Collaborative Regret Minimization for Piecewise-Stationary Multi-armed Bandit

Xiaotong Cheng* and Setareh Maghsudi*†
*Department of Computer Science, University of Tübingen, Germany
†Fraunhofer Heinrich Hertz Institute, Berlin, Germany

***Abstract*—We study a structured multi-agent multi-armed bandit (MAB) problem in a non-stationary environment. Agents in the system face the same piecewise-stationary MAB problem. Consequently, they share information so far allowed by the graph links to accelerate learning. Each agent aims at minimizing the regret of sequential decision-making, which is the expected total loss of not playing the optimal arm at each time step. We propose a solution to that problem, RBO-Coop-UCB, which involves an efficient multi-agent UCB algorithm with a Bayesian change point detector as its core, enhanced by a collaboration mechanism for performance improvement. Theoretically, we establish an upper bound for the expected group regret of RBO-Coop-UCB. Numerical experiments on real-world datasets demonstrate that our proposed method outperforms the state-of-the-art algorithms.**

***Index Terms*—Change point detection, distributed learning, multi-armed bandit, multi-agent cooperation**

I. Introduction

Multi-armed bandit (MAB) is a sequential optimization problem, which is applicable in several real-world application areas such as online advertisement [1], wireless communications [2], and personalized medicine [3]. In the seminal MAB, an agent selects an action at each round, aiming to develop an action selection policy to minimize its cumulative regret over all rounds. The majority of research on the MAB problem focuses on single-agent policies, neglecting the social elements of the applications of the MAB framework; nevertheless, the ever-increasing significance of networked systems and large-scale information networks encourages studying multi-agent MAB problems as a framework to optimize decision-making in distributed systems. Reference [4] studies the distributed multi-agent MAB (MAMAB) problem and proposes an online indexing policy based on distributed bipartite matching. In [5], the authors introduce the notion of sociability to model the likelihood probabilities that one agent observes its neighbors' choices and rewards in the network graph.

Most previous research on MAB focuses on either the stochastic- or adversarial environment, undermining the "intermediate" setting, where the reward distribution of each arm is piecewise-constant and shifts at some unknown time steps called the *change points*. The cutting-edge research on piecewise-stationary MAB includes two categories: (i) Passively adaptive methods, where the agent makes decisions based on the most recent observations while unaware of the underlying distribution changes [6]–[8]; and (ii) Actively adaptive methods, where a change point detector subroutine to monitor the reward distributions are incorporated [9]–[11]. Several studies show the superior performance of the latter category over the former [11], [12]; As such, we focus on active adaptation.

The research works mentioned above study multi-agent MAB and non-stationary MAB; nevertheless, the research community neglects the combination of the two problems to a great extent. On the one hand, multi-agent MAB problems consider the social components and distributed structure of the applications of the MAB framework, which is essential in recent technological development [13]. On the other hand, the assumption of a stationary environment rarely holds in practice. Piecewise-stationary bandit algorithms address the challenges caused by non-stationary environments. Therefore, studying the two problems simultaneously is crucial.

In this paper, we take the first step to unify these two independent strands of bandit research by formulating a piecewise-stationary MAMAB problem. Our main contributions are as follows:

- We propose an efficient running consensus algorithm for piecewise-stationary MAMAB, called *RBO-Coop-UCB*. The method integrates a change point detector, namely, restarted Bayesian online change point detector (RBOCPD) [14] and a MAMAB algorithm in its core.
- We incorporate an effective cooperation mechanism in RBO-Coop-UCB, not only in arm selection but also in the restart decision part. The cooperation framework is generic and applicable to enhance various actively adaptive policies in piecewise-stationary bandit problems.
- For any networked multi-agent systems, we establish the group regret bound $\mathcal{O}(KNM\log T + K\sqrt{MT\log T})$, where K, M, N and T respectively denote the number of agents, arms, change points and time steps. To the best of our knowledge, this is the first regret-bound analysis for bandit policies that include RBOCPD.
- By experiments on a real-world dataset, we show that our proposed algorithm, RBO-Coop-UCB, outperforms the state-of-the-art policies.

II. Problem Formulation

We consider a K agent piecewise-stationary MAB problem. We use $\mathcal{M}$ to denote the time-invariant action set and $\mathcal{K}$ to denote the agent set. Besides, f_t^m is the reward distribution

This work was supported by the German Research Foundation (DFG) under Grant MA 7111/6-1 and MA 7111/7-1, and by the German Federal Ministry of Education and Research (BMBF) under Grant 01IS20051.

of arm m at time t with mean μ_t^m. At each time step, each agent $k \in \mathcal{K}$ pulls one arm $m \in \mathcal{M}$ and obtains a reward sampled from f_t^m. The agents form a network modeled by an undirected graph $\mathcal{G}(\mathcal{K}, \mathcal{E})$, where $\mathcal{E} = \{e(k,j)\}_{k,j \in \mathcal{K}}$ is the edge set. Agents k and j are neighbors if $e(k,j) \in \mathcal{E}$ and $e(k,k) \in \mathcal{E}$, $\forall k \in \mathcal{K}$. Such a pair can observe each others' selected arms and sampling reward. We use I_t^k to show the action of agent k and X_t^m is the sampling reward of arm m at time t.

Assumption 1. *[14] Let N denote the overall number of piecewise-stationary segments observed until time T, $N = 1 + \sum_{t=1}^{T-1} \mathbb{1}\{f_t^m \neq f_{t+1}^m$ for some $m \in \mathcal{M}\}$. The reward distributions of arms are piecewise-stationary Bernoulli processes $\mathcal{B}(\mu_t^m)$ such that there exists a non-decreasing change point sequence $(\nu_n)_{n \in [1,N-1]} \in \mathbb{N}^{N-1}$ verifying*

$$\begin{cases} \forall n \in [1, N-1],\, \forall t \in [\nu_n, \nu_{n+1}),\, \forall m \in \mathcal{M}, \quad \mu_t^m = \mu_n^m \\ \nu_1 = 1 < \nu_2 < \ldots < \nu_N = T. \end{cases}$$

Each agent k measures its performance by its (dynamic) regret, i.e., the cumulative difference between the expected reward obtained by the optimal arm I_t^* at time t, and the expected reward obtained by action I_t^k selected by agent k

$$R_T^k = \sum_{t=1}^{T} [\mathbb{E}(X_t^{I_t^*}) - \mathbb{E}(X_t^{I_t^k})]. \tag{1}$$

In the multi-agent setting, we study the network performance in terms of the regret experienced by the entire network $R_T = K \sum_{t=1}^{T} \mathbb{E}(X_t^{I_t^*}) - \sum_{k=1}^{K} \sum_{t=1}^{T} \mathbb{E}(X_t^{I_t^k})$.

III. The RBO-Coop-UCB Algorithm

Our decision-making policy, *RBO-Coop-UCB*, combines a network UCB algorithm with a change-point detector running on each arm, which is based on Bayesian change point detection strategy. Our work is closely related to the studies on multi-agent cooperative bandit algorithms [5], [13], [15] and the Bayesian change point detection [14]. It has three building blocks: (1) A cooperative UCB structure, which guides the systems to learn the optimal arm in each piecewise-stationary segment; (2) A change point detector [14] described in Section III-A; and (3) A cooperation mechanism for change point detection to filter false alarms.

A. Restarted Bayesian Online Change Point Detector

Let r_t be the number of time steps since the last change point (current run length), and $\boldsymbol{X}_{1:t} = (X_1, \ldots, X_t)$ the data observed so far, which is generated from the piecewise-stationary Bernoulli process. The seminal Bayesian strategy computes $p(r_t|\boldsymbol{X}_{1:t})$ as the posterior distribution over the current runlength r_t [16] and uses the message-passing algorithm to recursively infer the run length distribution $p(r_t|\boldsymbol{X}_{1:t}) \propto \sum_{r_{t-1}} p(r_t|r_{t-1}) p(X_t|r_{t-1}, \boldsymbol{X}_{1:t-1}) p(r_{t-1}|\boldsymbol{X}_{1:t-1})$. In RBOCPD, it assumes that each possible value of r_t corresponds to one specific run length forecaster. The loss $l_{s,t}$ of the forecaster s at time t, which is related to $p(X_t|r_{t-1}, \boldsymbol{X}_{1:t-1})$, follows as [14]

$$l_{s:t} = -\log Lp(X_t|\boldsymbol{X}_{s:t-1}), \tag{2}$$

where $Lp(\cdot)$ is the Laplace predictor. $Lp(X_{t+1}|\boldsymbol{X}_{s:t})$ takes a sequence $\boldsymbol{X}_{s:t} \in \{0,1\}^{n_{s:t}}$ as input with $n_{s:t} = t - s + 1$ being the length of $\boldsymbol{X}_{s:t}$. It predicts the value of the next observation $X_{t+1} \in \{0,1\}$ as

$$Lp(X_{t+1}|\boldsymbol{X}_{s:t}) = \begin{cases} \frac{\sum_{i=s}^{t} X_i + 1}{n_{s:t}+2}, & \text{if } X_{t+1} = 1, \\ \frac{\sum_{i=s}^{t} (1 - X_i) + 1}{n_{s:t}+2}, & \text{if } X_{t+1} = 0, \end{cases} \tag{3}$$

where $\forall X \in \{0,1\}, Lp(X|\phi) = \frac{1}{2}$ corresponds to the uniform prior given to the process generating μ_c.

The weight $\vartheta_{r,s,t}$ of forecaster s at time t for starting time r is the posterior $\vartheta_{r,s,t} = p(r_t = t - s|\boldsymbol{X}_{s:t})$, where

$$\vartheta_{r,s,t} = \begin{cases} \frac{\eta_{r,s,t}}{\eta_{r,s,t-1}} \exp(-l_{s,t}) \vartheta_{r,s,t-1}, & \forall s < t \\ \eta_{r,t,t} \times \mathcal{V}_{r,t-1}, & s = t \end{cases} \tag{4}$$

by using the hyperparameter $\eta_{r,s,t}$, which is related to $p(r_t|r_{t-1})$. $\mathcal{V}_{r,t-1}$ is the initial weight $\mathcal{V}_{r:t-1} = \exp\left(-\hat{L}_{r:t-1}\right)$ for some starting time r, where $\hat{L}_{r:t-1} = \sum_{r'=r}^{t-1} l_{r':t-1}$ is the cumulative loss incurred by the forecaster r from r until $t-1$. Based on (2), the cumulative loss $\hat{L}_{r:t-1} = \sum_{r'=r}^{t-1} -\log Lp(x_{t-1}|\boldsymbol{x}_{r':t-2})$. The change point detection (restart) decisions are made based on the forecaster weight. For any starting time $r \leq t$,

$$\textbf{Restart}_{r:t} = \mathbb{1}\{\exists s \in (r,t] : \vartheta_{r,s,t} > \vartheta_{r,r,t}\}. \tag{5}$$

The intuition behind (5) is the following. At each step without change, the forecaster distribution concentrates around the forecaster launched at the starting time r. Thus, if distribution $\vartheta_{r,s,t}$ changes, then a certain change point appears.

The restarted Bayesian online change point detector is summarized in Algorithm 1

Algorithm 1 RBOCPD [14]: RBO($\boldsymbol{X}_{1:t}, \eta_{1,s,t}$)

Require: Observations $\boldsymbol{X}_{1:t}$ and hyperparameter $\eta_{1,s,t}$.
$r \leftarrow 1, \vartheta_{r,1,1} \leftarrow 1, \eta_{r,1,1} \leftarrow 1$.
for $i = 1, 2, \ldots, t$ **do**
  Calculate $\vartheta_{r,s,i}$ of each $s \in (r, i]$ according to (4).
  Calculate **Restart**$_{r:i}$ according to (5).
  if Restart$_{r:i} = 1$ **then**
    return True
  end if
end for
return False

B. RBO-Coop-UCB Algorithm

Our proposed algorithm, *RBO-Coop-UCB*, is a Network UCB algorithm that allows for some restarts during the decision-making process. The agents run the RBO-Coop-UCB policy in parallel. To guarantee sufficient samples from each arm for change point detection, each arm will be selected several

times in the forced exploration steps. Let I_t^k and $X_t^{k,m}$ denote the selected arm and the reward of agent k at time t, respectively and $X_t^{k,m}$ be the i.i.d copies of X_t^m. The total number of times that agent k observes rewards from option m is $N_t^{k,m} = \sum_{t'=1}^{t} \sum_{j=1}^{K} \mathbb{1}\{I_{t'}^j = m\}\mathbb{1}\{e(k,j) \in \mathcal{E}\}$. The empirical rewards of agent k by pulling arm m at time t yields

$$\hat{\mu}_t^{k,m} = \frac{S_t^{k,m}}{N_t^{k,m}}, \tag{6}$$

where $S_t^{k,m} = \sum_{t'=1}^{T} \sum_{j=1}^{K} X_{t'}^{k,m} \mathbb{1}\{I_{t'}^k = m\}\mathbb{1}\{e(k,j) \in \mathcal{E}\}$ is the accumulated reward from option m in t rounds. At every step, if an agent is in a forced exploration phase, it selects each arm several times to ensure sufficient number of observations for each arm; otherwise, it chooses an arm according to the sampling rule described in Definition 1.

Definition 1. *At time t, agent k follows the sampling rule*

$$\mathbb{1}\{I_t^k = m\} = \begin{cases} 1, & \text{if } Q_t^{k,m} = \max\{Q_t^{k,1}, \ldots, Q_t^{k,M}\} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

with $Q_t^{k,m} = \hat{\mu}_t^{k,m} + C_t^{k,m}$, $C_t^{k,m} = \sqrt{\frac{\xi(\alpha^k+1)\log(t-\tau^k)}{N_t^{k,m}}}$, and τ^k is the last change point detected by agent k. $\xi \in (0,1]$ is a constant. Besides, $\alpha^k = \frac{\eta_k - \eta_k^{avg}}{\eta_k}$ is an agent-based parameter with η_k being the number of k's neighbors. Finally, $\eta_k^{avg} = \frac{1}{\eta_k}\sum_{e(k,j)\in\mathcal{E}} \eta_j$ is the average degree of neighbors. We assume that $\forall k \in \mathcal{K}$, $\eta_k^{avg} \leq 2\eta_k$, therefore $\forall k \in \mathcal{K}$, $\alpha_k \in (-1, 1)$.

Remark 1. *Different values of α^k imply different exploration rates for the agents. Those with more neighbors benefit from more observations and thus less uncertainty for the expected reward estimation. That increases their exploitation potential and reduces the usefulness of the broadcast information, which, in turn, decreases the neighbors' exploitation potential [15]. To improve the group performance by regulating the exploitation potential across the network, we propose the heterogeneous explore-exploit strategies with sampling rule in Definition 1.*

For each agent k, the set $\mathcal{X}_t^{k,m} = \boldsymbol{X}_{\tau^k:N_t^{k,m}}$ contains all of the observed feedback from arm m since the last change point τ^k. Note that, at each step t, the feedback includes not only the sampling reward $X_t^{k,m}$ of agent k but also those of its neighbors, which agent k collects as $\boldsymbol{X}_t^{k,m}$. The agent uses $\mathcal{X}_t^{k,m}$ to run the RBOCPD (Algorithm 1). Each agent k will receive a binary restart signal $r_t^{k,m}$ afterwards, $r_t^{k,m} = 1$ when there is a change point and zero otherwise. The restart signal $r_t^{k,m}$ is calculated according to (5). According to the cooperation mechanism described in Definition 2, each agent observes its neighbors' restart signals before deciding about the restart.

Definition 2. *Each agent has different observation set (size and value) as its neighbors. To reduce the missed detections caused by asynchronous detection, we design an effective cooperation mechanism for the restart decision. It has a restart memory time window that records the previous restart of the agents' neighbors in a short time d. The cooperation restart detection considers the majority voting of restart among neighbors in that period,*

$$\sum_{j\in\mathcal{N}_k} \mathbb{1}\{\exists i \in [N_{t-d}^{j,m}, N_t^{j,m}], r_i^{j,m} > 0\} \geq \lceil \frac{\eta_k}{2} \rceil$$
$$\rightarrow \textbf{\textit{Restart}}_t^k = \textbf{\textit{True}}. \tag{8}$$

Hence the slower detector receives the restart information from the faster ones to avoid missing change points.

Remark 2. *The length of restart memory time window d depends on the detection delay of RBOCPD, as described by Theorem 2. Although each agent maintains a change point detector with its own observations, all detectors follow the same principle. Therefore, for every change point, the maximum detection time difference is bounded. Thus, setting the restart memory time window d based on the time delay is guaranteed to include all possible correct detections and does not increase the regret.*

Algorithm 2 RBO-Coop-UCB $\forall k \in \mathcal{K}$

Initialization $\forall m \in \mathcal{M}$, $\mathcal{X}^{k,m} \leftarrow \phi$; $N_0^{k,m} \leftarrow 0$; $S_0^{k,m} \leftarrow 0$; $\tau^k \leftarrow 0$.
for $t = 1, 2, \cdots, T$ **do**
  if $(t - \tau^k) \mod \lfloor \frac{M}{p} \rfloor \in \mathcal{M}$ **then**
    Select arm $I_t^k \leftarrow (t - \tau^k) \mod \lfloor \frac{M}{p} \rfloor$;
  else
    Select arm I_t^k according to sampling rule (7).
  end if
  Play arm I_t^k and receive the reward X_t^{k,I_t^k}.
  Observe neighbors' option and rewards.
  Update $\hat{\mu}_t^{k,m}$ according to (6).
  $\mathcal{X}^{k,m} = \mathcal{X}^{k,m} \cup \{X_t^{j,m}\}$, if $j \in \mathcal{N}_k$, $\mathbb{1}\{I_t^j = m\} = 1$.
  $r_t^{k,m} = \text{RBO}_k(\mathcal{X}^{k,m}, \eta_{\tau^k,s,N_t^{k,m}})$ (from Algorithm 1).
  if Restart$_t^k$ from (8) **then**
    $\tau^k \leftarrow t$, $\forall m \in \mathcal{M}$, $\mathcal{X}^{k,m} \leftarrow \phi$; $N_t^{k,m} \leftarrow 0$; $S_t^{k,m} \leftarrow 0$ (restart agent k's UCB)
  end if
end for

IV. Performance Analysis

In this section, we first overview the performance guarantees for the Restarted Bayesian online change point detector [14]. Based on that, we analyze the regret bound.

Theorem 1 (False alarm rate)**.** *Assume that $\boldsymbol{X}_{r:t} \sim \mathcal{B}(\mu)$. Let $\alpha > 1$. If $\eta_{r,s,t}$ is small enough such that [14] $\forall t \in [r, \nu_n), s \in (r, t]$:,$\eta_{r,s,t} < \frac{\sqrt{n_{r:s-1} \times n_{s:t}}}{10(n_{r:t}+1)} \times (\frac{\log 2 \log^4(\alpha)\delta^2}{4n_{r:t}\log^2(n_{r:t})\log(\alpha n_{r:s-1})\log(n_{r:s-1})\log(\alpha n_{s:t})\log(n_{s:t})})^{\alpha}$, then with probability higher than $1 - \delta$, no false alarm occurs in the interval $[r, \nu_n)$:*

$$\mathbb{P}_\theta\{\exists t \in [r, \nu_n), \textbf{\textit{Restart}}_{r:t} = 1\} \leq \delta. \tag{9}$$

Definition 3 (Relative gap $\Delta_{r,s,t}$)**.** *Let $\Delta \in [0,1]$. The relative gap $\Delta_{r,s,t}$ for the forecaster s at time t takes the following form (depending on the position of s) [14]:*

$$\Delta_{r,s,t} = \left(\frac{n_{r:\nu_n-1}}{n_{r:s-1}}\mathbb{1}\{\nu_n \le s \le t\} + \frac{n_{\nu_n:t}}{n_{s:t}}\mathbb{1}\{s < \nu_n\}\right)\Delta$$

Theorem 2 (Detection delay)**.** *Let $\boldsymbol{x}_{r:\nu_n-1} \sim \mathcal{B}(\mu_1), \boldsymbol{x}_{\nu_n:t} \sim \mathcal{B}(\mu_2)$ and $f_{r,s,t} = \log n_{r:s} + \log n_{s:t+1} - \frac{1}{2}\log n_{r:t} + \frac{9}{8}$. Also, $\Delta = |\mu_1 - \mu_2|$ is the change point gap. If $\eta_{r,s,t}$ is large enough such that [14]*

$$\eta_{r,s,t} > \exp\left(-2n_{r,s-1}(\Delta_{r,s,t} - \mathcal{C}_{r,s,t,\delta})^2 + f_{r,s,t}\right), \quad (10)$$

Then the change point ν_n is detected (with a probability at least $1-\delta$) with a delay not exceeding $\mathcal{D}_{\Delta,r,\nu_n}$ such that

$$\mathcal{D}_{\Delta,r,\nu_n} = \min\{d \in \mathbb{N}^* : d > \frac{(1 - \frac{\mathcal{C}_{r,\nu_n,d+\nu_n-1,\delta}}{\Delta})^{-2}}{2\Delta^2} \times \frac{-\log \eta_{r,\nu_n,d+\nu_n-1} + f_{r,\nu_n,d+\nu_n-1}}{1 + \frac{\log \eta_{r,\nu_n,d+\nu_n-1} - f_{r,\nu_n,d+\nu_n-1}}{2n_{r,\nu_n-1}(\Delta - \mathcal{C}_{r,\nu_n,d+\nu_n-1,\delta})^2}}\}, \quad (11)$$

where

$$\mathcal{C}_{r,s,t,\delta} = \frac{\sqrt{2}}{2}\Bigg(\sqrt{\frac{1+\frac{1}{n_{r:s-1}}}{n_{r:s-1}}\log\left(\frac{2\sqrt{n_{r:s}}}{\delta}\right)} + \sqrt{\frac{1+\frac{1}{n_{s:t}}}{n_{s:t}}\log\left(\frac{2n_{r:t}\sqrt{n_{s:t}+1}\log^2(n_{r:t})}{\log(2)\delta}\right)}\Bigg). \quad (12)$$

In the following we first propose an assumption and then we present the regret analysis.

Assumption 2. *Define $d_n^{k,m} = \lceil \frac{M}{p}\mathcal{D}_{\Delta,(\nu_{n-1}+d_{n-1}^{k,m}),\nu_n} + \frac{M}{p}\rceil$, where $\mathcal{D}_{\Delta,r,\nu_n}$, $\mathcal{C}_{r,\nu_n,d+\nu_n-1,\delta}$, $f_{r,\nu_n,d+\nu_n-1}$ and Δ are calculated according to Theorem 2. Then we assume that for all $n \in \{1,\dots,N\}$, $k \in \mathcal{K}$, $m \in \mathcal{M}$, $\nu_n - \nu_{n-1} \ge 2\max(d_n^{k,m}, d_{n-1}^{k,m})$.*

Assumption 2 is a standard assumption in non-stationary bandit problems [9]. It guarantees that the length between two change points is sufficiently long so that with high probability, they are detectable with a reasonable delay. As the detection delay $\mathcal{D}_{\Delta,r,\nu_n}$ is asymptotically order optimal [14], $d_n^{k,m}$ is bounded by $\mathcal{O}(\log T)$.

Theorem 3. *Running Algorithm 2 with assumption 1 and 2, the expected cumulative regret of RBO-Coop-UCB with exploration probability p and confidence level $1-\delta$ satisfies*

$$R_T^k \le \sum_{i=1}^{N} \tilde{C}_i^k + \Delta^* T(p + 2MN\sigma + M\delta), \quad (13)$$

where $\tilde{C}_i^k = \Delta_i^*[M\lceil\frac{8\log T}{(\Delta_i^{\min})^2}\rceil + M(1+\frac{\pi^2}{3})]$, Δ_i^* and $\Delta_i^{\min}$ are the highest and lowest reward difference in the i-th stationary segment, δ is the false alarm rate in Theorem 2 in [14], $\sigma < \delta$ is the maximum false alarm under cooperation.[1]

Proof Sketch. $\tilde{C}_i^k$ indicates the bandit algorithm regret, $pT\Delta^*$ refers to the regret caused by the exploration and the other parts calculate the regret caused by the bad event including false alarm and missed detection. To prove the regret, first consider the stationary scenario with $N=1$, $\nu_0 = 0$ and $\nu_1 = T$, then the problem reduces to the multi-agent MAB and the regret is $R_T^k \le \Delta_1^*[M\sigma T + pT + M\lceil\frac{8\xi\log T}{(\Delta_1^{\min})^2}\rceil + M(1+\frac{\pi^2}{3})]$. Based on Theorem 1 and 2, we can bound the regret caused by the bad event. Then the regret in piecewise-stationary scenario can be proven recursively. □

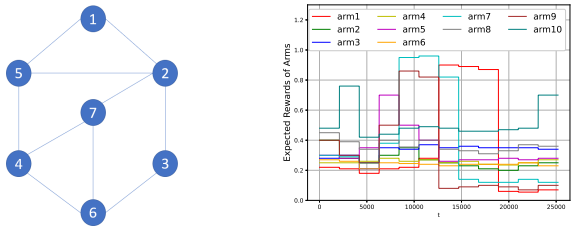
[1]We omit the proofs due to space considerations.

Corollary 3.1. *Choosing $\delta = \frac{1}{T}$, $p = \sqrt{\frac{M\log T}{T}}$, the upper bound of regret will be*

$$R_T \le \mathcal{O}(KNM\log T + K\sqrt{MT\log T}) \quad (14)$$

V. Experiments

Our proposed algorithm for Bernoulli distributions is easily extendable to other distributions with bounded support by using the sample $x \sim \mathcal{B}(y_t)$ after obtaining the scaled reward $y_t \in [0,1]$. In this section, we evaluate our algorithm using a real-world dataset for digital marketing shown in [17]. There are 12 piecewise-stationary segments of the bounded reward distribution and the underlying expected rewards of arms are shown in Fig. 1b.



(a) Observation network

(b) Average rewards of arms in the Digital Marketing dataset.

Fig. 1: Setting of Experiment (Digital Marketing dataset).

We compare RBO-Coop-UCB with six benchmarks from the literature and its variant. Specially, DUCB [6] and SW-UCB [6] are passively adaptive algorithms while M-UCB [17], GLR-UCB [9], [18] are actively adaptive algorithms. For consistency, we implement each piecewise-stationary algorithm with a cooperative version (information-sharing) and a non-cooperative version (each agent runs independently in parallel). We select UCB [19] and EXP3 [20] from the stochastic- and adversarial bandit literature. RBO-Coop-UCB and GLR-Coop-UCB share a similar structure, including information sharing and cooperative change point detection. The only difference between RBO-Coop-UCB and GLR-Coop-UCB is the implemented change point detector. The hyperparameter in our experiments are as follows. In DUCB, we have the discount factor $\gamma = 1 - \sqrt{N/T}/4$; In SW-UCB, we select the sliding window length $\tau = 2\sqrt{T\log T/N}$; In M-UCB, we have $\delta = \max_{i\in N, m\in\mathcal{M}} |\mu_i^m - \mu_{i+1}^m|$, window size $\omega = 800$, $b = [\omega\log(2MT^2)/2]^{1/2}$ and $\gamma = 0.05\sqrt{\frac{(N-1)(2b+3\sqrt{\omega})}{2T}}$; For algorithms with GLR and RBO, $p = \sqrt{\frac{\log T}{T}}$, while in GLR $\delta = \frac{10}{T}$ and in RBO, $\eta_{r,s,t} = \frac{10}{T}$.

We further pre-process the dataset by setting a network with $K = 7$ agents and the communication network is shown in Fig. 1a. Fig. 2 shows the average regret of all algorithms and Fig. 3 shows the change point detection signals based on ten independent experiments.

According to Fig. 2, RBO-Coop-UCB has the lowest regret among all algorithms, which shows the effectiveness of our proposal. Compared with the algorithms with cooperation among agents (solid lines), most other methods (dashed lines) suffer higher regrets. That proves that an information-sharing framework can improve performance. Finally, most algorithms that track the time variations have a regret lower than UCB and EXP3. In general, it is excessively ideal to assume the stochastic bandit model and apply a UCB-based algorithm and too conservative to assume the adversarial bandit model and use an EXP3-type method [11]; consequently, most of piecewise-stationary bandit algorithms improve the performance by taking the piecewise-stationary environment into account.

The performance of RBOCPD and GLRCPD in change point detection of ten independent experiments is compared in Fig. 3. The change point detection performance of RBO-Coop is better than GLR-Coop due to its sensitive detection and short detection delay. Besides, it is observable that RBO-UCB has more false alarms while our proposed cooperation mechanism reduces false alarms significantly and maintains comparable performance in detecting real change points. Thus RBO-Coop detects more change points than GLR-Coop does, and guarantees fewer false alarms, leading to a lower regret. Therefore, our proposed algorithm, RBO-Coop-UCB has the best performance compared to the state-of-the-art algorithms.

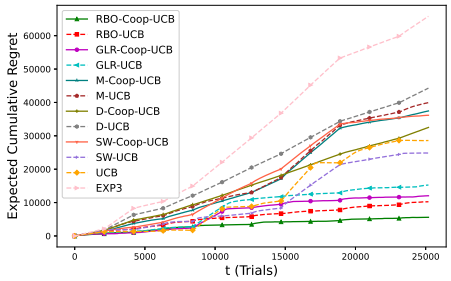
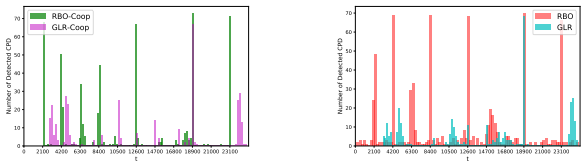


Fig. 2: Expected cumulative regret for different benchmarks.



(a) RBO-Coop and GLR-Coop (b) RBO and GLR

Fig. 3: Change Point detection in different algorithms.

VI. Conclusion

In this paper, we propose an algorithm (RBO-Coop-UCB) which can solve the MAMAB problem in a piecewise-stationary environment. RBO-Coop-UCB considers a cooperative UCB framework with information sharing among agents with change point detector based on Bayesian strategy. We prove that the group regret of RBO-Coop-UCB algorithm is upper bounded by $\mathcal{O}(KNM \log T + K\sqrt{MT \log T})$. Numerical analysis shows our proposed algorithm outperforms other state-of-the-art algorithms.

References

[1] L. Tang, R. Rosales, A. Singh, and D. Agarwal, "Automatic ad format selection via contextual bandits," in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013, pp. 1587–1594.

[2] S. Maghsudi and S. Stańczak, "Joint channel selection and power control in infrastructureless wireless networks: A multiplayer multiarmed bandit framework," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4565–4578, 2014.

[3] O. Atan, S. Ghoorchian, S. Maghsudi, and M. v. d. S. Schaar, "Data-driven online recommender systems with costly information acquisition," *IEEE Transactions on Services Computing*, pp. 1–1, 2021.

[4] D. Kalathil, N. Nayyar, and R. Jain, "Decentralized learning for multiplayer multiarmed bandits," *IEEE Transactions on Information Theory*, vol. 60, no. 4, pp. 2331–2345, 2014.

[5] U. Madhushani and N. E. Leonard, "Heterogeneous stochastic interactions for multiple agents in a multi-armed bandit problem," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 3502–3507.

[6] A. Garivier and E. Moulines, "On upper-confidence bound policies for switching bandit problems," in *International Conference on Algorithmic Learning Theory*. Springer, 2011, pp. 174–188.

[7] O. Besbes, Y. Gur, and A. Zeevi, "Stochastic multi-armed-bandit problem with non-stationary rewards," *Advances in neural information processing systems*, vol. 27, pp. 199–207, 2014.

[8] L. Wei and V. Srivatsva, "On abruptly-changing and slowly-varying multiarmed bandit problems," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 6291–6296.

[9] L. Besson and E. Kaufmann, "The generalized likelihood ratio test meets klUCB: an improved algorithm for piecewise non-stationary bandits," *Proceedings of Machine Learning Research vol XX*, vol. 1, p. 35, 2019.

[10] P. Auer, P. Gajane, and R. Ortner, "Adaptively tracking the best bandit arm with an unknown number of distribution changes," in *Conference on Learning Theory*. PMLR, 2019, pp. 138–158.

[11] H. Zhou, L. Wang, L. Varshney, and E.-P. Lim, "A near-optimal change-detection based algorithm for piecewise-stationary combinatorial semi-bandits," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 6933–6940.

[12] J. Mellor and J. Shapiro, "Thompson sampling in switching environments with Bayesian online change detection," in *Artificial Intelligence and Statistics*. PMLR, 2013, pp. 442–450.

[13] P. Landgren, V. Srivastava, and N. E. Leonard, "Distributed cooperative decision making in multi-agent multi-armed bandits," *Automatica*, vol. 125, p. 109445, 2021.

[14] R. Alami, O. Maillard, and R. Féraud, "Restarted Bayesian online change-point detector achieves optimal detection delay," in *International Conference on Machine Learning*. PMLR, 2020, pp. 211–221.

[15] U. Madhushani and N. E. Leonard, "Heterogeneous explore-exploit strategies on multi-star networks," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 1192–1197.

[16] R. P. Adams and D. J. MacKay, "Bayesian online changepoint detection," *stat*, vol. 1050, p. 19, 2007.

[17] Y. Cao, W. Zheng, B. Kveton, and Y. Xie, "Nearly optimal adaptive procedure for piecewise-stationary bandit: a change-point detection approach," *AISTATS,(Okinawa, Japan)*, 2019.

[18] O.-A. Maillard, "Sequential change-point detection: Laplace concentration of scan statistics and non-asymptotic delay bounds," in *Algorithmic Learning Theory*. PMLR, 2019, pp. 610–632.

[19] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2, pp. 235–256, 2002.

[20] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM journal on computing*, vol. 32, no. 1, pp. 48–77, 2002.