# The Non-stationary Stochastic Multi-armed Bandit Problem

Robin Allesiardo, Raphaël Féraud, Odalric-Ambrym Maillard

**Noname manuscript No.**
(will be inserted by the editor)

# The Non-stationary Stochastic Multi-armed Bandit Problem

**Robin Allesiardo · Raphaël Féraud · Odalric-Ambrym Maillard**

Received: date / Accepted: date

**Abstract** We consider a variant of the stochastic multi-armed bandit with $K$ arms where the rewards are not assumed to be identically distributed, but are generated by a non-stationary stochastic process. We first study the *unique best arm* setting when there exists one unique best arm. Second, we study the general *switching best arm* setting when a best arm switches at some unknown steps. For both settings, we target problem-dependent bounds, instead of the more conservative problem free bounds. We consider two classical problems: 1) Identify a best arm with high probability (best arm identification), for which the performance measure by the sample complexity (number of samples before finding a near optimal arm). To this end, we naturally extend the definition of sample complexity so that it makes sense in the switching best arm setting, which may be of independent interest. 2) Achieve the smallest cumulative regret (regret minimization) where the regret is measured with respect to the strategy pulling an arm with the best instantaneous mean at each step.

## 1 Introduction

The theoretical framework of the multi-armed bandit problem formalizes the fundamental exploration/exploitation dilemma that appears in decision making problems facing partial information. At a high level, a set of $K$ arms is available to a player. At each turn, she has to choose one arm and receives a reward corresponding to the played arm, without knowing what would have been the received reward had she played another arm. The player faces the dilemma of *exploring*, that is playing an arm whose mean reward is loosely estimated in order to build a better estimate or *exploiting*, that is playing a seemingly best arm based on current mean estimates in order to maximize her cumulative reward. The accuracy of the player policy at time horizon $T$ is typically measured in terms of *sample complexity* or of *regret*. The sample complexity is the number of plays required to find an approximation of the best arm with high probability. In that case, the player can stop playing after identifying this arm. The regret is the difference between the cumulative rewards of the player and the one that could be acquired by a policy assumed to be optimal.

The **stochastic** multi-armed bandit problem assumes the rewards to be generated independently from stochastic distribution associated with each arm. Stochastic algorithms usually assume distributions to be constant over time like with the Thompson Sampling (TS) [17], UCB [2] or Successive Elimination (SE) [6]. Under this assumption of *stationarity*, TS and UCB achieve optimal upper-bounds on the cumulative regret with logarithmic dependencies on $T$ . SE algorithm achieves a near optimal sample complexity.

In the **adversarial** multi-armed bandit problem, rewards are chosen by an adversary. This formulation can model any form of non-stationarity. The EXP3 algorithm [3, 14] achieves an optimal regret of $O(\sqrt{T})$ against an oblivious opponent that chooses rewards before the beginning of the game, with respect to the best policy that pulls the same arm over the totality of the game. This weakness is partially overcame by EXP3.S [3], a variant of EXP3, that forgets the past adding at each time step a proportion of the mean

This paper extends the work presented in the DSAA'2015 Long Presentation paper "EXP3 with Drift Detection for the Switching Bandit Problem." [1]. The algorithms SER3 and SER4 are original and presented for the first time.

Robin Allesiardo and Raphaël Féraud
Orange Labs
E-mail: firstname.lastname@orange.com

Robin Allesiardo and Odalric-Ambrym Maillard
Team TAO, CNRS - Inria Saclay, Île de France - LRI
E-mail: odalricambrym.maillard@inria.fr.

gain and achieves controlled regret with respect to policies that allow arm switches during the run.

The **switching bandit** problem introduces non-stationarity within the *stochastic* bandit problem by allowing means to change at some time-steps. As mean rewards stay stationary between those changes, this setting is also qualified as *piecewise-stationary*. *Discounted* UCB [13] and *sliding-window* UCB [8] are adaptations of UCB to the switching bandit problem and achieve a regret bound of $O(\sqrt{MT \log T})$, where $M-1$ is the number of distribution changes. It is also worth citing META-EVE [10] that associates UCB with a mean change detector, resetting the algorithm when a change is detected. While no analysis is provided, it has demonstrated strong empirical performances.

**Stochastic and Adversarial**. Several variants combining stochastic and adversarial rewards have been proposed by Seldin & Slivkins [15] or Bubeck & Slivkins [5]. For instance, in the setting with *contaminated rewards*, rewards are mainly drawn from stationary distributions except for a minority of mean rewards chosen in advance by an adversary. In order to guarantee their proposed algorithm EXP3++ [15] achieves logarithmic guarantees, the adversary is constrained in the sense it cannot lowered the gap between arms more than a factor $1/2$. They also proposed another variant called *adversarial with gap* [15] which assumes the existence of a round after which an arm persists to be the best. These works are motivated by the desire to create generic algorithms able to perform bandit tasks with various reward types, stationary, adversary or mainly stationary. However, despite achieving good performances on a wide range of problems, each one needs a specific parametrization (i.e. an instance of EXP3++ parametrized for stationary rewards may not perform well if rewards are chosen by an adversary).

**Our contribution**. We consider a generalization of the stationary stochastic, piecewise-stationary and adversarial bandit problems. In this formulation, rewards are drawn from stochastic distributions of arbitrary means defined before the beginning of the game. Our first contribution is for the unique best arm setting. We introduce a deceptively simple variant of the SUCCESSIVE ELIMINATION (SE) algorithm, called SUCCESSIVE ELIMINATION WITH RANDOMIZED ROUND-ROBIN (SER3) and we show that the seemingly minor modification – a randomized round-robin procedure – leads to a dramatic improvement of the performance over the original SE algorithm. We identify a notion of gap that generalizes the gap from stochastic bandits to the non-stationary case, and derive *gap-dependent* (also known as problem-dependent) sample complexity and regret bounds, instead of the more classical and less informative *problem-free* bounds. We show for instance in Theorem 1 and Corollary 1 that SER3 achieves a non-trivial problem dependent sample complexity scaling with $\Delta^{-2}$ and a cumulative regret in $O(K \log(TK/\Delta)/\Delta)$ after $T$ steps, in situations where SE may even suffers from a linear regret, as supported by numerical experiments (see Section 5). This result positions, under some assumptions, SER3 as an alternative to EXP3 when the rewards are non-stationary.

Our second contribution is to manage best arm switches during the game. First, we extend the definition of the sample complexity in order to analyze the best arm identification algorithms when best arm switches during the game. SER4 takes advantages of the low regret of SER3 by resetting the reward estimators randomly during the game and then starting a new phase of optimization. Against an optimal policy with $N-1$ switches of the optimal arm (but arbitrarily many distribution switches), this new algorithm achieves an expected sample complexity of $O(\Delta^{-2}\sqrt{NK\delta^{-1}\log(K\delta^{-1})})$, with probability $1-\delta$, and an expected cumulative regret of $O(\Delta^{-1}\sqrt{NTK\log(TK)})$ after $T$ time steps. A second algorithm for the non stationary stochastic multi-armed bandit with switches is an alternative to the passive approach used in SER4 (the random resets). Second, the algorithm EXP3.R takes advantage of the exploration factor of EXP3 to evaluate unbiased estimations of the mean rewards. Combined with a drift detector, this active approach resets the weights of EXP3 when a change of best arm is detected. We finally show that EXP3.R also obtains competitive problem-dependent regret minimization guarantees in $O\left(3NCK\sqrt{TK\log T}\right)$, where $C$ depends on $\Delta$.

## 2 Setting

We consider a generalization of the stationary stochastic, piecewise-stationary and adversarial bandit problems where the adversary chooses before the beginning of the game a sequence of *distributions* instead of directly choosing a sequence of rewards. This formulation generalizes the adversarial setting since choosing arbitrarily a reward $y_k(t)$ is equivalent to drawing this reward from a distribution of mean $y_k(t)$ and a variance of zero. The stationary stochastic formulation of the bandit problem is a particular case, where the distributions do not change.

### 2.1 The problem

Let $[K] = 1, ..., K$ be a set of $K$ arms. The reward $y_{k_t}(t) \in [0,1]$ obtained by the player after playing the arm $k_t$ is drawn from a distribution of mean $\mu_{k_t}(t) \in [0,1]$. The instantaneous gap between arms $k$ and $k'$ at time $t$ is:

$$\Delta_{k,k'}(t) \stackrel{\text{def}}{=} \mu_k(t) - \mu_{k'}(t)\,. \tag{1}$$

The player competes against an optimal policy, assumed as optimal (per example, always playing the arm with the highest mean reward). Let $k^*(t)$ be the arm played by the optimal policy at time $t$.

### 2.2 The notion of sample complexity

In the literature [12], the sample-complexity of an algorithm is the number of samples needed by this algorithm to find a policy achieving a specific level of performance with high probability. We denote $\delta \in (0, 1]$ the probability of failure. For instance, for the best arm identification in the stochastic stationary bandit (that is when $\forall k \forall t\,, \mu_k(t) = \mu_k(t+1)$ and $k^*(t) = k^*(t+1)$), the sample complexity is the number of sample needed to find, with a probability at least $1 - \delta$, the arm $k^*$ with the maximum mean reward. Analysis in sample complexity are useful for situations where the knowledge of the optimal arm is needed to make one impactfull decision, for example to choose which one of several possible products to manufacture or for building hierarchical models of contextual bandits in a greedy way [7], reducing the exploration space.

**Definition 1 (Sample complexity)** Let A be an algorithm. An arm $k$ is epsilon optimal if $\mu_k \geq \mu^* - \epsilon$, with $\epsilon \in [0, 1]$. The sample-complexity of A performing a best arm identification task is the number of observations needed to find an $\epsilon$-optimal arm with a probability of at least $1 - \delta$.

The usual notion of sample complexity - the minimal number of observations required to find a near optimal arm with high probability - is well adapted to the case when there exists a unique best arm during all the game, but makes little sense in the general scenario when the best arm can change. Indeed, after a best arm change, a learning algorithm requires some time steps before recovering. Thus, we provide in section 4 a meaningful extension of the sample complexity definition to the *switching best arm* scenario. This extended notion of sample complexity now takes into account not only the number of time-steps required by the algorithm to identify a near optimal arm, but more generally the number of time steps required before recovering a near optimal arm after each change.

### 2.3 The notion of regret

When the decision process does not lead to one final decision, minimizing the sample complexity may not be an appropriate goal. Instead, we may want to maximize the cumulative gain obtained through the game which is equivalent to minimize the difference between the choices of an optimal policy and those of the player. We call this difference, the regret. We define the *pseudo cumulative regret* as the difference of mean rewards between the arms chosen by the optimal policy and those chosen by the player.

**Definition 2 (Pseudo cumulative regret)**

$$\sum_{t=1}^{T} \mu_{k^*(t)}(t) - \mu_{k_t}(t)\,. \tag{2}$$

Usually, in the stochastic bandit setting, the distributions of rewards are stationary and the instantaneous gap $\Delta_{k,k'}(t) = \mu_k(t) - \mu_{k'}(t)$ is the same for all the time-steps.

There exists a non reciprocate relation between the minimization of the sample-complexity and the minimization of the pseudo cumulative regret. For instance, the algorithm UCB has an order optimal regret, but it does not minimize the sample-complexity. UCB will continue to play sub-optimal arms, but with a decreasing frequency as the number of plays increases. However, an algorithm with an optimal sample complexity, like MEDIAN ELIMINATION [6], will also have an optimal pseudo cumulative regret (up to some constant factors). More details on the relation between both lower bounds can be found in [4, 9].

Therefore, the algorithms presented in this paper slightly differ according to the quantity to minimize, the regret or the sample complexity. For instance, when the target is the regret minimization, after identifying the best arm, the algorithms continue to sample it whereas in the case of sample complexity minimization, the algorithms stop the sampling process when the best arm is identified. When best arm switches are considered, algorithms minimizing the sample complexity enter a waiting state after identifying the current best arm and do not sample the sequence for exploitation purposes (sampling the optimal arm still increases the sample complexity). However, they still have to parsimoniously collect samples for each actions in order to detect best arm changes and face a new trade-off between the rate of sampling and the time needed to find the new best arm after a switch.

## 3 Non-stationary Stochastic Multi-armed Bandit with Unique Best Arm.

In this section, we present the algorithm SUCCESSIVE ELIMINATION WITH RANDOMIZED ROUND-ROBIN (SER3, see algorithm 1), a randomized version of SUCCESSIVE ELIMINATION which tackles the best arm identification problem when rewards are non-stationary.

### 3.1 A modified Successive Elimination algorithm

We elaborate on several notions required to understand the behavior of the algorithm and to relax the constraint of stationarity.

#### *3.1.1 The elimination mechanism*

The elimination mechanism was introduced by SUCCESSIVE ELIMINATION [6]. Estimators of the rewards are built by sequentially sampling the arms. After $\tau_{\min}$ turns of round-robin, the elimination mechanism starts to occur. A lower-bound of the reward of the best empirical arm is computed and compared with an upper-bound of the reward of all other arms. If the lower-bound is higher than one of the upper-bounds, then the associated arm is eliminated and stop being considered by the algorithm. Processes of sampling and elimination are repeated until the elimination of all arms except one.

**Algorithm 1** SUCCESSIVE ELIMINATION WITH RANDOMIZED ROUND-ROBIN (SER3)

**input:** $\delta \in (0, 0.5]$, $\epsilon \in [0, 1)$, $\tau_{\min} = \log \frac{K}{\delta}$
**output:** an $\epsilon$-approximation of the best arm
$S_1 = [K]$, $\forall k$, $\hat{\mu}_k(0) = 0$, $t = 1$, $\tau = 1$
**While** $|S_\tau| > 1$
  Shuffle $S_\tau$
  **For** each $k \in S_\tau$ **do**
    Play $k$
    $\hat{\mu}_k(\tau) = \frac{\tau-1}{\tau}\hat{\mu}_k(\tau-1) + \frac{y_k(t)}{\tau}$
    $t = t + 1$
  **End for**
  $k_{\max} = \arg\max_{k \in S} \hat{\mu}_k(\tau)$
  **If** $\tau \geq \tau_{\min}$
    Remove from $S_{\tau+1}$ all $k$ such as:

$$\hat{\mu}_{\max}(\tau) - \hat{\mu}_k(\tau) + \epsilon \geq \sqrt{\frac{2}{\tau}\log\left(\frac{4K\tau^2}{\delta}\right)} \tag{3}$$

  **End if**
  **If** $|S_\tau| = 1$ **and** the algorithm performs a sample complexity minimization task
    **Return** the last element of $S_\tau$
  **End if**
  $\tau = \tau + 1$
**End while**

#### *3.1.2 Hoeffding inequality*

SUCCESSIVE ELIMINATION assumes that the rewards are drawn from stochastic distributions that are identical over time (rewards are identically distributed). However, the Hoeffding inequality used by this algorithm does not require stationarity and only requires independence. We remember the Hoeffding inequality:

**Lemma 1 (Hoeffding inequality [11])** *If $X_1, X_2, ..., X_\tau$ are $\tau$ independent random variables and $0 \leq X_i \leq 1$ for all $(i = 1, 2, ..., \tau)$, then for $\epsilon_\tau > 0$*

$$P\left(\left|\sum_{i=1}^{t}\frac{X_i}{\tau} - \mathbb{E}\left[\sum_{i=1}^{t}\frac{X_i}{\tau}\right]\right| \geq \epsilon_\tau\right) \leq 2\exp\left(-2\epsilon_\tau^2\tau\right) .$$

Thus, we can use this inequality to calculate confidence bounds of empirical means computed with rewards drawn from non identical distributions.

#### *3.1.3 Randomization of the Round-Robin*

We illustrate the need of randomization with an example tricking a deterministic algorithm (see figure 1).

| $\mu_k(t)$ | $t=1$ | $t=2$ | $t=3$ | $t=4$ | $t=5$ | $t=6$ |
|---|---|---|---|---|---|---|
| $k=1$ | 0.6 | 1 | 0.6 | 1 | 0.6 | 1 |
| $k=2$ | 0.4 | 0.8 | 0.4 | 0.8 | 0.4 | 0.8 |

Fig. 1: A sequence of mean rewards tricking a deterministic bandit algorithm.

The best arm seems to be $k = 1$ as $\mu_1(t)$ is greater than $\mu_2(t)$ at every time-step $t$. However, by sampling the arms with a deterministic policy playing sequentially $k = 1$ and then $k = 2$, after $t = 6$ the algorithm has only sampled rewards from a distribution of mean $0.6$ for $k = 1$ and of mean $0.8$ for $k = 2$. After enough time following this pattern, an elimination algorithm will eliminate the first arm. Our algorithm SER3 adds a shuffling of the arm set after each round-robin cycle to SUCCESSIVE ELIMINATION and avoids this behavior.

#### *3.1.4 Uniqueness of the best arm*

The best arm identification task assumes a criteria identifying the best arm without ambiguity. We define the **optimal arm** as:

$$k^* = \arg\max_{k \in [K]} \sum_{t=1}^{T} \mu_k(t) . \tag{4}$$

As an efficient algorithm will find the best arm before the end of the run, we use assumption 1 to ensure its uniqueness at every time-step. First, we define some notations. A run of SER3 is a succession of round-robin. The set $[\tau] = \{(t_1, |S_1|), ..., (t_\tau, |S_\tau|)\}$ is a realization of SER3 and $t_i$ is the time step when the round-robin $i^{\text{th}}$ of size $|S_i|$ starts ($t_i = 1 + \sum_{j=1}^{i-1} |S_j|$). As arms are only eliminated, $|S_i| \geq |S_{i+1}|$. We denote $\mathbb{T}(\tau)$ the set containing all possible realizations of $\tau$ round-robin steps. Now, we can introduce assumption 1 that ensures the best arm is the same at any time-step.

**Assumption 1** (Positive mean-gap)**.** *For any* $k \in [K] - \{k^*\}$ *and any* $[\tau] \in \mathbb{T}(\tau)$ *with* $\tau \geq \tau_{\min}$*, we have:*

$$\Delta_k^*([\tau]) = \frac{1}{\tau} \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i+|S_i|-1} \frac{\Delta_{k^*,k}(j)}{|S_i|} > 0\,. \tag{5}$$

Assumption 1 is trivially satisfied when distributions are stationary, is quite weak (see e.g. figure 2(b)) and can tolerate a large noise when $\tau$ is high. As the optimal arm must distinguish itself from others, instantaneous gaps are more constrained at the beginning of the game. It is quite similar to the assumption used by Seldin & Slivkins [15] to be able to achieve logarithmic expected regret on *moderately contaminated rewards*, i.e., the adversary does not lower the averaged gap too much. Another analogy can be done with the *adversarial with gap* setting [15], $\tau_{\min}$ representing the time needed for the optimal arm to accumulate enough rewards and to distinguish itself from the suboptimal arms.

Figure 2(a) illustrates assumption 1. In this example the mean of the optimal arm $k^*$ is lower than the second one on time-steps $t \in \{5, 6, 7\}$. Thus, even if the instantaneous gap is negative during these time-steps, the mean gap $\Delta_k^*([\tau])$ stays positive. The parameter $\tau_{\min}$ protects the algorithm from local noise at the initialization of the algorithm. In order to ease the reading of the results in the next sections, we here assume $\tau_{\min} = \log \frac{K}{\delta}$.

Assumption 1 can be seen as a sanity-check assumption ensuring that the best-arm identification problem indeed makes sense. In section 4, we consider the more general switching bandit problem. In this case, assumption 1 may not be verified (see figure 2(b)), and is naturally extended by dividing the game in segments wherein assumption 1 is satisfied.

### 3.2 Analysis

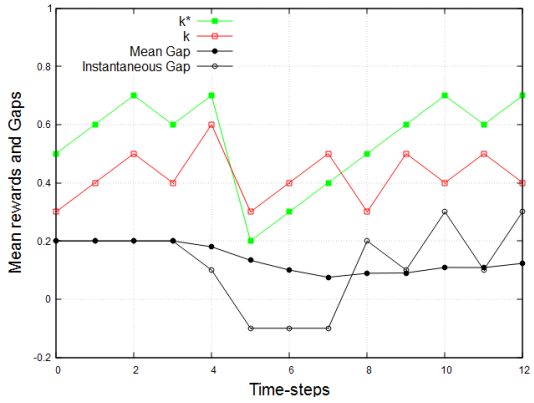
All theoretical results are provided for $\epsilon = 0$ and therefore accept only $k^*$ as the optimal arm.

**Theorem 1 (Sample-complexity of** SER3**)** *For* $K \geq 2$*,* $\delta \in (0, 0.5]$*, and* $\tau_{\min} = \log \frac{K}{\delta}$*, the sample-complexity of* SER3 *is upper bounded by:*

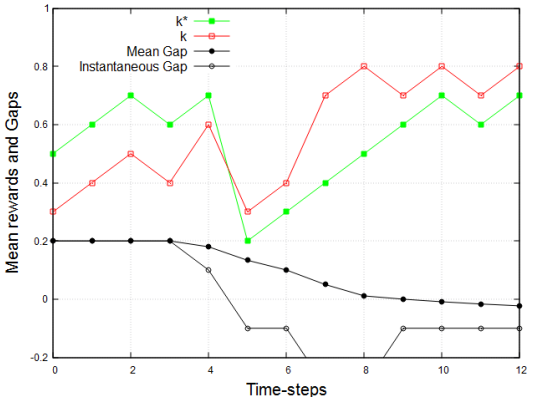
$$O\left(\frac{K}{\Delta^2} \log(\frac{K}{\delta\Delta})\right)$$

*where* $\Delta = \min_{[\tau],k} \frac{1}{\tau} \sum_{i=1}^{\tau} \sum_{t=t_i}^{t_i+|S_i|-1} \frac{\Delta_{k^*,k}(t)}{|S_i|}$.

The proof is given in Appendix B.1.




(a) Assumption 1 is satisfied as the mean gap remains positive.




(b) Assumption 1 is not satisfied. This sequence involves a best arm switch as the mean gap become non positive.

Fig. 2: Two examples of sequence of mean rewards.

Guarantee on the sample complexity can be transposed in guarantee on the pseudo cumulative regret. In that case, when only one arm remains in the set, the player continues to play this last arm until the end of the game.

**Corollary 1 (Expected pseudo cumulative regret of** SER3**).** *For* $K \geq 2$*, and* $\delta = 1/T$*, and* $\tau_{\min} = \log(KT)$*, the expected pseudo cumulative regret of* SER3 *is upper bounded by:*

$$\min\left(O\left(\frac{K-1}{\Delta} \log(\frac{KT}{\Delta})\right), O\left(\sqrt{TK \log \frac{T}{K}}\right)\right)$$

The proof is given in Appendix B.2.

These guarantees are the same as the original SUCCESSIVE ELIMINATION performed with a deterministic round-robin on arms with stationary rewards. Indeed, when reward distributions are stationary, we have for all $t$ and all $[\tau]$:

$$\frac{1}{\tau} \sum_{i=1}^{\tau} \sum_{t=t_i}^{t_i+|S_i|-1} \frac{\Delta_{k^*,k}(t)}{|S_i|} = \Delta_{k^*,k}(t) = \Delta_{k^*,k}(t+1)\,. \tag{6}$$

However, in a non-stationary environment satisfying assumption 1 SUCCESSIVE ELIMINATION will eliminate the optimal arm if the adversary knows the order of its round-robin before the beginning of the run and exploits this knowledge against the learner, thus resulting in a linear cumulative regret. Our modification of the SE algorithm allows SER3 to perform on *near adversarial* sequence of reward while achieving a gap dependent logarithmic pseudo cumulative regret.

**Remark:** These logarithmic guarantees result from assumption 1 that allows to stop the exploration of eliminated arms. They do not contradict the lower bound for non-stationary bandit whose scaling is in $\Omega(\sqrt{T})$ [8] as it is due to the cost of the constant exploration for the case where the best arm changes.

### 3.3 Non-stationary Stochastic Multi-armed Bandit with Budget

We study the case when the sequence from which the rewards are drawn does not satisfy assumption 1.

The sequence of mean rewards is build by the adversary in two steps. First, the adversary choose the mean rewards $\mu_k(1), ..., \mu_k(T)$ associated with each arm in such a way that assumption 1 is satisfied. The adversary can then apply a malus $b_k(t) \in [0, \mu_k(t)]$ to each mean reward to obtain the final sequence. The mean reward of the arm $k$ at time $t$ is $\mu_k(t) - b_k(t)$. The budget spent by the adversary for the arm $k$ is $B_k = \sum_{t=1}^{T} b_k(t)$. We denote $B \geq \arg\max_k B_k$ the upper-bound on the budget of the adversary.

The algorithm SER3 can be modified to perform a best arm identification task when assumption 1 is not satisfied but $B$ is known. To achieve that, we replace the condition of elimination (Inequality (3) in Algorithm 1) is replaced by the following:

$$\hat{\mu}_{\max}(\tau) - \hat{\mu}_k(\tau) + \epsilon \geq \frac{B}{\tau} + 2\sqrt{\frac{1}{2\tau}\log\left(\frac{4K\tau^2}{\delta}\right)}$$

This new algorithm is called SUCCESSIVE ELIMINATION WITH ROUND ROBIN RANDOMIZED AND BUDGET (SER3.B).

**Theorem 2** *For $K \geq 2$, $\delta \in (0, 0.5]$, and $\tau_{\min} = \log\frac{K}{\delta}$, the sample complexity of* SER3.B *is upper-bounded by:*

$$O\left(\frac{K}{\Delta^2}\left(\log\frac{K}{\delta\Delta} + B\right)\right)$$

*where $\Delta = \min_{[\tau],k} \frac{1}{\tau}\sum_{i=1}^{\tau}\sum_{t=t_i}^{t_i+|S_i|-1}\frac{\Delta_{k^*,k}(t)}{|S_i|}$.*

The proof is given in Appendix B.1.

## 4 Non-stationary Stochastic Multi-armed Bandit with Best Arm Switches

The switching bandit problem has been proposed by Garivier et al. [8] and assumes means to be stationary between switches. In particular, the algorithm SW-UCB is built on this assumption and is a modification of UCB using only the rewards obtained inside a sliding window. In our setting, we allow mean rewards to change at every time-steps and consider that a best arm switch occurs when the arm with the highest mean change. This setting provides an alternative to the adversarial bandit with budget, when $B$ is very high or unknown.

The **optimal policy** is the sequence of couples (optimal arm, time when the switch occurred):

$$\{(k_1^*, 1), ..., (k_N^*, T_N)\}, \tag{7}$$

with $k_n^* \neq k_{n+1}^*$ and $\Delta_{k_n^*,k}(t) > 0$ for any $k \in [K] - \{k_n^*\}$ and any $t \in [T_n, T_{n+1})$. The optimal policy starts playing the arm $k_n^*$ at the time-step $T_n$. Time-steps $T_n$ when switches occur are unknown to the player.

### 4.1 Successive Elimination with Randomized Round-Robin and Resets (SER4)

The Definition 1 of the sample complexity is not adapted to the switching bandit problem. Indeed this definition is used to measure the number of observations needed by an algorithm to find one unique best arm. When the best arm changes during the game, this definition is too limiting. In subsection 4.1.1 we introduce a generalization of the sample complexity for the case of switching policies.

#### *4.1.1 The sample complexity of the best arm identification problem with switches*

A cost associated is added to the usual sample complexity. This cost is equal to the number of iterations after a switch during which the player does not know the optimal arm and does not sample.

**Definition 3 (Sample complexity with switches)** Let A be an algorithm. The sample-complexity of A performing a best arms identification task for a segmentation $\{T_n\}_{n=1..N}$ of $[1:T]$, with $T_1 = 1 < T_2 < \cdots < T_N < T$, is:

$$\sum_{n=1}^{N}\sum_{t=T_n}^{T_{n+1}-1}\max(s(t), 1_{[\![k_t \neq k_n^*]\!]}), \tag{8}$$

where $s(t)$ is a binary variable equal to 1 if and only if the time-step $t$ is used by the sampling process of A, $k_t$ is the arm identified as optimal by A at time t, $k_n^*$ is the optimal arm over the segment $n$ and $T_{N+1} = T + 1$.

In order to clarify definition 3, we detail the different states achievable by an algorithm of best arms identification and their impact on the sample complexity. Two states are achievable during a task of minimization of the sample complexity:

- $s(t) = 1$ if the algorithm is sampling an arm during the time-step $t$. In the case of SER4, $s(t) = 1$ when $|S_\tau| \neq 1$ and the sample complexity increases by one.
- $s(t) = 0$ if the algorithm submits an arm as the optimal one during the time-step $t$. In the case of SER4, $s(t) = 0$ when $|S_\tau| = 1$. The sample complexity increases by one if $k_t \neq k^*(t)$.

In the context of SER4, the sample complexity is the number of time-steps during which the arm set does not only contain the optimal arm.

#### *4.1.2 Algorithm*

In order to allow the algorithm to choose another arm when a switch occurs, at each turn, estimators of SER3 are reseted with a probability $\varphi \in [0, 1]$ and a new task of best arm identification is started. We name this algorithm SUCCESSIVE ELIMINATION WITH RANDOMIZED ROUND-ROBIN AND RESETS (SER4).

**Algorithm 2** SUCCESSIVE ELIMINATION WITH RANDOMIZED ROUND-ROBIN AND RESETS (SER4)

**input:** $\delta \in (0, 1]$, $\epsilon \in [0, 1)$, $\varphi \in [0, 1)$
$S_1 = [K]$, $\forall k$, $\hat{\mu}_k(0) = 0$, $t = 1$, $\tau = 1$
**While** $t \leq T$
  Shuffle $S_\tau$
  **For** each $k \in S_\tau$ **do**
    **If** $|S_\tau| \neq 1$ **or If** the algorithm performs a regret minimization task
      Play $k$
      $\hat{\mu}_k(\tau) = \frac{\tau-1}{\tau}\hat{\mu}_k(\tau - 1) + \frac{y_{k_t}(t)}{\tau}$
    **End if**
    $t = t + 1$
  **End for**
  $k_{\max} = \arg\max_{k \in S} \hat{\mu}_k(\tau)$
  Remove from $S_{\tau+1}$ all $k$ such as:

$$\hat{\mu}_{\max}(\tau) - \hat{\mu}_k(\tau) + \epsilon \geq 2\sqrt{\frac{1}{2\tau}\log\left(\frac{4K\tau^2}{\delta}\right)}$$

  $\tau = \tau + 1$
  $t = t + 1$
  **With a probability** $\varphi$
    $S_t = [K]$
    $\forall k$, $\hat{\mu}_k(t) = 0$
    $\tau = 1$
  **End with a probability**
**End while**

#### *4.1.3 Analysis.*

We now provide the performance guarantees of the SER4 algorithm, both in terms of sample complexity and of pseudo cumulative regret.

The following results are given in expectation and in high probability. The expectations are taken with regard to the randomization of the resets. The sample complexity or the pseudo cumulative regret achieved by the algorithm between each resets (given by the analysis of SER3) are still results in high probability.

**Theorem 3 (Expected sample complexity of** SER4**)** *For $K \geq 2$, $\delta = 1/T$, $\tau_{\min} = \log\frac{K}{\delta}$ and $\varphi \in (0, 1]$, the expected sample complexity of* SER4 *w.r.t. the randomization of resets is upper bounded by:*

$$O\left(\frac{\varphi K}{\delta \Delta^2}\log(\frac{K}{\delta\Delta}) + \frac{N}{\varphi}\right)$$

*with a probability of at least $1 - \delta$.*

The proof is given in Appendix B.3.

We tune $\varphi$ in order to minimize the sample complexity.

**Corollary 2.** *For $K \geq 2$, $\delta = 1/T$, $\tau_{\min} = \log\frac{K}{\delta}$, $\Delta \geq \frac{1}{KT}$ and $\varphi = \sqrt{\frac{N\delta}{K\log(\frac{K}{\delta})}}$, the expected sample complexity of* SER4 *w.r.t. the randomization of resets is upper bounded by:*

$$O\left(\frac{1}{\Delta^2}\sqrt{\frac{NK\log(\frac{K}{\delta})}{\delta}}\right).$$

**Remark 2:** transposing Theorem 3 for the case where $\epsilon \in [\frac{1}{KT}, 1]$ is straightforward. This allows to tune the bound by setting $\varphi = \epsilon\sqrt{(N\delta)/(K\log(TK))}$.

This result can also be transposed in bound on the expected cumulative regret. We consider that the algorithm continues to play the last arm of the set until a reset occurs.

**Corollary 3 (Expected cumulative regret of** SER4**).** *For $K \geq 2$, and $\delta = 1/T$, $\tau_{\min} = \log(KT)$, $\Delta \geq \frac{1}{KT}$ and $\varphi = \sqrt{\frac{N}{TK\log(KT)}}$, the expected cumulative regret of* SER4 *w.r.t. the randomization of resets is upper bounded by:*

$$\min\left(O\left(\frac{\sqrt{NTK\log(KT)}}{\Delta}\right), O\left(T^{2/3}\sqrt{NK\log\frac{T}{K}}\right)\right). \tag{9}$$

The proof is given in Appendix B.4.

**Remark 3:** A similar dependency in $\sqrt{T}\Delta^{-1}$ appears also in SW-UCB (see Theorem 1 in [8]), and is standard in this type of results.

## 4.2 EXP3 with Resets

SER4 and other algorithms from the state of the art [3, 8, 13] use a passive approach through forgetting the past. In this subsection, we propose an active strategy which consists in resetting the reward estimations when a change of the best arm is detected. A supposed advantage of this approach is to let the algorithm converge on a longer time period, as it is reset only when a switch is detected, and thus build a more accurate estimate of the reward distributions. First, we describe the adversarial bandit algorithm EXP3 [3], which will be used by the proposed algorithm EXP3.R between detections. We then describe the drift detector used to detect changes of the best arm. Finally, we combine the both to obtain the EXP3.R algorithm.

**Algorithm 3** EXP3

The parameter $\gamma \in [0, 1]$ controls the exploration and the probability to choose an action $k$ at round $t$ is:

$$p_k(t) = (1-\gamma)\frac{w_k(t)}{\sum_{i=1}^{k} w_i(t)} + \frac{\gamma}{K}\,, \tag{10}$$

where the weight $w_k(t)$ of each action $k$ is computed from the unbiased cumulative reward estimator $\hat{X}_k(t)$:

$$w_k(t) = \exp(\frac{\gamma}{K}\hat{X}_k(t))\,, \tag{11}$$

with

$$\hat{X}_k(t) = \sum_{j=t_r}^{t} \frac{x_k(j)}{p_k(j)}[\![k = k(j)]\!]\,, \tag{12}$$

where $t_r$ is the time steps when the algorithm is initialized.

### *4.2.1 The EXP3 algorithm*

The EXP3 algorithm (see Algorithm 3) minimizes the regret against the best arm using an unbiased estimation of the cumulative reward at time $t$ for computing the choice probabilities of each action. While this policy can be viewed as optimal in an actual adversarial setting, in many practical cases the non-stationarity within a time period exists but is weak and is only noticeable between different periods. If an arm performs well in a long time period but is extremely bad on the next period, the EXP3 algorithm can need a number of trial equal to the first period's length to switch its most played arm.

### *4.2.2 The detection test*

The detection test (see Algorithm 4) uses confidence intervals to estimate the expected reward in the previous time period. The action distribution in EXP3 is a mixture of uniform and Gibbs distributions. We call $\gamma$-observation an observation selected though the uniform distribution. Parameters $\gamma$, $H$ and $\delta$ define the minimal number of $\gamma$-observations by arm needed to call a test of accuracy $\epsilon$ with a probability $1-\delta$. They will be fixed in the analysis (see Corollary 4) and the correctness of the test is proven in Lemma 2. We denote $\bar{\mu}^k(I)$ the empirical mean of the rewards acquired from the arm $k$ on the interval $I$ using only $\gamma$-observations and $\Gamma_{\min}$(I) the smallest number of $\gamma$-observations among each action on the interval $I$. The detector is called only when $\Gamma_{\min}(I) \geq \frac{\gamma H}{K}$. The detector raises an alert when the action $k_{\max}$ with the highest empirical mean $\hat{\mu}^k(I-1)$ on the interval $I-1$ is eliminated by an other on the current interval.

**Algorithm 4** DriftDetection(I)

**Parameters:** Current interval $I$
$k_{\max} = \arg\max_k \hat{\mu}^k(I-1)$
$\epsilon = \sqrt{\frac{K\log(\frac{1}{\delta})}{2\gamma H}}$
**return** $[\![\exists k, \quad \hat{\mu}^k(I) - \hat{\mu}^{k_{\max}}(I) \geq 2\epsilon]\!]$

### *4.2.3 The EXP3.R algorithm*

Coupled with a detection test, the EXP3 algorithm has several advantages. First in a non-stationary environment, we need a constant exploration to detect changes where a suboptimal arm becomes optimal and this exploration is naturally given by the algorithm. Second, the number of breakpoints is higher than the number of best arm changes ($M \geq N$). This means that the number of resets needed by EXP3 is lower than the one needed by a stochastic bandit algorithm such as UCB. Third, EXP3 is robust against test failures (non detection) or local non-stationarity. We call EXP3.R the algorithm obtained by combining EXP3 and the drift detector. First, one instance of EXP3 is initialized and used to select actions. When the count of $\frac{\gamma H}{K}$ $\gamma$-observations per arm is fulfilled, the detection test is called. If in the corresponding interval, the empirical mean of an arm exceeds by $2\epsilon$ the empirical mean of the current best arm then a drift detection is raised. In this case, weights of EXP3 are reset. Then a new interval of collect begins, preparing the next test. These steps are repeated until the run ends (see Algorithm 5).

### *4.2.4 Analysis*

In this section we analyze the drift detector and then we bound the expected regret of the EXP3.R algorithm.

**Algorithm 5** EXP3 with Resets

**Parameters:** Reals $\delta, \gamma$ and Integer $H$
$I = 1$
**for each** $t = 1, ..., T$ **do**
  Run EXP3 on time step t
  **if** $\Gamma_{\min}(I) \geq \frac{\gamma H}{K}$ **then**
    **if** $DriftDetection(I)$ **then**
      Reset EXP3
    **end if**
    $I = I + 1$
  **end if**
**end for**

**Assumption 2 (Accuracy of the drift detector).** *During each of the segments $S$ where $k_S^*$ is the optimal arm, the gap between $k_S^*$ and any other arm is of at least $4\epsilon$ with*

$$\epsilon = \sqrt{\frac{K \log(\frac{1}{\delta})}{4\gamma H}}\,. \tag{13}$$

Lemma 2 guarantees that when Assumption 1 holds and the interval $I$ is included into the interval $S$ then, with high probability, the test will raise a detection if and only if the optimal action $k_S^*$ eliminates a sub-optimal action.

**Lemma 2 (Arm switches are detected)** *When Assumption 2 holds and $I \subseteq S$, then, with a probability $1 - 2\delta$, for any $k \neq k_S^*$:*

$$\hat{\mu}^{k_S^*}(I) - \hat{\mu}^{k}(I) \geq 2\sqrt{\frac{K \log(\frac{1}{\delta})}{2\gamma H}} \Leftrightarrow \mu^{k_S^*}(I) \geq \mu^{k}(I)\,. \tag{14}$$

The proof is given in Appendix B.5.

Theorem 4 bounds the expected cumulative regret of EXP3.R.

**Theorem 4 (Expected cumulative regret of** EXP3.R**)** *For any $K > 0$, $0 < \gamma \leq 1$, $0 \leq \delta < \frac{1}{2}$, $H \geq K$ and $N \geq 1$ when Assumption 2 holds, the expected cumulative regret of* EXP3.R *is*

$$G^* - \boldsymbol{E}[G_{EXP3.R}] \leq (e-1)\gamma T + \frac{\left(N - 1 + \frac{K\delta T}{H} + K\delta\right) K \log(K)}{\gamma} + (N-1)HK\left(\frac{1}{1-2\delta} + 1\right)\,. \tag{15}$$

The proof is given in Appendix B.6.

In Corollary 4 we optimize parameters of the bound obtained in Theorem 4.

**Corollary 4 (Expected cumulative regret of** EXP3.R**).** *For any $K \geq 1$, $T \geq 10$, $N \geq 1$ and $C \geq 1$ when Assumption 1 holds, the expected cumulative regret of* EXP3.R *run with input parameters*

$$\delta = \sqrt{\frac{\log T}{KT}}\,, \gamma = \sqrt{\frac{K \log K \log T}{T}} \text{ and } H = C\sqrt{T \log T} \tag{16}$$

*is*

$$G^* - \boldsymbol{E}[G_{EXP3.R}]) \leq (e-1)\sqrt{TK \log K \log T} + N\sqrt{TK \log K} + (C+1)K\sqrt{T \log K} + 3NCK\sqrt{T \log T}\,. \tag{17}$$

The proof is given in Appendix B.7.

Accordingly to $C$, the precision $\epsilon$ is:

$$\epsilon = \sqrt{\frac{1}{2C}}\sqrt{\frac{\log\sqrt{\frac{KT}{\log T}}}{\log T}\sqrt{\frac{K}{\log K}}}\,. \tag{18}$$

Notice that, when $T$ increases, $\sqrt{\frac{\log\sqrt{\frac{KT}{\log T}}}{\log T}\sqrt{\frac{K}{\log K}}}$ tends towards a constant.

## 5 Numerical Experiments

We compare our algorithm with the state-of-the-art. For each problem, $K = 20$ and $T = 10^7$. The instantaneous gap between the optimal arm and the others is constant, $\Delta = 0.05$, i.e. the mean of the optimal arm is $\mu^*(t) = \mu(t) + \Delta$. During all experiments, probabilities of failure of SUCCESSIVE ELIMINATION (SE), SER3 and SER4 are set to $\delta = 0.05$. Constant explorations of all algorithms of the EXP3 family are set to $\gamma = 0.05$. Results are averaged over 50 runs. On problems 1 and 2, variances are low (in the order of $10^3$) and thus not showed. On problem 3, variances are plotted as the gray areas under the curves.

### 5.1 Problem 1: Sinusoidal means

The index of the optimal arm $k^*$ is drawn before the game and does not change. The mean of all suboptimal arm is $\mu(t) = \cos(2\pi t/K)/5 + 0.5$.

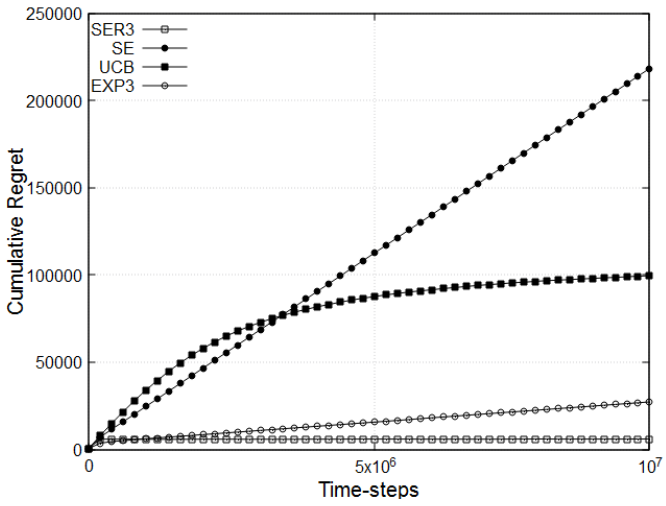



Fig. 3: Cumulative regret of SER3, SE, UCB and EXP3 on the Problem 1.

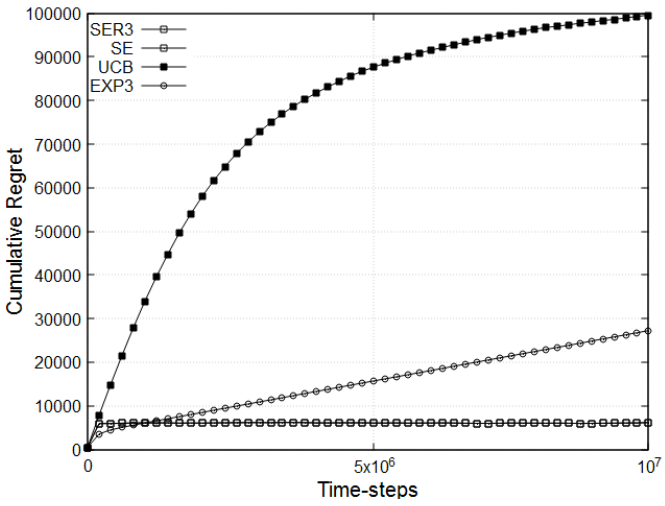



Fig. 4: Cumulative regret of SER3, UCB and EXP3 on the Problem 2.

This problem challenges SER3 against SE, UCB and EXP3. SER3 achieves a low cumulative regret, successfully eliminating sub-optimal arms at the beginning of the run. Contrarily, SE is tricked by the periodicity of the sinusoidal means and eliminates the optimal arm. The deterministic policy of UCB is not adapted to the non-stationarity of rewards and thus the algorithm suffers from a high regret. The unbiased estimators of EXP3 enable the algorithm to quickly converge on the best arm. However, EXP3 suffers from a linear regret due to its constant exploration until the end of the game.

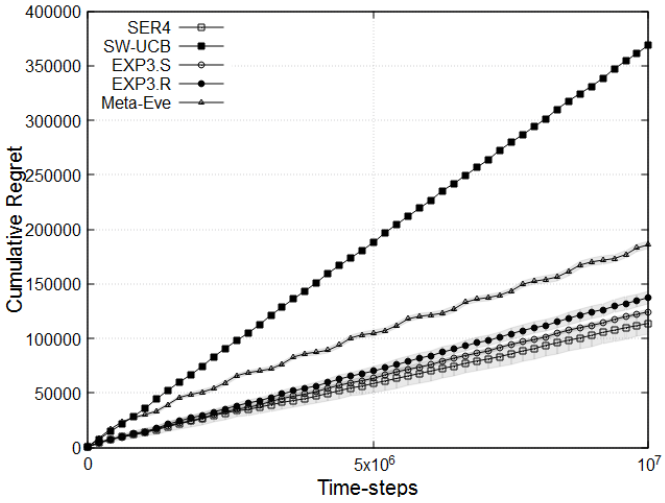
### 5.2 Problem 2: Decreasing means with positive gap

The optimal arm $k^*$ does not change during the game. The mean of all suboptimal arms is $\mu(t) = 0.95 - \min(0.45, 10^{-7}t)$.

On this problem, SER3 is challenged against SE, UCB and EXP3. SER3 achieves a low cumulative regret, successfully eliminating sub-optimal arms at the beginning of the run. Contrarily to problem 1, mean rewards evolve slowly and SUCCESSIVE ELIMINATION (SE) achieves the same level of performance than SER3. Similarly to problem 1, UCB achieves a high cumulative regret. The cumulative regret of EXP3 is low at the end of the game but would still increase linearly with time.

### 5.3 Problem 3: Decreasing means with arm switches

At every turn, the optimal arm $k^*(t)$ changes with a probability of $10^{-6}$. In expectation, there are 10 switches by run. The mean of all suboptimal arms is $\mu(t) = 0.95 - \min(0.45, 10^{-7}(t[\text{mod } 10^6])$.




(a) Cumulative regret of SER4, SW-UCB, EXP3.S, EXP3.R and META-EVE on the Problem 3.

Fig. 5

On problem 3, SER4 is challenged against SW-UCB, EXP3.S, EXP3.R and META-EVE. The probability of reset of SER4 is $\varphi = 5^{-5}$. The size of the window of SW-UCB is $10^5$. The historic considered by EXP3.R is $H = 4 \cdot 10^5$ and the regularization parameter of EXP3.S is $\alpha = 10^{-5}$.

SER4 obtains the lowest cumulative regret, confirming the random resets approach to overcome switches of best arm. SW-UCB suffers from the same issues as UCB in previous problems and obtains a very high regret. Constant changes of mean cause META-EVE to reset very frequently and to obtain a lower regret than SW-UCB. EXP3.S and EXP3.R achieves both low regrets but EXP3.R suffers from

the large size of historic needed to detect switches with a gap of $\Delta$. We can notice that the randomization of resets in SER4, while allowing to achieve the best performances on this problem, involve a highest variance. Indeed, on some runs, a reset may occur lately after a best arm switch whereas the use of windows or regularization parameters will be more consistent.

## 6 Conclusion

We proposed a new formulation of the multi-armed bandit problem that generalize the stationary stochastic, piecewise-stationary and adversarial bandit problems. This formulation allows to manage difficult cases, where the means rewards and/or the best arm may change at each turn of the game. We studied the benefit of *random shuffling* in the design of sequential elimination bandit algorithms. We showed that the use of *random shuffling* extends their range of application to a new class of best arm identification problems involving non-stationary distributions, while achieving the same level of guarantees than SE with stationary distributions. We introduced SER3 and extended it to the switching bandit problem with SER4 by adding a probability of restarting the best arm identification task. We extended the definition of the sample complexity to include switching policies. Up to our knowledge, we proved the first sample complexity based upper-bound for the best arm identification problem with arm switches. The upper-bound over the cumulative regret of SER4 depends only of the number $N-1$ of arm switches, as opposed to the number of distribution changes $M-1$ in SW-UCB ($M \geq N$ can be of order $T$ in our setting). The algorithm EXP3.R also achieves a competitive regret bound. The adversarial nature of EXP3 makes it robust to non-stationarity and the detection test accelerates the switch when the optimal arm changes while allowing convergence of the bandit algorithm during periods where the best arm does not change.

## 7 Acknowledgement

This work was supported by

- Team TAO (CNRS - Inria Saclay, Île de France - LRI)
- Team Profiling and Data-mining (Orange Labs)

## References

1. Allesiardo, Robin, & Féraud, Raphaël. 2015. EXP3 with Drift Detection for the Switching Bandit Problem. *In: 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA 2015)*.
2. Auer, Peter, Cesa-Bianchi, Nicolò, & Fischer, Paul. 2002a. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, **47**(2-3), 235–256.
3. Auer, Peter, Cesa-Bianchi, Nicolò, Freund, Yoav, & Schapire, Robert E. 2002b. The Nonstochastic Multiarmed Bandit Problem. *SIAM J. Comput.*, **32**(1), 48–77.
4. Bubeck, S., & Cesa-Bianchi, N. 2012. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *In: Foundations and Trends in Machine Learning*.
5. Bubeck, Sébastien, & Slivkins, Aleksandrs. 2012. The Best of Both Worlds: Stochastic and Adversarial Bandits. *Pages 42.1–42.23 of: COLT 2012 - The 25th Annual Conference on Learning Theory, June 25-27, 2012, Edinburgh, Scotland*.
6. Even-Dar, Eyal, Mannor, Shie, & Mansour, Yishay. 2006. Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems. *Journal of Machine Learning Research*, **7**.
7. Féraud, Raphaël, Allesiardo, Robin, Urvoy, Tanguy, & Clérot, Fabrice. 2016. Random Forest for the Contextual Bandit Problem. *AISTATS*.
8. Garivier, Aurélien, & Moulines, Eric. 2011. On Upper-Confidence Bound Policies for Non-stationary Bandit Problems. *Pages 174–188 of: Algorithmic Learning Theory*.
9. Garivier, Aurélien, Kaufmann, Emilie, & Lattimore, Tor. December, 2016. On Explore-Then-Commit Strategies. *NIPS 2016*, **30**.
10. Hartland, C., Baskiotis, N., Gelly, S., Teytaud, O., & Sebag, M. 2006. Multi-armed Bandit, Dynamic Environments and Meta-Bandits. *In: Online Trading of Exploration and Exploitation Workshop, NIPS*.
11. Hoeffding, Wassily. 1963. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, **58**(301), 13–30.
12. Kaufmann, Emilie, Cappé, Olivier, & Garivier, Aurélien. Jan. 2016. On the Complexity of Best Arm Identification in Multi-Armed Bandit Models. *Journal of Machine Learning Research*, **17**(1), 1–42.
13. Kocsis, L., & Szepesvári, C. 2006. Discounted UCB. *In: 2nd PASCAL Challenges Workshop*.
14. Neu, Gergely. 2015. Explore no more: improved high-probability regret bounds for non-stochastic bandits. *NIPS*.
15. Seldin, Yevgeny, & Slivkins, Aleksandrs. 2014. One Practical Algorithm for Both Stochastic and Adversarial Bandits. *In: 31th Intl. Conf. on Machine Learning (ICML)*.
16. Serfling, R.J. 1974. Probability Inequalities for the Sum in Sampling without Replacement. *Pages 39–48 of: The Annals of Statistics, Vol 2, No.1*.
17. Thompson, W.R. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, **25**, 285–294.
18. Yu, Jia Yuan, & Mannor, Shie. 2009. Piecewise-stationary Bandit Problems with Side Observations. *In:*

*Proceedings of the 26th Annual International Conference on Machine Learning*. ICML.

## A Summary of the contributions

We provide in Table 1 and 2 a brief summary of the existing results regarding the performance of a few algorithms, together with the contributions of this article, that are indicated in bold.

In both tables, $T$ is the time horizon, assumed to be know, $K$ the number of arms, $\Delta$ is the gap, and $\delta$ is the probability of success of the algorithm. $C$ is quantity similar to the gap, described in subsection 4.2.4. Finally, $M$ is the number of breakpoints (the mean reward of an arm changes) and $N$ the number of best arm switches.

Table 1: Overview of the different bandit algorithms for policies with unique best arm

| Algorithms | Regret | Sample Complexity | Non Stationarity |
|---|---|---|---|
| *State of the art* | | | |
| UCB | $O\left(\Delta^{-1}K\log(T)\right)$ | X | No |
| SE | $O\left(\Delta^{-1}K\log(TK/\Delta)\right)$ | $O\left(\Delta^{-2}K\log(TK/\Delta)\right)$ | No |
| EXP3 | $O\left(\sqrt{KT\log K}\right)$ | X | Yes |
| EXP3++ | $O(\Delta^{-1}K\log^3 T)+O(\Delta^{-3})$ | X | Yes |
| *Our contribution* | | | |
| SER3 | $O\left(\Delta^{-1}K\log(TK/\Delta)\right)$ | $O\left(\Delta^{-2}K\log(TK/\Delta)\right)$ | Yes |

Table 2: Overview of the different bandit algorithms for policies with switching best arm

| Algorithms | Regret | Sample Complexity | Non Stationarity between breakpoints |
|---|---|---|---|
| *State of the art* | | | |
| SW-UCB | $O\left(\Delta^{-1}\sqrt{MT\log T}\right)$ | X | No |
| EXP3.S | $O\left(\sqrt{NKT\log(KT)}\right)$ | X | Yes |
| *Our contributions* | | | |
| SER4 | $O\left(\Delta^{-1}\sqrt{NKT\log(KT)}\right)$ | $O\left(\Delta^{-2}\sqrt{NK\delta^{-1}\log(K\delta^{-1})}\right)$ | Yes |
| EXP3.R | $O\left(3NCK\sqrt{TK\log T}\right)$ | X | Yes |

## B Technical results

### B.1 Proof of Theorem 1 and Theorem 2

*Proof.* Theorem 1 is a special case of Theorem 2. For Theorem 1, for every $k$ and every $t$, $B=0$, $B_k=0$ and $b_k(t)=0$.

The proof consists of three main steps. The first step makes explicit the conditions leading to the elimination of an arm from the set. The second step shows that the optimal arm will not be eliminated with high probability. Finally, the third step shows that a sub-optimal arm will be eliminated after at most a critical number of steps $\tau^*$, which then allows to derive an upper-bound on the sample complexity.

**Step 1. Conditions for the elimination of an arm.**
From Hoeffding's inequality, for any deterministic round-robin length $\tau$ and arm $k$ we have:

$$P\left(|\hat{\mu}_k - \mathbb{E}[\hat{\mu}_k]| \geq \epsilon_\tau\right) \leq 2\exp\left(-2\epsilon_\tau^2\tau\right) .$$

where $\mathbb{E}$ denotes the expectation with respect to the distribution $D_y$. By setting

$$\epsilon_t = \sqrt{\frac{1}{2\tau}\log\left(\frac{4K\tau^2}{\delta}\right)}, \text{ we have:}$$

$$P\left(|\hat{\mu}_k - \mathbb{E}[\hat{\mu}_k(\tau)]| \geq \epsilon_t\right) \leq 2\exp\left(-2\sqrt{\frac{1}{2\tau}\log\left(\frac{4K\tau^2}{\delta}\right)}^2\tau^2\right) = \frac{\delta}{2K\tau^2} .$$

Applying Hoeffding's inequality for each round-robin size $\tau \in \mathbb{N}^\star$, applying a standard union bound and using that $\sum_{\tau=1}^{\infty} 1/\tau^2 = \pi^2/6$, the following inequality holds simultaneously for any $\tau$ with a probability at least $1 - \frac{\delta\pi^2}{12K}$:

$$\hat{\mu}_k(\tau) - \epsilon_\tau \leq \mathbb{E}[\hat{\mu}_k] \leq \hat{\mu}_k(\tau) + \epsilon_\tau \,. \tag{19}$$

Let $S_i \subset \{1, \ldots, K\}$ be the set containing all the arms that are not eliminated by the algorithm at the start of the $i^{\text{th}}$ round-robin. By construction of the algorithm, an arm $k'$ remains in the set of selected arms as long as for each arm $k \in S_\tau - \{k'\}$:

$$\hat{\mu}_k(\tau) - \epsilon_\tau < \hat{\mu}_{k'}(\tau) + \epsilon_\tau \text{ and } \tau \geq \tau_{\min} \tag{20}$$

Combining (19) and the left inequality of (20), it holds on an event $\Omega$ of high probability

$$\mathbb{E}[\hat{\mu}_k(\tau)] - 2\epsilon_\tau < \mathbb{E}[\hat{\mu}_{k'}(\tau)] + 2\epsilon_\tau \,. \tag{21}$$

We denote $t_\tau$, the time-step where the $\tau^{\text{th}}$ round-robin starts ($t_\tau = 1 + \sum_{i=1}^{\tau-1} |S_i|$). Let us remind that $\mathbb{T}(\tau)$ is the set containing all possible realizations of $\tau$ sequences of round-robin. Each arm $k$ is played one time during each round-robin phase and thus $\tau$ observations per arm are available after $\tau^{\text{th}}$ round-robin phases. The empirical mean reward $\hat{\mu}_k(\tau)$ of each arm $k$ after the $\tau^{\text{th}}$ round-robin is:

$$\hat{\mu}_k(\tau) = \sum_{r \in \mathbb{T}(\tau)} \frac{1_{[\![r=[\tau]]\!]}}{\tau} \sum_{j=1}^{t_\tau + |S_\tau| - 1} y_k(j) 1_{[\![k=k_j]\!]} \,. \tag{22}$$

Decomposing the second sum in round-robin phases and taking the expectation with respect to the reward distribution $D_y$ we have:

$$\mathbb{E}_{D_y}[\hat{\mu}_k(\tau)] = \sum_{r \in \mathbb{T}(\tau)} \frac{[\![r=[\tau]]\!]}{\tau} \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i + |S_\tau| - 1} (\mu_k(j) - b_k(t)) [\![k = k_j]\!] \,. \tag{23}$$

Taking the expectation of equation (23) with respect to the randomization of the round-robin we have:

$$\mathbb{E}[\hat{\mu}_k(\tau)] = \left( \sum_{r \in \mathbb{T}(\tau)} \frac{[\![r=[\tau]]\!]}{\tau} \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i + |S_\tau| - 1} \frac{\mu_k(j)}{|S_i|} \right) - \frac{B_k}{\tau} \,. \tag{24}$$

Now, under the event $\Omega$ for which (21) holds for $k$ and $k'$, we deduce by using (24) that

$$\sum_{r \in \mathbb{T}(\tau)} \frac{[\![r=[\tau]]\!]}{\tau} \left( \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i + |S_\tau| - 1} \frac{\mu_k(j)}{|S_i|} - \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i + |S_\tau| - 1} \frac{\mu_{k'}(j)}{|S_i|} \right) < 4\epsilon_\tau + \frac{B_k}{\tau} - \frac{B_{k'}}{\tau} + \frac{B}{\tau} \,. \tag{25}$$

Let us introduce the following mean-gap quantity

$$\Delta_{k,k'}([\tau]) = \sum_{r \in \mathbb{T}(\tau)} \frac{1_{[\![r=[\tau]]\!]}}{\tau} \left( \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i + |S_\tau| - 1} \frac{\mu_k(j)}{|S_i|} - \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i + |S_\tau| - 1} \frac{\mu_{k'}(j)}{|S_i|} \right) .$$

Replacing the value of $\epsilon_t$ in (25), it comes

$$\Delta_{k,k'}([\tau]) < 4\sqrt{\frac{1}{2\tau} \log\left(\frac{4K\tau^2}{\delta}\right)} + \frac{B_k}{\tau} - \frac{B_{k'}}{\tau} + \frac{B}{\tau} \,,$$

$$\Delta_{k,k'}([\tau])^2 < \frac{8}{\tau} \log\left(\frac{4K\tau^2}{\delta}\right) + \frac{B_k}{\tau} - \frac{B_{k'}}{\tau} + \frac{B}{\tau} \,. \tag{26}$$

An arm will be eliminated if (26) becomes false and if $\tau \geq \tau_{\min}$.

**Step 2. The optimal arm is not eliminated.**
For $k' = k^*$ et $k \neq k^*$, in the worst case $B_k = 0$ and $B_{k'} = B$. After injecting those quantities in (26), we have :

$$\Delta_{k,k'}([\tau])^2 < \frac{8}{\tau} \log\left(\frac{4K\tau^2}{\delta}\right). \tag{27}$$

By assumption ($\Delta_{k,k^*}([\tau])$ is negative after $\tau_{\min}$), (27) is always true when $\tau \geq \tau_{\min}$,implying that the optimal arm will always remain in the set with a probability of at least $1 - \frac{\delta}{K}$ for all $\tau$.

**Step 3. The elimination of sub-optimal arms.**
If the arm $k'$ still remain in the set, it will be eliminated if inequality (26) is not satisfied and if $\tau^* \geq \tau_{\min}$.

Let us consider $k = k^*$, $k' \neq k^*$, and define the quantity

$$\Delta_k([\tau]) = \sum_{r \in \mathbb{T}(\tau)} \frac{1_{[\![r=[\tau]]\!]}}{\tau} \left( \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i+|S_\tau|-1} \frac{\mu_k(j)}{|S_i|} - \sum_{i=1}^{\tau} \sum_{j=t_i}^{t_i+|S_\tau|-1} \frac{\mu_{k'}(j)}{|S_i|} \right).$$

In the worst case, $B_{k^*} = B$ et $B_k = 0$. Using equation (26) we obtain the condition to invalidate to eliminate the arm of index $k$:

$$\Delta_{k,k'}([\tau])^2 < \frac{8}{\tau} \log\left(\frac{4K\tau^2}{\delta}\right) + \frac{2B}{\tau}. \tag{28}$$

Let us also introduce for convenience the critical value

$$\tau_1^* = \frac{64^2}{\Delta_k([\tau])^2} \log\left(\frac{16K}{\delta \Delta_k([\tau])}\right).$$

Notice that $\tau_1^* \geq \tau_{\min}$, satisfying one of the two conditions needed to eliminate an arm.

We introduce the following quantity

$$C_1(t) = \frac{8}{\tau} \log\left(\frac{4K\tau^2}{\delta}\right).$$

For $\tau = \tau_1^*$, we derive the following bound

$$\begin{aligned}
C_1(\tau_1^*) &= \frac{8\Delta_k([\tau])^2}{64^2 \log \frac{16K}{\delta \Delta_k([\tau])}} \left( \log \frac{4K}{\delta} + 2 \log \frac{64}{\Delta_k([\tau])^2} + 2 \log\log \frac{16K}{\delta \Delta_k([\tau])} \right), \\
&= \frac{8\Delta_k([\tau])^2}{64^2 \log \frac{16K}{\delta \Delta_k([\tau])}} \left( \log \frac{4K}{\delta} - 4 \log \Delta_k([\tau]) + 24 \log 2 + 2 \log\log \frac{16K}{\delta \Delta_k([\tau])} \right), \\
&\leq \frac{8\Delta_k([\tau])^2}{64^2 \log \frac{16K}{\delta \Delta_k([\tau])}} \left( 4 \log \frac{16K}{\delta \Delta_k([\tau])} + 24 \log 2 + 2 \log\log \frac{16K}{\delta \Delta_k([\tau])} \right).
\end{aligned}$$

We remark that for $X > 8$ we have

$$24 \log 2 + 2 \log\log X < 8 \log X.$$

Hence, provided that for $K \geq 2$, $\delta \in (0, 0.5]$ and $\Delta_k([\tau]) > 0$, we have $\frac{4K}{\delta \Delta_k([\tau])} > 8$ and

$$\begin{aligned}
C_1(\tau_1^*) &\leq \frac{8\Delta_k([\tau])^2}{64^2 \log \frac{16K}{\delta \Delta_k([\tau])}} \left( 16 \log \frac{16K}{\delta \Delta_k([\tau])} \right) \\
&\leq \frac{\Delta_k([\tau])^2}{512}.
\end{aligned} \tag{29}$$

As $C_1(\tau_1^*)$ is strictly decreasing with regard to $t$, (29) is true for all $\tau > \tau_1^*$.

When $t > \tau_1^*$, it exists $C_2(t)$ such as:

$$\Delta_k([\tau])^2 = C_1(t) + C_2(t).$$

For invalidating 28, we must find a value $\tau_2^* > \tau_1^*$ such as:

$$\tau_2^* \geq \frac{4B}{C_2(t_2^*)} \tag{30}$$

As $C_2(\tau) = \Delta_k([\tau])^2 - C_1(\tau)$, we have $C_2(\tau) \geq \Delta_k([\tau])^2 - \frac{\Delta_k([\tau])^2}{512}$ and:

$$\tau \geq \frac{2048B}{511\Delta_k([\tau])^2}$$

For $\tau = \tau_2^*$ with:

$$\tau_2^* = \frac{64^2}{\Delta_k([\tau])^2} \log\left(\frac{16K}{\delta\Delta_k([\tau])}\right) + \frac{5B}{\Delta_k([\tau]^2)}\,. \tag{31}$$

(30) is true, invalidating (28) and invalidating (26) and involving the elimination of the suboptimal arms $k$ with a probability at least $1 - \delta/K$.

We conclude the proof by summing over all the arms, taking the union bound and lower-bounding all $\Delta_k([\tau])$ by

$$\Delta = \min_{[\tau]\in\mathbb{T}(\tau),k} \sum_{r\in\mathbb{T}(\tau)} \frac{[\![r=[\tau]]\!]}{\tau} \left(\sum_{i=1}^{\tau}\sum_{j=t_i}^{t_i+|S_\tau|-1} \frac{\mu_k(j)}{|S_i|} - \sum_{i=1}^{\tau}\sum_{j=t_i}^{t_i+|S_\tau|-1} \frac{\mu_{k'}(j)}{|S_i|}\right)\,. \tag{32}$$

□

B.2 Proof of Corollary 1

*Proof.* We first provide the proof of the distribution dependent upper-bound.

The pseudo cumulative regret of the algorithm is:

$$R(T) = \sum_{k\neq k^*}\sum_{i=1}^{\tau}\sum_{t=t_i}^{t_i+|S_i|-1} \Delta_{k^*,k}(t) 1_{[\![k=k_t]\!]}\,. \tag{33}$$

Taking in each round-robin the expectation of the corresponding random variable $k_t$ with respect to the randomization of the round-robin (denoted by $\mathbb{E}_{k_t}$), it comes:

$$\begin{aligned}\mathbb{E}[R(T)] &= \mathbb{E}\left[\sum_{k\neq k^*}\sum_{i=1}^{\tau}\sum_{t=t_i}^{t_i+|S_i|-1} \mathbb{E}_{k_t}[\Delta_{k^*,k}(t)1_{[\![k=k_t]\!]}]\right] \\ &= \mathbb{E}\left[\sum_{k\neq k^*}\sum_{i=1}^{\tau}\sum_{t=t_i}^{t_i+|S_i|-1} \frac{\Delta_{k^*,k}(t)}{|S_i|}\right].\end{aligned}$$

$$\mathbb{E}[R(T)] = \mathbb{E}\left[\sum_{k\neq k^*} \tau \underbrace{\frac{1}{\tau}\sum_{i=1}^{\tau}\sum_{t=t_i}^{t_i+|S_i|-1} \frac{\Delta_{k^*,k}(t)}{|S_i|}}_{\Delta_k^*}\right] = \mathbb{E}\left[\sum_{k\neq k^*} \tau\Delta_k^*\right]. \tag{34}$$

The penultimate step of the proof of Theorem 1 allows us to upper-bound $\tau$ with the previously introduced critical value $\tau^*$ on an event of high probability $1-\delta$, while the cumulative regret is controlled by the trivial upper bound $T$ on the complementary event of probability not higher than $\delta$, leading to:

$$\mathbb{E}[R(T)] \leq \sum_{k\neq k^*} \frac{64}{\Delta_k^2} \log\left(\frac{4K}{\delta\Delta_k}\right)\Delta_k + \delta T\,. \tag{35}$$

We conclude the proof of the distribution dependent upper-bound by setting $\delta = 1/T$ and :

$$\mathbb{E}[R(T)] = O\left(\frac{K-1}{\Delta}\log(\frac{KT}{\Delta})\right)\,, \tag{36}$$

with $\Delta = \min_{[\tau],k} \frac{1}{\tau} \sum_{i=1}^{\tau} \sum_{t=t_i}^{t_i+|S_i|-1} \frac{\Delta_{k^*,k}(t)}{|S_i|}$.

We now upper-bound the regret in the worst case in order to derive a distribution independent bound. To this end, we consider a sequence that ensures that, with high probability, no suboptimal arm is eliminated by the algorithm at the end of the $T$ rounds, while maximizing the instantaneous regret. According to (21) an arm is not eliminated as long as

$$\mathbb{E}[\hat{\mu}_k(\tau)] - \mathbb{E}[\hat{\mu}_{k'}(\tau)] < 4\epsilon_\tau \,. \tag{37}$$

By injecting (37) in (34) and replacing $\epsilon_\tau$ by its value $\sqrt{\frac{2}{\tau}\log\left(\frac{4K\tau^2}{\delta}\right)}$ we obtain:

$$\mathbb{E}[R(T)] < \sum_{k \neq k^*} \tau 4 \sqrt{\frac{2}{\tau}\log\left(\frac{4K\tau^2}{\delta}\right)} + \delta T \,. \tag{38}$$

The non-elimination of sub-optimal arms involves $\tau = \frac{T}{K}$ and by setting $\delta = \frac{1}{T}$ we obtain the distribution independent upper-bound:

$$\mathbb{E}[R(T)] < (K-1)\frac{T}{K} 4 \sqrt{\frac{K}{T}\log(\frac{4T^3}{K})} + 1 \,, \tag{39}$$

$$\mathbb{E}[R(T)] = O\left(\sqrt{TK\log\frac{T}{K}}\right) \,. \tag{40}$$

□

### B.3 Proof of Theorem 3

*Proof.* In order to prove Theorem 3, we consider the following quantities:

- The expected number of times when the estimators are reseted: $N_{\text{reset}} = \varphi T$.
- The sample complexity needed to find the best arm between each reset is $S_{\text{SER3}} = O\left(\frac{K}{\Delta^2}\log(\frac{K}{\delta\Delta})\right)$.
- The time before a reset, that follows a negative binomial distribution of parameters $r = 1$ and $p = 1 - \varphi$. Its expectation is upper-bounded by $1/\varphi$.
- The number of arm switches: $N - 1$.

The sample complexity of SER4 is the total number of time-steps spent sampling an arm added to the time between each switch and reset.

Taking the expectation with respect to the randomization of resets, we obtain an upper-bound on the expected number of suboptimal plays given by

$$O\left(\frac{\varphi TK}{\Delta^2}\log\left(\frac{K}{\delta\Delta}\right) + \frac{N}{\varphi}\right) \,. \tag{41}$$

The first term is the expectation of the total number of time-steps required by the algorithm in order to find the best arms at its initialization and then after each reset of the algorithm. The second term is the expected total number of steps lost by the algorithm when not resetting the algorithm after the $N-1$ arm switches.

We obtain the final statement of the Theorem by setting $T = \frac{1}{\delta}$. □

### B.4 Proof of Corollary 3

*Proof.* Converting Corollary 2 into a distribution dependent upper-bound on the cumulative regret is straightforward by setting $\delta = \frac{1}{T}$, replacing the sample complexity in the proof of Theorem 3 by the cumulative regret and using the upper-bound of Corollary 1.

$$\mathbb{E}[R(T)] = O\left(\frac{\varphi TK}{\Delta}\log\left(\frac{KT}{\Delta}\right) + \frac{N}{\varphi}\right) \,. \tag{42}$$

Setting $\varphi = \sqrt{\frac{N}{TK\log(KT)}}$ and assuming $\Delta \geq \frac{1}{KT}$ we obtain the final statement of the theorem:

$$\mathbb{E}[R(T)] = O\left(\frac{\sqrt{NTK\log(KT)}}{\Delta}\right). \tag{43}$$

We also derive below a distribution independent upper-bound. We introduce some notations, $N_{\text{reset}}$ is the number of resets, $\tau_i^{\text{reset}}$ is the number of round-robin phases between the $i^{\text{th}}$ and the $(i+1)^{\text{th}}$ resets and $L_n$ is the number of timesteps before a reset after the $n^{\text{th}}$ arm switch.

When the resets are fixed, the expected cumulative regret is:

$$\mathbb{E}[R(T)] < \mathbb{E}\left[\sum_{i=1}^{N_{\text{reset}}+1}(K-1)\tau_i^{\text{reset}}4\sqrt{\frac{2}{\tau_i^{\text{reset}}}\log(\frac{4(\tau_i^{\text{reset}})^2}{\delta})} + \sum_{n=1}^{N} L_n + \delta T\right], \tag{44}$$

$$\mathbb{E}[R(T)] < \mathbb{E}\left[\sum_{i=1}^{N_{\text{reset}}+1}\underbrace{(K-1)4\sqrt{2\tau_i^{\text{reset}}\log(\frac{4(\tau_i^{\text{reset}})^2}{\delta})}}_{f(\tau_i^{\text{reset}})}\right] + \mathbb{E}\left[\sum_{n=1}^{N} L_n\right] + \delta T\,. \tag{45}$$

At this point, we note that $\{\tau_i^{\text{reset}}\}_i$ is an i.i.d sequence of random variables and that $N_{\text{reset}}$ is a random stopping time with respect to this sequence. Moreover, $f$ is a concave function. We can thus apply Wald's equation followed by Jensen's inequality and deduce that

$$\begin{aligned}\mathbb{E}[\sum_{i=1}^{N_{\text{reset}}+1} f(\tau_i^{\text{reset}})] &\leq \mathbb{E}[N_{\text{reset}}+1]\mathbb{E}[f(\tau_1^{\text{reset}})] \\ &\leq \mathbb{E}[N_{\text{reset}}+1]f(\mathbb{E}[\tau_1^{\text{reset}}])\,.\end{aligned}$$

We upper-bound $\log(\frac{4(\tau_i^{\text{reset}})^2}{\delta})$ by $\log(\frac{4T^2}{\delta K^2})$ and set $\delta = \frac{1}{T}$. As $\mathbb{E}[N_{\text{reset}}] = \varphi T$, $\mathbb{E}[\tau_1^{\text{reset}}] = \frac{1}{\varphi K}$ and $\mathbb{E}[L_n] \leq \frac{1}{\varphi}$, we have

$$\mathbb{E}[R(T)] < 4(\varphi T+1)\sqrt{\frac{2}{\varphi}K\log\left(\frac{4T^3}{K^2}\right)} + \frac{N}{\varphi} + 1\,. \tag{46}$$

The previous equation makes appear a trade-off in $\varphi$, and we set $\varphi = \frac{\sqrt{N}}{T^{2/3}}$.

Finally we have shown that

$$\mathbb{E}[R(T)] = O\left(T^{2/3}\sqrt{NK\log\frac{T}{K}}\right). \tag{47}$$

□

### B.5 Proof of Lemma 2

*Proof.* We justify our detection test by considering an observation of a reward through $\gamma$-exploration as a drawing in an urn without replacement. More specifically, when all the necessary observations are collected, the detection test procedure is called. During the interval, rewards were draw from $m$ different distributions of mean $\mu_0^k(I), ..., \mu_m^k(I)$ . We denote $t_i$ the steps where the mean reward starts being $\mu_i^k(I)$ and $t_{m+1}$ the time step of the call. When the test is called, all $x_k(t)$ have a probability $(t_{i+1}-t_i)/(t_{m+1}-t_0)$ to be drawn from the distribution of mean $\mu_i^k(I)$. The mean $\mu^k(I)$ of the urn corresponding to the action $k$ is:

$$\mu^k(I) = \sum_{i=1}^{m}\frac{t_{i+1}-t_i}{t_{m+1}-t_0}\mu_i^k(I)\,. \tag{48}$$

At each time step, by assumption , the mean reward of the best arm is away by $4\epsilon$ from any suboptimal arms. Consequently, the difference between the mean reward of the urn of the optimal arm $k^*$ and that of an another arm $k$ is at least $4\epsilon$ if the best arm doesn't change during the interval.

$$\mu^k(I) \leq \sum_{i=1}^{m} \frac{t_{i+1} - t_i}{t_{m+1} - t_0}(\mu_i^{k_S^*} - 4\epsilon) \leq \mu^{k_S^*}(I) - 4\epsilon\,. \tag{49}$$

The following arguments prove the equivalence between the detection and the optimality of $k_S^*$ with high probability.

Applying the Serfling inequality [16], we have:

$$P(\hat{\mu}^{k_S^*}(I) + \epsilon \geq \mu^{k_S^*}(I)) \leq e^{\frac{-2n\epsilon^2}{1-\frac{n-1}{U}}} \leq e^{-2n\epsilon^2} = \delta \tag{50}$$

and

$$P(\hat{\mu}^k(I) - \epsilon \leq \mu^k(I)) \leq \delta\,, \tag{51}$$

where $n = \frac{\gamma H}{K}$is the number of observation and $U$ the size of the urn.

$$\mu^{k_S^*}(I) - \mu^k(I) \geq 4\epsilon \tag{52}$$

and with probability at least $1 - 2\delta$,

$$\hat{\mu}^{k_S^*}(I) + \epsilon \geq \mu^{k_S^*}(I) \tag{53}$$

and

$$\hat{\mu}^k(I) - \epsilon \leq \mu^k(I) \tag{54}$$

Summing (55), (53) and (54) we obtain:

$$\hat{\mu}^{k_S^*}(I) - \hat{\mu}^k(I) \geq 2\epsilon\,? \tag{55}$$

This ensures, with high probability, a positive test if $\hat{\mu}^{k_{\max}}$ is not the optimal arm.

Reciprocally, we also have

$$\hat{\mu}^k(I) - \hat{\mu}^{k_S^*}(I) \leq -2\epsilon\,. \tag{56}$$

ensuring, with high probability, a negative test if $\hat{\mu}^{k_{\max}}$ is the optimal arm.

□

### B.6 Proof of Theorem 4

*Proof.* First we obtain the main structure of the bound. In the following, $L(T)$ denotes the expected number of intervals after a best action change occurs before detection and $F(T)$ denotes the expected number of false detections up to time $T$. Using the same arguments as [18] we deduce the form of the bound with drift detector from the classical EXP3 bound. If there are $N-1$ changes of best arm. Therefore the expectation of the number of resets over an horizon $T$ is upper bounded by $N - 1 + F(T)$. The regret of EXP3 on these periods is $(e-1)\gamma T + \frac{K \log K}{\gamma}$ [3]. While our optimal policy plays the arm with the highest mean, the optimal policy of EXP3 plays the arm associated with the actual highest cumulative reward. As

$$\sum_{t=T_S}^{T_{S+1}-1} x_{k_S^*}(t) \leq \max_k \sum_{t=T_S}^{T_{S+1}-1} x_k(t)\,, \tag{57}$$

the gain of our optimal policy is upper bounded by the gain the EXP3 optimal policy. Summing over each periods we obtain $(e-1)\gamma T + \frac{(N-1+F(T))K \log K}{\gamma}$.

The regret also include the delay between a best arm change and its detection. To evaluate the expected size of the intervals between each call of the detection test, we consider an hypothetical algorithm that collects only the observations of one arm and then proceeds on the next arm until collecting all the observations. The $\gamma$-observation are drawn with a probability $\frac{\gamma}{K}$

and $\frac{\gamma H}{K}$ observations are needed per action. The expectation of the number of failures before collecting $\frac{\gamma H}{K}$ $\gamma$-observations follows a negative binomial distribution of expectation

$$\frac{\gamma H}{K}(1-\frac{\gamma}{K})\frac{K}{\gamma} = H - \frac{\gamma H}{K}\,. \tag{58}$$

The expectation of the number of steps at the end of the collect is the number of success plus the expected number of failures:

$$\frac{\gamma H}{K} + H - \frac{\gamma H}{K} = H\,. \tag{59}$$

Summing over all arms gives a total expectation of $HK$. Because our algorithm collects $\gamma$-observations from any arm at any step, on a same sequence of drawings, our algorithm will collect the required observations before the hypothetical algorithm. By consequence, the expectation of the time between each query of the detection test is upper bounded by $HK$ and lower bounded by $H$, the expected time of collect for one arm. There are $N-1$ best action changes and the detections occur at most $\lceil L(T)\rceil HK$ time steps after the drifts. Finally, there are also at most $N-1$ intervals where the optimal arm switches. In these intervals we don't have any guarantee on the test behavior due to this change. In the worst case, the test doesn't detect the drift and we set the instantaneous regret to 1.

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{(N-1+F(T))K\log K}{\gamma} + (N-1)HK(\lceil L(T)\rceil + 1)\,. \tag{60}$$

We now bound $F(T)$ and $L(T)$. Confidence intervals hold with probability $1-\delta$ and they are used $K$ times at each detection test. The maximal number of calls of the test up to time horizon $T$ is $\frac{T}{H}+1$. Using the union bound we deduce $F(T) \leq K\delta(\frac{T}{H}+1)$. $L(T)$ is the first occurrence of the event DETECTION after a drift. When a drift occurs, Lemma 2 ensures the detection happens with a probability $1-2\delta$. We have $L(T) \leq \frac{1}{1-2\delta}$.

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{\left(N-1+\frac{K\delta T}{H}+K\delta\right)K\log K}{\gamma} + (N-1)HK\left(\frac{1}{1-2\delta}+1\right)\,. \tag{61}$$

□

B.7 Proof of Corollary 4

*Proof.* We set $\delta = \sqrt{\frac{\log T}{KT}}$ and $H = C\sqrt{T\log T}$ in Theorem 4

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{(N-1+(C+1)\sqrt{K})K\log K}{\gamma} + 3(N-1)CK\sqrt{T\log T}\,. \tag{62}$$

Finally, setting $\gamma = \sqrt{\frac{K\log K\log T}{T}}$ we obtain:

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\sqrt{TK\log K\log T} + N\sqrt{TK\log K} + (C+1)K\sqrt{T\log K} + 3NCK\sqrt{T\log T}\,. \tag{63}$$

□