EXP3 with Drift Detection for the Switching Bandit Problem

Robin Allesiardo*†, Raphaël Féraud*
*Orange Labs, 22300 Lannion, FRANCE
†TAO - INRIA, LRI, Université Paris-Sud, CNRS, 91405 Orsay, FRANCE

***Abstract*—The multi-armed bandit is a model of exploration and exploitation, where one must select, within a finite set of arms, the one which maximizes the cumulative reward up to the time horizon T. For the *adversarial* multi-armed bandit problem, where the sequence of rewards is chosen by an oblivious adversary, the notion of best arm during the time horizon is too restrictive for applications such as ad-serving, where the best ad could change during time range. In this paper, we consider a variant of the *adversarial* multi-armed bandit problem, where the time horizon is divided into unknown time periods within which rewards are drawn from stochastic distributions. During each time period, there is an optimal arm which may be different from the optimal arm at the previous time period. We present an algorithm taking advantage of the constant exploration of EXP3 to detect when the best arm changes. Its analysis shows that on a run divided into N periods where the best arm changes, the proposed algorithms achieves a regret in $O(N\sqrt{T \log T})$.**

I. Introduction

The multi-armed bandit problem is a repeating game where a player chooses one of the K arms (or actions) to play and receives a reward corresponding to the chosen action. In order to find the most profitable action, the player needs to explore different choices but also needs to exploit the best action identified so far in order to maximize her cumulative reward. The quality of the player policy at time horizon T is measured in terms of regret. The regret is the difference between the cumulative rewards of the player and the one that could be acquired by a policy assumed optimal such as "*play only the arm with the highest expected reward*".

In the *stochastic* formulation, the rewards are generated independently from unknown distributions associated with each arm. The oldest approach, the Thompson Sampling algorithm [1], is based on a Bayesian approach. At each time step, it draws an action by sampling from estimated posterior reward distributions. This policy achieves logarithmic expected regret [2] and is asymptotically optimal [3]. The UCB algorithm [4] computes an *upper confidence bound* for each arm and plays the arm with the highest upper confidence bound. This elegant and efficient algorithm achieves logarithmic bound uniformly over time which is optimal.

In the *adversarial* formulation, the rewards are chosen in advance by an adversary. This setting was studied extensively in the seminal work of Auer et al [5] and Cesa-Bianchi and Lugosi [6]. The most popular algorithms for the adversarial multiarmed bandit problem are from the EXP3 family [5]. EXP3 achieves a regret in $O(\sqrt{T})$, which is optimal. The switching bandit problem considers breakpoints in the reward sequences chosen in advance by an adversary [7]. Between breakpoints, the rewards are drawn from stochastic distributions. This setting is well suited for applications such as ad-serving (where the apparition of new ads may have a significant impact on rewards of others).

Summary of the Contributions. In order to handle the switching bandit problem, our contribution consists in combining the adversarial bandit algorithm EXP3 with a concept drift detector. This algorithm, called EXP3.R, takes advantage of the exploration factor of EXP3 to evaluate unbiased estimations of the mean rewards. When a change of the best arm is detected, the weights of EXP3 are reset by reinitializing the algorithm. The provided analysis shows that our algorithm achieves a regret bound in $O(N\sqrt{T \log T})$.

II. Previous Works

The drawback of EXP3 is that it is built to find the best arm on the entire run. The notion of best arm during the time horizon is too restrictive for applications such as ad-serving. To handle this restriction, EXP3.S [5] uses a regularization method on the reward estimators to forget the past and ease arm switches. In addition, the analysis of EXP3.S allows the optimal policy to play N different arms during the run and shows a regret bound in $O(\sqrt{NT \log T})$. In DiscountedUCB [8], a discount factor γ is used to adapt UCB algorithm to the switching bandit problem. The use of a sliding windows to compute the index of UCB is another reasonable approach for this problem. These two adaptations of UCB were analyzed in [7]. They achieve a regret bound in $O(\sqrt{MT \log T})$, where $M-1$ is the number of distribution changes. To achieve such bounds, parameters of the algorithms have to be tuned with the knowledge of N or M. The lower bound of $\Omega(\sqrt{T})$ for the switching bandit problem has been demonstrated in [7] and previously described algorithms match it up to a factor of $\sqrt{\log T}$. Another approach proposed for the *scratch games* problem is to reset EXP3 at each birth or death of arms [9]. The extension of this approach to any arbitrary arm switch of the optimal policy necessitates to know when abrupt changes occur.

The change point detection has been intensively studied [10] for various applications including spam detection, fraud detection, meteorology, or finance. The monitored concept depends on the application. For instance, in online classification, the variations of the performance of the model are often used to detect a concept drift: in [11], the authors assume that an error of classification is a Bernoulli random variable to detect changes and in [12] the performance of the model on the training set and on a validation set is used to detect the drift. To deal with the partial information nature of the bandit problem, in Meta-Eve [13] the mean reward of the estimated best action is monitored. Page-Hinkley statistics are used to test if the time serie of means rewards of the best action can be attributed to a single statistical law or not. The drawback of this approach is that it does not handle the case of a suboptimal action becoming the best action. Confidence intervals are used in [14] to detect changes in the mean reward of each action. This algorithm achieves a regret bound in $O(N \log T)$ where $N-1$ is the number of changes during the run. This bound is obtained by the use of "side observations", i.e. information on the rewards of unplayed arms acquired with no impact on the regret. The Kullback-Leiber divergence can also be used as a drift detector [15], [16].

III. Optimal Policy and Cumulative Regret

Setting. Following SW-UCB [7], we define the switching bandit problem as a set of K possible actions, where $1 \leq k \leq K$ is the index of each action, and a sequence of T reward vectors $\mathbf{x}(t) = (x_1(t), ..., x_K(t)) \in \{0,1\}^K$. Each reward $x_k(t)$ is drawn from a Bernoulli distribution of mean $\mu^k(t)$. The $M-1$ time steps where $\exists k, \mu(t)^k \neq \mu^k(t+1)$ are called breakpoints. An adversary chooses the time steps when the breakpoints occur and then he draws the sequences of rewards. The action played at time t is denoted $k(t)$. The goal of an algorithm A is to maximize the cumulative gain at time horizon T defined by:

$$G_A = \sum_{t=1}^{T} x_{k(t)}(t)\,. \tag{1}$$

The optimal policy. The sequence of reward vectors is divided into $N \leq M$ sequences called segments. S is the index of the segment including the rounds $[T_S, T_{S+1})$. A segment S begins when $\arg\max_k \mu^k(T_S) \neq \arg\max_k \mu^k(T_S - 1)$. The optimal arm on the segment S is denoted k_S^*, where:

$$k_S^* = \arg\max_k \sum_{t=T_S}^{T_{S+1}-1} \mu^k(t)\,. \tag{2}$$

The gain of the optimal policy up to time T is denoted G^* and defined by:

$$G^* = \sum_{S=1}^{N} \sum_{t=T_S}^{T_{S+1}-1} x_{k_S^*}(t)\,. \tag{3}$$

The cumulative regret of an algorithm A measured against this optimal policy is:

$$R(A) = G^* - G_A\,. \tag{4}$$

Notice that

$$G^* \leq \sum_{S=1}^{N} \max_k \sum_{t=T_S}^{T_{S+1}-1} x_k(t)\,, \tag{5}$$

i.e. after the drawing by the adversary, the reward sequence of any arm might be greater than the sequence of the arm with the highest mean reward.

IV. Algorithm

While other algorithms use a passive approach through forgetting the past [5], [8], [7], we propose an active strategy, which consists in resetting the reward estimations when a change of the best arm is detected. First, we describe the adversarial bandit algorithm EXP3 [5], 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 1 EXP3

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

$$p_k(t) = (1-\gamma)\frac{w_k(t)}{\sum_{i=1}^{k} w_i(t)} + \frac{\gamma}{K}\,. \tag{6}$$

The weight $w_k(t)$ of each action k is:

$$w_k(t) = \exp\left(\frac{\gamma}{K}\sum_{j=t_r}^{t}\frac{x_k(j)}{p_k(j)}\llbracket k = k(j)\rrbracket\right)\,, \tag{7}$$

where t_r is the time steps of initialization or reset.

The EXP3 algorithm (see Algorithm 1) 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 may require a number of trial equal to the first period's length to switch his most played arm.

Coupled with a detection test, the EXP3 algorithm has several good properties. First in a non-stationary environment, we need a constant exploration to detect changes where a sub-optimal 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 required by

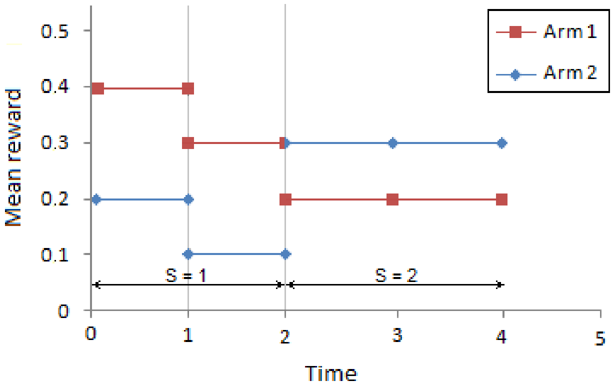

Fig. 1. Three drifts occur on this game but the optimal arm changes only one time. Here, $M = 3$ and $N = 2$.

an adversarial algorithm is lower than the one required by a stochastic bandit algorithm such as UCB. EXP3 can handle sequences where the reward distributions of arm change but the best arm remains the same. Hence, we will reset the algorithm only when the best arm changes (see Figure 1). Third, due to its adversarial nature, EXP3 is robust against test failures (non detection) or local non-stationarity.

The detection test (see Algorithm 2) 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 γ-observation an observation selected though the uniform distribution. Parameters γ, H and δ induce the minimal number of γ-observations by arm required to call a test of accuracy 2ϵ with a probability $1-\delta$. They will be fixed in the analysis (see Corollary 1) and the test validity is proved in Lemma 1. We denote $\hat{\mu}^k(I)$ the empirical mean of the rewards acquired from the arm k on the interval I using only γ-observations and Γ_{min}(I) the smallest number of γ-observations among each action on the interval I. The detector is called only when $\Gamma_{\text{min}}(I) \geq \frac{\gamma H}{K}$. The detector raises a detection when the upper confidence bound of the mean of action k_{max} with the highest probability $p_k(t)$ (see Algorithm 1) is smaller than the lower confidence bound of the mean of another action on the current interval.

Algorithm 2 DriftDetection()

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

The EXP3.R algorithm is obtained by combining EXP3 and the drift detector. First, one instance of EXP3 is initialized and used to select actions. The detection test is called when $\Gamma_{\text{min}}(I) \geq \frac{\gamma H}{K}$. If in the corresponding interval, the empirical mean of an arm exceeds by 2ϵ the empirical mean of the current best arm then a drift detection is raised. In this case, weights of EXP3 are reset (by setting $t_r = t$ in Algorithm 1). Then a new interval of collect begins, preparing the next test. These steps are repeated until the run ends (see Algorithm 3).

Algorithm 3 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 $\Gamma_{\text{min}}(I) \geq \frac{\gamma H}{K}$ then
      if $DriftDetection(I)$ then
        Reset EXP3
      end if
      $I = I + 1$
    end if
  end for
```

With an accuracy ϵ, only differences higher than 4ϵ can be detected with high probability. We follow [14] and use Assumption 1 to ensure all changes of the best arm are detected with high probability.

Assumption 1: During each of the M stationary periods, the difference between the mean reward of the optimal arm and any other is at least 4ϵ with

$$\epsilon = \sqrt{\frac{K\log(\frac{1}{\delta})}{2\gamma H}}\,. \tag{8}$$

A. Analysis

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

Lemma 1 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 1: When Assumption 1 hold and $I \subseteq S$, then, with a probability at least $1 - 2\delta$, for any $k \neq k^*_S$:

$$\hat{\mu}^{k^*_S}(I) - \hat{\mu}^k(I) \geq 2\sqrt{\frac{K\log(\frac{1}{\delta})}{2\gamma H}} \Leftrightarrow \mu^{k^*_S}(I) \geq \mu^k(I)\,. \tag{9}$$

Proof 1: We justify our detection test by considering an observation of a reward through γ-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 drawn from $1 \leq m \leq M$ different distributions of mean $\mu^k_0(I), ..., \mu^k_m(I)$. We denote t_i the steps where the mean reward starts being $\mu^k_i(I)$ 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 $\mu^k_i(I)$. The mean $\mu^k(I)$ of the urn corresponding to the action k is:

$$\mu^k(I) = \sum_{i=1}^{m} \frac{t_{i+1} - t_i}{t_{m+1} - t_0}\mu^k_i(I)\,. \tag{10}$$

At each time step, by Assumption 1, the mean reward of the best arm is away by 4ϵ from any suboptimal arms. Consequently, the difference between the mean reward of the urn of the optimal arm k^* and that of another arm k is at least 4ϵ if the best arm does not change during the interval.

$$\mu^k(I) \leq \sum_{i=1}^{m} \frac{t_{i+1} - t_i}{t_{m+1} - t_0} (\mu_i^{k_S^*} - 4\epsilon) \leq \mu^{k_S^*}(I) - 4\epsilon\,. \quad (11)$$

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

Applying the Serfling inequality [17], we have:

$$P(\hat{\mu}^k(I) + \epsilon \geq \mu^k(I))) \leq e^{\frac{-2n\epsilon^2}{1-\frac{n-1}{U}}} \leq e^{-2n\epsilon^2}\,, \quad (12)$$

where $n = \frac{\gamma H}{K}$ is the number of observation and U the size of the urn. We denote $P(\hat{\mu}^k(I) + \epsilon \geq \mu^k(I)))$ as δ. Using Assumption 1 and the union bound, we have $\hat{\mu}^k(I) + \epsilon \geq \mu^k(I)$ and $\hat{\mu}^{k_S^*}(I) - \epsilon \leq \mu^{k_S^*}(I)$ with probability at least $1 - 2\delta$. As consequence, if $\hat{\mu}^{k_S^*}(I) - \hat{\mu}^k(I) \geq 2\epsilon$ then with high probability $\mu^{k^*}(I) \geq \mu^k(I)$. The detection test uses the upper bound of the confidence interval for the current best arm and the lower bound for the candidate arm. When both hold, in the worst case, the estimators are away from the true mean each by ϵ, then the detector test add the two confidence intervals of ϵ. The equation (11) ensures that all changes of the best arm are detected. □

Theorem 1 bounds the expected cumulative regret of EXP3.R.

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

$$\begin{aligned} G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq\ & (e-1)\gamma T \\ & + \frac{\left(N - 1 + \frac{K\delta T}{H} + K\delta\right) K \log(K)}{\gamma} \\ & + (N-1)HK\left(\frac{1}{1-2\delta} + 1\right). \end{aligned} \quad (13)$$

Proof 2: 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 [14] we deduce the form of the bound with drift detector from the classical EXP3 bound. If there is $N-1$ changes of best arm. Therefore, the expectation of the number of resets over a horizon T is upper bounded by $N - 1 + F(T)$. The regret of EXP3 on these periods is $(e-1)\gamma T + \frac{K \log K}{\gamma}$ [5]. 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

$$\sum_{t=T_S}^{T_{S+1}-1} x_{k_S^*}(t) \leq \max^{k} \sum_{t=T_S}^{T_{S+1}-1} x_k(t)\,, \quad (14)$$

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 a hypothetical algorithm that collects only the observations of one arm and then proceeds on the next arm until collecting all the observations. The γ-observation are drawn with a probability $\frac{\gamma}{K}$ and $\frac{\gamma H}{K}$ observations are required per action. The expectation of the number of failures before collecting $\frac{\gamma H}{K}$ γ-observations follows a negative binomial distribution of expectation

$$\frac{\gamma H}{K}(1 - \frac{\gamma}{K})\frac{K}{\gamma} = H - \frac{\gamma H}{K}\,. \quad (15)$$

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

$$\frac{\gamma H}{K} + H - \frac{\gamma H}{K} = H\,. \quad (16)$$

Summing over all arms gives a total expectation of HK. Because our algorithm collects γ-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 do not have any guarantee on the test behavior due to this change. In the worst case, the test does not detect the drift and we set the instantaneous regret to 1.

$$\begin{aligned} & G^* - \mathbf{E}[G_{\text{EXP3.R}}]) \leq (e-1)\gamma T \\ & + \frac{(N-1+F(T))K \log K}{\gamma} + (N-1)HK(\lceil L(T) \rceil + 1)\,. \end{aligned} \quad (17)$$

We now bound $F(T)$ and $L(T)$. As detection tests are computed with observations obtained on different intervals, their results are independent. Confidence intervals hold with probability at least $1 - \delta$ and they are used K times at each detection test. The maximal number of call 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 1 ensures the detection happens with a probability at least $1-2\delta$. We have $L(T) \leq \frac{1}{1-2\delta}$.

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{\left(N - 1 + \frac{K\delta T}{H} + K\delta\right) K \log K}{\gamma} + (N-1)HK\left(\frac{1}{1-2\delta} + 1\right). \quad (18)$$

□

In Corollary 1 we optimize parameters of the bound obtained in Theorem 1.

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

$$\delta = \sqrt{\frac{\log T}{KT}}, \gamma = \sqrt{\frac{K \log K \log T}{T}} \text{ and } H = C\sqrt{T \log T} \quad (19)$$

is

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}]) \leq (e-1)\sqrt{TK \log K \log T} + N\sqrt{TK \log K} + (C+1)K\sqrt{T \log K} + 3NCK\sqrt{T \log T}. \quad (20)$$

Accordingly to C, the precision ϵ is:

$$\epsilon = \sqrt{\frac{1}{2C}}\sqrt{\frac{\log\sqrt{\frac{KT}{\log T}}}{\log T}\sqrt{\frac{K}{\log K}}}. \quad (21)$$

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

Proof 3: We set $\delta = \sqrt{\frac{\log T}{KT}}$ and $H = C\sqrt{T \log T}$ in Theorem 1

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{(N - 1 + (C+1)\sqrt{K})K \log K}{\gamma} + 3(N-1)CK\sqrt{T \log T}. \quad (22)$$

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

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\sqrt{TK \log K \log T} + N\sqrt{TK \log K} + (C+1)K\sqrt{T \log K} + 3NCK\sqrt{T \log T}. \quad (23)$$

□

B. Discussion.

The obtained bound of $O(N\sqrt{T \log T})$ matches the lower bound up to a factor of $\sqrt{\log T}$. In comparison, the cumulative regrets of SW-UCB and EXP3.S are respectively upper bounded by $O(\sqrt{MT \log T})$ and $O(\sqrt{NT \log T})$. To reach these bounds, the knowledge of the maximal number of changes is necessary, otherwise the bounds become $O(M\sqrt{T \log T})$ and $O(N\sqrt{T \log T})$. In practical uses, algorithms are deployed during long periods, N is small and $H \ll T$.

V. Simulations

We consider two problems which are composed of 10^8 round with a choice of 20 arms. During these experimentations, EXP3.R is compared with five algorithms (Figure I) and the cumulative regret is averaged over 100 independent runs. Parameters of the different algorithms are shown in Table 1. They are tuned on the first problem.

On the two following problems, an index defines the optimal arm and will be incremented at fixed time-steps. The first problem considers *constant arms*. These arms are called *constant* because their mean rewards never change. The behavior of these constant arms is not in favor of Meta-Eve. When a constant arm is optimal, the UCB used as a sub-routine by Meta-Eve will quickly converge on this arm and may not see changes when the optimal index will be incremented. On the second problem, mean rewards of all the arms will change frequently, even when the index of the optimal arm is not incremented, illustrating a case where M is much higher than N.

Problem 1. One arm in three has a mean reward of 0.7, including the first arm and are called *constant arms*. An index, defining the optimal arm, is initialized on a *constant arm* and is incremented all 2×10^7 rounds. The mean reward of the optimal arm stays 0.7 if the index is on a *constant arm* but else become 0.8. Other arms have 0.2 as mean reward.

Unsurprisingly EXP3 and UCB, not built for arm switch, achieve a high regret. Meta-Eve is competitive but suffers from arms with mean rewards of 0.7. When such arm is optimal, because the algorithm does not explore, the detection test may not see the drift. Variance of this algorithm on this simulation is very high (see Table I). The runs where the drift is detected obtains a low regret and runs where the drift is unseen obtain a high regret. Behaviors of EXP3.S and EXP3.R on this problem are very similar, but the discount factor of EXP3.S hinders the convergence conducting to an higher regret. Finally, SW-UCB achieves the lowest cumulative regret than EXP3.R taking advantage of the long periods of stationarity.

Problem 2. An index, defining the optimal arm, is initialized on a random arm and is incremented all 2×10^7 rounds. The reward of the optimal arm changes all 10^5 rounds following this cycle: 0.6, 0.8 then 0.5. Thus, within the periods of lenght 2×10^7 the optimal arm never changes, even if its mean reward

Algorithm	Parameter	Value	Problem 1	Problem 2
EXP3	γ	10^{-2}	$6.1 \times 10^6 \pm 10^5$	$8 \times 10^6 \pm 10^4$
EXP3.S	γ α	10^{-2} 2×10^{-5}	$8.8 \times 10^5 \pm 10^4$	$\mathbf{3.5 \times 10^5 \pm 5 \times 10^4}$
EXP3.R	γ δ H	10^{-2} 10^{-3} 3×10^5	$\mathbf{7.1 \times 10^5 \pm 10^5}$	$\mathbf{2.9 \times 10^5 \pm 10^5}$
UCB	$\emptyset$	$\emptyset$	$4.8 \times 10^6 \pm 10^6$	$7.2 \times 10^6 \pm 10^6$
SW-UCB	W	8×10^5	$\mathbf{2.2 \times 10^5 \pm 10^4}$	$3.5 \times 10^6 \pm 10^5$
Meta-Eve	δ λ α β	10^{-3} 100 $1 + 10^{-2}$ $1 - 10^{-2}$	$1.3 \times 10^6 \pm 8 \times 10^5$	$1.1 \times 10^6 \pm 9.5 \times 10^5$

TABLE I

DIFFERENT ALGORITHMS TESTED WITH EXP3.R AND THEIR CUMULATIVE REGRET ON TWO PROBLEMS. THE CUMULATIVE REGRET IS AVERAGED OVER 100 INDEPENDENT RUNS.

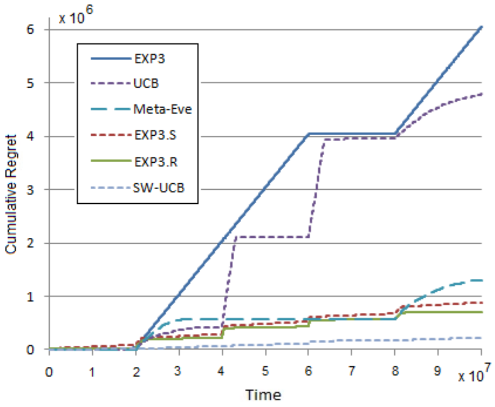

Fig. 2. Cumulative regret over the time of different algorithms on Problem 1.

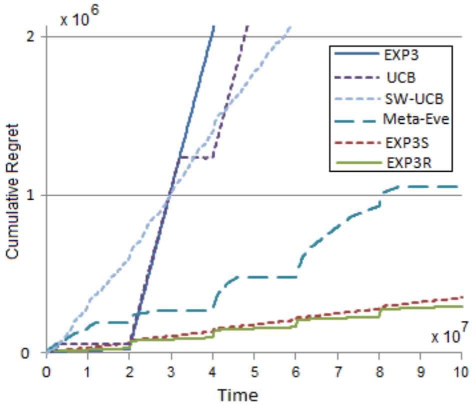

Fig. 3. Cumulative regret over the time of different algorithms on Problem 2.

changes during time. Suboptimal arms have as reward 0.1 less than the optimal arm.

EXP3 and UCB achieve a high regret like in the previous experiment. The variance of META-EVE is still very high and constant drifts without changes of best arm prevent the tuning of λ. EXP3.S is still close to EXP3.R but is penalized even more on this problem by the discount factor. Thanks to its active strategy, EXP3.R achieves a low regret during intervals with no change of optimal arm. Finally, the recurrent drifts prevent totally the SW-UCB algorithm to converge.

Non-stationarity introduces another exploration/exploitation dilemma. In addition to find the best arm, algorithms must be able to cope with changes in reward distributions. In some cases, a low regret within stationary periods may become a handicap and prevent the detection of changes in suboptimal reward distributions, like in META-EVE. As demonstrated in [7] the regret in the non-stationary multi-armed bandit problem is lower bounded by $\sqrt{T}$. Thereby, all algorithms showing a lesser upper bound on the stationary case may be tricked by certain kinds of drift, showed on Problem 2 with META-EVE. Parameters like discount factors or windows are hard to tune if the non-stationarity is aperiodic. When changes are close, algorithms need to forget the past quickly but when changes are faraway they would benefit from a larger memory. The advantage of an active strategy is to allow the algorithm to converge on stationary periods and to reset the algorithm only when a change is detected.

VI. CONCLUSION

The proposed algorithm, EXP3.R achieving a regret bound in $O(N\sqrt{T \log T})$. EXP3.R is also competitive with other state-of-the art algorithms, simulations showing promising results. 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 stationary periods. Further works may be concerned with the use of this algorithm as a meta-bandit [18] for tuning parameters and managing set of contextual bandit algorithms in a non-stationary environment.

References

[1] Thompson, W.: On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. Biometrika **25** (1933) 285–294

[2] Agrawal, S., Goyal, N.: Analysis of thompson sampling for the multi-armed bandit problem. In: Proceedings of the 25th Annual Conference on Learning Theory (COLT). (June 2012)

[3] Kaufmann, E., Korda, N., Munos, R.: Thompson sampling: An asymptotically optimal finite-time analysis. In Bshouty, N., Stoltz, G., Vayatis, N., Zeugmann, T., eds.: Algorithmic Learning Theory. Volume 7568 of Lecture Notes in Computer Science., Springer Berlin Heidelberg (2012) 199–213

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

[5] Auer, P., Cesa-Bianchi, N., Freund, Y., Schapire, R.E.: The nonstochastic multiarmed bandit problem. SIAM J. Comput. **32**(1) (2002) 48–77

[6] Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning, and Games. Cambridge University Press, New York, NY, USA (2006)

[7] Garivier, A., Moulines, E.: On upper-confidence bound policies for non-stationary bandit problems. In: Algorithmic Learning Theory. (2011) 174–188

[8] Kocsis, L., Szepesvári, C.: Discounted ucb. In: 2nd PASCAL Challenges Workshop, Venice, Italy (April 2006)

[9] Feraud, R., Urvoy, T.: Exploration and exploitation of scratch games. Machine Learning **92**(2-3) (2013) 377–401

[10] Hoens, T., Polikar, R., Chawla, N.: Learning from streaming data with concept drift and imbalance: an overview. Progress in Artificial Intelligence **1**(1) (2012) 89–101

[11] Gama, J., Medas, P., Castillo, G., Rodrigues, P.: Learning with drift detection. In Bazzan, A., Labidi, S., eds.: Advances in Artificial Intelligence SBIA 2004. Volume 3171 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2004) 286–295

[12] Last, M.: Online classification of nonstationary data streams. Intell. Data Anal. **6**(2) (April 2002) 129–147

[13] Hartland, C., Baskiotis, N., Gelly, S., Teytaud, O., Sebag, M.: Multi-armed bandit, dynamic environments and meta-bandits. In: Online Trading of Exploration and Exploitation Workshop, NIPS, Whistler, Canada (December 2006)

[14] Yu, J.Y., Mannor, S.: Piecewise-stationary bandit problems with side observations. In: Proceedings of the 26th Annual International Conference on Machine Learning. ICML '09, New York, NY, USA, ACM (2009) 1177–1184

[15] Y. Yao, L.F., Chen, F.: Concept drift visualization. Journal of Information and Computational Science **10**(10) (2013)

[16] Borchani, H., Larraaga, P., Bielza, C.: Mining concept-drifting data streams containing labeled and unlabeled instances. In Garca-Pedrajas, N., Herrera, F., Fyfe, C., Bentez, J., Ali, M., eds.: Trends in Applied Intelligent Systems. Volume 6096 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2010) 531–540

[17] Serfling, R.: Probability inequalities for the sum in sampling without replacement. In: The Annals of Statistics, Vol 2, No.1, pages = 39–48, year = 1974,

[18] Allesiardo, R., Feraud, R., Bouneffouf, D.: A neural networks committee for the contextual bandit problem. In: Neural Information Processing - 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part I. (2014) 374–381