The Gossiping Insert-Eliminate Algorithm for Multi-Agent Bandits

Ronshee Chawla UT Austin
Abishek Sankararaman UT Austin
Ayalvadi Ganesh University of Bristol
Sanjay Shakkottai UT Austin

Abstract

We consider a decentralized multi-agent Multi Armed Bandit (MAB) setup consisting of N agents, solving the same MAB instance to minimize individual cumulative regret. In our model, agents collaborate by exchanging messages through pairwise gossip style communications. We develop two novel algorithms, where each agent only plays from a subset of all the arms. Agents use the communication medium to recommend only arm-IDs (not samples), and thus update the set of arms from which they play. We establish that, if agents communicate $\Omega(\log(T))$ times through any connected pairwise gossip mechanism, then every agent's regret is a factor of order N smaller compared to the case of no collaborations. Furthermore, we show that the communication constraints only have a second order effect on the regret of our algorithm. We then analyze this second order term of the regret to derive bounds on the regret-communication tradeoffs. Finally, we empirically evaluate our algorithm and conclude that the insights are fundamental and not artifacts of our bounds. We also show a lower bound which gives that the regret scaling obtained by our algorithm cannot be improved even in the absence of any communication constraints. Our results demonstrate that even a minimal level of collaboration among agents greatly reduces regret for all agents.

1 Introduction

Multi Armed Bandit (MAB) is a classical model ((Lattimore and Szepesvári, 2018),(Bubeck et al., 2012)), that captures the explore-exploit trade-off in making online decisions. MAB paradigms have found applications in many large scale systems such as ranking on search engines (Yue and Joachims, 2009), displaying advertisements on e-commerce web-sites (Chakrabarti et al., 2009), model selection for classification (Li et al., 2016) and real-time operation of wireless networks (Avner and Mannor, 2016). Oftentimes in these settings, the decision making is distributed among many agents. For example, in the context of web-servers serving either search ranking or placing advertisements, due to the the volume and rate of user requests, multiple servers are deployed to perform the same task (Cesa-Bianchi et al., 2019). Each server, makes decisions (which can be modeled as a MAB (Yue and Joachims, 2009)) on rankings or placing advertisements and also collaborate with other servers by communicating over a network (Cesa-Bianchi et al., 2019). In this paper, we study a multi-agent MAB model in which agents collaborate to reduce individual cumulative regret.

Model Overview - Our model generalizes the problem setting described in (Sankararaman et al., 2019). Concretely, our model consists of N agents, each playing the same instance of a K armed stochastic MAB, to minimize its cumulative regret. At each time, every agent pulls an arm and receives a stochastic reward independent of everything else (including other agents choosing the same arm at the same time). Additionally, an agent can choose after an arm pull, to receive a message from another agent through an *information pull*. Agents have a *communication budget*, which limits how many times an agent can pull information. If any agent $i \in \{1, \cdots, N\}$ chooses to receive a message through an information-pull, then it will contact another agent j chosen independent of everything else, at random from a distribution $P(i, \cdot)$ (unknown to the agents) over $\{1, \cdots, N\}$. The agents thus cannot actively choose from whom they can receive information, rather they receive from another randomly chosen agent. The $N \times N$ matrix P with its i^{th} row being the distribution $P(i, \cdot)$ is denoted as the gossip matrix. Agents take actions (arm-pulls, information-pulls and messages sent) only as a function of their past his-

Proceedings of the 23[rd]International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.

tory of arm-pulls, rewards and received messages from information-pulls and is hence *decentralized*.

Model Motivations - The problem formulation and the communication constraints aim to capture key features of many settings involving multiple agents making distributed decisions. We highlight two examples in which our model is applicable. The first example is a setting consisting of N computer servers (or agents), each handling requests for web searches from different users on the internet (Buccapatnam et al., 2015; Li et al., 2010). For each keyword, one out of a set of M ad-words needs to be displayed, which can be viewed as choosing an arm of a MAB. Here, each server is making decisions on which ad to display (for the chosen keyword) independently of other servers. Further, the rewards obtained by different servers are independent because the search users are different at different servers. The servers can also communicate with each other over a network in order to collaborate to maximize revenue (i.e., minimize cumulative regret).

A second example is that of collaborative recommendation systems, e.g., where multiple agents (users) in a social network are jointly exploring restaurants in a city (Sankararaman et al., 2019). The users correspond to agents, and each restaurant can be modeled as an arm of a MAB providing stochastic feedback. The users can communicate with each other over a social network, personal contact or a messaging platform to receive recommendation of restaurants (arms) from others to minimize their cumulative regret, where regret corresponds to the loss in utility incurred by each user per restaurant visit. Furthermore, if the restaurants/customers can be categorized into a finite set of contexts (say, e.g. by price: low-cost/mid-price/high-end, type of cuisine: italian, asian, etc.), our model is applicable per context.

Key Contributions:

1. Gossiping Insert-Eliminate (GosInE) Algorithm - In our algorithms (Algorithm 1 and 3), agents only choose to play from among a small subset (of cardinality $\lceil \frac{K}{N} \rceil + 2$) of arms at each time. Agents in our algorithm accept the communication budget as an input and use the communication medium to *recommend* arms, i.e., agents communicate the arm-ID of their current estimated best arm. Specifically, agents do not exchange samples, but only recommend an arm index. On receiving a recommendation, an agent updates the set of arms to play from: it discards its estimated worst arm in its current set and replaces it by the recommended new arm.

Thus, our algorithm is *non monotone* with respect to the set of arms an agent plays from, as agents can discard an arm in a phase and then subsequently bring the arm back and play it in a later phase, if this previously discarded arm gets recommended by another agent. This is in contrast to most other bandit algorithms in the literature. On one hand, classical regret minimization algorithms such as UCB-α (Auer et al., 2002) or Thompson sampling (Thompson, 1933) allow sampling from any arm at all points in time (no arm ever discarded). On the other hand, pure explore algorithms such as successive rejects (Audibert and Bubeck, 2010) are monotone with respect to the arms, i.e., a discarded arm is never subsequently played again. The social learning algorithm in (Sankararaman et al., 2019) is also monotone, as the subset of arms from which an agent plays at any time is non-decreasing. In contrast, in this paper we show that even if an agent (erroneously) discards the best arm from its playing set, the recommendations ensure that with probability 1, the best arm is eventually back in the playing set.

2. Regret of GosInE Algorithm - Despite agents playing among a *time-varying* set of arms of cardinality $\lceil \frac{K}{N} \rceil + 2$, we show that the regret of any agent is (Theorems 1 and 2) $\mathrm{O}\Big(\Big(\frac{\lceil \frac{K}{N} \rceil + 1}{\Delta_2} \log(T)\Big)\Big) + C$. Here, Δ_2 is the difference in the mean rewards of the best and second best arm and C is a constant depending on communication constraints and independent of time. We show that the regret scaling holds *for any connected gossip matrix P and communication budget scaling as* $\Omega(\log(T))$. Thus, any agent's asymptotic regret is independent of the the gossip matrix P or the communication budget (Corollary 4). If agents never collaborate (communication budget of 0), the system is identical to each agent playing a standard K arm MAB, in which case the regret scales as $\mathrm{O}\big(\frac{K}{\Delta_2} \log(T)\big)$ (Lai and Robbins, 1985),(Auer et al., 2002). Thus, our algorithms reduce the regret of any agent by a factor of order N from the case of no collaborations. Furthermore, a lower bound in Theorem 3 (and the discussion in Section 6) shows that this scaling with respect to K and N cannot be improved by any algorithm, communication budget or gossip matrix. Specifically, we show that even if an agent has knowledge of the entire system history of arms pulled and rewards obtained by other agents, the regret incurred by every agent is only a factor of order N smaller than the case of no collaborations. Moreover, our regret scaling significantly improves over that of (Sankararaman et al., 2019), which applies only to the complete graph among agents, in which the regret scales as $\mathrm{O}\big(\frac{\lceil \frac{K}{N} \rceil + \log(N)}{\Delta_2} \log(T)\big)$. Thus, despite communication constraints, our algorithm leverages collaboration effectively.

3. Communication/Regret Trade-Off - The sec-

ond order constant term in our regret bound captures the trade-off between communications and regret. As an example, we show in Corollary 18 that, if the communication budgets scale polynomially, i.e., agents can pull information at-most $t^{1/\beta}$ times over a time horizon of t, for some $\beta > 1$, when the agents are connected by a ring graph (the graph with poorest connectivity), the constant term in the regret scales as $(N)^{\beta}$ (upto poly-logarithmic factor), whereas the regret scales as $(\log(N))^{\beta}$, in the case when agents are connected by the complete graph. Thus, we see that there is an exponential improvement (in the additive constant) in the regret incurred, when changing the network among agents from the ring graph to the complete graph. In general, we give through an explicit formula (in Corollary 18) that, if the gossip matrix P has smaller conductance (i.e., a poorly connected network), then the regret incurred by any agent is higher. Similarly, we also establish the fact that if the communication budget per agent is higher, then the regret incurred is lower (Corollary 20). We further conduct numerical studies that establish these are fundamental and not artifacts of our bounds.

2 Problem Setup

Our model generalizes the setting in (Sankararaman et al., 2019). In particular, our model, imposes communication budgets and allows for general gossip matrices P, while the model in (Sankararaman et al., 2019) considered only the complete graph among agents.

Arms of the MAB - We consider N agents, each playing the same instance of a K armed stochastic MAB to minimize cumulative regret. The K arms have unknown average rewards denoted by $\mu_1, \cdots, \mu_K$, where for every $i \in \{1, \cdots, K\}$, $\mu_i \in (0, 1)$. Without loss of generality, we assume $1 > \mu_1 > \mu_2 \geq \mu_3 \cdots \geq \mu_K \geq 0$. However, the agents are not aware of this ordering. For all $j \in \{2, \cdots, K\}$, denote by $\Delta_j := \mu_1 - \mu_j$. The assumption on the arm-means imply that $\Delta_j > 0$, for all $j \in \{2, \cdots, K\}$.

Network among Agents - We suppose that the agents are connected by a network denoted by a $N \times N$ *gossip matrix* P, where for each $i \in \{1, \cdots, N\}$, the i^{th} row $P(i, \cdot)$ is a probability distribution over $\{1, \cdots, N\}$. This matrix is fixed and unknown to the agents.

Agent Actions - We assume that time is slotted (discrete), with each time slot divided into an arm-pulling phase followed by an information-pulling phase. In the arm-pulling phase, all agents pull one of the K arms and observe a stochastic Bernoulli reward, independent of everything else. In the information pulling phase, if an agent has communication budget, it can decide to receive a message from another agent through an information pull. A non-negative and non-decreasing sequence $(B_t)_{t\in\mathbb{N}}$ specifies the communication budget, where no agent can pull information for more than B_t times in the first t time slots for all $t \geq 0$. If any agent $i \in \{1, \cdots, N\}$, chooses to pull information in the information-pulling phase of any time slot, it will contact another agent $j \in \{1, \cdots, N\}$ chosen independently of everything else, according to the probability distribution given by $P(i, \cdot)$. Thus, agents receive information from a randomly chosen agent according to a *fixed* distribution, rather than actively choosing the agents based on observed samples. When any agent j is contacted by another agent in the information pulling phase of a time-slot, agent j can communicate a limited $(\mathrm{O}\big(\log(NK)\big)$ number of bits. Crucially, the message length does not depend on the arm-means or on the time index.

Decentralized System - Each action of an agent, i.e., its arm pull, decision to engage in an information pull and the message to send when requested by another agent's information pull, can only depend on the agent's past history of arms pulled, rewards obtained and messages received from information pulls. We allow each agent's actions in the information pulling phase (such as whether to pull information and what message to communicate if asked for), to depend on the agent's outcome in the arm-pulling phase of that time slot.

Performance Metric - Each agent minimizes their expected cumulative regret. For an agent $i \in \{1, \cdots, N\}$ and time $t \in \mathbb{N}$, denote by $I_t^{(i)} \in \{1, \cdots, K\}$ to be the arm pulled by agent i in the arm-pulling phase of time slot t. The regret of agent $i \in \mathbb{N}$, after T time slots (arm-pulls) is defined as $R_T^{(i)} := \sum_{t=1}^{T}(\mu_1 - \mu_{I_t^{(i)}})$ and the expected cumulative regret is $\mathbb{E}[R_T^{(i)}]$[1].

3 Synchronous GosInE Algorithm

We describe the algorithm by fixing an agent $i \in \{1, \cdots, N\}$.

Input Parameters - The algorithm has three inputs *(i)* a communication budget $(B_t)_{t\in\mathbb{N}}$, *(ii)* $\alpha > 0$ and *(iii)* $\varepsilon > 0$. From this communication budget, we construct a sequence $(A_x)_{x\in\mathbb{N}}$ such that

$$A_x = \max\left(\min\{t \in \mathbb{N}, B_t \geq x\}, \lceil (1+x)^{1+\varepsilon} \rceil\right). \quad (1)$$

Every agent, only pulls information in time slots

[1] Expectation is with respect to all randomness, i.e., rewards, communications and possibly the algorithm.

$(A_x)_{x\in\mathbb{N}}$. This automatically respects the communication budget constraints. Since agents engage in information pulling at common time slots, we term the algorithm, synchronous. For instance, if $B_t = \lceil t^{1/3} \rceil$, for all $t \geq 1$. and $\varepsilon < 2$, then $A_x = \lfloor x^3 \rfloor$, for all $x \geq 1$. Similarly, if $B_t = t$, for all $t \geq 1$, i.e., if the budget is adequate to communicate in every time slot, then $A_x = \lceil (1+x)^{1+\varepsilon} \rceil$, for all $x \geq 1$. The parameter ϵ ensures that the time intervals between the instants when agents request for an arm are well separated. In particular, having $\varepsilon > 0$ ensures that the inter-communication times scale at least polynomially in time. As we shall see in the analysis, this only affects the regret scaling in the second order term.

Initialization - Associated with each agent $i \in \{1, \cdots, N\}$, is a *sticky*[2] set of arms -

$$\widehat{S}^{(i)} = \left\{ \left((i-1) \left\lceil \frac{K}{N} \right\rceil \mod K \right) + 1, \cdots, \left(i \left\lceil \frac{K}{N} \right\rceil - 1 \mod K \right) + 1 \right\}. \quad (2)$$

Notice that the cardinality $|\widehat{S}^{(i)}| = \lceil \frac{K}{N} \rceil$. In words, we are partitioning the total set of arms, into sets of size $\lceil \frac{K}{N} \rceil$ with the property that $\bigcup_{i=1}^{N} \widehat{S}^{(i)} = \{1, \cdots, K\}$. For instance, if $K = N$, then for all $i \in \{1, \cdots, N\}$, $\widehat{S}^{(i)} = \{i\}$. Denote by the set $U_0^{(i)} = \{i\lceil \frac{K}{N} \rceil \mod K\}$ and $L_0^{(i)} = \{i\lceil \frac{K}{N} \rceil + 1 \mod K\}$ and

$$S_0^{(i)} = \widehat{S}^{(i)} \cup U_0^{(i)} \cup L_0^{(i)}. \quad (3)$$

UCB within a phase - The algorithm proceeds in phases with all agents starting in phase 0. Each phase $j \geq 1$ lasts from time-slots $A_{j-1} + 1$ till time-slot A_j, both inclusive[3]. We shall fix a phase $j \geq 0$ henceforth in the description. For any arm $l \in \{1, \cdots, K\}$ and any time $t \in \mathbb{N}$, $T_l^{(i)}(t)$ is the total number of times agent i has pulled arm l, upto and including time t and by $\widehat{\mu}_l^{(i)}(t)$, the empirical observed mean[4]. Agent i in phase j, chooses arms from $S_j^{(i)}$ according to the UCB-α policy of (Auer et al., 2002) where the arm is selected from $\arg\max_{l \in S_j^{(i)}} \left(\widehat{\mu}_l^{(i)}(t-1) + \sqrt{\frac{\alpha \ln(t)}{T_l^{(i)}(t-1)}} \right)$.

Pull Information at the end of a phase - The message received (arm-ID in our algorithm) in the information-pulling phase of time slot A_j is denoted by $\mathcal{O}_j^{(i)} \in \{1, \cdots, K\}$. Every agent, when asked for a message in the information-pulling phase of time-slot A_j, will send the arm-ID it played the most in phase j.

[2]The choice of term *sticky* is explained in the sequel.

[3]We use the convention $A_{-1} = 0$

[4] $\widehat{\mu}_l^{(i)}(t) = 0$ if $T_l^{(i)}(t) = 0$

Update arms at the beginning of a phase - If $\mathcal{O}_j^{(i)} \in S_j^{(i)}$, then $S_{j+1}^{(i)} = S_j^{(i)}$. Else, agent i discards the least played arm in phase j from the set $S_j^{(i)} \setminus \widehat{S}^{(i)}$ and accepts the recommendation $\mathcal{O}_j^{(i)}$, to form the playing set $S_{j+1}^{(i)}$. Observe that the cardinality of $S_{j+1}^{(i)}$ remains unchanged. Moreover, the updating ensures that for all agents $i \in \{1, \cdots, N\}$ and all phases j, $\widehat{S}^{(i)} \subset S_j^{(i)}$, namely agents never drop arms from the set $\widehat{S}^{(i)}$. Hence, we term the set $\widehat{S}^{(i)}$, sticky.

The pseudo-code of the Algorithm described above is given in Algorithm 1.

3.1 Model Assumptions

We make two mild assumptions on the inputs, namely that the gossip matrix P is connected (Assumption **A.1**) and that the communication budgets $B_t = \Omega(\log(t))$ (Assumption **A.2**). These are specified in detail in Appendix A.

3.2 Regret Guarantee

The regret guarantee of Algorithm 1 is given in Theorem 1, which requires a definition. Let $N \in \mathbb{N}$ and a P be a $N \times N$ gossip matrix. Denote by the random variable $\tau_{spr}^{(P)}$ to be the *spreading time* of a rumor in a pull model, with a rumor initially in node 1 (cf (Shah, 2009)). Formally, consider a discrete time stochastic process where initially, node 1 has a rumor. At each time step, each node $j \in \{1, \cdots, N\}$ that does not possess the rumor, calls another node sampled independently of everything else from the probability distribution $P(j, \cdot)$. If a node j calls on a node possessing the rumor, node j will possess the rumor at the end of the call (at the end of current time step). The spreading time $\tau_{spr}^{(P)}$ is the stopping time when all nodes possess the rumor for the first time.

Theorem 1. *Suppose in a system of $N \geq 2$ agents connected by a communication matrix P satisfying assumption* **(A.1)** *and $K \geq 2$ arms, each agent runs Algorithm 1, with UCB parameter $\alpha > 3$ and communication budget $(B_t)_{t\in\mathbb{N}}$ and $\varepsilon > 0$ satisfying assumption* **(A.2)**. *Then the regret of any agent $i \in [N]$, after time any time $T \in \mathbb{N}$ is bounded by*

$$\mathbb{E}[R_T^{(i)}] \leq \underbrace{\left(\sum_{j=2}^{\lceil \frac{K}{N} \rceil + 2} \frac{1}{\Delta_j} \right) 4\alpha \ln(T) + \frac{K}{4}}_{\text{Collaborative UCB Regret}} + \underbrace{g((A_x)_{x\in\mathbb{N}}) + \mathbb{E}[A_{2\tau_{spr}^{(P)}}]}_{\text{Cost of Infrequent Pairwise Communications}}, \quad (4)$$

where $(A_x)_{x\in\mathbb{N}}$ is given in Equation (1) and

Algorithm 1 Synch GosInE Algorithm (at Agent i)

1: **Input** : Communication Budgets $(B_t)_{t\in\mathbb{N}}$ and UCB Parameter α, $\varepsilon > 0$
2: **Initialization**: $\widehat{S}^{(i)}, S_0^{(i)}$ according to Equations (2) and (3) respectively.
3: $j \leftarrow 0$
4: $A_j = \max\left(\min\{t \geq 0, B_t \geq j\}, \lceil (1+j)^{1+\varepsilon} \rceil\right)$ ▷ Reparametrize the communication budget
5: **for** Time $t \in \mathbb{N}$ **do**
6: | Pull - $\arg\max_{l \in S_i^{(j)}} \left(\widehat{\mu}_l^{(i)}(t-1) + \sqrt{\frac{\alpha \ln(t)}{T_l^{(i)}(t-1)}} \right)$
7: | **if** $t == A_j$ **then** ▷ End of Phase
8: | | $\mathcal{O}_j^{(i)} \leftarrow$ `GetArm`(i,j)
9: | | **if** $\mathcal{O}_j^{(i)} \notin S_i^{(j)}$ **then**
10: | | | $U_{j+1}^{(i)} \leftarrow \arg\max_{l \in \{U_j^{(i)}, L_j^{(i)}\}} (T_l(A_j) - T_l(A_{j-1}))$ ▷ The most played arm
11: | | | $L_{j+1}^{(i)} \leftarrow \mathcal{O}_j^{(i)}$
12: | | | $S_{j+1}^{(i)} \leftarrow \widehat{S}^{(i)} \cup L_{j+1}^{(i)} \cup U_{j+1}^{(i)}$ ▷ Update the set of playing arms
13: | | **else**
14: | | | $S_{j+1}^{(i)} \leftarrow S_j^{(i)}$.
15: | | $j \leftarrow j+1$
16: | | $A_j = \max\left(\min\{t \geq 0, B_t \geq j\}, \lceil (1+j)^{1+\varepsilon} \rceil\right)$ ▷ Reparametrize the communication budget

Algorithm 2 Synchronous Arm Recommendation

1: **procedure** GetArm$((i,j))$ ▷ *Input an agent i and Phase j*
2: | $m \sim P(i,\cdot)$ ▷ *Sample another agent*
3: **return** $\arg\max_{l \in S_m^{(j)}} \left(T_l^{(m)}(A_j) - T_l^{(m)}(A_{j-1}) \right)$ ▷ *Most Played arm in phase j by agent m*

$$g((A_x)_{x\in\mathbb{N}}) = A_{j^*} + \frac{2}{2\alpha-3}\left(\sum_{l \geq \frac{j^*}{2}-1} \frac{A_{2l+1}}{A_{l-1}^3}\right) \text{ where}$$

$$j^* = 2\max\left(A^{-1}\left(\left(N\binom{K}{2}\left(\left\lceil\frac{K}{N}\right\rceil+1\right)\right)^{\frac{1}{(2\alpha-6)}}\right) + 1, \min\left\{j \in \mathbb{N} : \frac{A_j - A_{j-1}}{2+\lceil\frac{K}{N}\rceil} \geq 1 + \frac{4\alpha\log(A_j)}{\Delta_2^2}\right\}\right),$$

where, $A^{-1}(x) = \sup\{y \in \mathbb{N} : A_y \leq x\}$, $\forall x \in \mathbb{R}_+$ *and* log *refers to natural logarithm throughout the paper.*

In Appendix B, we provide some discussion and derive insights from this theorem.

3.3 Proof Sketch

The proof of this theorem is carried out in Appendix C and we describe the main ideas here. We deduce in Proposition 2 that there exists a *freezing* time τ such that, all agents have the best arm by time τ and only recommend the best arm from henceforth, i.e., the set of arms of agents do not change after τ. The technical novelty of our proof is in bounding $\mathbb{E}[A_\tau]$, as this leads to the final regret bound (Proposition 2).

There are two key challenges in bounding this term. First, the choice of arm recommendation is based on the most played arm in the current phase, while the choice of arm to pull is based on samples even in the past phases, as the UCB considers all samples of an arm thus far. If the phase lengths are large (Equation (1) ensures this), Lemma 6 shows that the probability of an agent recommending a sub-optimal arm at the end of a phase is small, irrespective of the number of times it was played till the beginning of the phase. Second, the events that any agent recommends a sub-optimal arm in different phases are not independent, as the reward samples collected by this agent, leading to those decisions are shared. We show in Proposition 3 by establishing that after a random, almost surely finite time (denoted as $\widehat{\tau}_{stab}$ in Appendix C), agents never recommend incorrectly.

3.4 Initialization without Agent IDs

The initialization in Line 2 of Algorithm 1 relies on each agent knowing its identity. However, in many settings, it may be desirable to have algorithms that do not depend on the agent's identity. We outline a simple procedure to fix this (with guarantees) in Appendix N.

4 Asynchronous GosInE Algorithm

A synchronous system is not desirable in many cases as agents could get a large number of message requests during time slots $(A_j)_{j\geq 0}$. Consider an example where the gossip matrix P is a star graph, i.e., for all $i \neq 1$, $P(i,1) = 1$ and $P(1,i) = \frac{1}{N-1}$. In this situation, at

time slots $(A_x)_{x\in\mathbb{N}}$, the central node 1 will receive a (large) $N-1$ different requests for messages, which may be infeasible if agents are bandwidth constrained.

We present an asynchronous algorithm to alleviate this problem. This new algorithm is identical to Algorithm 1 with two main differences - *(i)* each agent chooses the number of time slots it stays in any phase j as a random variable independently of everything else, and *(ii)* when asked for a recommendation, agents recommend the most played arm in the *previous phase*. The first point, ensures that even in the case of the star graph described above, with high probability, eventually, no two agents will pull information in the same time slot. The second point ensures that even though the phase lengths are random, the *quality* of recommendations are good as they are based on large number of samples. We give a pseudo-code in Algorithm 3 in Appendix E, where lines 5 and 18 are new and lines 8 (agents have different phase lengths) and 9 (arm recommendation from previous phase) are modified from Algorithm 1.

Theorem 2. *Suppose in a system of $N \geq 2$ agents connected by a communication matrix P satisfying assumption* **(A.1)** *and $K \geq 2$ arms, each agent runs Algorithm 3, with UCB parameter $\alpha > 3$, $\delta > 0$ and communication budget $(B_t)_{t\in\mathbb{N}}$ and $\varepsilon > 0$ satisfying assumption* **(A.2)**. *Then the regret of any agent $i \in [N]$, after any time $T \in \mathbb{N}$ is bounded by*

$$\mathbb{E}[R_T^{(i)}] \leq \underbrace{\left(\sum_{j=2}^{\lceil \frac{K}{N}\rceil+2} \frac{1}{\Delta_j}\right) 4\alpha \ln(T) + \frac{K}{4} +}_{\text{Collaborative UCB Regret}}$$
$$\underbrace{(1+\delta)\mathbb{E}[A_{2\lfloor 2+\delta\rfloor \tau_{spr}^{(P)}}] + \widehat{g}((A_x)_{x\in\mathbb{N}},\delta)}_{\text{Cost of Asynchronous Infrequent Pairwise Communications}},$$

where $\widehat{g}((A_x)_{x\in\mathbb{N}},\delta) = 2(1+\delta)\left(A_{2\lceil 2+\delta\rceil j^*} + \left(\frac{2}{2\alpha-3}\right)\sum_{l\geq 3}\frac{A_{2l}}{A_{l-1}^3}\right)$, *where j^* given in Theorem 1 and $(A_x)_{x\in\mathbb{N}}$ is given in Equation (1).*

4.1 Proof Sketch

The proof of this theorem is carried out in Appendices F,G and H. In order to prove this, we find it effective to give a more general algorithm (Algorithm 5 in Appendix F) where the agents choose the phase lengths $\mathcal{P}_j$ as a Poisson distributed random variable. This algorithm does not satisfy the budget constraint exactly, but only in expectation, over the randomization used in the algorithm. We analyze this in Theorem 8 stated in Appendix F and proved in Appendix G. The main additional technical challenge is that the phase lengths of different agents are staggered. We crucially use the convexity of the sequence $(A_x)_{x\in\mathbb{N}}$ (Assumption **A.2**) in Proposition 6, along with more involved coupling argument to a rumor spreading process (Proposition 4). The proof of Theorem 2 is a corollary of the proof of Theorem 8 in Appendix H.

5 Lower Bound

In order to state the lower bound, we will restrict ourselves to a class of *consistent policies* (Lai and Robbins, 1985). A policy (or algorithm) is consistent, if for any agent $i \in [N]$, and any sub-optimal arm $l \in \{2,\cdots,K\}$, the expected number of times agent i plays arm l up-to time $t \in \mathbb{N}$ (denoted by $T_l^{(i)}(t)$) satisfies for all $a > 0$, $\lim_{t\to\infty} \frac{\mathbb{E}[T_l^{(i)}(t)]}{t^a} = 0$.

Theorem 3. *The regret of any agent $i \in [N]$ after playing arms for T times under any consistent policy played by the agents and any communication matrix P satisfies*

$$\liminf_{T\to\infty} \frac{\mathbb{E}[R_T^{(i)}]}{\ln(NT)} \geq \left(\frac{1}{N}\sum_{j=2}^{K} \frac{\Delta_j}{KL(\mu_j,\mu_1)}\right), \quad (5)$$

where for any $a, b \in [0,1]$, $KL(a,b)$ is the Kullback-Leibler distance between two Bernoulli distributions with mean a and b.

The proof of the theorem is carried out in Appendix J. The proof of this lower bound is based on a system where there are no communication constraints.

6 Insights

1. Insensitivity to Communication Constraints - The following corollary follows directly from Theorems 1 and 2.

Corollary 4. *Suppose in a system of $N \geq 2$ agents each running Algorithm 1 or 3 with parameters satisfying conditions in Theorems 1 and 2 respectively. Then, for every agent $i \in [N]$ and time $T \in \mathbb{N}$,*

$$\limsup_{T\to\infty} \frac{E[R_T^{(i)}]}{\ln(T)} \leq \left(\sum_{j=2}^{\lceil \frac{K}{N}\rceil+2} \frac{4\alpha}{\Delta_j}\right).$$

Thus, as long as the gossip matrix P is connected (Assumption **A.1**) and the communication budget over a horizon of T is at-least $\Omega(\log(T))$, (Assumption **A.2**), the asymptotic regret of any agent, is insensitive to P and the communication budget.

2. Benefit of Collaboration - As an example, consider a system where $K = N$ and arm-means such that

$\forall j \in \{2, \cdots, K\}$, $\Delta_j := \Delta > 0$. Let $\mathbf{\Pi}$ be any consistent policy for the agents in the sense of Theorem 3. Then Equation (5) and Corollary 4 implies that $\sup_{\boldsymbol{\pi}} \limsup_{T\to\infty} \frac{\mathbb{E}[R_T^{(i)}]}{\mathbb{E}_{\boldsymbol{\pi}}[R_T^{(i)}]} \leq \frac{4\alpha}{\mu_1(1-\mu_1)}$, where in the numerator is the regret obtained by our algorithms and the denominator is by the policy $\boldsymbol{\pi}$. As ratio of asymptotic regret in our algorithm and the lower bound is a constant *independent of the size of the system*, (does not grow with N), our algorithms benefit from collaboration. Recall that the lower bound is obtained from the full interaction setting where all agents communicate with every other agent, after every arm pull while in our model, every agent pulls information, a total of at most o(T) times over a time horizon of T. Thus, we observe that, despite communication constraints, any agent in our algorithm performs nearly as good as the best possible algorithm when agents have no communication constraints, i.e., the regret ratio is a constant independent of N.

3. Impact of Gossip Matrix P - The second order constant term in the regret bounds in Theorems 1 and 2 provides a way of quantifying the impact of P, based on its conductance, which we define now. Given an undirected finite graph G on vertex set V, denote for any vertex $u \in V$, $\deg(u)$ to be the degree of vertex u in G. For any set $H \subseteq V$, denote by $\mathrm{Vol}(H) = \sum_{u\in H} \deg(u)$. For any two sets $H_1, H_2 \subseteq V$, denote by $\mathrm{Cut}(H_1, H_2)$, to be the number of edges in G with one end in H_1 and the other in H_2. The conductance of G, denoted by ϕ is defined as

$$\phi := \min_{H\subset V: 0<\mathrm{Vol}(H)\leq \mathrm{Vol}(V)/2} \frac{\mathrm{Cut}(H, V\setminus H)}{\mathrm{Vol}(H)}.$$

In corollaries 17 and 18, we prove the intuitive fact that if the conductance of the gossip matrix is higher, then the regret (the second order constant term) is lower. In order to derive some intuition, we consider two examples here - one wherein the N agents are connected by a complete graph, and one wherein they are connected by the ring graph. The conductance of the complete graph is $\frac{N}{2(N-1)}$, while that of the ring graph is $\frac{2}{N}$. Let the communication budget scale polynomially in both systems, i.e., $B_t = \lfloor t^{1/\beta} \rfloor$, for some $\beta > 1$, for all $t \geq 1$. The cost of communications (as captured in Corollary 18) scales as $(4C\log(N))^\beta$ for the complete graph, but scales as $(4C\log(N)N)^\beta$ in the ring graph. This shows the reduction in regret that is possible by a 'more' connected gossip matrix, where the regret is reduced from order $(N\log(N))^\beta$ to $(\log(N))^\beta$ on moving from the ring graph to the complete graph. This is also demonstrated empirically in Figures 1 and 2.

4. Regret/Communication Trade-off - For a fixed problem instance and gossip matrix P, reducing the total number of information pulls, i.e., reducing the rate of growth of $(B_x)_{x\in\mathbb{N}}$ increases the per-agent regret. This can be inferred by examining the cost of communications in Equation (4), which we state in the Corollary 20 in Appendix M. This corollary makes precise the qualitative fact that if agents are allowed more communication budget, then they experience lesser regret. We demonstrate this empirically in Figure 3.

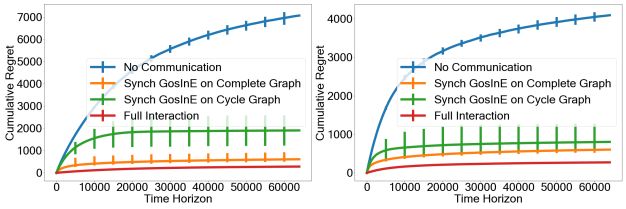

Figure 1: (N, K) are $(25, 75)$ and $(15, 50)$ respectively.

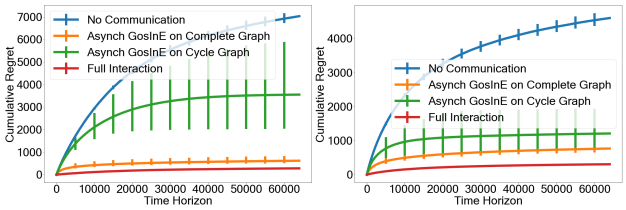

Figure 2: (N, K) are $(25, 75)$ and $(15, 50)$ respectively.

7 Numerical Results

We evaluate our algorithm and the insights empirically. Each plot is the regret averaged over all agents, produced after 30 and 100 random runs for Algorithms 1 and Algorithm 3 (with $\delta = 0.5$) respectively, along with 95% confidence intervals. We also plot the two benchmarks of no interaction among agents (where a single agent is running the UCB-4 algorithm of (Auer et al., 2002)) and the system corresponding to complete interaction, where all agents are playing the UCB-4 algorithm with entire system history of all arms pulled and rewards obtained by all agents as described in Section 5.

Synthetic Experiments - We consider a synthetic setup with $\Delta = 0.1$, $\mu_1 = 0.95, \mu_2 = 0.85$, rest of the arm means sampled uniformly in $(0, 0.85]$. In Figures 1 and 2, we consider the impact of gossip matrix by fixing the communication budget $B_t = \lfloor t^{1/3} \rfloor$ $(A_x = x^3)$ and varying P to be the complete and cycle graph among agents. We see that our algorithms are effective in leveraging collaboration in both settings and experiences a lower regret in the complete graph case as opposed to the cycle graph, as predicted by our insights.

In Figure 3, we compare the effect of communication

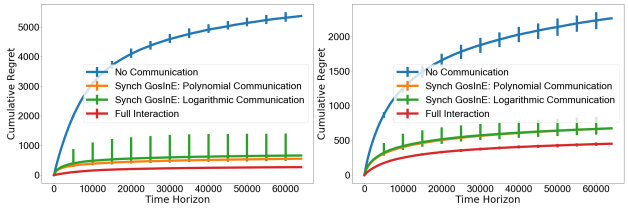

Figure 3: (N, K) as $(20, 70)$ and $(5, 20)$ and the graphs are complete and cycle respectively.

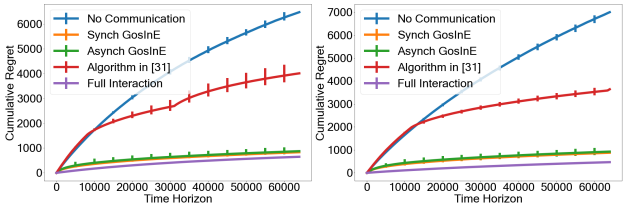

Figure 4: (N, K) are $(10, 30)$ and $(15, 40)$ respectively.

budget by considering two scenarios - polynomial budget $B_t = \lfloor t^{1/3} \rfloor$ ($A_x = x^3$) and logarithmic budget $B_t = \lfloor \log_2(t) \rfloor$ ($A_x = 2^x$). We see that even under a logarithmic communication budget, our algorithms achieve significant regret reduction.

Real Data - In Figure 4, we run our Algorithms on MovieLens data (Harper and Konstan, 2016) using the methodology in (Sankararaman et al., 2019). This dataset contains $6k$ movies rated by $4k$ users. We treat the movies as arms and estimate the arm-means from the data by averaging the ratings of a section of similar users (same age, gender and occupation and have rated at-least 30 movies). We further select only those movies that have at least 30 ratings by users in the chosen user category. We estimate the missing entries in the sub-matrix (of selected users and movies) using matrix completion (Hastie et al., 2015) and choose a random set of 30 and 40 movies, in Figure 4. We compare against (Sankararaman et al., 2019) (hyperparameter $\varepsilon = 0.1$) for the setting of complete graph among agents and communication budget $B_t = \lfloor t^{1/3} \rfloor$. We see that in all settings, our algorithm has superior performance and strongly benefits from limited collaboration.

8 Related Work

The closest to our work is (Sankararaman et al., 2019) which introduced a model similar to ours. However, the present paper improves on the algorithm in (Sankararaman et al., 2019) in three aspects: *(i)* our algorithm can handle any gossip matrix P, while that of (Sankararaman et al., 2019) can only handle complete graphs and *(ii)*, the algorithm in (Sankararaman et al., 2019), needs as an input, a lower bound on the arm gap between the best and the second best arm, while our algorithms do not require any such knowledge and *(iii)*, our regret scaling is superior even on complete graphs.

The multi-agent MAB was first introduced in the non-stochastic setting in (Awerbuch and Kleinberg, 2005) and further developed in (Cesa-Bianchi et al., 2019). However, there was no notion of communication budgets in these models. Subsequently, (Kanade et al., 2012) considered the regret/communication trade-off in the non-stochastic setting, different from our stochastic MAB model. In the stochastic setting, the papers of (Chakraborty et al., 2017),(Buccapatnam et al., 2015), (Martínez-Rubio et al., 2018), (Kolla et al., 2018), (Landgren et al., 2016) consider a collaborative multi agent model where agents minimize individual regret in a decentralized manner. In these models, communications is not an active decision made by agents, rather agents can observe neighbor's actions and are, therefore, different from our setup, where agents actively choose to communicate depending on a budget. The papers of (Hillel et al., 2013) and (Szörényi et al., 2013) study the benefit of collaboration in reducing simple regret, unlike the cumulative regret considered in our paper. The paper of (Korda et al., 2016) considers a distributed version of contextual bandits, in which agents could share information, whose length grows with time and thus different from our setup. There has also been a lot of recent interest in 'competitive' multi-agent bandits ((Anandkumar et al., 2011), (Liu et al., 2013), (Rosenski et al., 2016), (Avner and Mannor, 2014), (Kalathil et al., 2014), (Bistritz and Leshem, 2018),(Mansour et al., 2017),(Liu et al., 2019)), where if multiple agents choose the same arm in a time slot, then they experience a 'collision' and receive small reward (only a subset (possibly empty) gets a reward). This differs from our setup where even on collision, agents receive independent rewards.

9 Conclusions

We introduced novel algorithms for multi agent MAB, where agents play from a subset of arms and recommend arm-IDs. Our algorithms leverage collaboration effectively and in particular, its performance (asymptotic regret) is insensitive to the communication constraints. Furthermore, our algorithm exhibits a regret communication trade-off, namely achieves lower regret (finite time) with increased communications (budget or conductance of P), which we characterize through explicit bounds.

Acknowledgements

This work was partially supported by ONR Grant N00014-19-1-2566, NSF Grant SATC 1704778, ARO grant W911NF-17-1-0359 and the NSA SoS Lablet H98230-18-D-0007. AS also thanks François Baccelli for the support and generous funding through the Simons Foundation Grant (#197892) awarded to the University of Texas at Austin.

References

Animashree Anandkumar, Nithin Michael, Ao Kevin Tang, and Ananthram Swami. Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*, 29(4):731–745, 2011.

Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. 2010.

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

Orly Avner and Shie Mannor. Concurrent bandits and cognitive radio networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 66–81. Springer, 2014.

Orly Avner and Shie Mannor. Multi-user lax communications: a multi-armed bandit approach. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2016.

Baruch Awerbuch and Robert D Kleinberg. Competitive collaborative learning. In *International Conference on Computational Learning Theory*, pages 233–248. Springer, 2005.

Ilai Bistritz and Amir Leshem. Distributed multi-player bandits-a game of thrones approach. In *Advances in Neural Information Processing Systems*, pages 7222–7232, 2018.

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

Swapna Buccapatnam, Jian Tan, and Li Zhang. Information sharing in distributed stochastic bandits. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 2605–2613. IEEE, 2015.

Nicolo Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Delay and cooperation in nonstochastic bandits. *The Journal of Machine Learning Research*, 20(1):613–650, 2019.

Deepayan Chakrabarti, Ravi Kumar, Filip Radlinski, and Eli Upfal. Mortal multi-armed bandits. In *Advances in neural information processing systems*, pages 273–280, 2009.

Mithun Chakraborty, Kai Yee Phoebe Chua, Sanmay Das, and Brendan Juba. Coordinated versus decentralized exploration in multi-agent multi-armed bandits. In *IJCAI*, pages 164–170, 2017.

Flavio Chierichetti, Silvio Lattanzi, and Alessandro Panconesi. Almost tight bounds for rumour spreading with conductance. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 399–408. ACM, 2010.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4): 19, 2016.

Trevor Hastie, Rahul Mazumder, Jason D Lee, and Reza Zadeh. Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, 16(1):3367–3402, 2015.

Eshcar Hillel, Zohar S Karnin, Tomer Koren, Ronny Lempel, and Oren Somekh. Distributed exploration in multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 854–862, 2013.

Dileep Kalathil, Naumaan Nayyar, and Rahul Jain. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, 2014.

Varun Kanade, Zhenming Liu, and Bozidar Radunovic. Distributed non-stochastic experts. In *Advances in Neural Information Processing Systems*, pages 260–268, 2012.

Ravi Kumar Kolla, Krishna Jagannathan, and Aditya Gopalan. Collaborative learning of stochastic bandits over a social network. *IEEE/ACM Trans. Netw.*, 26(4):1782–1795, August 2018. ISSN 1063-6692. doi: 10.1109/TNET.2018.2852361. URL https://doi.org/10.1109/TNET.2018.2852361.

Nathan Korda, Balázs Szörényi, and Li Shuai. Distributed clustering of linear bandits in peer to peer networks. In *Journal of machine learning research workshop and conference proceedings*, volume 48, pages 1301–1309. International Machine Learning Societ, 2016.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 167–172. IEEE, 2016.

Tor Lattimore and Csaba Szepesvári. Bandit algorithms. *preprint*, 2018.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670. ACM, 2010.

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *arXiv preprint arXiv:1603.06560*, 2016.

Haoyang Liu, Keqin Liu, Qing Zhao, et al. Learning in a changing world: Restless multi-armed bandit with unknown dynamics. *IEEE Trans. Information Theory*, 59(3):1902–1916, 2013.

Lydia T Liu, Horia Mania, and Michael I Jordan. Competing bandits in matching markets. *arXiv preprint arXiv:1906.05363*, 2019.

Yishay Mansour, Aleksandrs Slivkins, and Zhiwei Steven Wu. Competing bandits: Learning under competition. *arXiv preprint arXiv:1702.08533*, 2017.

David Martínez-Rubio, Varun Kanade, and Patrick Rebeschini. Decentralized cooperative stochastic multi-armed bandits. *arXiv preprint arXiv:1810.04468*, 2018.

Jonathan Rosenski, Ohad Shamir, and Liran Szlak. Multi-player bandits–a musical chairs approach. In *International Conference on Machine Learning*, pages 155–163, 2016.

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.

Devavrat Shah. Gossip algorithms. *Foundations and Trends® in Networking*, 3(1):1–125, 2009.

Balázs Szörényi, Róbert Busa-Fekete, István Hegedűs, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Gossip-based distributed stochastic bandit algorithms. In *Journal of Machine Learning Research Workshop and Conference Proceedings*, volume 2, pages 1056–1064. International Machine Learning Societ, 2013.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

Yisong Yue and Thorsten Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1201–1208. ACM, 2009.