limited communication. *Journal of Machine Learning Research*, 23(212):1–24, 2022.
- Multi-Armed Bandits. Introduction to multi-armed bandits.
- Etienne Boursier and Vianney Perchet. Sic-mmab: Synchronisation involves communication in multiplayer multi-armed bandits. *Advances in Neural Information Processing Systems*, 32, 2019.
- Etienne Boursier and Vianney Perchet. A survey on multi-player bandits. *Journal of Machine Learning Research*, 25(137):1–45, 2024.
- Arnaud Casteigts, Yves Métivier, John Michael Robson, and Akka Zemmari. Deterministic leader election takes $\theta(d + \log n)$ bit rounds. *Algorithmica*, 81:1901–1920, 2019.
- Yae Jee Cho, Samarth Gupta, Gauri Joshi, and Osman Yağan. Bandit-based communication-efficient client selection strategies for federated learning. In *2020 54th Asilomar Conference on Signals, Systems, and Computers*, pages 1066–1069. IEEE, 2020.
- Devdatt P Dubhashi and Alessandro Panconesi. *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press, 2009.
- Glenn Ellison and Drew Fudenberg. Word-of-mouth communication and social learning. *The Quarterly Journal of Economics*, 110(1):93–125, 1995.
- Aurélien Garivier, Pierre Ménard, and Gilles Stoltz. Explore first, exploit next: The true shape of regret in bandit problems. *Mathematics of Operations Research*, 44(2):377–399, 2019.
- Daniel Golovin, Matthew Faulkner, and Andreas Krause. Online distributed sensor selection. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 220–231, 2010.
- Jun Huang, Haoran Zhang, and Tarek Raïssi. Distributed interval estimation methods for multiagent systems. *IEEE Systems Journal*, 17(2):1843–1852, 2022.
- Dileep Kalathil, Naumaan Nayyar, and Rahul Jain. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, 2014.
- Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.
- Tor Lattimore, Koby Crammer, and Csaba Szepesvári. Linear multi-resource allocation with semi-bandit feedback. *Advances in Neural Information Processing Systems*, 28, 2015.
- Jian Li, Tongbao Chen, and Shaohua Teng. A comprehensive survey on client selection strategies in federated learning. *Computer Networks*, page 110663, 2024.
- Tan Li and Linqi Song. Privacy-preserving communication-efficient federated multi-armed bandits. *IEEE Journal on Selected Areas in Communications*, 40(3):773–787, 2022.
- Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

- Michael Molloy and Bruce Reed. *Graph colouring and the probabilistic method*, volume 23. Springer Science & Business Media, 2002.
- Reza Olfati-Saber, J Alex Fax, and Richard M Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.
- Abishek Sankararaman, Ayalvadi Ganesh, and Sanjay Shakkottai. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–35, 2019.
- Chengshuai Shi and Cong Shen. Federated multi-armed bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9603–9611, 2021.
- Chengshuai Shi, Wei Xiong, Cong Shen, and Jing Yang. Heterogeneous multi-player multi-armed bandits: Closing the gap and generalization. *Advances in neural information processing systems*, 34:22392–22404, 2021.
- Rayadurgam Srikant and Lei Ying. *Communication networks: An optimization, control and stochastic networks perspective*. Cambridge University Press, 2014.
- Liang Tang, Yexi Jiang, Lei Li, Chunqiu Zeng, and Tao Li. Personalized recommendation via parameter-free contextual bandits. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 323–332, 2015.
- Chao Tao, Qin Zhang, and Yuan Zhou. Collaborative learning with limited interaction: Tight bounds for distributed exploration in multi-armed bandits. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 126–146, 2019. doi: 10.1109/FOCS.2019.00017.
- Po-An Wang, Alexandre Proutiere, Kaito Ariu, Yassir Jedra, and Alessio Russo. Optimal algorithms for multiplayer multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 4120–4129. PMLR, 2020.
- Xuchuang Wang and Lin Yang. Achieving near-optimal individual regret low communications in multi-agent bandits. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.
- You-Gan Wang. Sequential allocation in clinical trials. *Communications in Statistics-Theory and Methods*, 20(3):791–805, 1991.
- Yuanhao Wang, Jiachen Hu, Xiaoyu Chen, and Liwei Wang. Distributed bandit learning: Near-optimal regret with efficient communication. *arXiv preprint arXiv:1904.06309*, 2019.
- Lin Xiao, Stephen Boyd, and Sanjay Lall. A scheme for robust distributed sensor fusion based on average consensus. In *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.*, pages 63–70. IEEE, 2005.
- Mengfan Xu and Diego Klabjan. Regret lower bounds in multi-agent multi-armed bandit. *arXiv preprint arXiv:2308.08046*, 2023.
- Mengfan Xu and Diego Klabjan. Decentralized randomly distributed multi-agent multi-armed bandit with heterogeneous rewards. *Advances in Neural Information Processing Systems*, 36, 2024.
- Feng Yan, Shreyas Sundaram, SVN Vishwanathan, and Yuan Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. *IEEE Transactions on Knowledge and Data Engineering*, 25(11):2483–2493, 2012.
- Lin Yang, Xuchuang Wang, Mohammad Hajiesmaili, Lijun Zhang, John C.S. Lui, and Don Towsley. Cooperative multi-agent bandits: Distributed algorithms with optimal individual regret and communication costs. In *Coordination and Cooperation for Multi-Agent Reinforcement Learning Methods Workshop*, 2024.
- Jingxuan Zhu and Ji Liu. Distributed multi-armed bandits. *IEEE Transactions on Automatic Control*, 2023.
- Zhaowei Zhu, Jingxuan Zhu, Ji Liu, and Yang Liu. Federated bandit: A gossiping approach. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5(1):1–29, 2021.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The abstract reflects our work and contributions greatly.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: by [\[Yes\]](#)

The limitations are written in Appendix G.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: The proof of each theorem and the first proposed lemma are given in Appendix D.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: The experimental parameters are given in Section 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: This work does not have any dataset, but we provide all code in the supplemental material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The experimental details are given in Section 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The confidence interval is defined in Section 3.5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [No]

Justification: Our work is theoretical, and the experiment is not the core. Besides, our experiment is a small simulation that could work on any computer with VS Code. Hence, we think it is unnecessary.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: This paper is a theoretical work that has nothing to do with human morality.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: This paper is a theoretical work that has nothing to do with human society.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper is not connected with models that have a high risk.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: The codes in this paper are all from the authors.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: This research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Appendix / Symbol explanation

For all symbols in this article, we give explanations of them in Table 3.

Symbol/Term	Definition
$\mathcal{G}(\mathcal{N}, \mathcal{E}, \mathcal{A})$	A graph to describe a multi-agent system
$\mathcal{N} = \{1, \dots, N\}$	Set of agents in a multi-agent system
$\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$	The edge set in graph $\mathcal{G}$
$\mathcal{A} = [a_{i,j}]_{N \times N}$	The weight matrix to describe the relations between agents
$\mathcal{N}_j$	Neighborhood of agent j , excluding agent j
$\mathbf{W} = [\omega_{a,b}]_{N \times N}$	Communication matrix
D	The diameter of graph $\mathcal{G}$
λ_2	The second largest eigenvalue of $\mathbf{W}$
C	A symbol to describe whether the graph is balance
$\mathcal{K} = \{1, \dots, K\}$	Set of arms in a multi-armed bandit (MAB) problem
T	Total number of time slots
$A_j(t)$	Arm chosen by agent j at time slot t
$X_{A_j(t),j}(t)$	Random reward received by agent j after pulling arm $A_j(t)$ at time slot t
$X_{i,j}(t)$	Random reward for arm i observed by agent j at time slot t
$\mu_{i,j}$	Mean reward for arm i observed by agent j , bounded in $[0, 1]$
$X_i(t)$	Global reward for arm i at time slot t
a	Constant that determines the size of an epoch
p^r	The sample count of any arm in $\mathcal{S}_j^r$ in epoch r_j , $p^r = a(r_j + 1)$
$X_{i,j}^r$	The average local reward of $X_{i,j}$ in epoch r_j
μ_i	Global mean reward for arm i
$\tilde{\mu}_{i,j}(t)$	The global estimate of agent j on arm i at time slot t
i^*	The unique optimal arm with the largest global mean reward
$i^{\max}$	$i^{\max} = \arg \max_{i \in \mathcal{S}_j} \tilde{\mu}_{i,j}$
$\mathcal{S}_j^r$	The candidate arm set of agent j at time slot t
$\mathcal{B}_j^r$	The elimination arm set of agent j at time slot t
$\hat{r}_i$	The epoch label attached to arm i
$\Delta_i = \mu_{i^*} - \mu_i$	Reward gap between the optimal arm and arm i
$\tau_{i,j}^r$	The sample count of agent j on arm i until epoch r_j
τ_i^r	The global sample count on arm i until epoch r_j
δ	The violation probability of confidence interval
$U_{i,j}^r(\delta)$	The radius of confidence interval
$\mathbb{E}[R(T)]$	Expected group regret
$\mathbb{E}[R_j^T(\mathcal{A})]$	Expected individual regret of agent j

Table 3: Summary of symbols and Definitions

B Appendix / some knowledge of graphs

Throughout this paper, we consider a FMAB problem with N agents operating in a time-invariant network. The network is represented by a communication graph $\mathcal{G}(\mathcal{N}, \mathcal{E}, \mathcal{A})$, which consists of three components:

1. $\mathcal{N} = \{1, \dots, N\}$ is the set of agents in the network, corresponding to the number of agents in the distributed system.
2. $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ is the edge set, which determines the connectivity between agents.
3. $\mathcal{A} = [a_{j,j'}]_{N \times N}$ is the adjacency matrix of the graph $\mathcal{G}$, where $a_{j,j'}$ denotes the weight of the edge between agents j and j' .

Notably, the adjacency matrix represents the importance of one agent to its neighbors and encodes neighborhood information in $\mathcal{G}$. Specifically, $a_{j,j'}$ is the weight from agent j' to agent j . Since the graph is directed, we have $a_{j,j'} \neq a_{j',j}$.

The graph has no self-loops, meaning that $a_{j,j} = 0$ for all $j \in \mathcal{N}$. An edge between agents j and j' exists if and only if $a_{j,j'} \neq 0$, i.e., $(j, j') \in \mathcal{E}$.

For each agent j , its neighborhood is denoted as $\mathcal{N}_j = \{j' \mid j' \in \mathcal{N}, a_{j,j'} \neq 0, j' \neq j\}$. Finally, we define the diameter of the graph $\mathcal{G}$ as D , which represents the longest distance between any two agents in the network.

For graph $\mathcal{G}$, its corresponding Laplacian matrix $\mathcal{L}$ is defined as follows

$$\mathcal{L} = \begin{cases} -a_{j,j'}, & j \neq j' \\ \sum_{j'=1}^N a_{j,j'}, & j = j' \end{cases}$$

The maximum degree of graph $\mathcal{G}$ is defined as $\epsilon = \max_i (\sum_{j' \neq i} a_{i,j'})$. Then, for any constant $\beta \in (0, 1/\epsilon]$, the Perron matrix $W = I - \beta\mathcal{L}$ could be obtained. The Perron matrix $\mathbf{W} = [\omega_{i,j}]_{N \times N}$ is a doubly random matrix and both the sum of row elements and column elements in $\mathbf{W}$ is 1. In the multi-agent bandit setting, it is widely used to solve the consensus problem Olfati-Saber et al. [2007].

C Appendix / Preliminaries of the problem

Lemma 2. [Yan et al., 2012] For a doubly random matrix $\mathbf{W}$, it is an irreducible, doubly stochastic matrix with strictly positive diagonal entries, then there exists a positive constant C such that

$$\sum_{j=1}^N |\omega_{i,j}^k - \frac{1}{N}| < C\lambda_2^k, \quad (11)$$

where $\omega_{i,j}^k$ represents the element in the i -th row and j -th column of the matrix W^k , k represents the iteration step and λ_2 is the second largest value of matrix $\mathbf{W}$. C is equal to 1 if the graph $\mathcal{G}$ is balanced, otherwise, $C = \sqrt{N}$.

Lemma 3. [Molloy and Reed, 2002] Assume that $X(t) - \mu$ is independent, σ sub-Gaussian random variable. Then for any $\epsilon \geq 0$,

$$\begin{aligned} \mathbb{P}(\hat{\mu} \geq \mu + \epsilon) &\leq \exp(-\frac{n\epsilon^2}{2\sigma^2}), \\ \mathbb{P}(\hat{\mu} \leq \mu - \epsilon) &\geq \exp(-\frac{n\epsilon^2}{2\sigma^2}), \end{aligned}$$

where $\hat{\mu} = \frac{1}{n} \sum_{t=1}^n X(t)$ and n is the sample count.

Lemma 4. [Lattimore and Szepesvári, 2020] Suppose that X_i is σ_i^2 sub-Gaussian and X_i are all independent for $i \in \{1, \dots, N\}$. Then we have $\frac{1}{N} \sum_{i=1}^N X_i$ is $\frac{\sum_{i=1}^N \sigma_i^2}{N^2}$ sub-Gaussian.

Lemma 5. [Dubhashi and Panconesi, 2009] If a random variable X has a finite mean and $a \leq X \leq b$ almost surely, then X is $\frac{1}{4}(b-a)^2$ sub-Gaussian.

D Appendix / Missed proofs

D.1 Proof of Lemma 1

Proof. The goal is to obtain an unbiased estimation $\bar{\mu}_{i,j}^r$ of the global mean μ_i . To achieve the goal, we could divide the problem into three parts: $|\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r|$, $|\tilde{\mu}_{i,j}^r - \hat{\mu}_i^r|$ and $|\hat{\mu}_i^r - \mu_i|$. According to the triangle inequality, we have

$$|\bar{\mu}_{i,j}^r - \mu_i| \leq |\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r| + |\tilde{\mu}_{i,j}^r - \hat{\mu}_i^r| + |\hat{\mu}_i^r - \mu_i|. \quad (12)$$

The first term $|\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r|$ is bounded according to the definition of information quantization introduced in Section 3.3. We have

$$|\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r| \leq \frac{1}{r_j + 2}.$$

Since all agents make decisions synchronously, we remove the subscript j from the epoch number r_j . The consensus process mainly acts at the end of each epoch; therefore, we consider the average reward $\bar{X}_{i,j}^r$ during the total epoch instead of the real-time reward $X_{i,j}(t)$ in each time slot t . To facilitate matrix-based operations, we stack equation (5) as follows

$$\tilde{\mu}_i^r = (1 - \sigma_i(r))\mathbf{W}\tilde{\mu}_i^{r-1} + \sigma_i(r)\bar{\mathbf{X}}_i^r, \quad (13)$$

where $\tilde{\mu}_i^r$ and $\bar{\mathbf{X}}_i^r$ are defined as

$$\begin{aligned} \tilde{\mu}_i^r &= [\tilde{\mu}_{i,1}^r, \dots, \tilde{\mu}_{i,N}^r]^T, \\ \bar{\mathbf{X}}_i^r &= [\bar{X}_{i,1}^r, \dots, \bar{X}_{i,N}^r]^T. \end{aligned}$$

The coefficient $\sigma_i(r_j)$ represents the ratio of the total amount of data aggregated by variable $\bar{X}_{i,j}^r$ to the overall data volume. The iteration of (5) occurs at the end of each epoch r_j , which implies that the sample count could be connected to epoch r . Hence, we define $\sigma_i(r_j) = \frac{2a(r_j+1)}{a(r_j+1)(r_j+2)} = \frac{2}{r_j+2}$, where a is the pre-trained sample count at the beginning of the algorithm.

According to the state above, iterating (13) at each epoch r yields that

$$\begin{aligned} \tilde{\mu}_i^r &= \frac{r}{r+2}\mathbf{W}\tilde{\mu}_i^{r-1} + \frac{2}{r+2}\bar{\mathbf{X}}_i^r \\ &= \frac{2}{a(r+1)(r+2)}\mathbf{W}^r\tilde{\mu}_i^0 + \frac{2}{a(r+2)(r+1)}\sum_{k=1}^r\mathbf{W}^{r-k}a(k+1)\bar{\mathbf{X}}_i^k \\ &\stackrel{(a)}{=} \frac{2}{a(r+2)(r+1)}\sum_{k=0}^r\mathbf{W}^{r-k}a(k+1)\bar{\mathbf{X}}_i^k, \end{aligned} \quad (14)$$

where equation (a) is obtained by setting $\tilde{\mu}_i^0 = \bar{\mathbf{X}}_i^0$.

Then, separate agent j 's global estimate from the vector $\tilde{\mu}_i^r$, we have

$$\tilde{\mu}_{i,j}^r = \frac{2}{a(r+2)(r+1)}\sum_{k=0}^r\sum_{j'=1}^N\omega_{j,j'}^{r-k}a(k+1)\bar{X}_{i,j'}^k. \quad (15)$$

For the fully connected graph, each agent can observe all other agents' observations. Hence, their global estimates $\hat{\mu}_i^r$ on arm i are the same, which is written as

$$\begin{aligned} \hat{\mu}_i^r &= \frac{2}{a(r+2)(r+1)}\sum_{\tau=0}^t\frac{1}{N}\sum_{j=1}^NX_{i,j}(\tau) \\ &\stackrel{(a)}{=} \frac{2}{a(r+2)(r+1)}\sum_{k=0}^r\frac{1}{N}\sum_{j=1}^Na(k+1)\bar{X}_{i,j}^k, \end{aligned} \quad (16)$$

where equation (a) is obtained because the concentration accrues at the end of each epoch, i.e., $t = a(r+1)(r+2)/2$.

The error between $\tilde{\mu}_{i,j}^r$ and $\hat{\mu}_i^r$ is

$$\begin{aligned} |\tilde{\mu}_{i,j}^r - \hat{\mu}_i^r| &= \frac{2}{a(r+1)(r+2)}\sum_{k=0}^r\sum_{j'=1}^N(\omega_{j,j'}^{r-k} - \frac{1}{N})a(k+1)\bar{X}_{i,j'}^k \\ &\stackrel{(a)}{\leq} \frac{2}{(r+1)(r+2)}\sum_{k=0}^rC\lambda_2^{r-k}(1+k) \\ &\stackrel{(b)}{\leq} \frac{2C}{(r+1)(r+2)}\times\frac{\lambda_2^{r+2}-\lambda_2}{(1-\lambda_2)^2} + \frac{2C}{(r+2)(1-\lambda_2)} \\ &\leq \frac{2C}{(1-\lambda_2)(r+2)}, \end{aligned} \quad (17)$$

where equation (a) yields from Lemma 2 and equation (b) is derived based on the properties of arithmetic and geometric sequences.

For a federated setting with N agents, the global reward is defined as $X_i(t) = \frac{1}{N} \sum_{j=1}^N X_{i,j}(t)$, where $X_{i,j}(t) \in [0, 1]$ represents the local reward observed by agent j . According to Lemma 5, each $X_{i,j}(t)$ is $\frac{1}{4}$ -sub-Gaussian. By applying Lemma 4, which addresses linear combinations of sub-Gaussian variables, it follows that the global reward $X_i(t)$ is $\frac{1}{4N}$ -sub-Gaussian.

By using Lemma 3 and setting $\varepsilon = \sqrt{\frac{\log \delta^{-1}}{2N\tau}}$ and $\sigma^2 = \frac{1}{4N}$, the probability that the estimate $\hat{\mu}_i^r$ exceeds the confidence interval after round r is given by

$$\begin{aligned} \mathbb{P} \left(\hat{\mu}_i(t) \geq \mu_i + \sqrt{\frac{\log \delta^{-1}}{aN(r+1)(r+2)}} \right) &\leq \delta, \\ \mathbb{P} \left(\hat{\mu}_i(t) \leq \mu_i - \sqrt{\frac{\log \delta^{-1}}{aN(r+1)(r+2)}} \right) &\leq \delta, \end{aligned} \quad (18)$$

i.e., with probability at least $1 - 2\delta$, we have

$$|\hat{\mu}_i^r - \mu_i| \leq \sqrt{\frac{\log \delta^{-1}}{aN(r+1)(r+2)}}. \quad (19)$$

Combining the upper bounds of the three terms, we obtain the following result based on equation (12):

$$|\bar{\mu}_{i,j}^r - \mu_i| \leq \sqrt{\frac{\log \delta^{-1}}{aN(r+1)(r+2)}} + \frac{2C}{(1-\lambda_2)(r+2)} + \frac{1}{r+2}.$$

which is the radius of the confidence interval. $\square$

D.2 Proof of Theorem 1

Proof. To establish bounds on both regret and communication cost, the key step is to characterize the sample complexity of all agents. Hence, the proof is divided into three parts: (1) bounding the total number of samples collected by each agent, (2) deriving the corresponding regret bounds, and (3) analyzing the total communication cost.

Bound the sample counts: Since Algorithm 1 is embedded in each agent and executed in a similar manner, we focus our analysis on a representative agent j . Agent j usually distinguishes whether an arm is suboptimal at the end of each epoch. Recall that for all arms $i \in \mathcal{K} \setminus \{i^*\}$, $\Delta_i > 0$. Agents learn the gap $\Delta_i > 0$ and continuously eliminate the arms $i : \Delta_i > 0$. To accelerate the update of the candidate set $\mathcal{S}_j$, detection is performed at the end of each epoch. Once suboptimal arm i is identified through this detection, agent j will add it into the elimination arm set $\mathcal{B}_j^r$ and later initiate a communication phase that broadcasts the elimination set $\mathcal{B}_j$ across the distributed system in waves.

Since that $\tilde{\mu}_{i,j}^r$ is to estimate μ_i , the reward gap Δ_i for each agent $j \in \mathcal{N}$ is related to $U_{i,j}^r$. The gap between $\tilde{\mu}_{i_{\max},j}^r$ and $\tilde{\mu}_{i,j}^r$ is

$$2U_{i,j}^r \stackrel{(7)}{\geq} \tilde{\mu}_{i_{\max},j}^r - \tilde{\mu}_{i,j}^r \stackrel{(a)}{\geq} \Delta_i - 2U_{i,j}^r \quad (20)$$

where inequality (a) is because $\tilde{\mu}_{i_{\max},j}^r \geq \tilde{\mu}_{i^*,j}^r \geq \mu_{i^*} - U_{i,j}^r$ and $\tilde{\mu}_{i,j}^r \leq \mu_i + U_{i,j}^r$.

Denote $A_{i,j,t}$ as the event of agent j pulling arm i at the time slot t , then we have

$$\mathbb{P} \left(\bigcap_{i,j,t} A_{i,j,t} \right) = 1 - \mathbb{P} \left(\bigcup_{i,j,t} \neg A_{i,j,t} \right) \geq 1 - \sum_{i,j,t} \mathbb{P}(\neg A_{i,j,t}) \geq 1 - 2tNK\delta.$$

At the end of each epoch r_j , we take a detection for each arm in the candidate set $\mathcal{S}_j^r$. The corresponding sample count is equal to $\tau_{i,j} = a(r_j + 1)(r_j + 2)/2$. Equation (20) is written as

$$\Delta_i \leq 4U_{i,j}^r \leq 4\sqrt{\frac{\log \delta^{-1}}{aN(r+1)(r+2)}} + \frac{8C}{1-\lambda_2} + \frac{4}{r_j + 2},$$

i.e.,

$$\begin{aligned}\Delta_i^2 - \frac{8\Delta_i(\frac{2C}{1-\lambda_2} + 1)}{r+2} + \frac{16(\frac{2C}{1-\lambda_2} + 1)^2}{(r+2)^2} &\leq \frac{16 \log \delta^{-1}}{aN(r+1)(r+2)}, \\ (r+1)(r+2 - \frac{8(\frac{2C}{1-\lambda_2} + 1)}{\Delta_i}) &\leq \frac{16 \log \delta^{-1}}{aN\Delta_i^2}, \\ r &\leq \frac{4\sqrt{\log \delta^{-1}}}{\sqrt{aN\Delta_i}} - 2 + \frac{8(\frac{2C}{1-\lambda_2} + 1)}{\Delta_i}.\end{aligned}$$

According to the definition of the epoch label $\hat{r}_i$ in equation (8), it follows that any suboptimal arm i will be added into the elimination set $\mathcal{B}_j^r$ no later than $\left\lceil \frac{4\sqrt{\log \delta^{-1}}}{\sqrt{aN\Delta_i}} - 2 + \frac{8(\frac{2C}{1-\lambda_2} + 1)}{\Delta_i} \right\rceil$. In the next epoch $\left\lceil \frac{4\sqrt{\log \delta^{-1}}}{\sqrt{aN\Delta_i}} - 2 + \frac{8(\frac{2C}{1-\lambda_2} + 1)}{\Delta_i} \right\rceil + 1$, agent j will broadcast arm i to all neighbors (Communication phase I).

As the number of time slots per epoch grows over time, the number of epochs needed for information to propagate from a single agent to the entire network decreases accordingly. The maximum communication epoch required is determined by

$$\min_{\tilde{r}} \sum_{r=1}^{\tilde{r}} a(r+1) \geq D, \quad (21)$$

where $\tilde{r}$ represents the most epoch number. From the optimization problem (21), the epoch number is at most $\left\lceil \sqrt{\frac{2D}{a}} + 2 \right\rceil$ to broadcast the message to all agents. In Algorithm 1, the epoch for sampling suboptimal arm i is denoted by $\bar{r}_i = \left\lceil \frac{4\sqrt{\log \delta^{-1}}}{\sqrt{aN\Delta_i}} - 2 + \frac{8(\frac{2C}{1-\lambda_2} + 1)}{\Delta_i} \right\rceil + \left\lceil \sqrt{\frac{2D}{a}} + 2 \right\rceil$.

Bound the regrets: According to the basic regret decomposition identity, the individual regret in equation (1) of agent j could be written as

$$\begin{aligned}R_j(T) &= T\mu_{i^*} - \sum_{t=1}^T \mathbb{E}[X_{A_j(t)}(t)] = T\mu_{i^*} - \sum_{t=1}^T \mu_{A_j(t)} \\ &\leq \sum_{i:\Delta_i>0} \Delta_i \tau_{i,j}(T) = \sum_{i:\Delta_i>0} \frac{a\Delta_i(\bar{r}_i+1)(\bar{r}_i+2)}{2} \\ &\leq \sum_{i:\Delta_i>0} \frac{a\Delta_i}{2} \left(\frac{4\sqrt{\log \delta^{-1}}}{\sqrt{aN\Delta_i}} + \bar{d}_i \right)^2 \\ &\leq \sum_{i:\Delta_i>0} \left(\frac{16 \log T}{N\Delta_i} + \frac{2\bar{d}_i\sqrt{2a \log T}}{\sqrt{N}} + \frac{a\bar{d}_i^2}{2} \right),\end{aligned}$$

where

$$\bar{d}_i = \frac{16C + 8 - 8\lambda_2}{\Delta_i(1 - \lambda_2)}.$$

According to the definition of group regret and the individual regret bound, equation (2) is written as

$$R(T) = \sum_{j=1}^N R_j(T) \leq \sum_{i:\Delta_i>0} \left(\frac{16 \log T}{\Delta_i} + 2\bar{d}_i\sqrt{2aN \log T} + \frac{aN\bar{d}_i^2}{2} \right).$$

□

D.3 Proof of Theorem 2

Proof. The communication in Algorithm 1 has two types, i.e., Communication phase I and II. In Communication phase I, agents mainly share the indexes of sub-optimal arms and their corresponding

epoch label $\hat{r}_i$. In Communication phase II, agents mainly share differential information $\delta_{i,j}^r$ about the global estimates.

Consider the size of each message: There are three types of information that need to be communicated in this setting: (a) the arm set $\mathcal{B}_j^r$, (b) the epoch label $\hat{r}_i$, and (c) the global estimate $\delta_{i,j}^r$. The message size for type (a) requires only $O(\log K)$ bits. For type (b), the size depends on the epoch label $\hat{r}_i$, which is upper bounded by $\bar{r}_i$ and can be encoded using $O(\sqrt{\log T})$ bits, as shown in the proof of Theorem 1. For type (c), the global estimate $\delta_{i,j}^r$ lies in $[0, 1]$ and is adaptively quantized. It is the most important part that needs to be optimized, as it dominates the total communication cost. The key challenge is to transmit the order of the global estimate from $O(\log(T))$ to $O(1)$.

After agent j computes the global estimate $\tilde{\mu}_{i,j}^r$ at the end of epoch r_j , it will be transmitted into a more concise version $\bar{\mu}_{i,j}^r$ with $\lceil 1 + \log_2 r_j \rceil$ bits. Then, the difference is calculated from $\delta_{i,j}^r = \bar{\mu}_{i,j}^r - \bar{\mu}_{i,j}^{r-1}$. Hence, we could bound $\delta_{i,j}^r$ to reduce the total communication cost. The difference $\delta_{i,j}^r$ holds that

$$\begin{aligned} |\delta_{i,j}^r| &= |\bar{\mu}_{i,j}^r - \bar{\mu}_{i,j}^{r-1}| = |\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r - \bar{\mu}_{i,j}^{r-1} + \tilde{\mu}_{i,j}^{r-1} + \tilde{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^{r-1}| \\ &\leq |\bar{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^r| + |\bar{\mu}_{i,j}^{r-1} - \tilde{\mu}_{i,j}^{r-1}| + |\tilde{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^{r-1}| \stackrel{(a)}{\leq} \frac{1}{r} + \frac{1}{r-1} + |\tilde{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^{r-1}|, \end{aligned} \quad (22)$$

where equation (a) yields from the quantization process in Algorithm 2. Due to that $\bar{\mu}_{i,j}^r$ contains $\lceil 1 + \log_2 r_j \rceil$ bits, which implies that there is a quantization error at most $1/r_j$ compared with the true estimate value $\tilde{\mu}_{i,j}^r$. For the term $|\tilde{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^{r-1}|$, we have

$$\begin{aligned} |\tilde{\mu}_{i,j}^r - \tilde{\mu}_{i,j}^{r-1}| &= |\tilde{\mu}_{i,j}^r - \hat{\mu}_i^r - \tilde{\mu}_{i,j}^{r-1} + \hat{\mu}_i^{r-1} + \hat{\mu}_i^r - \hat{\mu}_i^{r-1}| \\ &\leq |\tilde{\mu}_{i,j}^r - \hat{\mu}_i^r| + |\tilde{\mu}_{i,j}^{r-1} - \hat{\mu}_i^{r-1}| + |\hat{\mu}_i^r - \hat{\mu}_i^{r-1}| \\ &\leq \frac{2C}{(1-\lambda_2)(r_j+2)} + \frac{2C}{(1-\lambda_2)(r_j+1)} + |\hat{\mu}_i^r - \hat{\mu}_i^{r-1}|, \end{aligned}$$

where

$$\hat{\mu}_i^r - \hat{\mu}_i^{r-1} = \frac{\sum_{t=0}^{\frac{a(r_j+1)(r_j+2)}{2}} X_i(t)}{a(r_j+1)(r_j+2)/2} - \frac{\sum_{t=0}^{\frac{ar_j(r_j+1)}{2}} X_i(t)}{ar_j(r_j+1)/2}.$$

Due to that $X_i(t)$ is $\frac{1}{2\sqrt{N}}$ -sub-Gaussian random variable, $\hat{\mu}_i^r - \hat{\mu}_i^{r-1}$ could be considered as $\sqrt{\frac{1}{2aN(r_j+1)(r_j+2)}}$ -sub-Gaussian random variable.

Thus, we can further derive that, for a dummy variable $x \geq \sqrt{\ln 2}$,

$$\begin{aligned} \mathbb{P} \left(|\hat{\mu}_i^r - \hat{\mu}_i^{r-1}| \geq \sqrt{\frac{x^2}{2aN(r_j+1)(r_j+2)}} \right) &\leq 2 \exp \left[-\frac{\frac{a(r_j+1)(r_j+2)}{2} \times \frac{x^2}{2a(r_j+1)(r_j+2)}}{2 \times \frac{1}{4N}} \right] \\ &= 2 \exp[-x^2/2], \end{aligned}$$

which implies that

$$\begin{aligned} \mathbb{P} \left(|\delta_{i,j}^r| \geq \frac{1}{r_j} + \frac{1}{r_j-1} + \frac{2C}{(1-\lambda_2)(r_j+2)} + \frac{2C}{(1-\lambda_2)(r_j+1)} + \sqrt{\frac{x^2}{2aN(r_j+2)^2}} \right) &\leq 2 \exp[-x^2/2] \\ \Rightarrow \mathbb{P} \left(|\delta_{i,j}^r| \geq \frac{2 + \frac{4C}{1-\lambda_2} + \frac{x}{\sqrt{2aN}}}{r_j+2} \right) &\leq 2 \exp[-x^2/2] \\ \stackrel{(a)}{\Rightarrow} \mathbb{P} \left(L_{i,j}^r \geq 3 + \log_2 r_j + \log_2 \frac{2 + 4C/(1-\lambda_2) + x/\sqrt{2aN}}{r_j+2} \right) &\leq 2 \exp[-x^2/2] \\ \Rightarrow \mathbb{P} \left(L_{i,j}^r \geq 3 + \log_2(2 + 4C/(1-\lambda_2) + x/\sqrt{2aN}) \right) &\leq 2 \exp[-x^2/2] \\ \stackrel{(b)}{\Rightarrow} \mathbb{P} \left(L_{i,j}^r \leq 3 + \log_2(2 + 4C/(1-\lambda_2) + x/\sqrt{2aN}) \right) &\geq 1 - 2 \exp[-x^2/2], \end{aligned}$$

where $L_{i,j}^r$ in implication (a) represents the length of the truncated version $|\delta_{i,j}^r|$ and it is bounded by

$$L_{i,j}^r + \lfloor \log_2(1/|\delta_{i,j}^r|) \rfloor \leq \lceil 1 + \log_2 r_j \rceil. \quad (23)$$

Denote that $l = 3 + \log_2(2 + 4C/(1 - \lambda_2) + x/\sqrt{2aN})$, i.e., $x = \sqrt{2aN}(2^{l-3} - 2 - \frac{4C}{1-\lambda_2})$, and (b) could be written as

$$\mathbb{P}(L_{i,j}^r \leq l) \geq 1 - 2 \exp[-x^2/2] \geq 1 - 2 \exp \left[-aN(2^{l-3} - 2 - \frac{4C}{1-\lambda_2})^2 \right]. \quad (24)$$

With the above results and viewing $L_{i,j}^r$ as a random variable, we have that its cumulative distribution function (CDF) $F_{L_{i,j}^r}(l)$ satisfies the following property:

$$\begin{aligned} \forall l \geq \hat{l} &= \left\lceil 3 + \log_2(2 + 4C/(1 - \lambda_2) + \sqrt{\frac{\ln 2}{2aN}}) \right\rceil, \\ F_{L_{i,j}^r}(l) &= 1 - 2 \exp \left[-aN \left(2^{l-3} - 2 - \frac{4C}{1-\lambda_2} \right)^2 \right]. \end{aligned}$$

Using the property of CDF, we can bound the expectation of $L_{i,j}^r$ as

$$\begin{aligned} \mathbb{E}[L_{i,j}^r] &= \sum_{l=0}^{\infty} (1 - F_{L_{i,j}^r}(l)) \\ &\leq 1 + \hat{l} + \sum_{l=\hat{l}}^{\infty} (1 - F_{L_{i,j}^r}(l)) \\ &\leq 1 + \hat{l} + \int_{l=\hat{l}}^{\infty} 2 \exp \left[-aN \left(2^{l-3} - 2 - \frac{4C}{1-\lambda_2} \right)^2 \right] dl \\ &\leq 2 + \hat{l}. \end{aligned} \quad (25)$$

Thus, we have that in expectation, the truncated version of $|\delta_{i,j}^r|$ has a length that is less than $2 + \hat{l}$ bits.

Consider the total number of communication rounds in communication phase I. Communication phase I is a novel phase, used to broadcast the indices of sub-optimal arms. When an arm is identified as sub-optimal, the arm index will be exchanged at most $\left\lceil \frac{D}{a(r_j+1)} \right\rceil a(r_j+1)$ rounds. The total communication rounds on the arm i is at most $(K-1)(N-1)^2 \left\lceil \frac{D}{a(r_j+1)} \right\rceil a(r_j+1)$ because agent j_1 may share the information with j_2 even if the information is from j_2 initially. Considering N agents pulling K arms, the number broadcast round during the total horizon T in phase I is at most $(K-1)(N-1)^2 \left\lceil \frac{D}{a(r_j+1)} \right\rceil a(r_j+1)$.

Consider the total number of communication rounds in communication phase II. According to Theorem 1, each suboptimal arm is pulled for at most $\bar{r}_i$ epochs. In the structure of Algorithm 1, agents exchange their differential information $\delta_{i,j}^r$ at the end of each epoch. Therefore, the total number of communication rounds in Communication Phase II is upper bounded by $N \sum_{i=1}^K \bar{r}_i$.

According to the discussion above, the total communication cost of the entire system is bounded by

$$\begin{aligned} C(T) &\leq (2 + \hat{l})N \sum_{i=1}^K \bar{r}_i + (\log_2 \hat{r}_i + \log_2 K)(K-1)(N-1)^2 \left\lceil \frac{D}{a(r_j+1)} \right\rceil a(r_j+1) \\ &\leq O \left(\sum_{i:\Delta_i>0} \sqrt{N} \Delta_i^{-1} \sqrt{\log T} \right). \end{aligned} \quad (26)$$

□

D.4 Proof of Theorem 3

Proof. We will present the proof of the lower bound in three parts.

Problem setting. First of all, we need to consider a more relaxed scenario. Consider a multi-agent system with N agents, which is a fully connected graph. For each agent j , suppose that there is an associated “replica” subsystem $\hat{j}$ that can provide samples of agents $\{1, \dots, N\} \setminus \{j\}$. When agent j samples arm i and obtains a reward $X_{i,j}$, replica subsystem $\hat{j}$ provides rewards $X_{i,k}, k \in \{1, \dots, N\} \setminus \{j\}$. This construction allows j to leverage peer observations without requiring global synchronization. In this construction, all agents in $\mathcal{N}$ could make different decisions, which means that the lower bound is compatible with all bandit algorithms.

Consensus estimation evaluation. Then, we need to ensure the convergence of consensus estimation in this setting. In each round, the agents obtain not only the global estimates from their neighbors but also information from their replica subsystems. From the replica subsystem, agent j could obtain a global reward and denote it by $\bar{X}_{i,j}^\tau$. The concentration function for the fully connected graph is

$$\hat{\mu}_i^\tau = \frac{1}{\tau + 1} \sum_{j=1}^N \sum_{k=0}^{\tau} \frac{1}{N} \bar{X}_{i,j}^k.$$

Considering that $X_{i,j}$ are all 1-sub-Gaussian variables, the linear combination reward $\hat{\mu}_i$ is a $\sqrt{\frac{\sum_{j=1}^N \sum_{i=1}^N \sigma^2}{N^4}} = \frac{1}{N}$ -sub-Gaussian variable. By using Lemma 3, the relationship between $\hat{\mu}_i^\tau$ and μ_i in this setting is

$$\begin{aligned} \mathbb{P} \left(\hat{\mu}_i^\tau \geq \mu_i + \sqrt{\frac{\log \delta^{-1}}{8N^2\tau}} \right) &\leq \delta, \\ \mathbb{P} \left(\hat{\mu}_i^\tau \leq \mu_i - \sqrt{\frac{\log \delta^{-1}}{8N^2\tau}} \right) &\leq \delta. \end{aligned}$$

The confidence interval $\sqrt{\frac{\log \delta^{-1}}{8N^2\tau}}$ processes better concentration performance than (6) because it has an additional factor $\frac{1}{\sqrt{N}}$. Hence, the consensus estimation performance in this setting is better than that of general federated bandit settings. Hence, we could derive more convincing lower bounds on regrets.

Lower bound proof. Let $\mathcal{M}$ be a set of distributions with finite means, and let $\mu : \mathcal{M} \rightarrow \mathbb{R}$ be the function that maps $P \in \mathcal{M}$ to its mean. Let $\mu_{i^*} \in \mathbb{R}$ and $P \in \mathcal{M}$ have $\mu(P) < \mu_{i^*}$ and define

$$d_i = d_{\inf}(P, \mu_{i^*}, \mathcal{M}) = \inf_{P' \in \mathcal{M}} \{D(P, P') : \mu(P') > \mu_{i^*}\},$$

where $D(P, P')$ is the relative entropy between P and P' .

Define two reward distributions as follows

$$\nu = (P_1, \dots, P_i, \dots, P_K),$$

$$\nu' = (P_1, \dots, P'_i, \dots, P_K).$$

Let all arms except arm i be the same in the two distributions. For arm i , let $\epsilon > 0$ be an arbitrary small value such that $D(P_i, P'_i) \leq d_i + \epsilon$ and $\mu(P'_i) > \mu_{i^*}$.

According to Lemma 15.1 in reference [Lattimore and Szepesvári, 2020], the divergence between ν and ν' is decomposed into

$$D(\mathbb{P}_\nu, \mathbb{P}_{\nu'}) = \sum_{k=1}^K \mathbb{E}[\tau_{k,j}(T)] D(P_i, P'_i) \stackrel{(a)}{\leq} \mathbb{E}[\tau_{i,j}(T)] (d_i + \epsilon),$$

where equation (a) is obtained based on $D(P_j, P'_j) = 0$ if $j \neq i$.

According to Bretagnolle-Huber inequality (Theorem 14.2 in Lattimore and Szepesvári [2020]), for any event A , we have

$$\mathbb{P}_\mu(A) + \mathbb{P}_{\mu'}(A^c) \geq \frac{1}{2} \exp(-D(\mathbb{P}_\nu, \mathbb{P}_{\nu'})) \geq \frac{1}{2} \exp(-\mathbb{E}[\tau_{i,j}(T)](d_i + \epsilon))$$

Choose $A = \{\tau_{i,j}(T) > T/2\}$, and let $R_T = R_T(\mathcal{A}, \nu)$ and $R'_T = R'_T(\mathcal{A}, \nu')$. Then

$$\begin{aligned} R_T + R'_T &\geq \frac{T}{2}(\mathbb{P}_\mu(A)\Delta_i + \mathbb{P}_{\mu'}(A^c)(\mu'_i - \mu_{i^*})) \\ &\geq \frac{T}{2} \min\{\Delta_i, \mu'_i - \mu_{i^*}\}(\mathbb{P}_\mu(A)\Delta_i + \mathbb{P}_{\mu'}(A^c)) \\ &\geq \frac{T}{2} \min\{\Delta_i, \mu'_i - \mu_{i^*}\} \exp(-\mathbb{E}[\tau_{i,j}(T)](d_i + \epsilon)). \end{aligned}$$

Rearranging and taking the limit inferior leads to

$$\begin{aligned} \liminf_{T \rightarrow \infty} \frac{\mathbb{E}[\tau_{i,j}(T)]}{\log T} &\geq \frac{1}{d_i + \epsilon} \liminf_{T \rightarrow \infty} \frac{\log \frac{T \min\{\Delta_i, \mu'_i - \mu_{i^*}\}}{4(R_T + R'_T)}}{\log T} \\ &\geq \frac{1}{d_i + \epsilon} (1 - \liminf_{T \rightarrow \infty} \frac{\log(R_T + R'_T)}{\log T}) \\ &= \frac{1}{d_i + \epsilon}, \end{aligned}$$

where the last equality follows from the definition of consistency, which says that for any $p > 0$, there exists a constant C_p such that for sufficiently large T , $R_T + R'_T \leq C_p T^p$, which implies that

$$\liminf_{T \rightarrow \infty} \frac{\log(R_T + R'_T)}{\log T} \leq p.$$

Considering $p > 0$ was arbitrary and $\epsilon > 0$ is limited to zero, we have

$$\liminf_{T \rightarrow \infty} \frac{\mathbb{E}[\tau_{i,j}(T)]}{\log T} \geq \frac{1}{d_i}.$$

In the proof of the lower bound regret, agents only use the information from their replica subsystems and the reward mean is a $\frac{1}{\sqrt{N}}$ -sub-Gaussian variable. According to Table 16.1 given in Lattimore and Szepesvári [2020], we have $d_i = \frac{N\Delta_i^2}{2}$. The individual regret of the problem is lower bounded by

$$\liminf_{T \rightarrow \infty} \frac{R_j(T)}{\log T} \geq \liminf_{T \rightarrow \infty} \sum_{i: \Delta_i > 0} \frac{\Delta_i \mathbb{E}[\tau_{i,j}(T)]}{\log T} \geq \sum_{i: \Delta_i > 0} \frac{\Delta_i}{d_i} \geq \sum_{i: \Delta_i > 0} \frac{2}{N\Delta_i}.$$

The group lower regret is

$$\begin{aligned} \liminf_{T \rightarrow \infty} \frac{R(T)}{\log T} &= \liminf_{T \rightarrow \infty} \frac{\sum_{j=1}^N R_j(T)}{\log T} \geq \liminf_{T \rightarrow \infty} \sum_{j=1}^N \sum_{i: \Delta_i > 0} \frac{\Delta_i \mathbb{E}[\tau_{i,j}(T)]}{\log T} \\ &\geq \sum_{j=1}^N \sum_{i: \Delta_i > 0} \frac{\Delta_i}{d_i} \geq \sum_{i: \Delta_i > 0} \frac{2}{\Delta_i}. \end{aligned}$$

□

E Appendix / Experiments on homogeneous settings

Under the conditions in Section 5, some experiments are conducted in this section. Our algorithm (EpoInc-SE) can also obtain a similar result compared to the optimal homogeneous bandit algorithm in Figure 2.

F Appendix / Experiments on large-scale multi-agent systems

For large-scale deployments, the agent number N would play an important role in communication cost and regrets. In order to further verify the performance of our algorithm, we conducted an experiment involving 100 agents. The main results are shown in Figure 3. The proposed algorithm (EpoInc-SE) also processes the advantage in communication cost, while it can obtain a near-optimal regret compared with other algorithms.

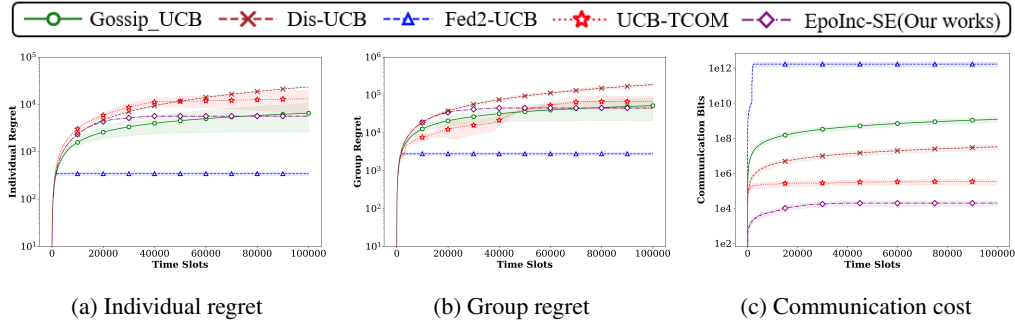


Figure 2: Performance in homogeneous settings.

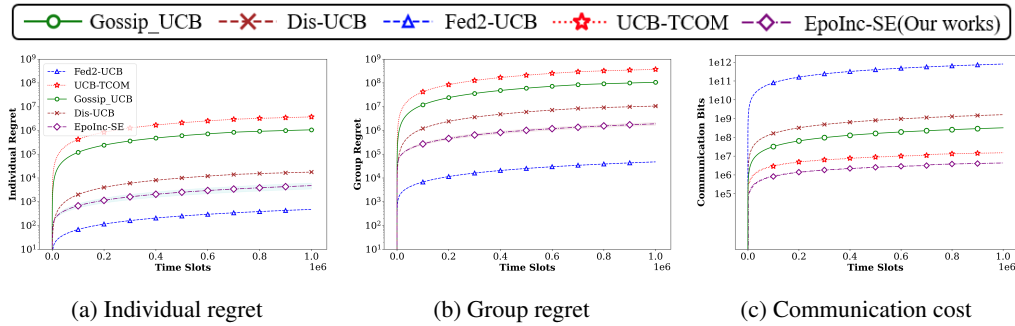


Figure 3: Performance in large-scale multi-agent systems.

G Appendix / Limitations

Although our algorithm has obtained a near-optimal result, it still has certain limitations as follows:

- The algorithm proposed in this work is the successive elimination algorithm, which relies on round-robin sampling. A similar result can not be introduced to UCB-based algorithms, which are the mainstream algorithms in bandit problems.
- For time-variant communication graphs, EpoInc-SE would be not work because the consensus estimation part EBCEs could not work in random graphs. This is because Lemma 2 relies on a Markov Decision process, which needs a fixed graph.
- In some practical distributed systems, information may be unavailable or costly to obtain. For the first condition, the information is unavailable to some agents, which implies that the graph $\mathcal{G}$ is disconnected with a clique-connected component. Then, agents cannot learn global estimates without bias because agents lack other agents' local observations. Thus, the regret would be linear $\Omega(T)$, which has been discussed in reference [Xu and Klabjan, 2023]. For the second condition, the communication cost cannot be upper-bounded because the cost of each message is unknown.