What Doubling Tricks Can and Can't Do for Multi-Armed Bandits

Lilian Besson[†] LILIAN.BESSON@CENTRALESUPELEC.FR
CentraleSupélec (campus of Rennes), IETR, SCEE Team,
Avenue de la Boulaie – CS 47601*, F-*35576 *Cesson-Sévigné, France*

Emilie Kaufmann EMILIE.KAUFMANN@UNIV-LILLE1.FR
CNRS & Université de Lille, Inria SequeL team
UMR 9189 *– CRIStAL, F-*59000 *Lille, France*

Abstract

An online reinforcement learning algorithm is *anytime* if it does not need to know in advance the horizon T of the experiment. A well-known technique to obtain an anytime algorithm from any non-anytime algorithm is the "Doubling Trick". In the context of adversarial or stochastic multi-armed bandits, the performance of an algorithm is measured by its regret, and we study two families of sequences of growing horizons (geometric and exponential) to generalize previously known results that certain doubling tricks can be used to conserve certain regret bounds. In a broad setting, we prove that a geometric doubling trick can be used to conserve (minimax) bounds in $R_T = \mathcal{O}(\sqrt{T})$ but *cannot* conserve (distribution-dependent) bounds in $R_T = \mathcal{O}(\log T)$. We give insights as to why exponential doubling tricks may be better, as they conserve bounds in $R_T = \mathcal{O}(\log T)$, and are close to conserving bounds in $R_T = \mathcal{O}(\sqrt{T})$.

Keywords: Multi-Armed Bandits; Anytime Algorithms; Sequential Learning; Doubling Trick.

1. Introduction

Multi-Armed Bandit (MAB) problems are well-studied sequential decision making problems in which an agent repeatedly chooses an action (the "arm" of a one-armed bandit) in order to maximize some total reward (Robbins, 1952; Lai and Robbins, 1985). Initial motivation for their study came from the modeling of clinical trials, as early as 1933 with the seminal work of Thompson (1933). In this example, arms correspond to different treatments with unknown, random effect. Since then, MAB models have been proved useful for many more applications, that range from cognitive radio (Jouini et al., 2009) to online content optimization (*e.g.*, news article recommendation (Li et al., 2010), online advertising (Chapelle and Li, 2011), A/B Testing (Kaufmann et al., 2014; Yang et al., 2017)), or portfolio optimization (Sani et al., 2012).

While the number of patients involved in a clinical study (and thus the number of treatments to select) is often decided in advance, in other contexts the total number of decisions to make (the horizon T) is unknown. It may correspond to the total number of visitors of a website optimizing its displays for a certain period of time, or to the number of attempted communications in a smart radio device. In such cases, it is thus crucial to devise *anytime algorithms*, that is algorithms that do no rely on the knowledge of this horizon T to sequentially select arms. A general way to turn any base algorithm into an anytime algorithm is the use of the so-called Doubling Trick, first proposed by Auer et al. (1995), that consists in repeatedly running the base algorithm with increasing horizons. Motivated by the frequent use of this technique and the absence of a generic study of its effect

© 2018 L. Besson & E. Kaufmann.

on the algorithm's efficiency, this paper investigates in details two families of doubling sequences (geometric and exponential), and shows that the former should be avoided for stochastic problems.

More formally, a MAB model is a set of K arms, each arm k being associated to a (unknown) *reward stream* $(Y_{k,t})_{t\in\mathbb{N}}$. Fix T a finite (possibly unknown) horizon. At each time step $t \in \{1,\dots,T\}$ an agent selects an arm $A(t) \in \{1,\dots,K\}$ and receives as a reward the current value of the associated reward stream, $r(t) := Y_{A(t),t}$. The agent's decision strategy (or *bandit algorithm*) $\mathcal{A}_T := (A(t), t \in \{1,\dots,T\})$ is such that $A(t)$ can only rely on the past observations $A(1), r(1), \dots, A(t-1), r(t-1)$, on external randomness and (possibly) on the knowledge of the horizon T. The objective of the agent is to find an algorithm $\mathcal{A}$ that maximizes the expected cumulated rewards, where the expectation is taken over the possible randomness used by the algorithm and the possible randomness in the generation of the rewards stream. In the oblivious case, in which the reward streams are independent of the algorithm's choice, this is equivalent to minimizing the *regret*, defined as

$$R_T(\mathcal{A}_T) := \max_{k\in\{1,\dots,K\}} \mathbb{E}\left[\sum_{t=1}^{T} \left(Y_{k,t} - Y_{A(t),t}\right)\right]. \tag{1}$$

This quantity, referred to as *pseudo-regret* in Bubeck et al. (2012), quantifies the difference between the expected cumulated reward of the best fixed action, and that of the strategy $\mathcal{A}_T$. For the general adversarial bandit problem (Auer et al., 2002b), in which the rewards streams are arbitrary (picked by an adversary), a *worst-case* lower bound has been given. It says that for every algorithm, there exists (stochastic) reward streams such that the regret is larger than $(1/20)\sqrt{KT}$ (Auer et al., 2002b). Besides, the EXP3 algorithm has been shown to have a regret of order $\sqrt{KT\log(K)}$.

Much smaller regret may be obtained in *stochastic* MAB models, in which the reward stream from each arm k is assumed to be *i.i.d.*, from some (unknown) distribution ν_k, with mean μ_k. In that case, various algorithms have been proposed with *problem-dependent* regret upper bounds of the form $C(\boldsymbol{\nu})\log(T)$, where $C(\boldsymbol{\nu})$ is a constant that only depend on the arms distributions. Different assumptions on the arms distributions lead to different problem-dependent constants. In particular, under some parametric assumptions (*e.g.*, Gaussian distributions, exponential families), *asymptotically optimal* algorithms have been proposed and analyzed (*e.g.*, kl-UCB (Cappé et al., 2013) or Thompson sampling (Agrawal and Goyal, 2012; Kaufmann et al., 2012)), for which the constant $C(\boldsymbol{\nu})$ obtained in the regret upper bound matches exactly that of the lower bound given by Lai and Robbins (1985). Under the non-parametric assumption that the ν_k are bounded in $[0,1]$, the regret of the UCB1 algorithm (Auer et al., 2002a) is of the above form with $C(\boldsymbol{\nu}) = 8 \times \sum_{k:\mu_k>\mu^*}(\mu^* - \mu_k)^{-1}$, where $\mu^* = \max_k \mu_k$ is the mean of the best arm. Like in this last example, all the available constants $C(\boldsymbol{\nu})$ become very large on "hard" instances, in which some arms are very close to the best arm. On such instances, $C(\boldsymbol{\nu})\log(T)$ may be much larger than the worst-case $(1/20)\sqrt{KT}$, and distribution-independent guarantees may actually be preferred.

The MOSS algorithm, proposed by Audibert and Bubeck (2009), is the first stochastic bandit algorithm to enjoy a problem-dependent logarithmic regret and to be optimal in a *minimax* sense, as its regret is proved to be upper bounded by $\sqrt{KT}$, for bandit models with rewards in $[0,1]$. However the corresponding constant $C(\boldsymbol{\nu})$ is proportional to $K/\Delta_{\min}$, where $\Delta_{\min} = \min_k(\mu^* - \mu_k)$ is the minimal gap, which worsen the constant of UCB1. Another drawback of MOSS is that it is *not* anytime. These two shortcoming have been overcame recently in two different works. On the one hand, the MOSS-anytime algorithm (Degenne and Perchet, 2016) is minimax optimal and anytime, but its problem-dependent regret does not improve that of MOSS. On the other hand, the kl-UCB^{++}

algorithm (Ménard and Garivier, 2017) is simultaneously minimax optimal and asymptotically optimal (*i.e.*, it has the best problem-dependent constant $C(\boldsymbol{\nu})$), but it is not anytime. A natural question is thus to know whether a Doubling Trick could overcome this limitation.

This question is the starting point of our comprehensive study of the Doubling Trick: can a single Doubling Trick be used to preserve both problem-dependent (logarithmic) regret and minimax (square-root) regret? We answer this question partially, by showing that two different types of Doubling Trick may actually be needed. In this paper, we investigate how algorithms enjoying regret guarantees of the generic form

$$\forall T \geq 1, \quad R_T(\mathcal{A}_T) \leq c\, T^{\gamma} (\log(T))^{\delta} + o(T^{\gamma} \left(\log(T)\right)^{\delta}\Big) \tag{2}$$

may be turned into an anytime algorithm enjoying *similar* regret guarantees with an appropriate Doubling Trick. This does not come for free, and we exhibit a "price of Doubling Trick", that is a constant factor larger than 1, referred to as a *constant manipulative overhead*.

The rest of the paper is organized as follows. The Doubling Trick is formally defined in Section 2, along with a generic tool for its analysis. In Section 3, we present upper and lower bounds on the regret of algorithms to which a geometric Doubling Trick is applied. Section 4 investigates regret guarantees that can be obtained for a "faster" exponential Doubling Trick. Experimental results are then reported in Section 5. Complementary elements of proofs are deferred to the appendix.

2. Doubling Tricks

The Doubling Trick, denoted by $\mathcal{DT}$, is a general procedure to convert a (possibly non-anytime) algorithm into an anytime algorithm. It is formally stated below as Algorithm 1 and depends on a non-decreasing diverging *doubling sequence* $(T_i)_{i\in\mathbb{N}}$ (*i.e.*, $T_i \to \infty$ for $i \to \infty$). $\mathcal{DT}$ fully restarts the underlying algorithm $\mathcal{A}$ at the beginning of each new sequence (at $t = T_i + 1$), and run this algorithm on a sequence of length $(T_i - T_{i-1})$.

```
Input: Bandit algorithm $\mathcal{A}$, and doubling sequence $(T_i)_{i\in\mathbb{N}}$.
Let $i = 0$, and initialize algorithm $\mathcal{A}^{(0)} = \mathcal{A}_{T_0}$.
for $t = 1, \dots, T-1$ do
    if $t > T_i$ then                // Next horizon $T_{i+1}$ from the sequence
        Let $i = i + 1$.
        Initialize algorithm $\mathcal{A}^{(i)} = \mathcal{A}_{T_i - T_{i-1}}$.        // Full restart
    end
    Play with $\mathcal{A}^{(i)}$: play arm $A'(t) := A^{(i)}(t - T_{i-1})$, observe reward $r(t) = Y_{A'(t),t}$.
end
```

Algorithm 1: The Generic Doubling Trick Algorithm, $\mathcal{A}' = \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$.

Related work. The Doubling Trick is a well known idea in online learning, that can be traced back to Auer et al. (1995). In the literature, the term Doubling Trick usually refers to the geometric sequence $T_i = 2^i$, in which the horizon is actually *doubling*, that was popularized by Cesa-Bianchi and Lugosi (2006) in the context of adversarial bandits. Specific doubling tricks have also been used for stochastic bandits, for example in the work of Auer and Ortner (2010), which uses the doubling sequence $T_i = 2^{2^i}$ to turn the UCB-R algorithm into an anytime algorithm.

Elements of regret analysis. For a sequence $(T_i)_{i\in\mathbb{N}}$, with $T_i \in \mathbb{N}$ for all i, we denote $T_{-1} = 0$, and T_0 is always taken non-zero, $T_0 > 0$ (*i.e.*, $T_0 \in \mathbb{N}^*$). We only consider *non-decreasing* and *diverging* sequences (that is, $\forall i,\ T_{i+1} \geq T_i$, and $T_i \to \infty$ for $i \to \infty$).

Definition 1 (Last Term L_T)
For a non-decreasing diverging sequence $(T_i)_{i\in\mathbb{N}}$ and $T \in \mathbb{N}$, we can define $L_T\big((T_i)_{i\in\mathbb{N}}\big)$ by

$$\forall T \geq 1, \quad L_T\big((T_i)_{i\in\mathbb{N}}\big) := \min\left\{i \in \mathbb{N} : T_i > T\right\}. \tag{3}$$

*It is simply denoted L_T when there is no ambiguity (*e.g.*, if the doubling sequence is chosen).*

$\mathcal{DT}(\mathcal{A})$ reinitializes its underlying algorithm $\mathcal{A}$ at each time T_i, and in generality the total regret is upper bounded by the regret on each sequence $\{T_i, \dots, T_{i+1} - 1\}$. By considering the last partial sequence $\{T_{L_T - 1}, \dots, T - 1\}$, this splitting can be used to get a generic upper bound (UB) by taking into account a larger last sequence (up to $T_{L_T} - 1$). And for stochastic bandit models, the *i.i.d.* hypothesis on the rewards streams makes the splitting on each sequence an equality, so we can also get the lower bound (LB) by excluding the last partial sequence. Lemma 2 is proved in Appendix A.1.

Lemma 2 (Regret Lower and Upper Bounds for $\mathcal{DT}$)
For any bandit model *and algorithm $\mathcal{A}$ and horizon T, one has the generic upper bound*

$$R_T(\mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})) \leq \sum_{i=0}^{L_T} R_{T_i - T_{i-1}}(\mathcal{A}_{T_i - T_{i-1}}). \tag{LB}$$

Under a stochastic bandit model*, one has furthermore the lower bound*

$$R_T(\mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})) \geq \sum_{i=0}^{L_{T-1}} R_{T_i - T_{i-1}}(\mathcal{A}_{T_i - T_{i-1}}). \tag{UB}$$

As expected, the key to obtain regret guarantees for a Doubling Trick algorithm is to carefully choose the doubling sequence $(T_i)_{i\in\mathbb{N}}$. Empirically, one can verify that sequences with slow growth gives terrible results, and for example using an arithmetic progression typically gives a linear regret. Building on this result, we will prove that if $\mathcal{A}$ satisfies a certain regret bound ($R_T = \mathcal{O}(T^\gamma)$, $\mathcal{O}((\log T)^\delta)$, or $\mathcal{O}(T^\gamma(\log T)^\delta)$) then an appropriate anytime version of $\mathcal{A}$ with a certain doubling trick can conserve the regret bound, with an explicit constant multiplicative overhead $\ell > 1$. In this paper, we study in depth two families of sequences: first geometric and then exponential growths.

3. What the Geometric Doubling Trick Can and Can't Do

We define geometric doubling sequences, and prove that they can be used to conserve bounds in $\mathcal{O}\big(T^\gamma(\log T)^\delta\big)$ with $\gamma > 0$ but cannot be used to conserve bounds in $\mathcal{O}\big((\log T)^\delta\big)$.

Definition 3 (Geometric Growth) *For $b \in \mathbb{R}$, $b > 1$ and $T_0 \in \mathbb{N}^*$, the geometric sequence $T_i = \lfloor T_0 b^i \rfloor, \forall i \in \mathbb{N}$ is non-decreasing and diverging, and satisfies furthermore*

$$\forall T < T_0,\ L_T = 0, \quad \textit{and} \quad \forall T \geq T_0,\ L_T = \left\lceil \log_b\left(\frac{T}{T_0}\right)\right\rceil, \tag{4}$$

$$\forall i > 0,\ \ T_0(b-1)b^{i-1} - 1 \leq T_i - T_{i-1} \leq 1 + T_0(b-1)b^{i-1}. \tag{5}$$

Asymptotically for i and $T \to \infty$, $T_i = \mathcal{O}\big(b^i\big)$ and $L_T \sim \log_b(T) = \mathcal{O}(\log T)$.

3.1. Conserving a Regret Upper Bound with Geometric Horizons

A geometric doubling sequence allows to conserve a minimax bound (*i.e.*, $R_T = \mathcal{O}(\sqrt{T})$). It was suggested, for instance, in (Cesa-Bianchi and Lugosi, 2006, Ex.2.9). We generalize this result in the following theorem, proved in Appendix A.2, by extending it from $\sqrt{T}$ bounds to bounds of the form $T^{\gamma}(\log T)^{\delta}$ for *any* $0 < \gamma < 1$ and $\delta \geq 0$. Note that no distinction is done on the case $\delta = 0$ neither in the expression of the constant overhead, nor in the proof.

Theorem 4 *If an algorithm $\mathcal{A}$ satisfies $R_T(\mathcal{A}_T) \leq c\, T^{\gamma}(\log T)^{\delta} + f(T)$, for $0 < \gamma < 1$, $\delta \geq 0$ and for $c > 0$, and an increasing function $f(t) = o\big(t^{\gamma}(\log t)^{\delta}\big)$ (at $t \to \infty$), then the anytime version $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ with the geometric sequence $(T_i)_{i\in\mathbb{N}}$ of parameters $T_0 \in \mathbb{N}^*, b > 1$ (*i.e., $T_i = \lfloor T_0 b^i \rfloor$) *with the condition* $T_0(b-1) > 1$ *if $\delta > 0$, satisfies,*

$$R_T(\mathcal{A}') \leq \ell(\gamma, \delta, T_0, b)\ c\, T^{\gamma}\, (\log T)^{\delta} + g(T), \tag{6}$$

with a increasing function $g(t) = o\big(t^{\gamma}(\log t)^{\delta}\big)$, and a constant overhead $\ell(\gamma, \delta, T_0, b) > 1$,

$$\ell(\gamma, \delta, T_0, b) := \left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^{\delta} \times \frac{b^{\gamma}(b-1)^{\gamma}}{b^{\gamma}-1}. \tag{7}$$

For a fixed γ and δ, minimizing $\ell(\gamma, \delta, T_0, b)$ does not always give a unique solution. On the one hand, if $\gamma \gtrsim 0.384$, there is a unique solution $b^*(\gamma) > 1$ minimizing the $\frac{b^{\gamma}(b-1)^{\gamma}}{b^{\gamma}-1}$ term, solution of $b^{\gamma+1} - 2b + 1 = 0$, but without a closed form if γ is unknown. On the other hand, for any γ, the term depending on δ tends quickly to 1 when T_0 increases.

Practical considerations. Empirically, when γ and δ are fixed and known, there is no need to minimize ℓ jointly. It can be minimized separately by first minimizing $\frac{b^{\gamma}(b-1)^{\gamma}}{b^{\gamma}-1}$, that is by solving $b^{\gamma+1} - 2b + 1 = 0$ numerically (*e.g.*, with Newton's method), and then by taking T_0 large enough so that the other term is close enough to 1.

For the usual case of $\gamma = 1/2$ and $\delta = 0$ (*i.e.*, bounds in $\sqrt{T}$), the optimal choice of b is $\frac{3+\sqrt{5}}{2}$ leading to $\ell \simeq 3.33$, and the usual choice of $b = 2$ gives $\ell \simeq 3.41$ (see Corollary 10 in appendix). Any large enough T_0 gives similar performance, and empirically $T_0 \gg K$ is preferred, as most algorithms explore each arm once in their first steps (*e.g.*, $T_0 = 200$ for $K = 9$ for our experiments).

3.2. A Regret Lower Bound with Geometric Horizons

We observe that the constant overhead in Eq. (7) from the previous Theorem 4 blows up when γ goes to zero, giving the intuition that no geometric doubling trick could be used to preserve a logarithmic bound (*i.e.*, with $\gamma = 0$, $\delta > 0$). This is confirmed by the lower bound given below.

Theorem 5 *If $\mathcal{A}$ satisfies $R_T(\mathcal{A}_T) \geq c\ (\log T)^{\delta}$, for $c > 0$ and $\delta > 0$, and under a certain* stochastic model*, then the anytime version $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ with the geometric sequence $(T_i)_{i\in\mathbb{N}}$ of parameters $T_0 \in \mathbb{N}^*, b > 1$* (i.e., *$T_i = \lfloor T_0 b^i \rfloor$) satisfies this lower bound for a certain constant $c' > 0$, under the same model,*

$$\forall T \geq 1, L_T \geq 2 \implies R_T(\mathcal{A}') \geq c'\ (\log T)^{\delta+1}. \tag{8}$$

This implies that $R_T(\mathcal{A}') = \Omega((\log T)^{\delta+1})$, which proves that a geometric sequence cannot *be used to conserve a logarithmic regret bound.*

Theorem 5 implies that a geometric sequence *cannot*[1] be used to conserve a finite-horizon lower bound like $R_T(\mathcal{A}_T) \geq c \log(T)$. A complementary lower bound, stated as Theorem 11 in Appendix B, shows that if the regret is lower bounded at finite horizon by $R_T(\mathcal{A}_T) \geq c\sqrt{T}$, then a comparable lower bound for the Doubling Trick algorithm $\mathcal{DT}(\mathcal{A})$, possibly with a larger constant.

This special case ($\delta = 1$) is indeed the most interesting, as in the stochastic case the regret of any uniformly efficient algorithm is at least logarithmic (Lai and Robbins (1985)), and efficient algorithm with logarithmic regret have been exhibited. If $R_T(\mathcal{A}_T)/\log T$ is bounded, then using a geometric sequence in the doubling trick algorithm is a bad idea as it guarantees a blow up in the regret, that is $R_T(\mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})) = \Omega((\log T)^2)$. This result is the reason we need to consider successive horizons growing faster than a geometric sequence (*i.e.*, such that $\log(T_i) \gg i$), like the exponential sequence, which is studied in Section 4.

3.3. Proof of Theorem 5

Let $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ and consider a fixed *stochastic* bandit problem. The lower bound (LB) from Lemma 2 gives

$$R_T(\mathcal{A}') \geq \sum_{i=0}^{L_T - 1} R_{T_i - T_{i-1}}(\mathcal{A}_{T = T_i - T_{i-1}})$$

We bound $T_i - T_{i-1} \geq T_0(b-1)b^{i-1} - 1$ for any $i > 0$, thanks to Definition 3, and we can use the hypothesis on $\mathcal{A}$ for each regret term.

$$\begin{aligned} &\geq \sum_{i=0}^{L_T-1} c(\log(T_i - T_{i-1}))^\delta \geq c \sum_{i=1}^{L_T-1} \left(\log\left(T_0(b-1)b^{i-1} - 1\right)\right)^\delta \\ &= c \sum_{i=0}^{L_T-2} \left(\log\left(T_0(b-1)b^{i} - 1\right)\right)^\delta \qquad (\text{with } i := i-1) \end{aligned}$$

Let $x_i := T_0(b-1)b^i > 0$. If we have $T_0(b-1) > 1$ (see below (♣) in Page 7 for a discussion on the other case), then Lemma 15 (Eq. (26)) gives $\log(x_i - 1) \geq \frac{\log(T_0(b-1)-1)}{\log(T_0(b-1))} \log(x_i)$ as $x_i > 1$. For lower bounds, there is no need to handle the constants tightly, and we have $x_i \geq b^i$ by hypothesis, so let call this constant $c' = c\left(\frac{\log(T_0(b-1)-1)}{\log(T_0(b-1))}\right)^\delta > 0$, and thus it simplifies to

$$\geq c' \sum_{i=0}^{L_T-2} \left(\log(b^i)\right)^\delta$$

A sum-integral minoration for the increasing function $t \mapsto t^\delta$ (as $\delta > 0$) gives $\sum_{i=0}^{L_T-2} \left(\log(b^i)\right)^\delta = (\log b)^\delta \sum_{i=1}^{L_T-2} i^\delta \geq (\log b)^\delta \int_0^{L_T-2} t^\delta \,\mathrm{d}t = \frac{(\log b)^\delta}{\delta+1} (L_T - 2)^{\delta+1}$ (if $L_T \geq 2$), and so

$$R_T(\mathcal{A}') \geq c' \frac{(\log b)^\delta}{\delta+1} (L_T - 2)^{\delta+1}$$

[1] Note that we only pay one extra $\log(T)$ factor, which is bad if δ is small, but not so bad for large δ.

For the geometric sequence, we know that $L_T \geq \log_b\left(\frac{T}{T_0}\right) \geq \log_b(T)$, and $\log_b(T) - 2 \sim \log_b(T)$ at $T \to \infty$ so there exists a constant $0 < c'' < 1$ such that $L_T - 2 \geq c'' \log_b(T)$ for T large enough ($\geq \frac{2}{b-1}$), and such that $L_T \geq 2$. And thus we just proved that there is a constant $c''' > 0$ such that

$$R_T(\mathcal{A}') \geq c'''(\log T)^{\delta+1} =: g(T).$$

So this proves that for T large enough, $R_T(\mathcal{A}') \geq g(T)$ with $g(T) = \mathcal{O}\big((\log T)^{\delta+1}\big)$, and so $R_T(\mathcal{A}') = \Omega((\log T)^{\delta+1})$, which also implies that $R_T(\mathcal{A}')$ *cannot* be a $\mathcal{O}\big((\log T)^{\delta}\big)$.

($\clubsuit$) If we do not have the hypothesis $T_0(b-1) > 1$, the same proof could be done, by observing that from $i \geq i_0$ large enough, we have $x_i \geq b^{i-i_0}$ (as soon as $b^{i_0} \geq \frac{1}{T_0(b-1)} > 0$, *i.e.*, $i_0 \geq \lceil -\log_b(T_0(b-1)) \rceil \geq 1$), and so the same arguments can be used, to obtain a sum from $i = i_0 + 1$ instead of from $i = 1$. For a fixed i_0, we also have $L_T - 2 - i_0 \geq c'' \log(T)$ for a (small enough) constant c'', and thus we obtain the same result. ■

4. What Can the Exponential Doubling Trick Do?

We define exponential doubling sequences, and prove that they can be used to conserve bounds in $\mathcal{O}\big((\log T)^{\delta}\big)$, unlike the previously studied geometric sequences. Furthermore, we provide elements showing that they may also conserve bounds in $\mathcal{O}(T^{\gamma})$ or $\mathcal{O}\big(T^{\gamma}(\log T)^{\delta}\big)$.

Definition 6 (Exponential Growth) *For $a, b \in \mathbb{R}$, $a, b > 1$, $T_0 \in \mathbb{N}^*$, and $\tau := \frac{T_0}{a} \in \mathbb{R}$, the exponential sequence $T_i := \lfloor \tau a^{b^i} \rfloor, \forall i \in \mathbb{N}$ is non-decreasing and diverging, and satisfies furthermore*

$$\forall T < T_0,\ L_T = 0, \ \text{ and } \ \forall T \geq T_0,\ L_T = \left\lceil \log_b\left(\log_a\left(\frac{T}{\tau}\right)\right)\right\rceil. \tag{9}$$

Asymptotically for i and $T \to \infty$, $T_i = \mathcal{O}\left(a^{b^i}\right)$ and $L_T \sim \log_b(\log_a(\frac{T}{\tau})) = \mathcal{O}(\log\log T)$.

4.1. Conserving a Regret Upper Bound with Exponential Horizons

An exponential doubling sequence allows to conserve a problem-dependent bound on regret (*i.e.*, $R_T = \mathcal{O}(\log T)$). This was already used in particular cases by Auer and Ortner (2010) and more recently by Liau et al. (2018). We generalize this result in the following theorem.

Theorem 7 *If an algorithm $\mathcal{A}$ satisfies $R_T(\mathcal{A}_T) \leq c\, T^{\gamma}(\log T)^{\delta} + f(T)$, for $0 \leq \gamma < 1$, $\delta \geq 0$, and for $c > 0$, and an increasing function $f(t) = o\big(t^{\gamma}(\log t)^{\delta}\big)$ (at $t \to \infty$), then the anytime version $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ with the exponential sequence $(T_i)_{i\in\mathbb{N}}$ of parameters $T_0 \in \mathbb{N}^*, a, b > 1$ (*i.e.,* $T_i = \lfloor \frac{T_0}{a} a^{b^i} \rfloor$), satisfies the following inequality,*

$$R_T(\mathcal{A}') \leq \ell(\gamma, \delta, T_0, a, b)\ c \left(T^b\right)^{\gamma} (\log T)^{\delta} + g(T). \tag{10}$$

with an increasing function $g(t) = o\big((t^b)^{\gamma}(\log t)^{\delta}\big)$, and a constant overhead $\ell(\gamma, \delta, T_0, a, b) > 0$,

$$\ell(\gamma, \delta, T_0, a, b) := \begin{cases} \left(\frac{a}{T_0}\right)^{(b-1)\gamma} \frac{b^{2\delta}}{b^{\delta}-1} > 0 & \textit{if } \delta > 0 \\ 1 + \frac{1}{(\log(a))(\log(b^{\gamma}))} > 1 & \textit{if } \delta = 0 \end{cases} \tag{11}$$

This result first shows that an exponential doubling trick can preserve a logarithmic regret bound ($O(\log(T))$, which corresponds to $\gamma = 0$ and $\delta = 1$), with a multiplicative constant overhead $\ell \geq 4$. It can further be applied to bounds of the generic form $R_T = \mathcal{O}\big(T^\gamma (\log T)^\delta\big)$, but with a *significant overhead* as T^γ becomes $T^{b\gamma}$, additionally to the constant multiplicative overhead $\ell > 0$. However, it is important to notice that for $\gamma > 0$, the constant ℓ can be made arbitrarily small (with a large enough first step T_0). This observation is encouraging, and let the authors think that a tighter upper bound could be proved.

Remark 8 *An interesting particular case of Theorem 7 is the following ($\gamma = 0, \delta = 1$ and $f(t) = 0$).*

$$R_T(\mathcal{A}_T) \leq c\ \log(T) \implies R_T(\mathcal{DT}(\mathcal{A}, (\lfloor T_0^{b^i} \rfloor)_{i\in\mathbb{N}})) \leq \frac{b^2}{b-1}\ c\ \log(T). \tag{12}$$

In this upper bound, the optimal choice of b is $b = 2$, which yields a constant multiplicative overhead of $\ell(\gamma = 0, b) = 4$. It can be observed that this overhead is twice smaller as the overhead of 8.0625 obtained by (Auer and Ortner, 2010, Sec.4)[2].

4.2. A Regret Lower Bound with Exponential Horizons

Assuming the upper bound of Theorem 7 obtained for $\gamma > 0$ are tight would lead to think that exponential doubling tricks cannot preserve minimax regret bounds of the form $O(\sqrt{T})$). If true, such a conjecture would need to be supported by a lower bound (a counterpart of Theorem 5). Theorem 9 provides such a lower bound, but as discussed below, combined with Theorem 7, this result rather advocates the use of exponential doubling tricks. Theorem 9 is proved in Appendix A.3.

Theorem 9 *If $\mathcal{A}$ satisfies $R_T(\mathcal{A}_T) \geq c\ T^\gamma$, for $c > 0$ and $0 < \gamma \leq 1$, and under a certain* stochastic model, *then the anytime version $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ with the exponential sequence $(T_i)_{i\in\mathbb{N}}$ of parameters $T_0 \in \mathbb{N}^*$, $a > 1$, $b > 1$* (i.e., $T_i = \lfloor \frac{T_0}{a} a^{b^i} \rfloor$)*, satisfies this lower bound for a certain constant $c' > 0$, under the same model,*

$$\forall T \geq 1, \quad R_T(\mathcal{A}') \geq c' \left(T^{\frac{1}{b}}\right)^\gamma. \tag{13}$$

We already saw that any exponential doubling trick can conserve logarithmic problem-dependent regret bounds. If we could take $b \to 1$ in the two previous Theorems 7 and 9, both results would match and prove that there exists an exponential doubling trick that can also be used to conserve minimax regret bounds. This argument is not so easy to formulate, as b cannot depend on T, but it supports our belief that exponential doubling tricks are good candidates for (asymptotically) preserving both problem-dependent and minimax regret bounds.

4.3. Proof of Theorem 7

Let $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$, and consider a fixed bandit problem. We first consider the harder case of $\delta > 0$, see below in Page 10 in (♠) for the other case. The lower bound (LB) from Lemma 2 gives

$$R_T(\mathcal{A}') \leq \sum_{i=0}^{L_T} R_{T_i - T_{i-1}}(\mathcal{A}_{T=T_i-T_{i-1}})$$

[2] In Auer and Ortner (2010), the authors obtained an overhead of $258/32 = 8.0625 \geq 8$, as the ratio between the constants for the $\log(T)$ terms, respectively 258 in Th.4.1 and 32 in Th.3.1.

We bound naively[3] $T_i - T_{i-1} \leq T_i \leq \frac{T_0}{a} a^{b^i}$, and we can use the hypothesis on $\mathcal{A}$ for each regret term, as f and $t \mapsto ct^\gamma (\log t)^\delta$ are non-decreasing for $t \geq 1$ (by hypothesis for f and by Lemma 16).

$$\leq \sum_{i=0}^{L_T} f(T_i) + c \sum_{i=0}^{L_T} (T_i)^\gamma \left(\log\left(T_i\right)\right)^\delta \leq g_1(T) + c \sum_{i=0}^{L_T} \left(\frac{T_0}{a} a^{b^i}\right)^\gamma \left(\log\left(\frac{T_0}{a} a^{b^i}\right)\right)^\delta$$

The first part is denoted $g_1(T) := \sum_{i=0}^{L_T} f(T_i)$ and is dealt with Lemma 17: the sum of $f(T_i)$ is a $o\Big(\sum_{i=0}^{L_T} T_i^\gamma (\log(T_i))^\delta\Big)$, as $f(t) = o\big(t^\gamma (\log t)^\delta\big)$ by hypothesis, and this sum of $T_i^\gamma (\log(T_i))^\delta$ is proved below to be bounded by $T^{b\gamma}(\log(T))^\delta$. So $g_1(T) = o\big(T^{b\gamma} (\log T)^\delta\big)$. The second part is $c\left(\frac{T_0}{a}\right)^\gamma \sum_{i=0}^{L_T} \left(a^{b^i}\right)^\gamma \left(\log\left(\frac{T_0}{a} a^{b^i}\right)\right)^\delta$. Define $\log^+(x) := \max(\log(x), 0) \geq 0$, so whether $\frac{T_0}{a} \leq 1$ or > 1, we always have $\log\left(\frac{T_0}{a} a^{b^i}\right) \leq \log^+\left(\frac{T_0}{a}\right) + \log\left(a^{b^i}\right)$. Then we can use Lemma 14 (Eq. (23)) to distribute the power on δ (as it is < 1). So $\left(\log\left(\frac{T_0}{a} a^{b^i}\right)\right)^\delta \leq \left(\log^+\left(\frac{T_0}{a}\right)\right)^\delta + (\log(a))^\delta \left(b^i\right)^\delta$ with the convention that $0^\delta = 0$ (even if $\delta = 0$), and so this gives

$$\leq g_1(T) + c\left(\frac{T_0}{a}\right)^\gamma \left[\left(\log^+\left(\frac{T_0}{a}\right)\right)^\delta \sum_{i=0}^{L_T} (a^{b^i})^\gamma + (\log(a))^\delta \sum_{i=0}^{L_T} (a^{b^i})^\gamma \left(b^i\right)^\delta\right]$$

If $\gamma = 0$ then the first sum is just $L_T + 1 = \mathcal{O}(\log(\log(T)))$ which can be included in $g_1(T) = o\big((\log T)^\delta\big)$ (still increasing), and so only the second sum has to be bounded, and a geometric sum gives $\sum_{i=0}^{L_T} \left(b^i\right)^\delta \leq \frac{b^\delta}{b^\delta - 1} (b^{L_T})^\delta$. But if $\gamma > 0$, we can naively bound the first sum by $\sum_{i=0}^{L_T} (a^{b^i})^\gamma \leq (L_T + 1)(a^{b^{L_T}})^\gamma$ Observe that $a^{b^{L_T}} = (a^{b^{L_T - 1}})^b \leq (a\frac{T}{T_0})^b$. So $a^{b^{L_T}} = \mathcal{O}\big(T^b\big)$ and $L_T + 1 = \mathcal{O}(\log(\log(T)))$, thus the first sum is a $\mathcal{O}\big(T^{b\gamma} \log(\log(T))\big) = o\big(T^{b\gamma} (\log T)^\delta\big)$ (as $\delta > 0$). In both cases, the first sum can be included in $g_2(T)$ which is still a $o\big(T^{b\gamma} (\log T)^\delta\big)$ Another geometric sum bounds the second sum by $\sum_{i=0}^{L_T} (a^{b^i})^\gamma \left(b^i\right)^\delta \leq (a^{b^{L_T}})^\gamma \sum_{i=0}^{L_T} \left(b^i\right)^\delta \leq \frac{b^\delta}{b^\delta - 1} (b^{L_T})^\delta$.

$$\leq g_1(T) + c_1 (a^{b^{L_T}})^\gamma (b^{L_T})^\delta$$

We identify a constant multiplicative overhead $c_1 := c\left(\frac{T_0}{a}\right)^\gamma \frac{b^\delta}{b^\delta - 1} (\log a)^\delta > 0$. The only term left which depends on L_T is $(a^{b^{L_T}})^\gamma (b^{L_T})^\delta$, and it can be bounded by using $b^{L_T} = b b^{L_T - 1} \leq b \log_a(a \frac{T}{T_0}) = b + b\log_a^+(\frac{T}{T_0}) \leq b + b\log_a(T)$ (as $T \geq 1$), and again with $a^{b^{L_T}} \leq (a\frac{T}{T_0})^b$. The constant part of b^{L_T} also gives a $\mathcal{O}\big(T^{b\gamma}\big)$ term, that can be included in $g(T) := g_2(T) + (a\frac{T}{T_0})^{b\gamma}$ which is still a $o\big(T^{b\gamma} (\log T)^\delta\big)$, and is still increasing as sum of increasing functions. So we can focus on the last term, and we obtain

$$\leq g(T) + c_1 \left(\frac{b}{\log(a)}\right)^\delta \left[\left(\frac{a}{T_0}\right)^b T^b\right]^\gamma (\log T)^\delta$$

$$\implies R_T(\mathcal{A}') \leq g(T) + \ell(\gamma, \delta, T_0, a, b)\; cT^{b\gamma} \left(\log T\right)^\delta \text{ with an increasing } g(t) = o\Big(t^{b\gamma} \left(\log t\right)^\delta\Big).$$

[3] Here, using the more subtle bound $T_i - T_i \leq \frac{T_0}{a} a^{b^{i-1}} (\alpha^{b^{i-1}}) + 1$, with $\alpha = a^{b-1}$, from Definition 6, does not seem to help as it becomes too complicated to handle clearly in the log terms.

So the constant multiplicative overhead ℓ depends on γ and δ as well as on T_0, a and b and is

$$\ell(\gamma,\delta,T_0,a,b) := \left(\frac{a}{T_0}\right)^{(b-1)\gamma} \times \frac{b^{2\delta}}{b^{\delta}-1} > 0, \quad \text{if } \delta > 0. \tag{14}$$

If $T_0 = a$, the overhead $\ell(\gamma,\delta,T_0,a,b)$ is minimal at $b^*(\delta) = 2^{1/\delta} > 1$ and for a minimal value of $\min_{b>1} \ell(\gamma,\delta,T_0,a,b) = 4$ (**for any** δ and γ). Finally, the a/T_0 part tends to 0 if $T_0 \to \infty$ so the overhead can be made as small as we want, simply by choosing a T_0 large enough (but constant *w.r.t.* T).

(♠) Now for the other case of $\delta = 0$, we can start similarly, but instead of bounding naively $\sum_{i=0}^{L_T} (a^{b^i})^\gamma$ by $(L_T+1)(a^{b^{L_T}})^\gamma$ we use Lemma 13 to get a more subtle bound: $\sum_{i=0}^{L_T} (a^{b^i})^\gamma \leq a^\gamma + (1 + \frac{1}{(\log(a))(\log(b^\gamma))})(a^{b^{L_T}})^\gamma$. The constant term gets included in $g(T)$, and for the non-constant part, $(a^{b^{L_T}})^\gamma$ is handled similarly. Finally we obtain the overhead

$$\ell(\gamma,0,T_0,a,b) := 1 + \frac{1}{(\log(a))(\log(b^\gamma))} > 1. \tag{15}$$

■

5. Numerical Experiments

We illustrate here the practical cost of using Doubling Trick, for two interesting non-anytime algorithms that have recently been proposed in the literature: *Approximated Finite-Horizon Gittins indexes*, that we refer to as AFHG, by Lattimore (2016) (for Gaussian bandits with known variance) and kl-UCB^{++} by Ménard and Garivier (2017) (for Bernoulli bandits).

We first provide some details on these two algorithms, and then illustrate the behavior of Doubling Tricks applied to these algorithms with different doubling sequences.

5.1. Two Index-Based Algorithms

We denote by $X_k(t) := \sum_{s<t} Y_{A(s),s} \mathbb{1}(A(s) = k)$ the accumulated rewards from arm k, and $N_k(t) := \sum_{s<t} \mathbb{1}(A(s) = k)$ the number of times arm k was sampled. Both algorithms $\mathcal{A}$ assume to know the horizon T. They compute an index $I_k^{\mathcal{A}}(t) \in \mathbb{R}$ for each arm $k \in \{1,\dots,K\}$ at each time step $t \in \{1,\dots,T\}$, and use the indexes to choose the arm with highest index, *i.e.*, $A(t) := \arg\max_{k\in\{1,\dots,K\}} I_k(t)$ (ties are broken uniformly at random).

- The algorithm AFHG can be applied for Gaussian bandits with variance V ($= 1$ for our experiments). Let $m(T,t) = T - t + 1 \geq 1$, and let

$$I_k^{\mathrm{AFHG}}(t) := \frac{X_k(t)}{N_k(t)} + \sqrt{\frac{V}{N_k(t)} \log\left(\frac{m(T,t)}{N_k(t)\log^{1/2}\left(\frac{m(T,t)}{N_k(t)}\right)}\right)}. \tag{16}$$

- The algorithm kl-UCB^{++} can be applied for bounded rewards in $[0,1]$, and in particular for Bernoulli bandits. The binary Kullback-Leibler divergence is $\mathrm{kl}(x,y) := x\log(x/y) + (1-x)\log((1-x)/(1-y))$ (for $0 < x,y < 1$), and let $\log^+(x) := \max(0,\log(x)) \geq 0$. Let the function $g(n,T) := \log^+\left(\frac{T}{Kn}(1 + \left(\log^+(\frac{T}{Kn})\right)^2)\right)$ for $n \leq T$, and finally let

$$I_k^{\text{kl-UCB}^{++}}(t) := \sup_{q\in[0,1]} \left\{ q : \mathrm{kl}\left(\frac{X_k(t)}{N_k(t)}, q\right) \leq \frac{g(N_k(t),T)}{N_k(t)} \right\}. \tag{17}$$

5.2. Experiments

We present some results from numerical experiments on Bernoulli and Gaussian bandits. More results are presented in Appendix E. We present in pages 12 and 13 results for $K = 9$ arms and horizon $T = 45678$ (to ensure that no choice of sequence were lucky and had $T_{L_T - 1} = T$ or too close to it). We ran $n = 1000$ repetitions of the random experiment, either on the same "easy" bandit problem $\boldsymbol{\mu}$ (with evenly spaced means), or on n different random instances $\boldsymbol{\mu}$ sampled uniformly in $[0,1]^K$, and we plot the average regret on n simulations. The black line without markers is the (asymptotic) lower bound in $\sum_{k \neq k^*} (\mathrm{kl}(\mu_k, \mu^*))^{-1} \log T$, from Lai and Robbins (1985). We consider $\mathrm{kl}\text{-}\mathrm{UCB}^{++}$ for Bernoulli bandits (Figures 2, 3) or AFHG for Gaussian bandits (Figures 4, 5),

Each doubling trick algorithm uses the same $T_0 = 200$ as a first guess for the horizon. We include both the non-anytime version that knows the horizon T, and different anytime versions to compare the choice of doubling trick. To compare against an algorithm that does not need the horizon, we also include $\mathrm{kl}\text{-}\mathrm{UCB}$ (Cappé et al., 2013) as a baseline for Bernoulli bandits and for Gaussian bandits (in the Gaussian version, the divergence used is $\mathrm{kl}(x,y) = (x-y)^2/2$, and the algorithm is referred to as UCB). We consider are a geometric doubling sequence with $b = 2$, and two different exponential doubling sequences: the "classical" $b = 2$ and a "slower" one with $b = 1.1$. Both use $a = T_0 = 200$, and the last one is using $a = 2$, $b = 2$. Despite what was proved theoretically in Theorem 7, using $a = T_0$ and a large enough T_0 improves regarding to using $a = 2$ and a leading (T_0/a) factor.

Another version of the Doubling Trick with "no restart", denoted $\mathcal{DT}_{\text{no-restart}}$, is presented in Appendix C, but it is only an heuristic and cannot be applied to any algorithm $\mathcal{A}$. Algorithm 2 can be applied to $\mathrm{kl}\text{-}\mathrm{UCB}^{++}$ or AFHG for instance, as they use T just as a numerical parameter (see Eqs. 16 and 17), but its first limitation is that it cannot be applied to $\mathrm{DMED}+$ (Honda and Takemura, 2010) or $\mathrm{EXP3}^{++}$ (Seldin and Lugosi, 2017), or any algorithms based on arm eliminations, for example. A second limitation is the difficulty to analyze this "no restart" variant, due to the unpredictable effect on regret of giving non-uniform prior information to the underlying algorithm $\mathcal{A}$ on each successive sequence. An interesting future work would be to analyze it, either in general or for a specific algorithm like $\mathrm{kl}\text{-}\mathrm{UCB}^{++}$. Despite its limitations, this heuristic exhibits as expected better empirical performance than $\mathcal{DT}$, as can be observed in Appendix E.

6. Conclusion

We formalized and studied the well-known "Doubling Trick" for generic multi-armed bandit problems, that is used to automatically obtain an anytime algorithm from any non-anytime algorithm. Our results are summarized in Table 1. We show that a geometric doubling can be used to conserve minimax regret bounds (in $\sqrt{T}$), with a constant overhead (typically ≥ 3.33), but cannot be used to conserve problem-dependent bounds (in $\log T$), for which a faster doubling sequence is needed. An exponential doubling sequence can conserve logarithmic regret bounds also with a constant overhead, but it is still an open question to know if minimax bounds can be obtained for this faster growing sequence. Partial results of both a lower and an upper bound, for bounds of the generic form $T^\gamma (\log T)^\delta$, let use believe in a positive answer.

It is still an open problem to know if an anytime algorithm can be both asymptotically optimal for the problem-dependent regret (*i.e.*, with the exact constant) and optimal in a minimax regret (*i.e.*, have a $\sqrt{KT}$ regret), but we believe that using a doubling trick on non-anytime algorithms like $\mathrm{kl}\text{-}\mathrm{UCB}^{++}$ cannot be the solution. We showed that it cannot work with a geometric doubling sequence, and conjecture that exponential doubling trick would never bring the right constant either.

Bound \ Doubling	Geometric, $T_i = \lfloor T_0 b^i \rfloor$	Exponential, $T_i = \lfloor \frac{T_0}{a} a^{b^i} \rfloor$
$(\log T)^\delta$	× *Known to fail* $R_T(\mathcal{DT}) \geq c'(\log T)^{1+\delta}$ if $R_T(\mathcal{A}_T) \geq c(\log T)^\delta$. (Theorem 5)	✓ *Known to work*, with overhead $\ell(\delta, b) = \frac{b^{2\delta}}{b^\delta - 1} > 1$. (Theorem 7)
T^γ	✓ *Known to work*, with overhead $\ell(\gamma, b) = \frac{b^\gamma(b-1)^\gamma}{b^\gamma - 1} > 1$. (Theorem 4)	? *Partial*, best known bound is $c_0'(T^{\frac{1}{b}})^\gamma \leq R_T(\mathcal{DT} \leq \ell c(T^b)^\gamma$ with overhead $\ell > 1$, if $c_0 T^\gamma \leq R_T(\mathcal{A}_T) \leq c T^\gamma$. (Theorems 7, 9)
$T^\gamma\ (\log T)^\delta$ for both $\gamma > 0, \delta > 0$	✓ *Known to work*, with overhead $\ell(\gamma, \delta, T_0, b) =$ $\left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta \frac{b^\gamma(b-1)^\gamma}{b^\gamma - 1} > 1$. (Theorem 4)	? *Partial*, best known bound is $R_T(\mathcal{DT}) \leq \ell c(T^b)^\gamma$ if $R_T(\mathcal{A}_T) \leq c T^\gamma$, with overhead $\ell \to 0$ for $T_0 \to \infty$. (Theorem 7)

Figure 1: Summary of *known positive* ✓ *and negative* × *and partial results ?*.

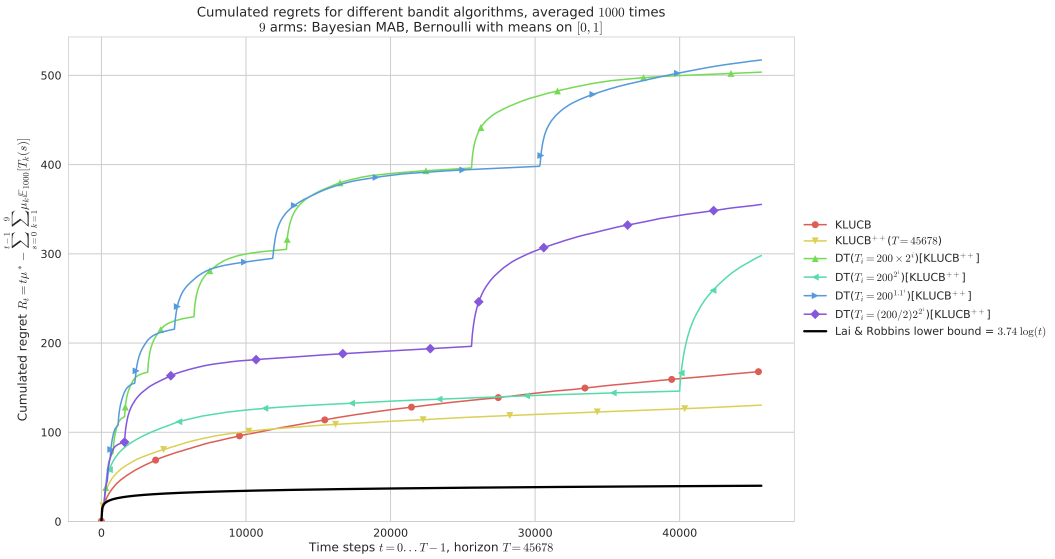

Figure 2: Regret for $\mathcal{DT}$, for $K = 9$ Bernoulli arms, horizon $T = 45678$, $n = 1000$ repetitions and $\boldsymbol{\mu}$ taken uniformly in $[0,1]^K$. Geometric doubling ($b = 2$) and slow exponential doubling ($b = 1.1$) are too slow, and short first sequences make the regret blow up in the beginning of the experiment. At $t = 40000$ we see clearly the effect of a new sequence for the best doubling trick ($T_i = 200 \times 2^i$). As expected, kl-UCB^{++} outperforms kl-UCB, and if the doubling sequence is growing fast enough then $\mathcal{DT}$(kl-UCB^{++}) can perform as well as kl-UCB^{++} (see for $t < 40000$).

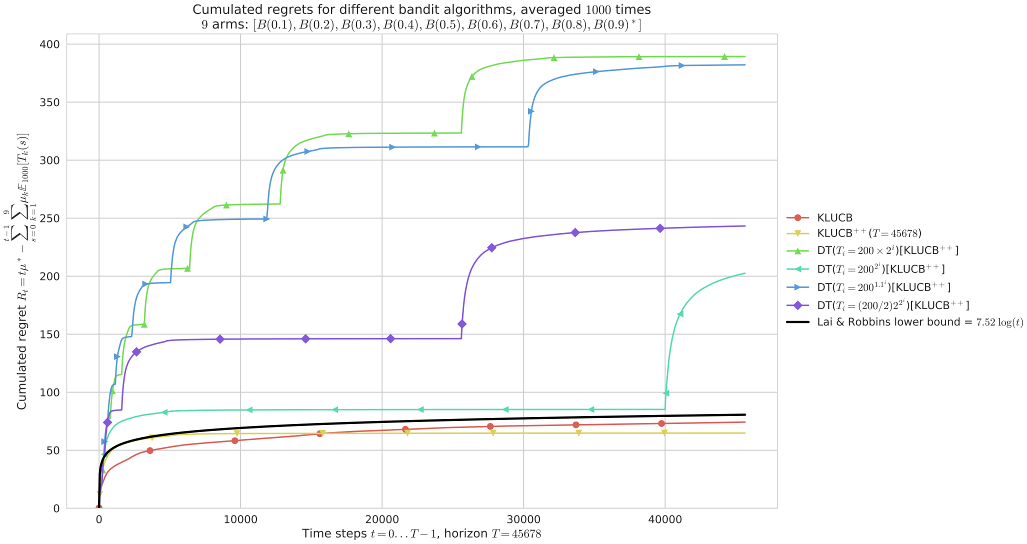

Figure 3: Similarly but for $\boldsymbol{\mu}$ evenly spaced in $[0,1]^K$ ($\{0.1, \ldots, 0.9\}$). Both kl-UCB and kl-UCB^{++} are very efficient on "easy" problems like this one, and we can check visually that they match the lower bound from Lai and Robbins (1985). As before we check that slow doubling are too slow to give reasonable performance.

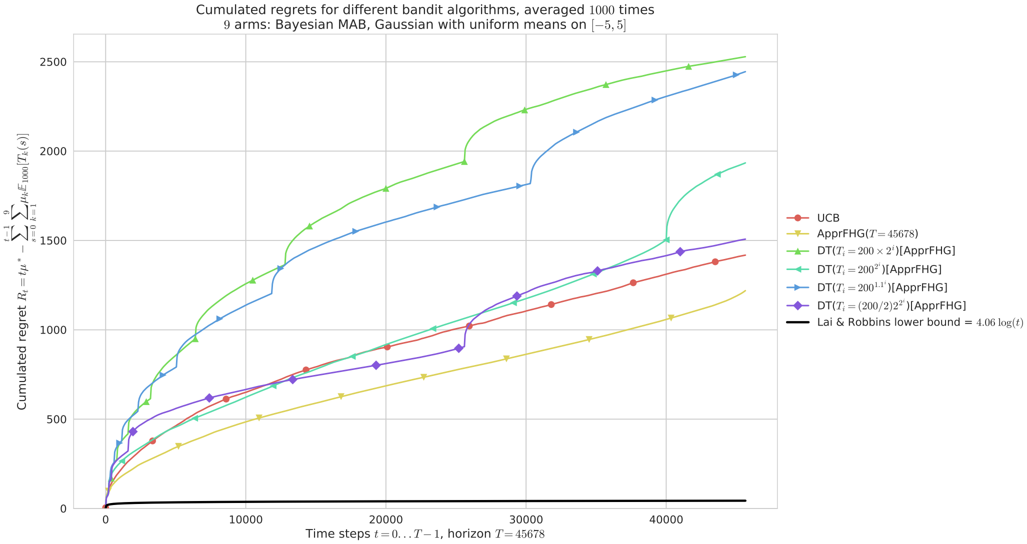

Figure 4: Regret for $K = 9$ Gaussian arms $\mathcal{N}(\mu, 1)$, horizon $T = 45678$, $n = 1000$ repetitions and $\boldsymbol{\mu}$ taken uniformly in $[-5, 5]^K$ and variance $V = 1$. On "hard" problems like this one, both UCB and AFHG perform similarly and poorly *w.r.t.* to the lower bound from Lai and Robbins (1985). As before we check that geometric doubling ($b = 2$) and slow exponential doubling ($b = 1.1$) are too slow, but a fast enough doubling sequence does give reasonable performance for the anytime AFHG obtained by Doubling Trick.

Acknowledgments

This work is supported by the French National Research Agency (ANR), under the project BADASS (grant coded: `N ANR-16-CE40-0002`), by the French Ministry of Higher Education and Research (MENESR) and ENS Paris-Saclay. The authors wish to thank the anonymous reviewers for their valuable comments that helped to improve the paper.

References

S. Agrawal and N. Goyal. Analysis of Thompson sampling for the Multi-Armed Bandit problem. In *Conference On Learning Theory*. PMLR, 2012.

J-Y. Audibert and S. Bubeck. Minimax policies for adversarial and stochastic bandits. In *Conference on Learning Theory*, pages 217–226. PMLR, 2009.

P. Auer and R. Ortner. UCB Revisited: Improved Regret Bounds For The Stochastic Multi-Armed Bandit Problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65, 2010.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. Schapire. Gambling in a Rigged Casino: The Adversarial Multi-Armed Bandit Problem. In *Annual Symposium on Foundations of Computer Science*, pages 322–331. IEEE, 1995.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time Analysis of the Multi-armed Bandit Problem. *Machine Learning*, 47(2):235–256, 2002a.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. Schapire. The Nonstochastic Multiarmed Bandit Problem. *SIAM journal on computing*, 32(1):48–77, 2002b.

S. Bubeck, N. Cesa-Bianchi, et al. Regret Analysis of Stochastic and Non-Stochastic Multi-Armed Bandit Problems. *Foundations and Trends® in Machine Learning*, 5(1), 2012.

O. Cappé, A. Garivier, O-A. Maillard, R. Munos, and G. Stoltz. Kullback-Leibler upper confidence bounds for optimal sequential allocation. *Annals of Statistics*, 41(3):1516–1541, 2013.

N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

O. Chapelle and L. Li. An Empirical Evaluation of Thompson Sampling. In *Advances in Neural Information Processing Systems*, pages 2249–2257. Curran Associates, Inc., 2011.

R. Degenne and V. Perchet. Anytime Optimal Algorithms In Stochastic Multi Armed Bandits. In *International Conference on Machine Learning*, pages 1587–1595, 2016.

A. Garivier, E. Kaufmann, and T. Lattimore. On Explore-Then-Commit Strategies. volume 29 of *Advances in Neural Information Processing Systems (NIPS)*, Barcelona, Spain, 2016.

J. Honda and A. Takemura. An Asymptotically Optimal Bandit Algorithm for Bounded Support Models. In *Conference on Learning Theory*, pages 67–79. PMLR, 2010.

W. Jouini, D. Ernst, C. Moy, and J. Palicot. Multi-Armed Bandit Based Policies for Cognitive Radio's Decision Making Issues. In *International Conference Signals, Circuits and Systems*. IEEE, 2009.

E. Kaufmann, N. Korda, and R. Munos. *Thompson Sampling: an Asymptotically Optimal Finite-Time Analysis*, pages 199–213. PMLR, 2012.

E. Kaufmann, O. Cappé, and A. Garivier. On the Complexity of A/B Testing. In *Conference on Learning Theory*, pages 461–481. PMLR, 2014.

T. L. Lai and H. Robbins. Asymptotically Efficient Adaptive Allocation Rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

T. Lattimore. Regret Analysis Of The Finite Horizon Gittins Index Strategy For Multi Armed Bandits. In *Conference on Learning Theory*, pages 1214–1245. PMLR, 2016.

L. Li, W. Chu, J. Langford, and R. E. Schapire. A Contextual-Bandit Approach to Personalized News Article Recommendation. In *International Conference on World Wide Web*, pages 661–670. ACM, 2010.

D. Liau, E. Price, Z. Song, and G. Yang. Stochastic Multi-Armed Bandits in Constant Space. In *International Conference on Artificial Intelligence and Statistics*, 2018.

P. Ménard and A. Garivier. A Minimax and Asymptotically Optimal Algorithm for Stochastic Bandits. In *Algorithmic Learning Theory*, volume 76, pages 223–237. PMLR, 2017.

H. Robbins. Some Aspects of the Sequential Design of Experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

A. Sani, A. Lazaric, and R. Munos. Risk-Aversion In Multi-Armed Bandits. In *Advances in Neural Information Processing Systems*, pages 3275–3283, 2012.

Y. Seldin and G. Lugosi. An Improved Parametrization and Analysis of the EXP3++ Algorithm for Stochastic and Adversarial Bandits. In *Conference on Learning Theory*, volume 65, pages 1–17. PMLR, 2017.

W. R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 25, 1933.

F. Yang, A. Ramdas, K. Jamieson, and M. Wainwright. A framework for Multi-A(rmed)/B(andit) Testing with Online FDR Control. In *Advances in Neural Information Processing Systems*, pages 5957–5966. Curran Associates, Inc., 2017.

Note: the simulation code used for the experiments is using Python 3. It is open-sourced at `https://GitHub.com/SMPyBandits/SMPyBandits` and fully documented at `https://SMPyBandits.GitHub.io`.

Appendix A. Omitted Proofs

We include here the proofs omitted in the main document.

A.1. Proof of Lemma 2, "Regret Lower and Upper Bounds for $\mathcal{DT}$"

Let $\mathcal{A}'$ denote $\mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$. For every $k \in \{1, \dots, K\}$,

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T}(X_{k,t} - X_{A(t),t})\right] &= \sum_{i=0}^{L_T-1} \mathbb{E}\left[\sum_{t=T_{i-1}}^{T_i} (X_{k,t} - X_{A(t),t})\right] + \mathbb{E}\left[\sum_{t=T_{L_T-1}}^{T} (X_{k,t} - X_{A(t),t})\right] \\
&\le \sum_{i=0}^{L_T-1} \max_{k\in\{1,\dots,K\}} \mathbb{E}\left[\sum_{t=T_{i-1}}^{T_i} (X_{k,t} - X_{A(t),t})\right] + \max_{k\in\{1,\dots,K\}} \mathbb{E}\left[\sum_{t=T_{L_T-1}}^{T} (X_{k,t} - X_{A(t),t})\right] \\
&\le \sum_{i=0}^{L_T-1} R_{T_i-T_{i-1}}\left(\mathcal{A}'\right) + R_{T-T_{L_T-1}}\left(\mathcal{A}'\right).
\end{aligned}$$

Thus, by definition of the regret

$$\begin{aligned}
R_T(\mathcal{A}') &\le \sum_{i=0}^{L_T-1} R_{T_i-T_{i-1}}(\mathcal{A}') + R_{T-T_{L_T-1}}(\mathcal{A}') \\
&= \sum_{i=0}^{L_T-1} R_{T_i-T_{i-1}}\left(\mathcal{A}_{T_i-T_{i-1}}\right) + R_{\underbrace{T - T_{L_T-1}}_{\le T_{L_T}-T_{L_T-1}}}\left(\mathcal{A}_{T_{L_T}-T_{L_T-1}}\right) \\
&\le \sum_{i=0}^{L_T} R_{T_i-T_{i-1}}\left(\mathcal{A}_{T_i-T_{i-1}}\right).
\end{aligned}$$

In the stochastic case, it is well known that the regret can be rewritten in the following way, introducing μ_k the mean of arm k and μ^* the mean of the best arm:

$$\begin{aligned}
R_T(\mathcal{A}') &= \mathbb{E}\left[\sum_{t=1}^{T}(\mu^* - \mu_{A(t)})\right] \\
&= \sum_{i=0}^{L_{T-1}} \mathbb{E}\left[\sum_{t=T_{i-1}}^{T_i} (\mu^* - \mu_{A(t)})\right] + \mathbb{E}\left[\sum_{t=T_{L_T-1}}^{T} (\mu^* - \mu_{A(t)})\right] \\
&= \sum_{i=0}^{L_T-1} R_{T_i-T_{i-1}}\left(\mathcal{A}_{T_i-T_{i-1}})\right) + \underbrace{R_{T-T_{L_T-1}}(\mathcal{A}')}_{\ge 0}.
\end{aligned}$$

and the lower bound follows. ■

A.2. Proof of Theorem 4, "Conserving a Regret Upper Bound with Geometric Horizons"

It is interesting to note that the proof is valid for both the easiest case when $\delta = 0$ (as it was known in Cesa-Bianchi and Lugosi (2006) for $\gamma = 1/2$) and the generic case when $\delta \geq 0$, with no distinction.

As far as the authors know, this result in its generality with $\delta \geq 0$ is new.

Proof Let $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$, and consider a fixed bandit problem. The upper bound (UB) from Lemma 2 gives

$$R_T(\mathcal{A}') \leq \sum_{i=0}^{L_T} R_{T_i - T_{i-1}}(A_{T=T_i-T_{i-1}})$$

We can use the hypothesis on $\mathcal{A}$ for each regret term, as f and $t \mapsto ct^\gamma(\log t)^\delta$ are non-decreasing for $t \geq 1$ (by hypothesis for f and by Lemma 16).

$$\leq \sum_{i=0}^{L_T} f(T_i - T_{i-1}) + cT_0^\gamma(\log T_0)^\delta + c\sum_{i=1}^{L_T} (T_i - T_{i-1})^\gamma \left(\log\left(T_i - T_{i-1}\right)\right)^\delta$$

The first part is denoted $g_1(T) := \sum_{i=0}^{L_T} f(T_i - T_{i-1}) + cT_0^\gamma(\log T_0)^\delta$, it is an increasing function as a sum of increasing functions, and it is dealt with by using Lemma 17: the sum of $f(T_i - T_{i-1})$ is a $o\Big(\sum_{i=0}^{L_T}(T_i - T_{i-1})^\gamma(\log(T_i - T_{i-1}))^\delta\Big)$, as $f(t) = o\big(t^\gamma(\log t)^\delta\big)$ by hypothesis, and this sum of $(T_i - T_{i-1})^\gamma(\log(T_i - T_{i-1}))^\delta$ is proved below to be bounded by $c'T^\gamma(\log(T))^\delta$ for a certain constant $c' > 0$, which gives $g_1(T) = o\big(T^\gamma(\log T)^\delta\big)$. For the second part, we bound $T_i - T_{i-1} \leq T_0(b-1)b^{i-1} + 1$ thanks to Definition 3. Moreover, as $\gamma < 1$ we can use Lemma 14 (Eq. (23)) to distribute the power on γ, so $(T_0(b-1)b^{i-1}+1)^\gamma \leq (T_0(b-1)b^{i-1})^\gamma + \mathbb{1}(\gamma \neq 0)$ (indeed if $\gamma = 0$ both sides are equal to 1). This gives

$$\leq g_1(T) + c(T_0(b-1))^\gamma \sum_{i=1}^{L_T}(b^{i-1})^\gamma \left(\log(T_i - T_{i-1})\right)^\delta + c\mathbb{1}(\gamma \neq 0)\sum_{i=1}^{L_T}\left(\log(T_i - T_{i-1})\right)^\delta$$

If $\gamma \neq 0$, the last sum is bounded by $\sum_i^{L_T}(\log T_i)^\delta \leq (\log T_0)^\delta(L_T + 1) + (\log b)^\delta \sum_i^{L_T} i^\delta$ which is a $\mathcal{O}\Big(L_T^{\delta+1}\Big) = \mathcal{O}\big((\log T)^{\delta+1}\big) = o\big(T^\gamma(\log T)^\delta\big)$ (as $\gamma > 0$, thanks to a geometric sum), and so it can be included in $g_2(T) = o\big(T^\gamma(\log T)^\delta\big)$. If $\gamma = 0$, there is only the first sum. We bound again $T_i - T_{i-1} \leq T_0(b-1)b^{i-1} + 1$ and use Lemma 15 to bound $\log(T_0(b-1)b^{i-1}+1)$ by $\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\log(T_0(b-1)b^{i-1})$ term (as $T_0(b-1) > 1$ by hypothesis).

$$\leq g_2(T) + c(T_0(b-1))^\gamma \sum_{i=1}^{L_T}(b^{i-1})^\gamma \left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\log(T_0(b-1)b^{i-1})\right)^\delta$$

We split the $\log(T_0(b-1)b^{i-1})$ term in two, and once again, the term with $\log(T_0(b-1))$ gives a $\mathcal{O}\big(b^{L_T-1}\big)$ (by a geometric sum), which gets included in $g_3(T) = o\big(T^\gamma(\log T)^\delta\big)$. We focus on the fastest term, and we can now rewrite the sum from $i = 0$ to $L_T - 1$,

$$\leq g_2(T) + c(T_0(b-1))^\gamma \left(\log(b)\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta \sum_{i=0}^{L_T-1}(b^i)^\gamma i^\delta$$

We naively bound i^δ by $(L_T-1)^\delta$, and use a geometric sum to have

$$\leq g_2(T) + c(T_0(b-1))^\gamma \left(\log(b)\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta (L_T-1)^\delta \frac{b^\gamma}{b^\gamma-1}(b^{L_T-1})^\gamma$$

Finally, observe that $L_T - 1 \leq \log_b(\frac{T}{T_0}) \leq \log_b(T)$, so the $(\log b)^\delta$ term simplifies, and observe that $b^{L_T-1} \leq \frac{T}{T_0}$ so the T_0^γ term also simplifies. Thus we get

$$\leq g_2(T) + c\left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta \frac{b^\gamma(b-1)^\gamma}{b^\gamma-1} T^\gamma(\log T)^\delta.$$

The constant multiplicative overhead ℓ depends on γ and δ as well as on T_0 and b, and is $\ell(\gamma,\delta,T_0,b) := \left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta \frac{b^\gamma(b-1)^\gamma}{b^\gamma-1} > 1$. ■

Minimizing the constant overhead? This constant overhead has two distinct part, $\ell(\gamma,\delta,T_0,b) = \ell_1(\delta,T_0,b)\ \ell_2(\gamma,b)$, with ℓ_1 depending on δ, T_0 and b (equal to 1 if $\delta=0$), and ℓ_2 depending on γ and b.

- Minimizing this constant overhead $\ell_1(\delta,T_0,b) := \left(\frac{\log(T_0(b-1)+1)}{\log(T_0(b-1))}\right)^\delta \geq 1$ is independent of δ (even if it is 0). If we assume b to be fixed, $\ell_1(\delta,T_0,b) \to 1$ when $T_0\to\infty$. Moreover, for any δ and $b>1$, $\ell_1(\delta,T_0,b)$ goes to 1 very quickly when T_0 is large enough. For instance, for $\gamma=\delta=\frac{1}{2}$ and $b=\frac{3+\sqrt{5}}{2}$ (see Corollary 10), then $\ell_1(\delta,T_0,b)\simeq 1.109$ for $T_0=2$, $\simeq 1.01$ for $T_0=10$ and $\simeq 1.0004$ for $T_0=100$.
- To minimize this constant overhead $\ell_2(\gamma,b) := \frac{b^\gamma(b-1)^\gamma}{b^\gamma-1} > 1$, we fix γ and study $h : b\mapsto \ell_2(\gamma,b)$. The function h is of class $\mathcal{C}^1$ on $(1,\infty)$ and $h(b)\to+\infty$ for $b\to 1^+$ and $b\to\infty$, so h has a (possibly non-unique) global minimum and attains it. Moreover $h'(b) = \frac{\gamma b^{\gamma-1}(b-1)^{\gamma-1}}{(b^\gamma-1)^2}\left(b^{\gamma+1}-2b+1\right)$ has the sign of $b^{\gamma+1}-2b+1$, which does not have a constant sign and does not have explicit root(s) for a generic γ. However, it is easy to minimizing $\ell_2(\gamma,b)$ for b numerically when γ is known and fixed (with, *e.g.*, Newton's method).

The result from Theorem 4 of course implies the result from (Cesa-Bianchi and Lugosi, 2006, Ex.2.9), in the special case of $\delta=0$ and $\gamma=\frac{1}{2}$ (for minimax bounds), as stated numerically in the following Corollary 10.

Corollary 10 *If $\gamma=\frac{1}{2}$ and $\delta=0$, the multiplicative overhead $\ell(\frac{1}{2},0,T_0,b)$ does not depend on T_0. It is then minimal for $b^*(\frac{1}{2})=\frac{3+\sqrt{5}}{2}\simeq 2.62$ and its minimum is $\sqrt{\frac{11+5\sqrt{5}}{2}}\simeq 3.33$. Usually $b=2$ is used, which gives an overhead of $\frac{\sqrt{2}}{\sqrt{2}-1}\simeq 3.41$, close to the optimal value.*

In particular, order-optimal and optimal algorithms for the minimax bound have $\gamma=\frac{1}{2}$ and $f(t)=0$, for which Theorem 7 gives a simpler bound

$$R_T(\mathcal{A}_T) \leq c\sqrt{T} \implies R_T(\mathcal{DT}(\mathcal{A},(T_0 2^i)_{i\in\mathbb{N}})) \leq \frac{\sqrt{2}}{\sqrt{2}-1}c\sqrt{T}. \tag{18}$$

A.3. Proof of Theorem 9, "Minimax Regret Lower Bound with Exponential Horizons"

Proof Let $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$, and consider a fixed *stochastic* bandit problem. The lower bound (LB) from Lemma 2 gives

$$R_T(\mathcal{A}') \geq \sum_{i=0}^{L_T-1} R_{T_i - T_{i-1}}(A_{T=T_i-T_{i-1}})$$

We can use the hypothesis on $\mathcal{A}$ for each regret term, and as $0 < \gamma \leq 1$, we can use Lemma 14 (Eq. (24)) to distribute the power on γ to ease the proof and obtain

$$\begin{aligned}
&\geq cT_0^\gamma + c\sum_{i=1}^{L_T-1} (T_i - T_{i-1})^\gamma \\
&\geq cT_0^\gamma + c\sum_{i=1}^{L_T-1} \left(T_i^\gamma - T_{i-1}^\gamma\right) \qquad \text{(it is a telescopic sum and simplifies)} \\
&\geq cT_{L_T-1}^\gamma
\end{aligned}$$

Observe that $T_{L_T-1} \geq (T_{L_T})^{\frac{1}{b}}$ by definition of the exponential sequence (Def. 6), and $T_{L_T} \geq T$ (Def. 1). For the $\log(T_{L_T-1})$ term, we simply have $\log(T_{L_T-1}) \geq \frac{1}{b}\log(T)$ so if $c' = c/b$, then we obtain what we want

$$R_T(\mathcal{A}') \geq c'\, T^{\frac{\gamma}{b}}.$$

This lower bound goes from $R_T(\mathcal{A}_T) = \Omega T^\gamma$ to $R_T(\mathcal{DT}(\mathcal{A})) = \Omega T^{\frac{\gamma}{b}}$, and it looks very similar to the upper bound from Theorem 7 where $R_T(\mathcal{DT}(\mathcal{A})) = \mathcal{O}\left(T^{b\gamma}\right)$ was obtained from $R_T(\mathcal{A}_T) = \mathcal{O}(T^\gamma)$. ■

Remark It does seem sub-optimal to lower bound T_{L_T-1} like this ($T_{L_T-1} \geq (T_{L_T})^{\frac{1}{b}}$), but we remind that T can be located anywhere in the discrete interval $\{T_{L_T-1}, \dots, T_{L_T} - 1\}$, so in the worst case when T is very close to T_{L_T} (and for large enough T), we indeed have $T_{L_T-1}^b \sim T_{L_T}$ and $T_{L_T} \sim T$, so with this approach, the lower bound $T_{L_T-1} \geq T^{\frac{1}{b}}$ cannot be improved.

Appendix B. Minimax Regret Lower Bound with Geometric Horizons

We include here a last result that partly replies to Theorem 4. It is more subtle that the lower bound in Theorem 5 but still provides an interesting insight: if b is not chosen carefully (*i.e.*, if $\ell_0(\gamma, b) > 1$), then the anytime version of $\mathcal{A}_T$ using a geometric Doubling Trick suffers a non-improvable constant multiplicative overhead compared to $\mathcal{A}_T$.

Theorem 11 *For* stochastic models, *if $\mathcal{A}$ satisfies $R_T(\mathcal{A}_T) \geq c\, T^\gamma$, for $0 < \gamma < 1$ and $c > 0$, then the anytime version $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ with the geometric sequence $(T_i)_{i\in\mathbb{N}}$ of parameters $T_0 \in \mathbb{N}^*, b > 1$ (*i.e., *$T_i = \lfloor T_0 b^i \rfloor$) satisfies*

$$L_T \geq 2 \implies R_T(\mathcal{A}') \geq \ell_0(\gamma, b)\, c\, T^\gamma + g_0(T). \tag{19}$$

with $g_0(t) = \mathcal{O}(\log t) = o(t^\gamma)$, and a constant overhead $\ell_0(\gamma, b)$ *depending only on γ and b,*

$$\ell_0(\gamma, b) = \frac{(b-1)^\gamma}{b^\gamma(b^\gamma - 1)} > 0. \tag{20}$$

$\ell_0(\gamma, b)$ is always > 0 and tends to 0 for $b \to \infty$, and some choice of b gives $\ell_0(\gamma, b) > 1$.

Proof Let $\mathcal{A}' := \mathcal{DT}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$, and consider a fixed *stochastic* bandit problem. Assume $L_T \geq 2$. The lower bound (LB) from Lemma 2 gives

$$R_T(\mathcal{A}') \geq \sum_{i=0}^{L_T-1} R_{T_i - T_{i-1}}(A_{T=T_i-T_{i-1}})$$

We bound $T_i - T_{i-1} \geq T_0(b-1)b^{i-1} - 1$ for $i > 0$, thanks to Definition 3, and we can use the hypothesis on $\mathcal{A}$ for each regret term. Additionally, we have $(T_i - T_{i-1})^\gamma \geq (T_0(b-1)b^{i-1} - 1)^\gamma \geq (T_0(b-1))^\gamma (b^{i-1})^\gamma - 1$ by Lemma 14 (Eq. (24), as $b > 1$ and $0 < \gamma < 1$), thus

$$\begin{aligned}
&\geq \sum_{i=0}^{L_T-1} c(T_i - T_{i-1})^\gamma \geq cT_0^\gamma + cT_0^\gamma(b-1)^\gamma \sum_{i=1}^{L_T-1} (b^{i-1})^\gamma - c\sum_{i=1}^{L_T-1} 1 \\
&\geq cT_0^\gamma + cT_0^\gamma(b-1)^\gamma \sum_{i=0}^{L_T-2} (b^i)^\gamma - c(L_T - 1)
\end{aligned}$$

We have $\sum_{i=0}^{L_T-1} (b^i)^\gamma = \frac{(b^{L_T-1})^\gamma - 1}{b^\gamma - 1}$ thanks to a geometric sum (with $\gamma > 0$) and thus

$$\geq cT_0^\gamma + cT_0^\gamma(b-1)^\gamma \frac{(b^{L_T-1})^\gamma - 1}{b^\gamma - 1} + c(1 - L_T)$$

Thanks to Definition 3, b^{L_T-1} satisfies $b^{L_T-1} \geq \frac{1}{b}\frac{T}{T_0}$. Let $g_0(T) := cT_0^\gamma\left(\frac{(b-1)^\gamma}{b^\gamma-1} - 1\right) + c(L_T - 1) = \mathcal{O}(1) + \mathcal{O}\Big(\log_b(\frac{T}{T_0})\Big) = \mathcal{O}(\log T) = o(T^\gamma)$ and $g_0(T) > 0$, then we have

$$\geq c\frac{(b-1)^\gamma}{b^\gamma(b^\gamma - 1)} T^\gamma - \left[cT_0^\gamma\left(\frac{(b-1)^\gamma}{b^\gamma - 1} - 1\right) + c(L_T - 1)\right].$$

We obtain as announced, $R_T(\mathcal{A}') \geq \ell_0(b)\; cT^\gamma + g_0(T)$ with $g_0(T) = \mathcal{O}(\log T) = o(T^\gamma)$. ■

Maximizing the constant overhead? To maximize[4] $\ell_0(\gamma, b) := \frac{(b-1)^\gamma}{b^\gamma(b^\gamma-1)} > 0$, we fix γ and study the function $h : b \mapsto \ell_0(\gamma, b)$. The function h is of class $\mathcal{C}^1$ on $(1, \infty)$ and $h(b) \to +\infty$ for $b \to 1^+$ and $h(b) \to 0$ for $b \to \infty$. Moreover $h'(b) = -\gamma\frac{(b-1)^{\gamma-1}}{b^{\gamma+1}(b^\gamma-1)^2}\left(-2b^\gamma + b^{\gamma+1} + 1\right)$ has the same sign as $f(b) := -\left(-2b^\gamma + b^{\gamma+1} + 1\right)$. The function f is of class $\mathcal{C}^1$, with $f(1) = 0$ and $f'(b) = -(\gamma+1)(b - \frac{2\gamma}{\gamma+1})b^{\gamma-1}$, and as $0 < \gamma < 1$, $\frac{2\gamma}{\gamma+1} < 1$ so $f'(b) < 0$ for all $b > 1$. Thus f is decreasing, and $\forall b > 1, f(b) < f(1) = 0$. So h' has a negative sign, and this allows to conclude that h is decreasing, and so $b \mapsto \ell_0(\gamma, b)$ has no global maximum at fixed γ, and $\ell_0 \to \infty$ if $b \to 1^+$.

[4] For the largest possible lower bound, we try to maximize the constant overhead in the lower bound.

Relationship with the upper bound. For any $b > 1$, we compare $\ell_0(\gamma, b)$ with $\ell(\gamma, b)$ and we see that, interestingly, $\ell_0(\gamma, b) = \ell_2(\gamma, b)/b^{2\gamma}$, with $\ell_2(\gamma, b)$ from Theorem 4. For the particular case of $\gamma = \frac{1}{2}$, this lower bound also leads to another interesting remark: if b is chosen to minimize the overhead in the upper bound (Theorem 4, $b^*(\frac{1}{2}) = \frac{3+\sqrt{5}}{2}$), then this lower bound gives $\ell_0(\frac{1}{2}, b^*(\frac{1}{2})) = 1 + \frac{\sqrt{2}}{2} \simeq 1.71 > 1$, which proves that this choice of geometric doubling trick cannot be used to conserve an optimal algorithm, *i.e.*, the constant overhead cannot be made as close to 1 as we want.

Appendix C. An Efficient Heuristic, the Doubling Trick with "No Restart"

Let $\mathcal{A}' := \mathcal{DT}_{\text{no-restart}}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$ denotes the following Algorithm 2. The only difference with $\mathcal{DT}$ (Algorithm 1) is that the history from all the steps from $t = 1$ to $t = T_i$ is used to reinitialize the new algorithm $\mathcal{A}^{(i)}$. To be more precise, this means that a fresh algorithm $\mathcal{A}^{(i)}$ is created, and then fed with successive observations $(A'(s), Y_{A'(s),s})$ for all $1 \leq s < t$, like if it was playing *from the beginning*. Note that $A^{(i)}$ could have chosen a different path of actions, but we give it the observations from the previous plays of $\mathcal{A}'$.

This obviously cannot be applied to any kind of algorithm $\mathcal{A}$, and for instance any algorithm based on arm elimination (*e.g.*, Explore-Then-Commit approaches like in Garivier et al. (2016)) will most surely fail with this approach. This second doubling trick algorithm $\mathcal{DT}_{\text{no-restart}}$ can be applied in practice if $\mathcal{A}$ is index-based and uses the horizon T as a simple numerical parameter in its indexes, like it is the case for instance for AFHG (cf. Eq. (16)). or kl-UCB^{++} (cf. Eq. (17)).

```
Input: Bandit algorithm $\mathcal{A}$, and sequence $(T_i)_{i\in\mathbb{N}}$
Let $i = 0$, and initialize algorithm $\mathcal{A}^{(0)} = \mathcal{A}_{T_0}$.
for $t = 1, \dots, T$ do
    if $t > T_i$ then                // Next horizon $T_{i+1}$ from the sequence
        Let $i = i + 1$.
        Initialize algorithm $\mathcal{A}^{(i)} = A_{T_i - T_{i-1}}$.
        Update internal memory of $\mathcal{A}^{(i)}$ with the history of plays and rewards from $\mathcal{A}_{i-1}$.
    end
    Play with $\mathcal{A}^{(i)}$: play arm $A'(t) := A^{(i)}(t - T_i)$, observe reward $r(t) = Y_{A'(t),t}$.
end
```

Algorithm 2: The Non-Restarting Doubling Trick Algorithm, $\mathcal{A}' = \mathcal{DT}_{\text{no-restart}}(\mathcal{A}, (T_i)_{i\in\mathbb{N}})$.

However, it is much harder to state any theoretical result on this heuristic $\mathcal{DT}_{\text{no-restart}}$. We conjecture that a regret upper bound similar to (UB) from Lemma 2 could still be obtained, but it is still an open problem that the authors do not know how to tackle for a generic algorithm. The intuition is that starting $\mathcal{A}^{(i)}$ with some previous observations from the (same) *i.i.d.* process $(Y_{k,s})_{k\in\{1,\dots,K\},s\in\mathbb{N}}$ can only improve the performance of $\mathcal{A}^{(i)}$, and lead to a smaller regret on the interval $\{T_i, \dots, T_{i+1} - 1\}$.

Appendix D. Basic but Useful Results

All the logarithm $\log$ are taken in base $\mathrm{e} = \exp(1)$ (*i.e.*, natural logarithm $\ln$, but $\log$ is preferred for readability). Logarithms in a basis $b > 1$ are denoted $\log_b(x) := \frac{\log x}{\log b}$, for any $x \in \mathbb{R}$, $x > 0$.

We remind that $\lfloor x \rfloor$ denotes the integer part of $x \in \mathbb{R}$, and for $x > 0$, that is the unique integer i such that $i \leq x < i + 1$. The only property we use is its definition and the fact that $\lfloor x \rfloor \leq x$. We also define $\lceil x \rceil := 1 + \lfloor x \rfloor$ for $x \geq 0$, which is the unique integer j such that $j - 1 \leq x < j$.

D.1. Weighted Geometric Inequality

Lemma 12 (*Weighted* Geometric Inequality) *For any $n \in \mathbb{N}^*$, $b > 1$ and $\delta > 0$, and if f is a function of class $\mathcal{C}^1$, non-decreasing and non-negative on $[0, \infty)$, we have*

$$\sum_{i=0}^{n} f(i)(b^i)^\delta \leq \frac{b^\delta}{b^\delta - 1} f(n)(b^n)^\delta. \tag{21}$$

Proof By hypothesis, f is non-decreasing, so $\forall i \in \{0, \dots, n\}$, $f(i) \leq f(n)$, and so by using the sum of a geometric sequence, we have

$$\sum_{i=0}^{n} f(i)(b^i)^\delta \leq f(n)\left(\sum_{i=0}^{n}(b^i)^\delta\right) \leq f(n)\frac{1}{b^\delta - 1}(b^{n+1})^\delta = \frac{b^\delta}{b^\delta - 1}\left(f(n)(b^n)^\delta\right).$$

$f(i) = 1$ gives the geometric inequality. Note that if we make $\delta \to 0$, the left sum converges to $\sum_{i=0}^{n-1} f(i)$ and the right term diverges to $+\infty$, making this inequality completely uninformative. ■

D.2. Another Sum Inequality

This second result is similar to the previous one but for a "doubly exponential" sequence, *i.e.*, a^{b^i}, as it also bounds a sum of increasing terms by a constant times its last term.

Lemma 13 *For any $n \in \mathbb{N}^*$, $a > 1$, $b > 1$ and $\gamma > 0$, we have*

$$\sum_{i=0}^{n}(a^{b^i})^\gamma \leq a^\gamma + \left(1 + \frac{1}{(\log(a))(\log(b^\gamma))}\right)(a^{b^n})^\gamma = \mathcal{O}\Big((a^{b^n})^\gamma\Big). \tag{22}$$

Proof We first isolate both the first and last term in the sum and focus on the from $i = 1$ sum up to $i = n - 1$. As the function $t \mapsto (a^{b^t})^\gamma$ is increasing for $t \geq 1$, we use a sum-integral inequality, and then the change of variable $u := \gamma b^t$, of Jacobian $\mathrm{d}t = \frac{1}{\log b}\frac{\mathrm{d}u}{u}$, gives

$$\sum_{i=1}^{n-1}(a^{b^i})^\gamma \leq \int_1^n a^{\gamma b^t}\,\mathrm{d}t \leq \frac{1}{\log(b^\gamma)}\int_{\gamma b}^{\gamma b^n}\frac{a^u}{u}\,\mathrm{d}u$$

Now for $u \geq 1$, observe that $\frac{a^u}{u} \leq a^u$, and as $\gamma b > 1$, we have

$$\leq \frac{1}{\log(b^\gamma)}\int_{\gamma b}^{\gamma b^n} a^u\,\mathrm{d}u \leq \frac{1}{\log(b^\gamma)}\frac{1}{\log(a)}a^{\gamma b^n} = \frac{1}{(\log(a))(\log(b^\gamma))}(a^{b^n})^\gamma.$$

Finally, we obtain as desired, $\sum\limits_{i=0}^{n}(a^{b^i})^\gamma \leq a^\gamma + (a^{b^n})^\gamma + \frac{1}{(\log(a))(\log(b^\gamma))}(a^{b^n})^\gamma$. ■

D.3. Basic Functional Inequalities

These functional inequalities are used in the proof of the main theorems.

Lemma 14 (Generalized Square Root Inequalities) *For any $x, y \geq 0$ and $0 < \delta < 1$,*

$$(x+y)^\delta \leq x^\delta + y^\delta. \tag{23}$$

And conversely for any $0 < \delta < 1$, and $x, y \geq 0$, if $x \geq y$ then

$$(x-y)^\delta \geq x^\delta - y^\delta. \tag{24}$$

Proof Fix $y \geq 0$. Let $f(x) := (x+y)^\delta - (x^\delta + y^\delta)$ for $x \geq 0$. First, $f(0) = y^\delta - y^\delta = 0$, and as $\delta > 0$, f is differentiable on $[0, \infty)$, with $f'(x) = (\log \delta)(x+y)^\delta - (\log \delta) x^\delta = (\log \delta)((x+y)^\delta - x^\delta)$, and as $\delta < 1$, $\log \delta < 0$, and $(x+y)^\delta \geq x^\delta$, so $f'(x) \leq 0$ for any $x \geq 0$. Therefore, f is non-increasing, and so $\forall x \geq 0$, $f(x) \leq f(0) = 0$, so f is non-positive, giving the desired inequality (for any $y \geq 0$ and any $x \geq 0$).

The second inequality is a direct application of the first one. Assume $x \geq y$, and let $x' = x - y \geq 0$, then $(x'+y)^\delta \leq (x')^\delta + y^\delta$. This gives $(x-y)^\delta = (x')^\delta \leq (x'+y)^\delta - y^\delta = x^\delta - y^\delta$. ■

Lemma 15 (Bounding $\log(x - \Delta)$) *Let $x_0 > 1$ and $0 < \Delta < x_0$ (e.g., $\Delta \leq 1$), then*

$$\forall x \geq x_0, \quad \frac{\log(x_0 - \Delta)}{\log(x_0)} \log(x) \leq \log(x - \Delta) \leq \log(x). \tag{25}$$

With $\Delta = 1$, it implies that if $T_0 > 1$, $b > 1$ satisfy $T_0(b-1) > 1$, then for any $i \in \mathbb{N}$, we have

$$\log(T_0(b-1)b^i - 1) \geq \frac{\log(T_0(b-1) - 1)}{\log(T_0(b-1))} \log(T_0(b-1)b^i). \tag{26}$$

Proof Let $f(x) := \frac{\log(x-\Delta)}{\log(x)}$, defined for $x \geq x_0$. It is of class $\mathcal{C}^1$, and by differentiating, we have $f'(x) = \frac{\log(x) - \log(x-\Delta)}{x(\log x)^2} > 0$ as log is increasing. So f is increasing, and its minimum is attained at $x = x_0$, *i.e.*, $\forall x \geq x_0$, $f(x) \geq f(x_0) = \frac{\log(x_0 - \Delta)}{\log(x_0)} > 0$, which gives Eq. (25).

The corollary is immediate but stated explicitly for clarity when used in page 6. ■

Lemma 16 *For any $\gamma > 0$ and $\delta > 0$, the function $f : x \mapsto x^\gamma (\log x)^\delta$ is increasing on $[1, \infty)$.*

Proof f is of class $\mathcal{C}^1$ on $[1, \infty)$. First, if $\gamma > 0$, we have $f'(x) = x^{\gamma-1} (\log x)^{\delta-1} (\delta + \gamma \log(x))$. So $f'(x) > 0$ if and only if $\delta + \gamma \log(x) \geq 0$, that it $x \geq \exp(-\frac{\delta}{\gamma})$. But $x \geq 1$ and < 0 so $f'(x)$ is always positive, and thus f is increasing. Then, if $\gamma = 0$, we have $f'(x) = \delta \frac{1}{x} (\log x)^{\delta-1} > 0$ as $x > 1$ gives $\log x > 0$ and so $(\log x)^{\delta-1} > 0$.

It is also true if $\gamma \geq 0$, $\delta \geq 0$ if not both are zero simultaneously. ■

D.4. Controlling an Unbounded Sum of Dominated Terms

This Lemma is used in the proofs of our upper bounds (Theorems 4 and 7), to handle the sum of $f(T_i)$ terms. In particular, it can be applied to (T_i) the geometric sequence and $g(t) = h(t) = t^{\gamma}$ or $g(t) = h(t) = t^{\gamma}(\log t)^{\delta}$ (for Theorem 4) for $\gamma > 0$ and $\delta \geq 0$; or (T_i) the exponential sequence, $g(t) = t^{\gamma}(\log t)^{\delta}$ and $h(t) = t^{b\gamma}(\log t)^{\delta}$ (for Theorem 7) for $\gamma \geq 0$ and $\delta \geq 0$. Note that it would be obvious if L_T was bounded for $T \to \infty$, but a more careful analysis has to be given as $L_T \to \infty$.

Lemma 17 *Let f, g and h be three positive, diverging and non-decreasing functions on $[1, \infty)$, such that $f(t) = o(g(t))$ for $t \to \infty$. Let a non-decreasing diverging sequence $(T_i)_{i\in\mathbb{N}}$, and a diverging sequence $(L_T)_{T\in\mathbb{N}}$* (i.e., *$T_i \to \infty$ for $i \to \infty$ and $L_T \to \infty$ if $T \to \infty$*)*, such that there exists a constant $c \geq 0$ satisfying $\forall T \geq 1, \sum_{i=0}^{L_T} g(T_i) \leq c \times h(T)$. Then the (unbounded) sum of dominated terms $f(T_i)$ is still dominated by $h(T)$,* i.e.,

$$f(t) \underset{T\to\infty}{=} o(g(t)) \;\textit{and}\; \exists c \geq 0, \sum_{i=0}^{L_T} g(T_i) \underset{\forall T \geq 1}{\leq} c \times h(T) \implies \sum_{i=0}^{L_T} f(T_i) \underset{T\to\infty}{=} o(h(T)) \,. \quad (27)$$

Proof By hypothesis, if f is dominated by g, formally $f(t) = o(g(t))$, then there exists a positive function $\varepsilon(t)$ such that $f(t) = g(t)\varepsilon(t)$ and $\varepsilon(t) \to 0$ for $t \to 0$. Fix $\eta > 0$, as small as we want, then there exists $T_\eta \geq 1$ such that $\forall t \geq T_\eta, \varepsilon(t) \leq \eta$. Let i_η such that $T_{i_\eta - 1} < T_\eta \leq T_{i_\eta}$. Now for any $T \geq T_\eta$ and large enough so that $L_T \geq T_\eta$, we can start to split the sum

$$\sum_{i=0}^{L_T} f(T_i) = \sum_{i=0}^{i_\eta - 1} f(T_i) + \sum_{i=i_\eta}^{L_T} f(T_i)$$

The first sum is naively bounded by $i_\eta \times f(T_{i_\eta - 1})$ as f is increasing, and for the second sum, for any $i \geq i_\eta$, $T_i \geq T_\eta$ and so $f(T_i) = \varepsilon(T_i)g(T_i) \leq \eta g(T_i)$, thus

$$\leq i_\eta f(T_{i_\eta - 1}) + \eta \times \left(\sum_{i=i_\eta}^{L_T} g(T_i) \right)$$

The sum is smaller than a sum on a larger interval, as $g(T_i) \geq 0$ for any i, and f is increasing so

$$\leq i_\eta f(T_\eta) + \eta \left(\sum_{i=0}^{L_T} g(T_i) \right)$$

But now, $f(T_\eta) \leq \eta g(T_\eta)$ by hypothesis, and this sum is smaller than $c \times h(T)$ also by hypothesis

$$\leq i_\eta \eta \times g(T_\eta) + \eta c \times h(T) = \eta \left(i_\eta g(T_\eta) + c \times h(T) \right)$$

Finally, we use the hypothesis that $h(T)$ is diverging and as η and T_η are both fixed, there exists a $\widetilde{T_\eta} \geq T_\eta$ large enough so that $i_\eta g(T_\eta) \leq h(T)$ for all $T \geq \widetilde{T_\eta}$. And so we have finally proved that

$$\forall \eta > 0, \;\; \exists \widetilde{T_\eta} \geq 1, \;\; \forall T \geq \widetilde{T_\eta}, \;\; \sum_{i=0}^{L_T} f(T_i) \leq \eta(c+1) \times h(T).$$

This concludes the proof and shows that $\sum\limits_{i=0}^{L_T} f(T_i) = o(h(T))$ as wanted. ■

Appendix E. Additional Experiments

We presents additional experiments, for Gaussian bandits and for the heuristic $\mathcal{DT}_{\text{no-restart}}$.

E.1. Experiments with Gaussian Bandits (with Known Variance)

We include here another figure for experiments on Gaussian bandits, see Fig. 5.

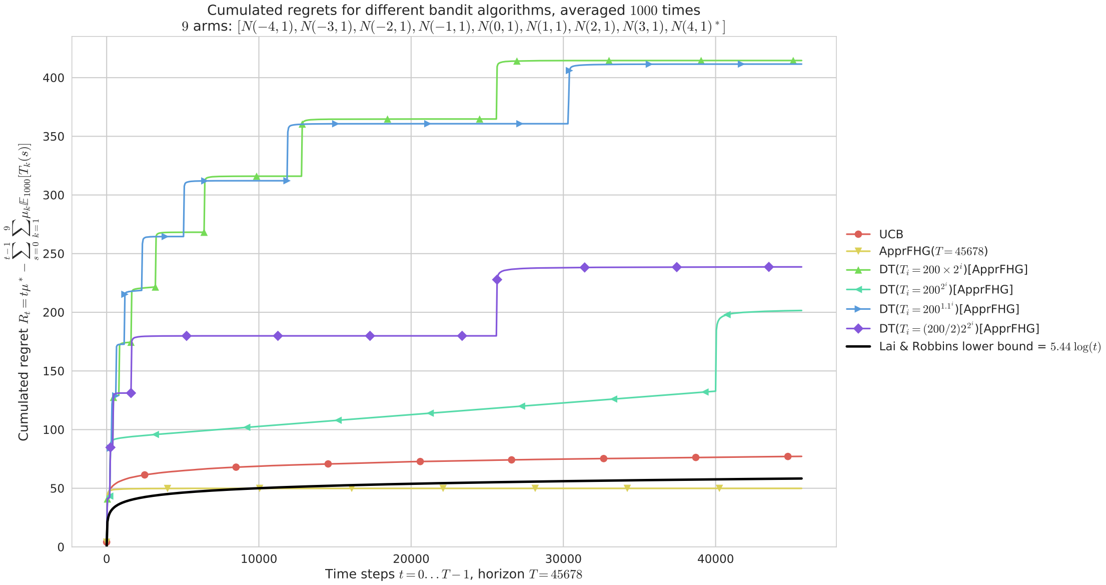

Figure 5: Regret for $\mathcal{DT}$, for $K = 9$ Gaussian arms $\mathcal{N}(\mu, 1)$, horizon $T = 45678$, $n = 1000$ repetitions and $\boldsymbol{\mu}$ uniformly spaced in $[-5, 5]^K$. On "easy" problems like this one, both UCB and AFHG perform similarly and attain near constant regret (identifying the best Gaussian arm is very easy here as they are sufficiently distinct). Each doubling trick also appear to attain near constant regret, but geometric doubling ($b = 2$) and slow exponential doubling ($b = 1.1$) are slower to converge and thus less efficient.

E.2. Experiments with $\mathcal{DT}_{\text{no-restart}}$

As mentioned previously, the $\mathcal{DT}_{\text{no-restart}}$ algorithm (Algorithm 2) is only an heuristic so far, as no theoretical guarantee was proved for it. For the sake of completeness, we also include results from numerical experiments with it, to compare its performance against the "with restart" version $\mathcal{DT}$.

As expected, $\mathcal{DT}_{\text{no-restart}}$ enjoys much better empirical performance, and in Figs. 6 and 7 we see that a geometric or a slow exponential doubling trick with no restart with kl-UCB^{++} can outperform kl-UCB and perform similarly to the non-anytime kl-UCB^{++}. But as observed before, the regret blows up after the beginning of each new sequence if the doubling sequence increase too fast (*e.g.*, exponential doubling). Despite what is proved theoretically in Theorem 5, here we observe that the geometric doubling is the only one to be slow enough to be efficient.

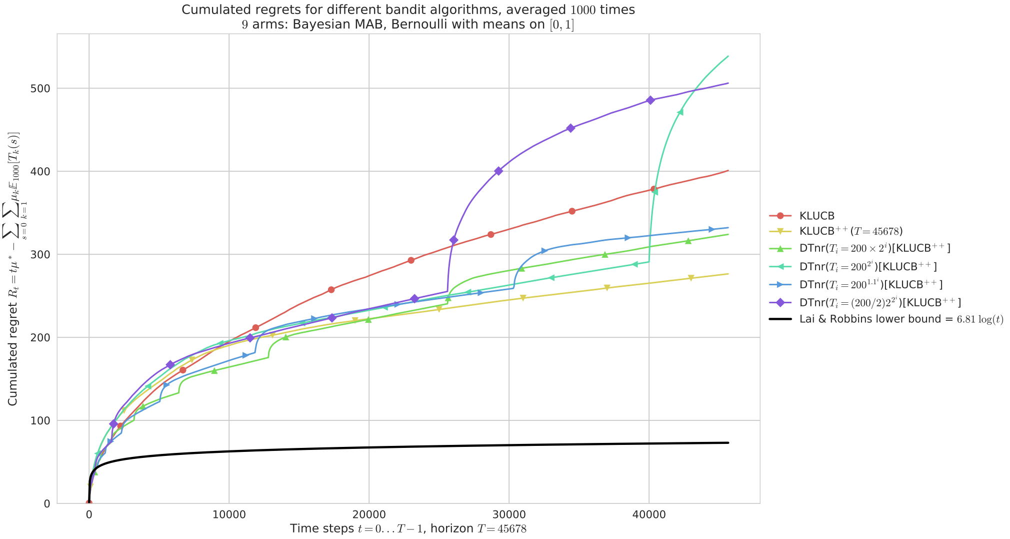

Figure 6: Regret for $K = 9$ Bernoulli arms, horizon $T = 45678$, $n = 1000$ repetitions and $\boldsymbol{\mu}$ taken uniformly in $[0, 1]^K$, for $\mathcal{DT}_{\text{no-restart}}$. Geometric doubling (*e.g.*, $b = 2$) and slow exponential doubling (*e.g.*, $b = 1.1$) are too slow, and short first sequences make the regret blow up in the beginning of the experiment. At $t = 40000$ we see clearly the effect of a new sequence for the best doubling trick ($T_i = 200 \times 2^i$). As expected, kl-UCB^{++} outperforms kl-UCB, and if the doubling sequence is growing fast enough then $\mathcal{DT}_{\text{no-restart}}(\text{kl-UCB}^{++})$ can perform as well as kl-UCB^{++}.

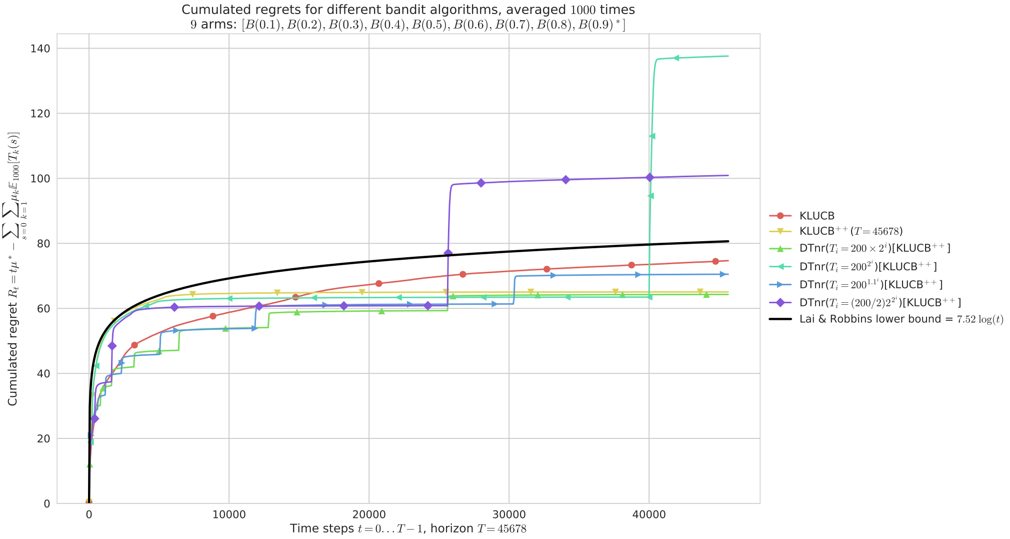

Figure 7: $K = 9$ Bernoulli arms with $\boldsymbol{\mu}$ evenly spaced in $[0, 1]^K$. On easy problems like this one, both kl-UCB and kl-UCB^{++} are very efficient, and here the geometric allows the $\mathcal{DT}_{\text{no-restart}}$ anytime version of kl-UCB^{++} to outperform both kl-UCB and kl-UCB^{++}.