Stochastic bandits with arm-dependent delays

Anne Gael Manegueu [1] **Claire Vernade** [2] **Alexandra Carpentier** [1] **Michal Valko** [3]

Abstract

Significant work has been recently dedicated to the *stochastic delayed bandits* because of its relevance in applications. The applicability of existing algorithms is however restricted by the fact that strong assumptions are often made on the delay distributions, such as full observability, restrictive shape constraints, or uniformity over arms. In this work, we weaken them significantly and only assume that there is a bound on the tail of the delay. In particular, we cover the important case where the delay distributions vary across arms, and the case where the delays are heavy-tailed. Addressing these difficulties, we propose a simple but efficient UCB-based algorithm called the `PatientBandits`. We provide both problems-dependent and problems-independent bounds on the regret as well as performance lower bounds.

1. Introduction

In realistic applications of *reinforcement learning* (RL), rewards come *delayed*. In a game, for instance, the consequences of the agent's actions are only observed at the end. This issue at the core of challenges in RL (Garcia et al., 1966), when the horizon is finite or geometrically discounted. Even much simpler (state-less) *bandit* setups, such as online advertising suffer from delayed feedback (Chapelle & Li, 2011; Chapelle, 2014). In particular, most systems do not optimize for clicks but for *conversions*, which are events implying a stronger commitment from the customer. However, different ads trigger different customer response time. Typically, more expensive—and rewarding—products require more time to convert on customers' side, and the system needs to be tuned to be robust to the delays.

[1]Otto-von-Guericke University of Magdeburg, DE [2]DeepMind, London, UK [3]DeepMind, Paris, FR. Correspondence to: Anne Manegueu <anne.manegueu@ovgu.de>, Claire Vernade <vernade@google.com>, Alexandra Carpentier <alexandra.carpentier@ovgu.de>, Michal Valko <valkom@deepmind.com>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

As a result, we study *stochastic delayed bandits* for which the delay distributions are *arm-dependent* and possibly *heavy-tailed*. We consider the realistic model for delayed conversions of Chapelle (2014) and Vernade et al. (2017) in which delays are only *partially observable*. Conversions are *binary* events that represent a strong commitment (buying, subscribing, ...). If a conversion happens, it is sent with some delay to the learner who observes both its reward and the corresponding delay. Otherwise, the reward is null by default but it has no specific delay, only the current waiting time. This models a typical e-commerce application: if a customer does not buy the recommended product, the recommendation system will not be informed. The nature of this setup brings two main challenges (1) the *censoring* due to partially observed delays, which forces the learner to deal with an unknown amount of missing feedback; and (2) the *identifiability*[1] issue due to arm-dependent delays.

Prior work for delayed bandits have bypassed the challenges above by assuming that the delays are *observed* (Joulani et al., 2013; Dudik et al., 2011), which removes the ambiguity, or *bounded by a fixed quantity* (Pike-Burke et al., 2018; Garg & Akash, 2019; Cesa-Bianchi et al., 2018), which gives other possibilities to deal with them. Another approach that has been proposed by Vernade et al. (2017) is to drop the artificial requirement of observability of delays, and instead impose that all delays have the *same* distribution across arms and that this distribution is *known*. We further discuss the relevant related work in Section 3. While the known approaches yield good results under their strong assumptions on delays, none of them provides a solution to the realistic problem that we are tackling.

Contributions This work is the first to consider a stochastic bandit setting with arm-dependent, unbounded, and possibly heavy-tailed delays with partially observable delays. We jointly address the challenges of Vernade et al. (2017), Zhou et al. (2019) and Thune et al. (2019). Unlike Vernade et al. (2017); Zhou et al. (2019), we make only mild assumptions on the delays. Furthermore, we give a precise characterization of the impact of the delays on the regret than that given in the more difficult, non-stochastic setting of Thune

[1]Ex.: Consider two instances for : (1) reward follows a $\mathrm{Bernoulli}(1)$ and delay is a Dirac in $+\infty$ and (2) reward follows a $\mathrm{Bernoulli}(0)$ and delay is a Dirac in 0. Both instances produce the same data but have strictly different parameters.

et al. (2019). Our algorithmic soltuion is `PatientBandits`, the right calibration of upper confidence bounds and prove that it attains *problem-dependent and minimax regret upper bounds*. In particular, we prove that:

- In the *asymptotic* regime, the presence of delays does not affect the regret by more than a constant factor with respect to what is achieved in standard bandits. In other words, the loss of information due to the delays *does not lead to a significant increase* of the regret with respect to standards bandits Our algorithm attains the problem-dependent upper bound of the standard bandits up to a constant multiplicative factor in many cases, e.g. in the homoscedastic Gaussian case.
- On the other hand, we prove that there is a *drop in performance* with respect to problem-independent guarantees as compared to standard bandits. This is *unavoidable* and we prove a lower bound to support it.
- Finally, we study the *impact of imperfect prior knowledge* for `PatientBandits`. Our algorithm takes a parameter that is related to an upper bound on the heaviness of the tails of the delay distributions. We provide a comprehensive study in which respect the precise knowledge of this parameter can be avoided.

2. Bandits with delayed feedback

We define our *stochastic delayed bandit* setting. Consider a sequential game of $T \in \mathbb{N}^\star$ rounds where an agent is interacting with an environment characterized by a finite set of $K \in \mathbb{N}^\star$ arms which we denote $[K] \triangleq \{1, ..., K\}$. An instance is characterized by a tuple $((\mathcal{V}_i, \mathcal{D}_i)_{i\in[K]})$, where each arm $i \in [K]$ is associated with both

- an unknown *reward* distribution $\mathcal{V}_i$ whose support is in $[0, 1]$, and with mean μ_i,
- and an unknown *delay* distribution $\mathcal{D}_i$ with cumulative distribution function (CDF) τ_i and support in $\mathbb{N}$, such that for any $d \geq 0$, $t \leq T$, if $D_t \sim \mathcal{D}_i$, then we have that $\mathbb{P}(D_t \leq d) = \tau_i(d)$.

At each round $t \leq T$, the learner chooses (pulls) an arm $I_t \in [K]$. A reward $C_t \sim \mathcal{V}_{I_t}$ and a delay $D_t \sim \mathcal{D}_{I_t}$ are generated *independently from each other.* Neither the reward nor the delay is necessarily displayed at the current round t. However, at each upcoming round $t + u$ for $1 \leq u \leq T - t$, the learner observes the updated quantity

$$X_{t,u} \triangleq C_t \mathbf{1}\{D_t \leq u\}, \tag{1}$$

corresponding to her pull at time t. Note that, conversely, at time t, the learner only observes the updated quantities corresponding to its past actions: $(X_{s,t-s})_{s\leq t} \triangleq (C_s \mathbf{1}\{D_s \leq t - s\})_{s\leq t}$. And therefore at round t, it disposes of the entire history information

$$\mathcal{H}_t \triangleq (X_{u,v})_{u<t, v\leq t-u}. \tag{2}$$

This setting is summarized in Figure 1.

Setting: K arms, horizon T, reward distributions $(\mathcal{V}_i)_{i\leq K}$, delay distributions $(\mathcal{D}_i)_{i\leq K}$
for $t = 1$ **to** T
- learner observes updated reward sequence $(X_{s,t-s})_{s\leq t}$, see Eq. 1
- learner chooses $I_t \in [K]$ based on $\mathcal{H}_t$, see Eq. 2
- reward $C_t \sim \mathcal{V}_{I_t}$ and delay $D_t \sim \mathcal{D}_{I_t}$ are generated independently but not necessarily displayed

end for

Figure 1. Delayed learning setting

Note that delays are only *partially observable*: at round t and for some $s \leq t$, if the learner observes $X_{s,t-s} = 0$, there is an *ambiguity*. Either the reward C_s is actually indeed 0, or the delay is not yet passed, i.e., $t - s < D_s$. This ambiguity is due to the *multiplicative noise* induced by the delays. Indeed, conditionally on the action taken at time $u < t$, $I_u \in [K]$, the expected observable payoff at round t is scaled by some delay and action dependent factor

$$\mathbb{E}[X_{u,t-u} \,|\, I_u = i] = \tau_i(t - u)\mu_i.$$

In other words, the delays induce temporarily missing data among the observations, but the learner cannot know exactly *how much feedback is missing*.

Indeed, the heavier the tail of the delay distribution of an arm, the longer it takes for the learner to be able estimate its mean well. This creates dramatic *identifiability* issues: if the best arm is more delayed than the others, its apparent value might seem lower for a while and only a learner that is patient enough shall rightfully identify it as the optimal action. To mitigate this issue and to give a chance to a learner to tune its patience level, we rely on the following assumption.

Assumption 1 (α-polynomial tails for the delay distributions)**.** *Let $\alpha > 0$ be some fixed quantity. We assume that $\forall m \in \mathbb{N}^*$ and $\forall i \in \{1, \dots, K\}$, it holds that*

$$|1 - \tau_i(m)| \leq m^{-\alpha}.$$

The smaller α, the more heavy-tailed the delay distribution, and the more difficult the setting. This assumption needs to hold uniformly across arms but does not impose they all have the same distribution, unlike required by Vernade et al. (2017). This is an important weakening of the restricted setting of the prior work, which we generalize.

For $i \in [K]$, we denote by $T_i(t) \triangleq \sum_{i=1}^{t} \mathbb{I}\{I_s = i\}$ the number of times that the arm i has been drawn up to round t. As $\mu^\star \triangleq \max_i \mu_i$ denotes the mean of the best arm(s), $\Delta_i \triangleq \mu^\star - \mu_i$ is the gap between the mean of the optimal arm(s) and the mean of arm i. The goal of the agent is to maximize its expected cumulative reward (i.e., $E[\sum_{t=1}^{T} C_t]$) after T rounds and therefore to minimize the expected regret,

$$\overline{R}_T = T\mu^* - \mathbb{E}\sum_{t=1}^{T} C_t = \sum_{i=1}^{K} \Delta_i \mathbb{E}[T_i(T)]. \qquad (3)$$

Remark 1. *In this paper, we consider the same concept of regret as for standard bandits, unlike what is done by Vernade et al. (2017). We believe it is a more relevant approach that allows for comparison with the vast existing prior work on the topic (see Section 3).*

3. Related work

The problem of learning with delayed feedback is ubiquitous in applications, including universal portfolios in finance (Cover, 2011), online advertising (Chapelle, 2014) and e-commerce (Yoshikawa & Imai, 2018). Therefore, there is a large body of theoretical results under different scenarios and assumptions on the delays. We review the contributions of prior work distinguishing between the full information setting and the bandit setting.

Full-information In online (convex) optimization, as opposed to our setting, the learner typically performs a gradient descent by estimating the gradient on the fly using the available information. Thus, delayed feedback forces the learner to make decisions in the face of additional uncertainty. This setting has been considered by Weinberger & Ordentlich (2002) in the case of prediction of individual sequences under the assumption that the *delays were fixed and known*. The study of online gradient type of algorithms under possibly random or adversarial delays is made under various hypotheses by Langford et al. (2009); Quanrud & Khashabi (2015); Joulani et al. (2016). In distributed learning, communication time between servers naturally induces delays, and this particular setting was studied by Agarwal & Duchi (2011); McMahan & Streeter (2014); Sra et al. (2015) who all proposed asynchronous or delay-tolerant versions of AdaGrad. Finally, an attempt to reduce the impact of delays was made by Mann et al. (2018) by allowing the learner to observe intermediate signals.

Bandits with observed delays Joulani et al. (2013) provide a clear overview of the impact of the delays for both the stochastic and adversarial setting. Using a method based on a non-delayed algorithm called `Base`, they succeed in extending the work of Weinberger & Ordentlich (2002) to the non-constant delays case. Following this work, Mandel et al. (2015) argued in favor of more randomization to improve exploration in delayed environments, although the guarantees remained unchanged. Taking a different path, closer to ours, the recent work of Zhou et al. (2019) relies on a *biased estimator* of the mean and corrects for it in the UCB using an estimator of the amount of missing information. They consider the contextual bandit setting and make a strong assumption on the delay distribution, imposing that 1) delays are fully observable, and 2) the delay distribution is the same for all arms and should concentrate nicely (bounded expectation). Due to these assumptions, their algorithm cannot be used for our setting and cannot be compared to ours. Thune et al. (2019) considered the adversarial bandit setting, where delays are observed right after (or before) sampling an arm. Unfortunately, their results and algorithms *do not* apply in our setting since we *do not* observe the delays before or right after sampling an arm.

Bandits with *partially* observed delays The delayed bandits with censored observations were introduced by Vernade et al. (2017), who build on the real-data analysis of Chapelle & Li (2011). They rely on the major assumption that delays are the same across arms and have a finite expectation. In this setting they prove an asymptotic problem-dependent lower bound that recovers the standard Lai & Robbins' lower bound. They propose an algorithm that uses as input the CDF of the delay distribution and matches this asymptotic lower bound. In other words, they prove that asymptotically, well-behaved delays have no impact on the regret. Following this work, Vernade et al. (2018); Arya & Yang (2019) extend its setting to the linear and contextual stochastic ones. Two more results consider the case where the delays are not observed at all but are bounded by a constant $D > 0$. Garg & Akash (2019) analyze the stochastic setting and Cesa-Bianchi et al. (2018) the adversarial setting and achieve a regret of order $\sqrt{TK \log K} + KD \log T$ and $\sqrt{DTK}$ respectively. Pike-Burke et al. (2018) further considers unbounded delays in adversarial setting but time under the assumption that only their expectation is bounded. Again, these results do not apply to our context as we do not assume that the delays are bounded; under Assumption 1 with $\alpha < 1$, the *delays can even have infinite means*.

4. The `PatientBandits` algorithm

In this section we describe an optimistic algorithm (Auer et al., 2002) that is able to cope with partially observed and potentially heavy-tailed delays. `PatientBandits` estimates high-probability upper confidence bounds on the parameter of each arm. As opposed to the standard UCB approach, it is hopeless to design conditionally unbiased estimators in this delayed setting. Therefore, the algorithm needs to also properly bound the *bias* for each arm adaptively. Throughout the paper, we use the notation

$A \wedge B \triangleq \min(A, B)$ and $A \vee B \triangleq \max(A, B)$.

A delay-corrected, high-probability UCB Delays being partially observable, the learner must build its estimators *with an unknown number of observations*. Indeed, since rewards are delayed, a certain proportion of the feedback of each arm is missing but it is impossible to know exactly how much because the *zeros are ambiguous*. Nonetheless, we show that it is possible to prove high-probability confidence bounds for the parameters of the problem, provided that we correctly handle this extra bias due to the delays. For this purpose, we rely on Assumption 1, that gives us a loose global bound on the tails of the distributions of the delays.

At a time $t \geqslant K+1$, given the history of pulls and observed rewards $\mathcal{H}_t$, we define the mean estimator,

$$\widehat{\mu}_i(t) \triangleq \frac{1}{T_i(t)} \sum_{u=1}^{t} X_{u,t-u} \mathbf{1}\{I_u = i\}, \tag{4}$$

where $T_i(t) \triangleq \sum_{u=1}^{t} \mathbf{1}\{I_u = i\}$. The key ingredient of our algorithm is the upper confidence bound. Our first major provides just that and is presented next.

Theorem 1. *Let $i \in [K]$ and $\alpha > 0$ satisfy Assumption 1. Then for any $t > K$ and $\delta > 0$, with probability $1 - \delta$*

$$|\widehat{\mu}_i(t) - \mu_i| \leq \left(\frac{2 \log \frac{2}{\delta}}{T_i(t)}\right)^{1/2} + 2T_i(t)^{-(\alpha \wedge 1/2)}. \tag{5}$$

Proof. The full proof is in Appendix A and relies on the following decomposition

$$|\widehat{\mu}_i(t) - \mu_i| \leq \left|\widehat{\mu}_i(t) - \frac{1}{T_i(t)} \sum_{u=1}^{t} \tau_i(t-u) \mu_i \mathbf{1}\{I_u = i\}\right| + \left|\frac{1}{T_i(t)} \sum_{u=1}^{t} \tau_i(t-u) \mu_i \mathbf{1}\{I_u = i\} - \mu_i\right|.$$

On the right-hand side, the first term is a standard deviation term. The probability that it is larger than $(2 \log(2\delta)/T_t(t))^{1/2}$ is uniformly bounded by $\delta/2$. The second term corresponds to the bias and is bounded by $2T_i(t)^{-\alpha \wedge 1/2}$, which comes from simply summing the $\tau_i(t-u)$ in the worst case, i.e., when all pulls of arm i are made in the last $T_i(t)$ rounds. □

A clear benefit of the above result is a simple and easy-to-compute *adaptive* upper bound on the parameter μ_i for our estimator. This UCB is similar to the standard UCB2 (Auer et al., 2002) except for the *extra bias term* that goes to zero with the number of pulls. In fact, it adaptively trades off bias and variance as a function of α: it is the largest for small values of $\alpha \leq 1/2$, that is when delays have very large tails. Indeed, α plays an important role in our algorithm presented in details next.

Algorithm 1 `PatientBandits`

Input: $\alpha > 0$, horizon T, number of arms K.
Initialisation: Pull each arm once and set for all $i \in [K]$: $T_i(t) = 1$ and initialise $\widehat{\mu}_i(t)$ according to Eq. 4.
for $t = K+1 ... T$ **do**
  Pull arm $I_t \in \arg\max_{i \in [K]} \text{UCB}_i(t)$
  Observe all feedback updates $(X_{s,t-s})_{s \leq t}$
end for

Algorithm `PatientBandits` is described in Algorithm 1. It receives as input the parameter $\alpha > 0$, the horizon T, and the number of arms K which we assume to be smaller than T. In the first phase of the game, all arms are pulled once. The player then pulls the arm from $[K]$ that has the highest UCB as defined in Theorem 1,

$$\text{UCB}_i(t) \triangleq \widehat{\mu}_i(t) + \left(\frac{2 \log(2KT^3)}{T_i(t)}\right)^{1/2} + 2T_i(t)^{-(\alpha \wedge 1/2)}.$$

The algorithm then pulls an arm I_t that maximises $\text{UCB}_i(t)$.

5. Analysis of `PatientBandits`

We analyse `PatientBandits` and provide a non-asymptotic lower-bound for delayed bandits. We first provide first a problem-dependent upper bound on the regret of `PatientBandits`. with proof in Appendix B and follows the lines of the usual analysis of UCB by Auer et al. (2002); see also Lattimore & Szepesvári (2019, Chapter 7).

Theorem 2. *Let $T > K \geq 1$ and $\alpha > 0$. Let $(\mathcal{V}_i, \mathcal{D}_i)_{i \in [K]}$ be the problem as defined in Section 2 such that Assumption 1 holds. If* `PatientBandits` *is run with parameters $\mathcal{P} = (\alpha, T, K)$, its cumulative regret is bounded as*

$$\overline{R}_T \leq \sum_{i : \Delta_i > 0} \left[\frac{64 \log(2T)}{\Delta_i} \vee \left(\frac{8}{\Delta_i}\right)^{\frac{1-\alpha}{\alpha} \vee 1}\right] + 2K.$$

The only term in Theorem 2 that depends on T is of the order of $\sum_{i:\Delta_i > 0} \log(T)/\Delta_i$. It is of the same order as the classical bound for UCB which is asymptotically optimal; see Lai & Robbins (1985). Note that this was expected, since Vernade et al. (2017) showed that delays should not have an asymptotic impact on the regret[2]. Our bound has an additional term of order $\sum_{i:\Delta_i > 0} (8/\Delta_i)^{\frac{1-\alpha}{\alpha} \vee 1}$. This term does not depend on T, so it is asymptotically negligible. But if $\alpha < 1/2$ and some of the gaps are very small, it can be large from a non-asymptotic perspective—see Section 7 for a discussion.

We now provide a problem-independent upper bound.

[2]Their result is stated for delays with finite expectation but remains valid in our setting.

Theorem 3. *Let $T > K \geq 1$ and $\alpha > 0$. If* `PatientBandits` *is run with parameters $\mathcal{P} = (\alpha, T, K)$, for any stochastic delayed instance such that Assumption 1 holds, it achieves*

$$\overline{R}_T \leq 2 \times 64^{(1-\alpha)\vee 1/2} T^{1-\alpha\wedge 1/2} (K \log(2T))^{\alpha\wedge 1/2} + 2K.$$

Up to logarithmic terms and multiplicative constants, the order of magnitude of this bound is $\max(\sqrt{KT}, K^\alpha T^{1-\alpha})$. Whenever $\alpha \geq 1/2$, the order of the bound is $\sqrt{KT}$; as is the case for UCB (up to logarithmic terms) and for more refined algorithms like `MOSS` (Audibert & Bubeck, 2009) *in the standard stochastic bandit setting (without delays)*. However if the delays are allowed to be more heavy-tailed, i.e., $\alpha < 1/2$, then the regret starts degrading with α as the upper bound is of order $K^\alpha T^{1-\alpha}$. We prove that this degradation of the (problem-independent) regret is *unavoidable*.

Theorem 4. *Consider $K = 2, T \geq K$, and $\alpha > 0$. There exists a Bernoulli stochastic delayed bandit problem satisfying Assumption 1, such that the expected regret of any algorithm $\overline{R}_T$ on this problem is larger than $T^{1-\alpha}/8$.*

Combining the above theorem with the standard problem independent lower bound for standard bandits (Lattimore & Szepesvári, 2018) we get that the order of magnitude of the worst case regret (for bandit instances satisfying Assumption 1) of any algorithm is larger than $\max(\sqrt{KT}, T^{1-\alpha})$. This matches (up to logarithmic terms) the upper bound in Theorem 3 *with respect to* T (not to K whenever $\alpha < 1/2$).

6. Adaptation to α

`PatientBandits` requires (a lower bound on) α as input. It is indeed natural to ask whether this prior information on the delays is necessary. In other words, can we design an algorithm that learns α as well or adapts to the delays on-the-fly? How much would the regret be impacted? We now give a precise answer to these questions, both in the asymptotic and non-asymptotic regime. In the latter, we prove a negative result in the general case. However, we propose a new assumption under which adaptivity is achievable.

6.1. Adaptation of the problem dependent regret to α

We first study possibilities of adaptation in the asymptotic regime. An immediate corollary of Theorem 2 is as follows.

Corollary 1. *Let $T > K \geq 1$ and consider a bandit problem with minimum gap $\overline{\Delta} = \min_{k:\Delta_k>0} \Delta_k$, and where all arms k satisfy Assumption 1 for α_k. Consider $T > e^{e^e}$ large enough so that (a) $\log\log(T)/\log(T) \leq \min_i \alpha_i$; (b) $8\overline{\Delta}^{-1} \leq \log T$. If* `PatientBandits` *is run with parameters $((\log\log(t)/\log(t))_{t\leq T}, T, K)$,*

$$\overline{R}_T \leq \sum_{i:\Delta_i>0} \left[\frac{128\log(2T)}{\Delta_i} \vee \left(\frac{8}{\Delta_i}\right)^{\frac{1-\alpha}{\alpha}\vee 1} \right] + 2K.$$

Therefore, for T large enough depending on instance dependent quantities, it is possible to run a slight variant of `PatientBandits` that takes as input the sequence $(\alpha_t = \log\log(t)/\log(t))_{t\geq 1}$ instead of a fixed $\overline{\alpha}$. As a result, for a fixed problem, asymptotically, knowing α is not necessary.

6.2. Impossibility result under Assumption 1 for adapting the problem independent regret to α

For a fixed horizon $T < \infty$, however, that is whole different story. Our second lower bound below states that if you give a input parameter α that is *too small* to a *good* algorithm, then it has a suboptimal regret, even in the simpler case where $K = 2$. Specifically, we define the class of *α-optimal algorithms* $\mathcal{A}_\alpha$ as the algorithms whose expected regret is smaller than $T^{1-\alpha}/8$ for all bandit instances satisfying Assumption 1 for a fixed α.

Theorem 5. *Consider $K = 2$, $T \geq K$, and fix $\alpha > 0$. For any $\beta \geq \alpha$, there exists a Bernoulli bandit instance satisfying Assumption 1 for β such that the expected regret $\overline{R}_T$ of any α-optimal algorithm is larger than $T^{1-\alpha}/8 > T^{1-\beta}/8$.*

In other words, an algorithm that performs optimally uniformly under Assumption 1 for a given α *cannot at all* adapt to $\beta \geq \alpha$.

6.3. New algorithm for adapting the problem independent regret to α under more restrictive assumptions

Yet, is adaptivity a lost cause? Under the weak Assumption 1, Theorem 5 above is quite disheartening. A structural reason for this is that it is impossible to estimate α under this assumption. We show that under a *slightly* more restrictive assumption this becomes possible.

Assumption 2. *Assume that there exists $0 < c \leq 1$ and $\overline{\mu} > 0$, such that $\min_k \mu_k > \overline{\mu}$ and for all $i \in [K]$,*

$$cm^{-\alpha} \leq |1 - \tau_i(m)| \leq m^{-\alpha}. \tag{6}$$

Assume also that $\alpha \geq \underline{\alpha}$ for some $\underline{\alpha} > 0$.

This assumption *does not mean* that the delay distributions of the arms are all the same. It means that the parameter α now globally characterizes the tails of the delay distributions. The challenge for estimating it is that delays are only *partially observable*. To further explain Assumption 2, let us consider small and a large delay d and D, such that $D > d > 0$. The conditional expectation of the difference

of the same reward after respectively d and D time steps have passed is

$$\mathbb{E}_{|I_t}[X_{t,D} - X_{t,d}] = \mu_{I_t}\tau_{I_t}(D) - \mu_{I_t}\tau_{I_t}(d) \in [c\mu_{I_t}d^{-\alpha} - \mu_{I_t}D^{-\alpha}, \mu_{I_t}d^{-\alpha}],$$

where $\mathbb{E}_{|I_t}$ is the *conditional* expectation with respect to the arm I_t pulled at time t, and where $c > 0$ comes from Assumption 2. Therefore, if $d \leq (c/2)^{1/\underline{\alpha}}D$, we now have that[3]

$$\mathbb{E}_{|I_t}[X_{t,D} - X_{t,d}] \in \left[\frac{c\overline{\mu}}{2}d^{-\alpha}, d^{-\alpha}\right],$$

where $\overline{\mu}, \underline{\alpha} > 0$ are defined in Assumption 2. We can now see that it is possible to estimate α up to a logarithmic factor using the logarithm of an estimator of $\mu_i\tau_i(D) - \mu_i\tau_i(d)$, if we properly choose d and D from some arm i sampled often enough. We formalize this idea next and start by introducing the necessary quantities followed by its analysis.

In the rest of this section, we denote $\overline{I}_t \triangleq \arg\max_k T_k(t)$. We only use the samples of this arm to estimate α at each round. For a simpler notation, let $\overline{T}_t \triangleq T_{\overline{I}_t}(t)$ and for some delay D, let

$$\overline{m}_{t,D} \triangleq \frac{1}{\overline{T}_{t-D}}\sum_{s=1}^{t-D} X_{s,D}\mathbf{1}\{I_s = \overline{I}_t\}$$

be the sample mean after waiting D steps. For the estimate, we set $D_t \triangleq \lfloor \overline{T}_t/2 \rfloor$ and $d_t \triangleq \left\lfloor (c/2)^{1/\underline{\alpha}}D_t \right\rfloor$. Subsequently, we define the estimator of α at round t as

$$\widehat{\alpha}_t \triangleq \min\left(-\frac{\log(\overline{m}_{t,D_t} - \overline{m}_{t,d_t})}{\log(\overline{T}_t)}, \frac{1}{2}\right).$$

Such an estimator is related to quantile or CDF-based estimators used in extreme value theory (De Haan & Ferreira, 2007; Carpentier & Kim, 2015). We define set the lower confidence bound on it as

$$\overline{\alpha}_t \triangleq \left[\widehat{\alpha}_t - \frac{\log\left(2^4\sqrt{\log(2KT^3)}/(c\overline{\mu})\right)}{\log(\overline{T}_t)}\right] \vee 0.$$

`Adapt-PatientBandits` then simply uses $\overline{\alpha}_t$ for the computation of the upper confidence bounds as in Eq. 5. The algorithm therefore does not need to know for which parameter α Assumption 2 is satisfied. We show the pseudocode `Adapt-PatientBandits` in Algorithm 2. We bound the expected regret of `Adapt-PatientBandits` in the following theorem.

Theorem 6. *Let $T > K \geq 1$ and $\alpha, \underline{\alpha}, c, \overline{\mu} > 0$, such that Assumption 2 holds. The expected regret of* `Adapt-PatientBandits` *is bounded as*

$$R_T = \widetilde{\mathcal{O}}(K^{\alpha}T^{1-\alpha\wedge 1/2}).$$

[3]Note that $c \leq 1$ so that with this definition of d and D, we have that $D \geq d$.

Algorithm 2 `Adapt-PatientBandits`

Input: $c, \underline{\alpha}, \overline{\mu}, T, K$
Initialisation: Pull each arm twice.
for $t = 2K+1, ..., T$ **do**
  Pull the arm $I_t \in \arg\sup_{i\in\{1,...,K\}} \text{UCB}_i(t-1)$ when using the parameter $\overline{\alpha}_t$ in the UCB.
  Observe all individual feedback $(X_{s,t-s})_{s\leq t}$.
end for

Comparing the above with Theorem 3, notice that `Adapt-PatientBandits` achieves a regret that depends on the *unknown* parameter α and is *of the same order* as the upper bound of Theorem 3 up to logarithmic term. This algorithm is therefore *minimax optimal* up to logarithmic terms relatively to the lower bound defined in Section 5.

7. Discussion

Comparison to bandits without delays As discussed in the Section 4, the major difference between `PatientBandits` and the standard UCB algorithm is the *extra bias term*. In the standard bandits, we have strictly more information than in our setting. When rewards are delayed, the learner has to deal with temporarily missing data: some actions have been taken but their rewards are missing until the delay has passed. In particular, when $\alpha < 1$, the delays are heavy-tailed, so their expectation is infinite, and this buffer of missing data always keeps growing in size with T, creating a non-negligible bias in the estimators. This difference is even more important when $\alpha < 1/2$, which corresponds to the situation where the bias term is larger than the standard deviation term. For such a instance, it is clear that a standard UCB would have a linear regret and we show it Section 8.

We now discuss optimality of `PatientBandits` by commenting both our problem-dependent (Theorem 2) and problem-independent (Theorem 3) guarantees.

- The problem-dependent bound for our setting is of the order $\sum_{k:\Delta_k>0}\log(T)/\Delta_k$. Up to a term that depends only on the $(\Delta_k)_k$ (and not on T, see Section 5), this is of the same order than the problem-dependent bound in standard bandits (Lai & Robbins, 1985) *and* it does *not* depend on the delay distributions, e.g. it does not depend on α.

- On the other hand, the problem-independent bound, is of order $T^{1-\alpha\wedge(1/2)}K^{\alpha\wedge(1/2)}$ up to logarithmic terms. This is light-years different from the problem-independent bound for standard bandits, which is of order $\sqrt{KT}$; see Lattimore & Szepesvári (2019, Chapter 7). In particular, the bound is of same order when

$\alpha \geq 1/2$, and is larger when $\alpha < 1/2$. However, Theorem 4 ensures that this rate is minimax optimal with respect to T, up to a multiplicative term $K^{\alpha \wedge (1/2)}$ and some logarithmic terms. This means that this *gap is the price to pay for having delays* that are potentially long, and can therefore pose strong bias issues.

Parameters `PatientBandits` needs α as as input, which is the *only* external prior information on the delays. It is a more delicate question to decide whether this prior information is necessary. Section 6 treat this subject extensively. In a nutshell, from an asymptotic perspective, the knowledge of α is not needed, but from a non-asymptotic perspective it is. Nonetheless, under a slightly stronger hypothesis on the delay distribution, see Assumption 2, there exists a *fully adaptive* algorithm Adapt-`PatientBandits`. with sublinear regret (Theorem 6) that matched the one of `PatientBandits` up to a logarithmic term.

8. Experiments

We now evaluate the empirical performance of `PatientBandits`. Throughout the section, we provide experiments where the delays of each arm i follows the Pareto Type I distribution with tail index α_i, where the rewards of each arm i follow a Bernoulli distribution with parameter μ_i. First, we investigate the performances of `PatientBandits` with respect to the parameters of the instance, $(\alpha_i, \Delta_i)_{i \in [K]}$, and to the hyperparameter of the algorithm α, which we here denote by $\overline{\alpha}$ to avoid confusion.

Second, we compare to strongest baseline that deals with *partially observed delays*, which is the censored version of `D-UCB` of Vernade et al. (2017) with various threshold windows. Their algorithm takes two parameters, the threshold m and the CDF τ of the delays.[4] The threshold m calibrates the time that the algorithm waits before updating reward.

8.1. Impact of the instance properties

Study of $\overline{\alpha}$ `PatientBandits` takes $\overline{\alpha}$ as an input. The choice of this parameter is a key for the implementation of `PatientBandits`. Ideally we would like to take $\overline{\alpha} = \min_i \alpha_i$ but in the absence of information on the delay distributions we cannot. We therefore illustrate the sensitivity of our method to the miscalibration of $\overline{\alpha}$. We consider a 2-arm setting with horizon $T = 3000$, with arm means $\mu = (0.5, 0.55)$ and with tail index $\alpha_1 = 1, \alpha_2 = 0.3$ respectively. We consider $\overline{\alpha} \in [0.02, 0.5]$ an in Figure 2 show the regret as function of $\overline{\alpha}$. Note that the regret first decreases with $\overline{\alpha}$ and then increases after approximately

[4] Vernade et al. (2017) assumed that the delay distributions are *know and homogeneous across arms*.

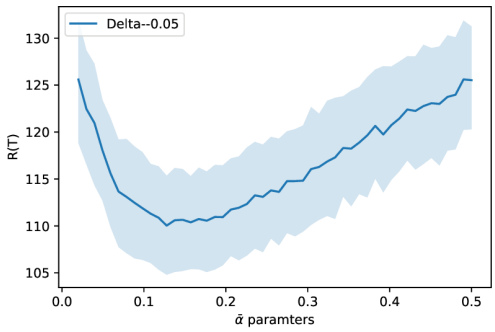

Figure 2. Regret at round $T = 3000$ of `PatientBandits` in function of $\overline{\alpha} \in [0.02, 0.5]$ for the bandit instance $\mu = (0.5, 0.55)$, $\alpha_1 = 1$, and $\alpha_2 = 0.3$. Results are averaged over 400 runs.

$\overline{\alpha} = 0.3$. This is precisely what is expected. For small $\overline{\alpha}$, the algorithm is consistent but explores too much and this induces a large regret. For $\overline{\alpha}$ larger than $\min_i \alpha_i$, the regret starts to increase again, since the bias coming from the delays are not sufficiently taken into account by the UCB.

Study of $(\alpha_i, \Delta_i)_{i \in [K]}$. We now investigate the dependency of the regret of `PatientBandits` on the delay parameters and arm gaps $(\alpha_i, \Delta_i)_{i \in [K]}$. We consider the following two-arm instance where we set $\mu = (0.4, 0.4 + \Delta)$ where we take $\Delta \in [0.02, ..., 0.6]$ - fixing the horizon to $T = 3000$. For each problem, we respectively choose $\alpha_1 = 1$ and $\alpha_2 \in \{0.2, 0.3, 0.4, 0.5, 0.8\}$ and run the `PatientBandits` policy with optimal parameter $\overline{\alpha} = \alpha_2$, so that we can see the impact of α_2 and Δ independently from calibration issues. The results represented in the Figure 3 display the influence of the arm gap Δ on the regret, for various values of α_2.

Figure 3 illustrates a standard phenomenon in stochastic bandits, and which also holds for delayed bandits: for small values of the arm gap Δ, the regret increases with Δ. This corresponds to the fact that for small Δ, the algorithm explores and is not able to focus on the most promising arms since the arms means are too close. Then at some point for larger values of Δ the regret starts decreasing, as predicted by the bound in Theorem 2. A phenomenon that is specific to delayed bandits is that the smaller α_2, the larger the regret. This is expected from Theorem 3, since the smaller α_2, the more delayed the rewards, and the harder the problem. A more subtle phenomenon, also illustrating Theorem 3, is that the smaller α_2, the larger the value and the position of the maximum of each curve - the maximum or the regret being bounded by the problem independent bound that depends here on $\overline{\alpha} = \alpha_2$.

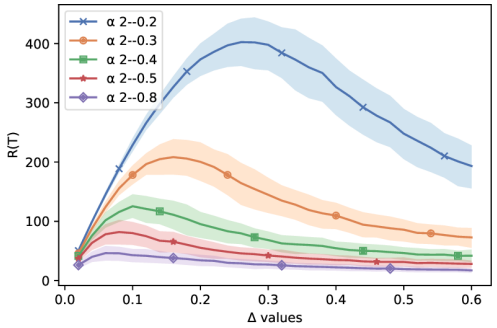

Figure 3. Regret of `PatientBandits` in function of the arm gap $\Delta \in [0.02, 0.6]$ where the bandit problem is characterized by $\mu = (0.5, 0.5 + \Delta)$ and $\alpha_1 = 1$ and where $\alpha_2 \in \{0.2, 0.3, 0.4, 0.5, 0.8\}$ - each curve corresponds to a different value of α_2. For each problem, `PatientBandits` was run with horizon $T = 3000$ and with parameter $\overline{\alpha} = \alpha_2$.

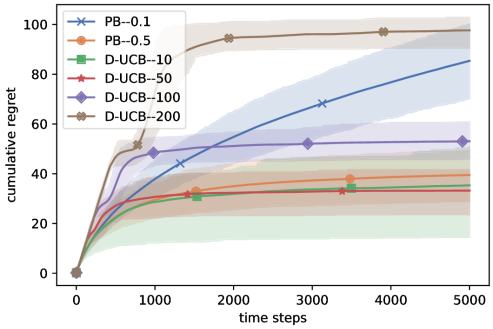

Figure 4. Regret of D-UCB and `PatientBandits` for $\mu = (0.6, 0.8)$ and with homog. delay distributions characterised by $\alpha_1 = \alpha_2 = 0.7$. We plot results for `PatientBandits` with parameters $\overline{\alpha} = (0.1, 0.5)$, and for D-UCB with parameters $m = (10, 50, 100, 200)$.

8.2. Comparison with with D-UCB.

We compare here the regret of `PatientBandits` with the one of D-UCB as a function of the horizon T, for different values of the parameters - respectively $\overline{\alpha}$ and m. Since D-UCB was designed for the context where the distribution of the delays is the same for all arms, i.e. α_i identical over arms, we consider that scenario as well as the more general case with heterogeneous α_i's.

Homogeneous delay distributions across arms. In the first scenario, we consider a two armed bandit problem with means $\mu = (0.6, 0.8)$. We set the same tail index for both arms, i.e. $\alpha_1 = \alpha_2 = 0.7$. We run `PatientBandits` for $\overline{\alpha} \in \{0.1, 0.5\}$. For D-UCB we consider various threshold parameters $m \in \{10, 50, 100, 200\}$, and feed D-UCB with the *exact delay distribution of the arms* - which gives D-UCB an important edge over our algorithm. The results are displayed in Figure 4 (regret as a function of time).

The performances of D-UCB and `PatientBandits` are comparable, in particular in the case of good calibration of the parameters, i.e. respectively $\overline{\alpha} = 0.5$, and $m = 50$. `PatientBandits` performs slightly worse in the best case, but note that D-UCB is tuned with the full knowledge of the CDF of the delays. An observation coming from Figure 4 is the presence of long lasting linear phases at the initial stage of learning of D-UCB for large m which tend to be caught up over time. This comes from the structure of the algorithm, and from the fact that it has to wait until $m + K$ time steps before it starts exploiting the observations - which is not the case for our strategy.

Non-homogeneous delay distributions. In the second scenario we still consider a two-armed bandit problem with means $\mu = (0.6, 0.8)$, and we set the parameters of the tail distribution of the delays as $\alpha_1 = 1, \alpha_2 = 0.3$. This is a 'difficult' scenario, since arm 2 which has the highest mean has also the lowest delay parameter. This means that its delays are more heavy tailed. We consider as before `PatientBandits` with parameters $\overline{\alpha} \in \{0.1, 0.5\}$, and the D-UCB for threshold parameters $m \in \{10, 50, 100, 200\}$. Regarding the CDF parameter of D-UCB, we provide a Pareto distribution with parameter 0.7. The results for all policies are displayed on Figure 5 (regret in function of horizon T). We observe that D-UCB has a very high regret,

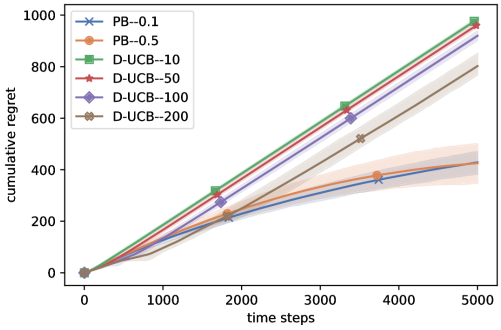

Figure 5. Regret of D-UCB and `PatientBandits` for $\mu = (0.6, 0.8)$ and with delay distributions that vary across arms, characterized by $\alpha_1 = 1$ and $\alpha_2 = 0.3$. We plot results for `PatientBandits` with parameters $\overline{\alpha} = (0.1, 0.5)$, and for D-UCB with parameters $m = (10, 50, 100, 200)$.

increasing linearly with T. This can be explained by the fact that D-UCB *cannot* adapt to delay distributions varying across arms. In other words, it does not take the heterogeneity of the delays into account and can be confused by this difficult situation where the best arm also corresponds to the longest delays. Consequently, D-UCB focuses only on observations that are substantially biased and misidentifies

the best arm. On the other hand, `PatientBandits` adapts to the heterogeneous delays and manages to identify the best arm, leading to a sub-linear regret.

8.3. Performance of `Adapt-PatientBandits`

We now compare the performance of the adaptive `Adapt-PatientBandits` to the one of `PatientBandits` with some values of parameter $\overline{\alpha}$.

As before, we consider a two-arm bandit instance with mean parameters $\mu = (0.6, 0.8)$. The delay distribution of the second arm is Pareto with parameter $\alpha_2 = \alpha = 0.3$. The first arm has its delay distribution characterised by the CDF:

$$\tau(m) = \begin{cases} 1 - 0.7/2^{\alpha} & \text{if } \ m = 0, \\ 1 - 0.7/(1+x)^{\alpha} & \text{if } \ m \in \mathbb{N}^*, \end{cases}$$

where we remind that $\alpha = 0.3$, so that Assumption 2 is satisfied with $c = 0.7$, but the delays of the optimal arm are stochastically dominated by those of the sub-optimal arm.

We compare the algorithm `Adapt-PatientBandits` launched with parameters $(\underline{\alpha} = 0.1, c = 0.7, \overline{\mu} = 0.6)$ with the algorithm `PatientBandits` launched with three different parameters parameters $\overline{\alpha} \in \{0.1, 0.3, 0.6\}$. The maximal horizon is here $T = 10000$ and the results are averaged over 100 runs. Figure 6 displays the regret of the algorithm as a function of time, and Figure 7 displays the estimated lower bound $\overline{\alpha}_t$ of α, used by `Adapt-PatientBandits`.

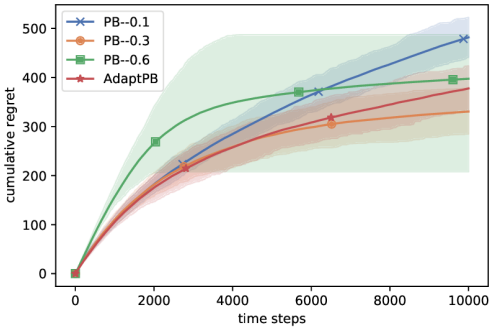

Figure 6. Regret of `Adapt-PatientBandits` and `PatientBandits` in function of the time, for $\mu = (0.6, 0.8)$ and delay distributions described in Section 8.3. The `PatientBandits` algorithm was run for $\overline{\alpha} = (0.1, 0.3, 0.6)$ and the `Adapt-PatientBandits` algorithm with parameters $(\underline{\alpha} = 0.1, c = 0.7, \overline{\mu} = 0.6)$.

The lower bound estimates $\overline{\alpha}_t$ eventually converges towards the right quantity, which is $\alpha = 0.3$. However this takes some time so that the performance of `Adapt-PatientBandits` is slightly worse than the one of `PatientBandits` that knows the oracle parameter $\overline{\alpha} = 0.3$. On the other hand, the performance of `Adapt-PatientBandits` is better than that of `PatientBandits` run non-oracle parameters $\overline{\alpha} \in \{0.1, 0.6\}$, indicating that `Adapt-PatientBandits` is more flexible than `PatientBandits` when Assumption 2 holds.

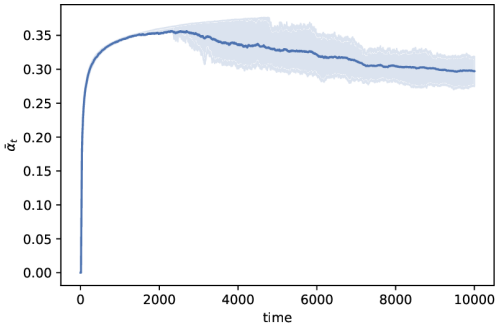

Figure 7. Lower bound estimate $\overline{\alpha}_t$ on α from `Adapt-PatientBandits`, in the same instance in Figure 6.

Conclusion In this paper, we extend the problem of learning with bandit feedback and partially observable delays to arm-dependent delay distributions with possibly unbounded expectations. We close many existing open problems left by Vernade et al. (2017; 2018), either with positive answers (Theorem 2) or negative answers (Theorem 5). The major difficulty faced by the learner in this setting is the identifiability issue due to missing rewards which induce a bias in the estimator of the real payoff. Under the assumption that the tail distribution of the delays is bounded - although it might be very heavy tailed - we designed a very simple UCB-based algorithm, termed `PatientBandits`. We proved that `PatientBandits` performs almost as well as the standard UCB in the classical, non-delayed case, from a problem dependent point of view. We also studied the problem of adaptivity to the delay distributions and concluded that this is not possible (Theorem 5) unless a global bound on the tails hold (Assumption 2). Closing the gap between the problem-independent bound and the lower bound may constitute the object of future studies.

Acknowledgements. The work of A. Carpentier is partially supported by the Deutsche Forschungsgemeinschaft (DFG) Emmy Noether grant MuSyAD (CA 1488/1-1), by the DFG - 314838170, GRK 2297 MathCoRe, by the DFG GRK 2433 DAEDALUS (384950143/GRK2433), by the DFG CRC 1294 'Data Assimilation', Project A03, and by the UFA-DFH through the French-German Doktorandenkolleg CDFA 01-18 and by the UFA-DFH through the French-German Doktorandenkolleg CDFA 01-18 and by the SFI Sachsen-Anhalt for the project RE-BCI. The work of A. Manegueu is supported by the Deutsche Forschungsgemeinschaft (DFG) CRC 1294 'Data Assimilation', Project A03.