A single algorithm for both restless and rested rotting bandits

Julien Seznec
Lelivrescolaire.fr
SCOOL, Inria Lille

Pierre Menard
SCOOL, Inria Lille

Alessandro Lazaric
FAIR Paris

Michal Valko
DeepMind Paris

Abstract

In many application domains (e.g., recommender systems, intelligent tutoring systems), the rewards associated to the actions tend to decrease over time. This decay is either caused by the actions executed in the past (e.g., a user may get bored when songs of the same genre are recommended over and over) or by an external factor (e.g., content becomes outdated). These two situations can be modeled as specific instances of the rested and restless bandit settings, where arms are *rotting* (i.e., their value decrease over time). These problems were thought to be significantly different, since Levine et al. (2017) showed that state-of-the-art algorithms for restless bandit perform poorly in the rested rotting setting. In this paper, we introduce a novel algorithm, Rotting Adaptive Window UCB (`RAW-UCB`), that achieves near-optimal regret in both rotting rested and restless bandit, without any prior knowledge of the setting (rested or restless) and the type of non-stationarity (e.g., piece-wise constant, bounded variation). This is in striking contrast with previous negative results showing that no algorithm can achieve similar results as soon as rewards are allowed to increase. We confirm our theoretical findings on a number of synthetic and dataset-based experiments.

1 Introduction

When we design sequential learner, we would like them to be as adaptive to environment as possible. This becomes a challenge when the environment only provides limited feedback, as in the *bandit* setting (Lai and Robbins, 1985; Lattimore and Szepesvári, 2020), where the learner receives only the feedback associated to the action it executed. Since the early stages of the research in bandits (Thompson, 1933; Whittle, 1980), one of the most desirable properties for a learners would be to adapt to actions whose *value changes over time* (Whittle, 1988), as it happens in non-stationary environments. In fact, from applications in medical trials (where the patient can become more resistant to antibiotics) to a modern applications in recommender systems (Chapelle and Li, 2011; Tracà and Rudin, 2015), assuming that the environment is *stationary is very limiting.*

However, modeling and managing non-stationary environments is obviously way more difficult (Lattimore and Szepesvári, 2020). That is why Auer et al. (2003) went as far as to consider the worst-case scenario, referred to as the *adversarial bandit* setting, where the learner should try to shield from the worst possible variation in rewards. Nonetheless, real-world environments are rarely adversarial and algorithms for adversarial bandits turn out to be too conservative for practical use. On the one hand, in order to manage such general family of environments, the performance of a learner is compared to the best *fixed* action in *hindsight*. This is arguably a weaker objective w.r.t. competing against the optimal strategy, as it is the case in stationary bandits. On the other hand, state-of-the-art adversarial algorithms (Audibert and Bubeck, 2009), which are proved to recover near-optimal regret rates on stationary problems, still under-perform in practice against optimal stationary algorithm (Zimmert and Seldin, 2019). In order to address these issues, prior work identified specific types of non-stationary environments, for which specifically designed algorithms can be used.

There are two main classes of non-stationary environments, depending on whether the change of rewards is triggered by the actions of the learner, the *rested bandits*, or it happens over time independently from the learner, the *restless bandits*. In this paper, we consider the specific case where the changes in the rewards are

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

arbitrary *non-increasing* functions of time and/or number of pulls (in contrast with typical restless bandit models, where the evolution of rewards was regulated by Markov chain processes). For instance, Warlop et al. (2018) model boredom effects in recommender systems as a rested bandit problem, but need to resort to a more general reinforcement learning framework to address the fact that rewards are decreasing while an action is repeatedly selected but may increase back if *enough time* has passed since the last time is chosen. Immorlica and Kleinberg (2018) and Pike-Burke and Grunewalder (2019) have recently modeled these recharging effects as a bandits problem. In the restless setting, Louëdec et al. (2016) models obsolescence of appearing arms (e.g. piece of news) with a known exponential rate. Komiyama and Qin (2014) study a parametric decay in restless bandits where rewards are linear combination of known decaying function. In the following, we briefly review the most relevant results available for restless bandit (where no rotting assumption has been studied before) and the rested rotting bandit settings.

Restless stochastic bandits Garivier and Moulines (2011) study the restless bandits case, where rewards are piece-wise stationary. If the number of stationary pieces Υ_T at the horizon T is known, the optimal strategy is included in a set of T^{Υ_T} switching experts. Hence one can use `Exp3.S`, an adversarial algorithm designed for this specific set of experts (Auer et al., 2003). Moreover, Garivier and Moulines (2011) show that two upper-confidence bound index algorithms with passive forgetting parameters, `SW-UCB` and `D-UCB`, are also able to reach nearly-minimax performance when they know in advance Υ_T and T. Recent research (Cao et al., 2019; Liu et al., 2018; Besson and Kaufmann, 2019) has focused on integrating change-detection algorithms with standard bandit learners (*e.g.* `UCB`) to actively forget past rewards whenever a significant variation in the reward distribution is detected. Among them, we mention `GLR-klUCB` (Besson and Kaufmann, 2019) which uses a parameter-free change-point detector. These algorithms actively explore sub-optimal actions to track potential increase in their value. Yet, their analysis assume that change-points are always big enough to be detectable with high-probability. Auer et al. (2019) introduce `AdSwitch`, a filtering algorithm with a planned active exploration scheme for sub-optimal actions. `AdSwitch` achieves the minimax rate while being agnostic to Υ_T without any extra assumption.

Besbes et al. (2014) introduced a restless bandits framework where the environment has a variation budget of V_T to change the rewards' values. In this setup, the best arm can change at each round and thus the optimal strategy is not necessary included in a "small" set of switching experts. Yet, they show that the best strategy with $\mathcal{O}\left(T^{1/3}\right)$ switches suffers low regret compared to the optimal strategy. Hence, `Exp3.S` matches the minimax rate $\mathcal{O}\left(T^{2/3}\right)$ with the knowledge of V_T. Cheung et al. (2019) and Russac et al. (2019) extended `SW-UCB` and `D-UCB` to show that they also match the minimax rate of the variation budget setting even in the more general linear bandits framework. Chen et al. (2019) show that `AdSwitch` also matches the minimax rate without the knowledge of V_T. They also analyse `ADA-ILTCB+`, an algorithm which achieves similar guarantee in the more general linear setting. Wei et al. (2016) extended these results to a non-stationary environment where both the means and the variances of the rewards may change.

Rested rotting bandits Finally, Heidari et al. (2016); Levine et al. (2017) and Seznec et al. (2019) studied *rested rotting bandits*, when the reward of an action decreases every time it is pulled. Seznec et al. (2019) recently proposed a nearly-optimal algorithm for this setting. Interestingly, the algorithm does not execute an *index policy* (defined later) which is a prevalent choice in bandit. Actually, a previous attempt of using an index policy by Levine et al. (2017) resulted in a sub-optimal performance.

Our contribution is threefold:

- We show that no learning strategy can achieve $o(T)$ worst case rate when we allow for both rested <u>and</u> restless decay (Section 2).
- We introduce a novel index policy `RAW-UCB` (Section 3) and prove that it achieves minimax rate regret for either restless (Section 4) <u>or</u> rested (Section 5) settings without any prior knowledge of the type of decay, the amount of change, or the horizon.
- `RAW-UCB` also recovers problem-dependent $\mathcal{O}(\log T)$ bounds in both setups. In the restless case[1], such bounds cannot be achieved when the reward can increase. Hence, it shows that the decreasing assumption do help the learner compared to the well-studied general case.

Also, we provide a rested simulated (Appendix G.1) and restless real-world (Section 6) benchmarks on which `RAW-UCB` gives the most consistent results in both setups.

[1]In the rested case, Heidari et al. (2016) shows that increasing reward is a much harder problem, even in the absence of noise.

2 Decreasing multi-armed bandits

At each round t, an agent chooses an arm $i_t \in \mathcal{K} \triangleq \{1, ..., K\}$ and receives a noisy reward o_t. The sample associated to each arm i is a σ^2-sub-Gaussian r.v. with expected value of $\mu_i(t, n)$ which depends on the number of times n it was pulled before and on the time t.

Let $\mathbf{H}_t \triangleq \{\{i(s), o_s\}, \forall s \leq t\}$ be the sequence of arms pulled and rewards observed until round t, then

$$o_t \triangleq \mu_{i_t}(t, N_{i_t,t-1}) + \varepsilon_t \,,$$

with $\mathbb{E}\left[\varepsilon_t | \mathbf{H}_{t-1}\right] = 0$ and $\forall \lambda \in \mathbb{R}$, $\mathbb{E}\left[e^{\lambda \varepsilon_t}\right] \leq e^{\frac{\sigma \lambda^2}{2}}$, where $N_{i,t} \triangleq \sum_{s=1}^{t} \mathbb{1}\left(i_s = i\right)$ is the number of pulls of arm i at time t. We call $\mu \triangleq \{\mu_i\}_{i \in \mathcal{K}}$ the set of reward functions.

Decreasing rewards Throughout all the paper, we consider the following assumption.

Assumption 1. *For each arm i, any number of pulls n, and time t, the functions $\mu_i(t, \cdot)$ and $\mu_i(\cdot, n)$ are non-increasing.*

We will use interchangeably the terms *decreasing*, *decaying* and *rotting* to refer to this Assumption. If $\mu_i(t, N_{i,t}) = \mu_i(N_{i,t})$, then i is called a rested arm. If $\mu_i(t, N_{i,t}) = \mu_i(t)$, then i is called a restless arm.

Learning problem A (deterministic) learning policy π is a function that maps history of observations to arms, i.e., $\pi(\mathcal{H}_t) \in \mathcal{K}$. In the following, we often use $\pi(t) \triangleq \pi(\mathcal{H}_{t-1})$ to denote the arm pulled at time t. The performance of a policy π is measured by the (expected) rewards accumulated over time,

$$J_T(\pi, \mu) \triangleq \sum_{t=1}^{T} \mu_{\pi(t)}\left(t, N_{\pi(t),t-1}\right) \,.$$

A (deterministic) oracle policy is a function which maps the set of reward functions and a round to an arm, i.e., $\pi(t, \mu) \in \mathcal{K}$. Thus, these oracles have access to the true (without noise) value of the rewards, including future value. Notice that at the horizon T, there are K^T distinct deterministic policies. Therefore, we call an optimal (oracle) policy, one which, at a given horizon T, maximizes the reward

$$\pi^*_T(t, \mu) \in \underset{\pi \in \mathcal{K}^T}{\arg\max}\, J_T(\pi, \mu) \,.$$

We define the regret as

$$R_T(\pi, \mu) \triangleq J_T(\pi^\star_T, \mu) - J_T(\pi, \mu).$$

Notice that this definition is more challenging than the regret w.r.t. the best fixed-arm policy commonly used as comparator in adversarial bandits. In the following, we often use shorter notation $\pi^*_T(t)$, $J_T(\pi)$, $R_T(\pi)$ where the considered problem μ is implicit.

Greedy oracle policy It is still unclear if 1) we can compute $\pi^\star_T$ in a tractable way; 2) if a learning policy can suffer low regret compared to this policy. We call π_O the oracle policy which selects greedily at each round t the largest available reward $i_t \in \arg\max_{i \in \mathcal{K}} \mu_i(t, N_{i,t-1})$.[2] We notice that this policy is optimal at any time in any restless non-stationary bandit problem $\mu(t)$. Heidari et al. (2016) show that it is also optimal in the rested rotting bandits problem. Thus, π_O answers positively to the first question for either rested or restless decay. In the next proposition, we show that the greedy oracle suffers linear worst-case regret when we allow for both restless and rested decay at the same time. Worse, we show that no learning policy can approach the performance of the optimal oracle at a $o(T)$ rate

Proposition 1. *In the no noise setting ($\sigma = 0$), there exists a rotting 2-arms bandits problem (satisfying Assumption 1) with reward value in $[0, 1]$, with one rested arm and one restless arm, and with at most one change-point before T each, such that the greedy oracle strategy π_O suffers a regret*

$$R_T\left(\pi_O\right) \geq \left\lfloor \frac{T}{4} \right\rfloor .$$

Moreover, for any learning strategy π_S, there exists a rotting 2-arms bandits problem (satisfying Assumption 1) with reward value in $[0, 1]$, with one rested arm and one restless arm, and with at most one change-point before T each, such that

$$R_T\left(\pi_S\right) \geq \left\lfloor \frac{T}{8} \right\rfloor .$$

Notice that the two reward functions of the constructed difficult problems are simple: either rested or restless, bounded and with at most one break-point. If we consider a 2-arm setup with one rested arm and one restless arm, a good strategy may be to select the restless arm even when its current value is the worst. Indeed, this value is only available now, while the good value of the rested arm will still be available in the future. Whether the restless rewards are interesting to the learner depends on the future behavior of the (currently best) rested arm. On the first hand, if it decays below the current value of the restless arm before the horizon T, then the learner should profit from the restless reward available right now. On the other hand, if the rested arm stays optimal until the end of the game then the learner should ignore the restless arm and follows the greedy oracle strategy. However, the learner does not know in advance if (and how much) an arm will decay and any anticipation she makes will

[2] We break the ties arbitrarily, for instance by selecting the smallest index in $\arg\max_{i \in \mathcal{K}} \mu_i\left(t, \mathcal{H}_t\right)$

turn to be bad in the worst case. We formalize these ideas in the proof in Appendix B and show that any strategy suffers linear regret in the worst case.

While learning with rested and restless rotting reward is impossible, we show in the next sections that a single policy reaches near-optimal guarantee in both separated setups.

3 The RAW-UCB algorithm

Notation For policy π, we define the average of the last h observations of arm i at time t as

$$\widehat{\mu}_i^h(t,\pi) \triangleq \frac{1}{h}\sum_{s=1}^{t-1} \mathbb{1}(\pi(s) = i \wedge N_{i,s} > N_{i,t-1} - h)\, o_s \quad (1)$$

and the average of the associated means as

$$\overline{\mu}_i^h(t,\pi) \triangleq \frac{1}{h}\sum_{s=1}^{t-1} \mathbb{1}(\pi(s) = i \wedge N_{i,s} > N_{i,t-1} - h)\, \mu_i(s, N_{i,s-1})\,.$$

A favorable event We use a similar high probability analysis than UCB1. We design a favorable event and we show in Prop. 2 that it holds with high probability.

Proposition 2. *For any round t and confidence $\delta_t \triangleq 2t^{-\alpha}$, let*

$$\xi_t^\alpha \triangleq \Big\{\forall i \in \mathcal{K},\ \forall n \le t-1,\ \forall h \le n, \\ |\widehat{\mu}_i^h(t,\pi) - \overline{\mu}_i^h(t,\pi)| \le c(h,\delta_t)\Big\} \quad (2)$$

be the event under which the estimates at round t are all accurate up to $c(h,\delta_t) \triangleq \sqrt{2\sigma^2\log(2/\delta_t)/h}$. Then, for a policy π which pulls each arms once at the beginning, and for all $t > K$,

$$\mathbb{P}\left[\overline{\xi_t^\alpha}\right] \le \frac{Kt^2\delta_t}{2} = Kt^{2-\alpha}\,.$$

Rotting Adaptive Window Upper Confidence Bound (RAW-UCB or π_{R}). At each round, RAW-UCB selects the arm with the largest following index,

$$\mathrm{ind}(i,t,\delta_t) \triangleq \min_{h \le N_{i,t-1}} \widehat{\mu}_i^h(t,\pi_{\mathrm{R}}) + c(h,\delta_t), \quad (3)$$

with $\delta_t \triangleq \frac{2}{t^\alpha}$. There is a bias-variance trade-off for the window choice: more variance for smaller size of the window h and more bias for larger h. The goal of RAW-UCB is to adaptively select the right window to compute the tightest UCB. RAW-UCB uses the indexes of UCB1 computed on all the slices of each arm's history which include the last pull. When the rewards are rotting, all these indexes are upper confidence bounds on the *next value*. Thus, RAW-UCB simply selects the tightest (minimum) one as index of the arm: it is a pure UCB-index algorithm. By contrast, when reward can increase, the learner can only derive upper-confidence bound on past values which are loosely related to the next value. Hence, all the UCB-index algorithms in the restless non-stationary literature need to add change-detection sub-routine, active random exploration or passive forgetting mechanism. In Lemma 1, we show a guarantee of RAW-UCB on the favorable event.

Algorithm 1 RAW-UCB

Input: $\mathcal{K}$, σ, α
1: **for** $t \leftarrow 1, 2, \dots, K$ **do** ▷ *Pull each arm once*
2: Pull $i_t \leftarrow t$; Receive o_t ; $N_{i_t} \leftarrow 1$
3: $\{\widehat{\mu}_{i_t}^h\}_h \leftarrow$ UPDATE($\{\widehat{\mu}_{i_t}^h\}_h, o_t$) ▷ *cf.* (1)
4: **end for**
5: **for** $t \leftarrow K+1, K+2, \dots$ **do**
6: Pull $i_t \in \arg\max_i \min_{h \le N_i} \widehat{\mu}_i^h + c(h,\delta_t)$ ▷ *cf.* (3)
7: Receive o_t ; $N_{i_t} \leftarrow N_{i_t} + 1$
8: $\{\widehat{\mu}_{i_t}^h\}_h \leftarrow$ UPDATE($\{\widehat{\mu}_{i_t}^h\}_h, o_t$) ▷ *cf.* (1)
9: **end for**

Lemma 1. *At round t on favorable event ξ_t^α, if arm i_t is selected, for any $h \le N_{i,t-1}$, the average of its h last pulls cannot deviate significantly from the best available arm at that round, i.e.,*

$$\overline{\mu}_{i_t}^h(t,\pi) \ge \max_{i \in \mathcal{K}} \mu_i(t, N_{i,t-1}) - 2c(h,\delta_t).$$

Seznec et al. (2019) show a slightly worse guarantee about the algorithm FEWA (π_{F}) for the rested rotting bandits. In Appendix C (see Lemma 2), we restate their result using only Assumption 1. FEWA uses the same statistics than RAW-UCB but in a rather complex expanding filtering mechanism which leads to a guarantee of only 4 confidence bounds. Lemma 1 is the only characterization we need for our analysis. Therefore, all our upper bounds will hold for both FEWA and RAW-UCB with their associated constant,

$$C_{\pi_{\mathrm{R}}} \triangleq 2\sqrt{2\alpha} \quad C_{\pi_{\mathrm{F}}} \triangleq 4\sqrt{2\alpha}. \quad (4)$$

Algorithmic complexity FEWA and RAW-UCB have $\mathcal{O}(Kt)$ per round time and space complexity. In Appendix D, we describe EFF-RAW-UCB (π_{ER}) and EFF-FEWA (π_{EF}), two algorithms which reduces the complexities to $\mathcal{O}(K\log_m(t))$. It is a refinement of the trick of Seznec et al. (2019) where we add a parameter $m > 1$ to trade-off between complexity and efficiency[3]. For $m = 2$, we prove Lemma 3 and Prop. 11, which are comparable with Lemma 1 and Prop. 2. Therefore, our analysis also holds for these algorithms with,

$$C_{\pi_{\mathrm{ER}}} \triangleq \frac{4\sqrt{\alpha}}{\sqrt{2}-1} \quad C_{\pi_{\mathrm{EF}}} \triangleq \frac{8\sqrt{\alpha}}{\sqrt{2}-1}. \quad (5)$$

[3] When $m < 1 + \frac{1}{T}$, EFF-RAW-UCB behave as RAW-UCB.

The efficient algorithms use less statistics than the original ones. Thus, the probability of the unfavorable event is bounded by $\mathcal{O}\left(t^{1-\alpha}\right)$ (see Prop. 11) which is smaller than $\mathcal{O}\left(t^{2-\alpha}\right)$ in Prop. 2. Hence, our theory holds for a wider range of α for the efficient algorithms.

4 Restless rotting bandits

In this section, the reward decreases independently of the user actions. Hence, we have that $\mu_i\left(t, n\right) = \mu_i\left(t\right)$.

Variation budget bandits

Setup. Besbes et al. (2014) introduce the limited variation budget bandits, a restless setting where at each round Nature can modify the reward value of any arm but with a limited total variation budget V_T at round T. We combine this assumption with Assumption 1,

Assumption 2. *$\mu_i : \mathbb{N}^\star \to [-V_T, 0]$ are decreasing functions of the time t with V_T a positive constant. Moreover, we have that,*

$$\sum_{t=1}^{T-1} \sup_{i\in\mathcal{K}} \left(\mu_i(t) - \mu_i(t+1)\right) \leq V_T\,. \tag{6}$$

Remark 1. *In the rotting scenario, the budget assumption is very similar to the bounded assumption. Indeed, any set of decreasing functions $\mu_i : \mathbb{N}^\star \to [-V, 0]$ satisfies Equation 6 with $V_T = KV$. Reciprocally, any set of functions satisfying Equation 6 with $\mu_i(1) \in [-V_T, 0]$ are bounded in $[-2V_T, 0]$.*

Lower bound. We show that our additional decreasing assumption does not change the minimax rate for budget bandits. This is an adaptation of the proof of Besbes et al. (2014) where we only use rotting function.

Proposition 3. *For any strategy π, there exists a rotting variation budget bandit scenario with means $\{\mu_i(t)\}_{i,t}$ satisfying Assumption 2 with a budget $V_T \geq \sigma\sqrt{\frac{K}{8T}}$ such that,*

$$\mathbb{E}\left[R_T(\pi)\right] \geq \frac{1}{16\sqrt{2}}\left(\sigma^2 V_T K T^2\right)^{1/3}.$$

Upper bound. `RAW-UCB` matches this lower bound up to poly-logarithmic factors without any knowledge of the horizon T nor the budget V_T.

Theorem 1. *Let $\pi \in \{\pi_{\mathrm{F}}, \pi_{\mathrm{R}}\}$ tuned with $\alpha \geq 4$ or $\pi \in \{\pi_{\mathrm{EF}}, \pi_{\mathrm{ER}}\}$ tuned with $\alpha \geq 3$ and $m = 2$. For any variation budget bandit scenario with means $\{\mu_i(t)\}_{i,t}$ satisfying Assumptions 2 with variation budget V_T, π suffers an expected regret,*

$$\mathbb{E}\left[R_T(\pi)\right] \leq 4\left(C_\pi^2 \sigma^2 V_T K T^2 \log T\right)^{1/3} + \widetilde{\mathcal{O}}\left(\left(\sigma V_T^2 K^2 T\right)^{1/3}\right).$$

The remaining terms are of second order when $KV_T \leq \mathcal{O}(T)$, which is a necessary condition for the problem to be learnable (see Proposition 3).

Piece-wise stationary bandits.

Setup. In this section, we also consider bounded functions. Hence, they also satisfy Assumption 2 (see Remark 1). However, we further restrained them to be piece-wise stationary,

Assumption 3. *Let V be a positive constant and Υ_T a positive integer. $\mu_i : \mathbb{N}^\star \to [-V, 0]$ are piece-wise stationary non-increasing functions of the time t with at most $\Upsilon_T - 1$ breakpoints.*

Formally, $\sum_{t=1}^{T-1} \mathbb{1}\left(\exists i \in \mathcal{K}, \mu_i(t) \neq \mu_i(t+1)\right) \leq \Upsilon_T - 1$. We call $\{t_k\}_{k\leq\Upsilon-1}$ the set of breakpoints with $t_0 = 0$, μ_i^k the value of $\mu_i(t)$ for $t \in \{t_k+1, \dots, t_{k+1}\}$. We call $i_k^\star \in \arg\max_{i\in\mathcal{K}} \mu_i^k$ (one of) the best arm in batch k and $\Delta_{i,k} \triangleq \mu_{i_k^\star}^k - \mu_i^k$ the gap to the best arm for arm i during batch k. Note that we relax all the assumptions related to the distance between consecutive breakpoints (e.g. Besson and Kaufmann (2019) and their Assumption 4 and 7; Liu et al. (2018) and their Assumption 1 and 2; Cao et al. (2019) and their Assumption 1).

Lower bound. We show that our additional Assumption 1 does not decrease the minimax rate of Garivier and Moulines (2011).

Proposition 4. *For any strategy π, there exists a rotting piece-wise stationary bandit scenario with means $\{\mu_i(t)\}_{i,t}$ satisfying Assumption 3 with $\Upsilon_T \leq \left(\frac{32V^2T}{K\sigma^2}\right)^{1/3}$ such that,*

$$\mathbb{E}\left[R_T(\pi)\right] \geq \frac{\sigma}{32}\sqrt{\Upsilon_T K T}\,.$$

The condition on Υ_T in Proposition 4 follows from Remark 1: if V is too small compared to Υ_T, then we have a budget constraint (with associated lower bound in Proposition 3) rather than a break-point constraint.

Upper bound. `RAW-UCB` matches the lower bound from Proposition 4 up to poly-logarithmic factors without any knowledge of the horizon T nor the number of breakpoints $\Upsilon_T - 1$.

Theorem 2. *Let $\pi \in \{\pi_{\mathrm{F}}, \pi_{\mathrm{R}}\}$ tuned with $\alpha \geq 4$ or $\pi \in \{\pi_{\mathrm{EF}}, \pi_{\mathrm{ER}}\}$ tuned with $\alpha \geq 3$ and $m = 2$. For any piece-wise stationary bandit scenario with means $\{\mu_i(t)\}_{i,t}$ satisfying Assumption 3 with $\Upsilon_T - 1$ change-points, π suffers an expected regret,*

$$\mathbb{E}\left[R_T(\pi)\right] \leq C_\pi \sigma\sqrt{\log T}\left(\sqrt{\Upsilon_T K T} + \Upsilon_T K\right) + 6KV.$$

Are rotting restless bandits easier? Learning at the minimax rate without knowing Υ_T or V_T was achieved in the non-rotting setup by significantly more complex algorithms. For instance, Auer et al. (2019) use a combination of filtering on the set of potentially good arms, forced exploration planning on identified bad arms, and full restart of the algorithm when a change is detected. This algorithmic complexity has a performance cost, as `AdSwitch` is guaranteed to achieve 56 times the leading term in Theorem 2. Moreover, these algorithms rely on doubling trick when the horizon is unknown, which also has a regret cost compared to intrinsically anytime algorithms (Besson and Kaufmann, 2018).

Yet, Proposition 3 and 4 show that the rotting assumption do not improve the minimax rate for the two considered setups. Interestingly both these lower bounds are matched by (tuned) `Exp3.S` (Auer et al., 2003), an algorithm originally designed for switching best arm in adversarial sequence of rewards. This is comparable to the fixed best arm world: adversarial and stochastic bandits share the same minimax rate which is matched in both setups by `Exp3`. The main interest of the stochastic assumption is to allow for *problem dependent analysis*. For the stochastic stationary bandits, it leads to a stronger $\mathcal{O}(\log(T))$ bounds. In the piece-wise stationary setting, Garivier and Moulines (2011) show that such bounds cannot be achieved without sacrificing the minimax optimality.

Proposition 5 (Theorem 31.2, Lattimore and Szepesvári (2020))**.** *If a policy π performs a regret $R_T(\pi,\mu)$ on a 2-arm stationary instance μ, one can find a piece-wise stationary instance μ' with only two breakpoints such that, for a sufficiently long horizon T, the regret is lower bounded by*

$$\mathbb{E}\left[R_T(\pi,\mu')\right] \geq \frac{T}{22R_T(\pi,\mu)}.$$

Corollary 1. *Let π a minimax policy on the (non-rotting) piece-wise stationary setups. Then, for a sufficiently large horizon T, there exists a universal constant C such that for all the 2-arm stationary problems μ,*

$$\mathbb{E}\left[R_T(\pi,\mu)\right] \geq C\sqrt{T}.$$

The proof of Proposition 5 is instructive. It builds a problem μ' on which the reward function equals the reward of the stationary problem μ except on a time span τ. During this time span, the best arm of μ keeps its value while the worst arm *increases* to become optimal. The size of τ is chosen inversely proportional to the average pulling rate of the bad arm in μ. Indeed, the lower the pulling rate of the bad arm, the longer the adversary can increase its value in μ' without being noticeable by the learner. Since the pulling rate of the bad arm in μ is proportional to $R_T(\mu)$, we get a lower bound proportional to $\tau \sim \frac{T}{R_T(\mu)}$.

The decreasing Assumption 1 excludes this μ' from the set of possible problems. Theorem 3 shows that not only `RAW-UCB` is able to recover the $\mathcal{O}(\log(T))$ on stationary problems but also recovers the same rate on each batch of a rotting piece-wise stationary problem.

Theorem 3. *Let $\pi \in \{\pi_{\mathrm{F}}, \pi_{\mathrm{R}}\}$ tuned with $\alpha \geq 4$ or $\pi \in \{\pi_{\mathrm{EF}}, \pi_{\mathrm{ER}}\}$ tuned with $\alpha \geq 3$ and $m = 2$. For any piece-wise stationary bandit scenario with means $\{\mu_i(t)\}_{i,t}$ satisfying Assumption 3 with $\Upsilon_T - 1$ change-points, π suffers an expected regret*

$$\mathbb{E}\left[R_T(\pi)\right] \leq \sum_{k=0}^{\Upsilon_T-1}\sum_{i\in\mathcal{K}} \frac{C_\pi^2\sigma^2\log T}{\Delta_{i,k}} + \mathcal{O}\left(\sigma\Upsilon_T K\sqrt{\log T}\right).$$

Like in `UCB1` 's analysis, Proposition 2 uses a union-bound with Hoeffding inequality. This technique leads to conservative theoretical tuning of confidence levels and to a suboptimal constant factor $C_\pi^2/2$. One can get the asymptotic optimal tuning for `UCB` on stationary gaussian bandits with a refined analysis which uses a specific concentration result on the deviation of the index (e.g. Lemma 8.2, Lattimore and Szepesvári (2020)). Yet, extending this result to our more complex meta-index and to our several setups is not straightforward and we leave it as future work. Interestingly, the experimental tuning $\alpha = 1.4$ is very close to the asymptotic tuning of `UCB` (see Section 6). It suggests that, besides our union bound considers more events than `UCB` in the theory, we do not have to be significantly more conservative on the confidence levels in practice.

Notice that Mukherjee and Maillard (2019) use a different assumption to get a similar problem-dependent bound. Indeed, they assume that all the arms change at the same time which also excludes μ' from the set of possible problems.

Proofs sketch (full proofs in Appendix E)

Lower bounds. Our proof technique make a strong connection between Proposition 3 and 4. Yet, we adapt existing proofs to the decreasing case (Garivier and Moulines, 2011; Besbes et al., 2014). Hence, we defer the full proof and its sketch to Appendix E.

Upper bounds. We start by separating the regret on the bad events $\overline{\xi}_t$ from the good events ξ_t. According to Proposition 2, the bad events $\overline{\xi}_t$ have low probability for appropriate α. For $\alpha = 4$, they weigh at most $\mathcal{O}(KV)$ in the expected regret. On the good events, we write:

$$R_T(\pi) = \sum_{t=1}^{T} \mu_{i_t^\star}(t) - \overline{\mu}_{i_t}^{h_t}(t,\pi) + \overline{\mu}_{i_t}^{h_t}(t,\pi) - \mu_{i_t}(t)\,. \quad (7)$$

Notice that Lemma 1 can bound the first difference for any h_t. When the reward is piece-wise stationary, we can select h_t such that we include all the pulls of arm i_t from the current stationary batch. If there is none, then it is the first pull of arm i_t in this batch. We handle these $\mathcal{O}(K\Upsilon_T)$ rounds separately (see Lemma 6 in Appendix E). In the other cases, we note that the second difference is null because $\overline{\mu}_{i_t}^{h_t}(t,\pi) = \mu_{i_t}(t) = \mu_i^k$ by the piece-wise stationary assumption. The remaining of the proofs of Theorem 2 and 3 are then very similar to the analysis of Auer et al. (2002) on each stationary batch. Indeed, the two confidence bounds guarantee of Lemma 1 is similar to `UCB1`'s guarantee.

In the variation budget setting, there is no stationary batches. Hence, we cannot choose an h_t which cancels the second difference in Equation 7. Yet, we still decompose the rounds in Υ batches of equal length for the analysis. We choose h_t such that we include all the pulls of arm i_t from the current batch. For the sum of the first differences in Equation 7, there is no difference with the piece-wise stationary case and we can bound

$$\sum_{t=1}^{T} \mu_{i_t^\star}(t) - \overline{\mu}_{i_t}^{h_t}(t,\pi) \leq \widetilde{\mathcal{O}}\left(\sqrt{K\Upsilon T}\right). \tag{8}$$

We call $\Delta_i^k \triangleq \mu_i(t_k) - \mu_i(t_{k+1})$, the total variation of arm i in batch k. The sum of second differences in Equation 7 can be bounded as follows: on each batch of $T\Upsilon^{-1}$ rounds, each second difference is bounded by $\max_{i\in\mathcal{K}} \Delta_i^k$. When we sum over the batches, we get

$$\sum_{t=1}^{T} \overline{\mu}_{i_t}^{h_t}(t,\pi) - \mu_{i_t}(t) \leq \frac{T}{\Upsilon}\sum_{k=0}^{\Upsilon-1} \max_{i\in\mathcal{K}} \Delta_i^k \leq \frac{TV_T}{\Upsilon}. \tag{9}$$

Indeed, in the middle term, we have a maximum on the summed variation of arm i in batch k. On the right-hand side, we have V_T which bounds the sum over the rounds of maximal variation of the arms (see Equation 6). Thus, the right-hand side is larger because the maximum of sums is smaller than the sum of maximums. We can then choose $\Upsilon = \widetilde{\mathcal{O}}\left(T^{1/3}V_T^{2/3}K^{-1/3}\right)$ to minimise the sum of Equation 8 and 9. It leads to the leading term of our Theorem 1. Notice that we still have to handle the first pull of each arm in each batch. If we bound roughly each first pull by V_T, we would get $K\Upsilon V_T \sim \widetilde{\mathcal{O}}\left(V_T^{5/3}\right)$ which would be the leading term for large V_T. Our Lemma 6 is more careful such that it leads to a second order term when $KV_T \leq o(T)$.

5 Rested rotting bandits

Setup We use the rotting setup of Seznec et al. (2019), which extends the one of Levine et al. (2017). This setup is *rested* non-stationary bandits: the change in arm's reward is triggered by the pulls. Hence, we have $\mu_i(t,n) = \mu_i(n)$. Thus, we note that $\overline{\mu}_i^h(t,\pi) = \overline{\mu}_i^h(N_{i,t-1}) = \frac{1}{h}\sum_{s=0}^{h-1}\mu_i(N_{i,t-1}-s)$. With a slight abuse of notations, we will also use $\widehat{\mu}_i^h(N_{i,t-1}) \triangleq \widehat{\mu}_i^h(t,\pi)$[4]. Let

$$L \triangleq \max_{i\in\mathcal{K}} \max_{n\in\{0,\dots,T-1\}} \mu_i(n) - \mu_i(n-1), \quad \text{with } \mu_i(-1) \triangleq \max_{j\in\mathcal{K}} \mu_j(0). \tag{10}$$

Hence, L bounds both the variation of μ_is between two consecutive pulls and the gaps between arms at the first pulls. This is an important quantity for the rested rotting analysis because the minimax rate for the noise-free case is $\mathcal{O}(KL)$ (Heidari et al., 2016).

Theoretical guarantees The analysis of `RAW-UCB` is straightforward from the analysis of `FEWA` due to their similarity. Thus, we recover the problem independent and dependent bounds (see Seznec et al. (2019) for a sketch of the proof, and App. F for a detailed analysis).

Proposition 6 (gap-free bound). *Let $\pi \in \{\pi_{\mathrm{F}}, \pi_{\mathrm{R}}\}$ tuned with $\alpha \geq 5$ or $\pi \in \{\pi_{\mathrm{EF}}, \pi_{\mathrm{ER}}\}$ tuned with $\alpha \geq 4$ and $m = 2$. For any rotting bandit scenario with means $\{\mu_i\}_i$ satisfying Assumption 1 with bounded decay L and any time horizon T, π suffers an expected regret,*

$$\mathbb{E}\left[R_T(\pi)\right] \leq C_\pi \sigma \sqrt{\log(T)}\left(\sqrt{KT} + K\right) + 6KL.$$

Proposition 7 (gap-dependent bound). *$\pi \in \{\pi_{\mathrm{F}}, \pi_{\mathrm{R}}\}$ tuned with $\alpha \geq 5$ (or $\pi \in \{\pi_{\mathrm{EF}}, \pi_{\mathrm{ER}}\}$ tuned with $\alpha \geq 4$ and $m = 2$) suffers an expected regret,*

$$\mathbb{E}\left[R_T(\pi)\right] \leq \sum_{i\in\mathcal{K}}\left(\frac{C_\pi^2\sigma^2\log(T)}{\Delta_{i,h_{i,T}^+-1}} + C_\pi\sigma\sqrt{\log(T)} + 6L\right)$$

with $h_{i,T}^+ \triangleq \max\left\{h \leq 1 + \frac{C_\pi^2\sigma^2\log T}{\Delta_{i,h-1}^2}\right\}$, and the pseudo-gap

$$\Delta_{i,h} \triangleq \min_{j\in\mathcal{K}} \mu_j\left(N_{j,T}^\star - 1\right) - \overline{\mu}_i^h\left(N_{i,T}^\star + h\right).$$

`RAW-UCB` matches the minimax rate (Prop. 6) up to poly-logarithmic factors. `RAW-UCB` improves over `FEWA`'s problem-dependent guarantee by a factor 4 (Prop. 7). Following Remark 1 of Seznec et al. (2019), one can identify $\Delta_{i,h} = \Delta_i$ in the stationary setting. It gives almost the same guarantee than in Theorem 3 when $\Upsilon_T = 1$ (stationary case). The difference comes from the increased α for the rested case. Indeed, in the rested case, the regret at each round t can be as bad as Lt. Hence, we reduce the probability of the bad event $\overline{\xi}_t$ (see Prop. 2). When the reward means are bounded (e.g. for Bernoullis), we can decrease the lower bound on α by one in Propositions 6 and 7.

[4] The average of the observations depends on the realization of the noise ε_t at time t. Yet, these h samples of noise are i.i.d. and thus do not perturb the analysis (see Prop. 2).

6 Real-word data experiment on Yahoo! Front Page

R6A - Yahoo! Front page today module user click log dataset This dataset was used for the Exploration and Exploitation Challenge[5] at ICML 2012 and inspired new algorithms. Among them we mention the work of Tracà and Rudin (2015) who noticed the non-stationary trend and took advantage of it. Since then the dataset continues to be a benchmark[6] for non-stationary bandits (Liu et al., 2018; Cao et al., 2019). It contains the history of clicks on news articles of 45 millions users in the first ten days of May 2009. We use three features in this dataset: *timestamp* (rounded every 5 minutes), *article_id*, and *click*.

A real decaying scenario Every day, between 6pm and 6am EST (12 hours), we notice a decreasing trend in click probability. It suggests that people in the US read less and less news during the evening and night. For every day, we keep all the articles which have been recommended at every timestamp during the 12 hours. For these articles, we use a rolling average window of 30000 in order to estimate the probability of click for each article at each timestamp [7]. We use the real total traffic for each timestamp. We highlight that *we do not enforce any of our assumptions* to create reward functions to be aligned with our setup. In particular, we do not enforce them to be piecewise constant nor to be decreasing. At each round, the learner receives 10 reward samples in order to reduce the cost of computation.

Algorithms and Parameters. We compare `RAW-UCB`, `FEWA`, `Exp3.S` and `GLR-UCB`. We refer to Appendix G for a discussion about missing algorithms and tuning. Note that our goal is to compare algorithms with the same tuning in the rested and restless benchmark.

Results We display the results for two different days. On day 2, there are several switches of optimal arms with many near-optimal ones: tracking the best arm is an "hard" problem. On day 7, one arm consistently dominates the others by far. Hence, it is an "easy" case where good algorithms should have a logarithmic regret rate. We show the six other days and running time in App. G.2.

[5]http://explochallenge.inria.fr/

[6]As it allows for offline evaluations as the actions were samples uniformly.

[7]For each timestamp, we average the values given by rolling average. These values are close to each other because the number of click opportunity per article in the same timestamp is small compared to 30000.

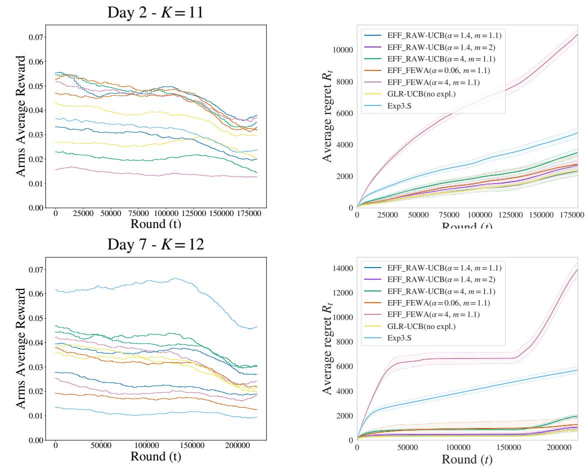

Figure 1: *Left:* rewards from the Yahoo! dataset for two days. *Right:* average regret over 500 runs.

`RAW-UCB` vs `FEWA`. The two algorithms compute the same statistics and share most of their analysis. Yet, `RAW-UCB` consistently outperforms `FEWA` on the full (rested and restless) benchmark. The difference between the two is even more significant in the restless case. Moreover, `RAW-UCB` is also simpler to implement and faster to run. Its theoretical tuning $\alpha = 4$ gets reasonable result, while theoretical `FEWA` is impractical. Finally, its empirical tuning $\alpha_{\mathrm{R}} = 1.4$ is similar to the asymptotic optimal tuning of `UCB` and shows good performance on both rested and restless problems. By contrast, `FEWA` with $\alpha_{\mathrm{F}} = 0.06$ shows worse performance with larger deviation on the restless benchmark.

`RAW-UCB` vs `Exp3.S`. In Appendix G.1, we show that random exploration of `Exp3.S` leads to high regret rate in rested rotting bandits. Unsurprisingly, `Exp3.S` recover more reasonable performance on the restless benchmark, on which it has theoretical guarantees. Yet, it is consistently outperformed by `RAW-UCB` when we tune the confidence bounds. It is particularly true on easy instance, e.g. on day 7. Indeed, on these cases, we expect logarithmic regret rate for `RAW-UCB`.

`RAW-UCB` vs `GLR-UCB` (no active exploration). `GLR-UCB` shows good results on the rested benchmark though it is less consistent than `RAW-UCB`. On the restless benchmark, `GLR-UCB` shows similar result than `RAW-UCB`. Yet, we highlight that 1) `GLR-UCB` needs the knowledge of the horizon to tune its change-detector; 2) we use an efficient version of `RAW-UCB` which runs ~ 10 times faster than `GLR-UCB`. In fact, the two algorithms are similar: they are UCB index policies, they recover logarithmic rate on easy restless rotting bandits problems and hence they would both suffer near-linear worst case regret rate in the general restless setting (when active exploration is turned off for `GLR-UCB`). The main difference is that `RAW-UCB` scans its history to select its rotting UCB's window, while `GLR-UCB` scans its history to detect significant changes and restart.

Acknowledgements We thank Lilian Besson for hosting and reviewing our numerical experiments code on his bandits package in python (Besson, 2018). The computational experiments were conducted using the Grid'5000 experimental testbed (Balouek et al., 2013).

References

Audibert, J.-Y. and Bubeck, S. (2009). Minimax policies for adversarial and stochastic bandits. In *Proceedings of the Conference on Learning Theory (COLT), 2009*, pages 217–226.

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

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (2003). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77.

Auer, P., Gajane, P., and Ortner, R. (2019). Adaptively Tracking the Best Bandit Arm with an Unknown Number of Distribution Changes. In Beygelzimer, A. and Hsu, D., editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 138–158, Phoenix, USA. PMLR.

Balouek, D., Amarie, A. C., Charrier, G., Desprez, F., Jeannot, E., Jeanvoine, E., Lèbre, A., Margery, D., Niclausse, N., Nussbaum, L., Richard, O., Perez, C., Quesnel, F., Rohr, C., and Sarzyniec, L. (2013). Adding Virtualization Capabilities to the Grid'5000 Testbed. In *Communications in Computer and Information Science*, volume 367 CCIS, pages 3–20. Springer Verlag.

Besbes, O., Gur, Y., and Zeevi, A. (2014). Stochastic Multi-Armed-Bandit Problem with Non-stationary Rewards. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 199–207. Curran Associates, Inc.

Besson, L. (2018). SMPyBandits: an Open-Source Research Framework for Single and Multi-Players Multi-Arms Bandits (MAB) Algorithms in Python. Online at: \url{GitHub.com/SMPyBandits/SMPyBandits}.

Besson, L. and Kaufmann, E. (2018). What Doubling Tricks Can and Can't Do for Multi-Armed Bandits.

Besson, L. and Kaufmann, E. (2019). The Generalized Likelihood Ratio Test meets klUCB: an Improved Algorithm for Piece-Wise Non-Stationary Bandits.

Bifet, A. and Gavaldà, R. (2007). Learning from time-changing data with adaptive windowing. In *Proceedings of the 7th SIAM International Conference on Data Mining*, pages 443–448.

Cao, Y., Wen, Z., Kveton, B., and Xie, Y. (2019). Nearly Optimal Adaptive Procedure with Change Detection for Piecewise-Stationary Bandit. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 418–427. PMLR.

Chapelle, O. and Li, L. (2011). An Empirical Evaluation of Thompson Sampling. In *Advances in Neural Information Processing Systems 24 (NIPS 2011)*, pages 2249–2257.

Chen, Y., Lee, C.-W., Luo, H., and Wei, C.-Y. (2019). A New Algorithm for Non-stationary Contextual Bandits: Efficient, Optimal and Parameter-free. In Beygelzimer, A. and Hsu, D., editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 696–726, Phoenix, USA. PMLR.

Cheung, W. C., Simchi-Levi, D., and Zhu, R. (2019). Learning to Optimize under Non-Stationarity. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1079–1087. PMLR.

Chow, Y. S. and Teicher, H. (1997). *Probability theory : independence, interchangeability, martingales.* Springer.

Garivier, A., Ménard, P., and Stoltz, G. (2019). Explore first, exploit next: The true shape of regret in bandit problems. *Mathematics of Operations Research*, 44(2):377–399.

Garivier, A. and Moulines, E. (2011). On upper-confidence bound policies for switching bandit problems. In *Proceedings of the 22nd International Conference on Algorithmic Learning Theory (ALT), 2011, Espoo, Finland.*, volume 6925 LNAI, pages 174–188. Springer, Berlin, Heidelberg.

Heidari, H., Kearns, M., and Roth, A. (2016). Tight Policy Regret Bounds for Improving and Decaying Bandits. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1562–1570.

Immorlica, N. and Kleinberg, R. (2018). Recharging bandits. In *Proceedings - Annual IEEE Symposium on Foundations of Computer Science, FOCS*, volume 2018-Octob, pages 309–319. IEEE Computer Society.

Komiyama, J. and Qin, T. (2014). Time-Decaying Bandits for Non-stationary Systems. In Liu, T.-Y., Qi, Q., and Ye, Y., editors, *Web and Internet Economics (WINE)*, pages 460–466, Cham. Springer International Publishing.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22.

Lattimore, T. and Szepesvári, C. (2020). *Bandit Algorithms*. Cambridge University Press UK.

Levine, N., Crammer, K., and Mannor, S. (2017). Rotting Bandits. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pages 3074–3083.

Liu, F., Lee, J., and Shroff, N. B. (2018). A change-detection based framework for piecewise-stationary multi-armed bandit problem. In McIlraith, S. A. and Weinberger, K. Q., editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 3651–3658. AAAI Press.

Louëdec, J., Rossi, L., Chevalier, M., Garivier, A., and Mothe, J. (2016). Algorithme de bandit et obsolescence : un modèle pour la recommandation (regular paper). In *Conférence francophone sur l'Apprentissage Automatique, Marseille, 05/07/2016-07/07/2016*, page (en ligne), http://www.lif.univ-mrs.fr. Laboratoire d'Informatique Fondamentale de Marseille.

Mukherjee, S. and Maillard, O.-A. (2019). Distribution-dependent and Time-uniform Bounds for Piecewise i.i.d Bandits.

Pike-Burke, C. and Grunewalder, S. (2019). Recovering Bandits. In H. Wallach and H. Larochelle and A. Beygelzimer and F. d'Alché-Buc and E. Fox and R. Garnett, editor, *Advances in Neural Information Processing Systems 32*, pages 14122—-14131. Curran Associates, Inc.

Russac, Y., Vernade, C., and Cappé, O. (2019). Weighted Linear Bandits for Non-Stationary Environments. In Wallach, H., Larochelle, H., Beygelzimer, A., d\textquotesingle Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 12040–12049. Curran Associates, Inc.

Seznec, J., Locatelli, A., Carpentier, A., Lazaric, A., and Valko, M. (2019). Rotting bandits are no harder than stochastic ones. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of Machine Learning Research, The 22nd International Conference on Artificial Intelligence and Statistics, 16-18 April 2019.*, volume 89 of *Proceedings of Machine Learning Research*, pages 2564–2572. PMLR.

Thompson, W. R. (1933). On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 25(3/4):285–294.

Tracà, S. and Rudin, C. (2015). Regulating Greed Over Time.

Warlop, R., Lazaric, A., and Mary, J. (2018). Fighting Boredom in Recommender Systems with Linear Reinforcement Learning. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 1757–1768. Curran Associates, Inc.

Wei, C.-Y., Hong, Y.-T., and Lu, C.-J. (2016). Tracking the Best Expert in Non-stationary Stochastic Environments. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 3972–3980. Curran Associates, Inc.

Whittle, P. (1980). Multi-Armed Bandits and the Gittins Index. *Journal of the Royal Statistical Society. Series B (Methodological)*, 42:143–149.

Whittle, P. (1988). Restless bandits: activity allocation in a changing world. *Journal of Applied Probability*, 25(A):287–298.

Zimmert, J. and Seldin, Y. (2019). An Optimal Algorithm for Stochastic and Adversarial Bandits. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 467–475. PMLR.