# Structure Adaptive Algorithms for Stochastic Bandits

**Rémy Degenne**[1] **Han Shao**[2] **Wouter M. Koolen**[3]

## Abstract

We study reward maximisation in a wide class of structured stochastic multi-armed bandit problems, where the mean rewards of arms satisfy some given structural constraints, e.g. linear, unimodal, sparse, etc. Our aim is to develop methods that are *flexible* (in that they easily adapt to different structures), *powerful* (in that they perform well empirically and/or provably match instance-dependent lower bounds) and *efficient* in that the per-round computational burden is small. We develop asymptotically optimal algorithms from instance-dependent lower-bounds using iterative saddle-point solvers. Our approach generalises recent iterative methods for pure exploration to reward maximisation, where a major challenge arises from the estimation of the sub-optimality gaps and their reciprocals. Still we manage to achieve all the above desiderata. Notably, our technique avoids the computational cost of the full-blown saddle point oracle employed by previous work, while at the same time enabling finite-time regret bounds. Our experiments reveal that our method successfully leverages the structural assumptions, while its regret is at worst comparable to that of vanilla UCB.

## 1. Introduction

Stochastic multi-armed bandits are online learning problems in which an algorithm sequentially chooses among a finite set of actions (it "pulls an arm") and receives a stochastic reward in return. The goal is to obtain the maximal cumulative reward over time. Introduced by Thompson (1933), bandits have been the subject of intense study for many years now, starting with asymptotic results in the 80s and 90s (Lai & Robbins, 1985; Graves & Lai, 1997) and moving to the finite time analysis of algorithms since the beginning of the new millennium, notably with the introduction of strategies based on the Optimism in Face of Uncertainty principle (Agrawal, 1995; Auer et al., 2002). For an overview of the field, we point the reader to (Bubeck & Cesa-Bianchi, 2012; Lattimore & Szepesvári, 2019).

Recently many studies focused on problems inspired from practical applications, in which additional prior information is available. For example the means of the arms might be known to be sparse (Kwon et al., 2017), to form a Lipschitz function (Magureanu et al., 2014) or have a linear structure (Dani et al., 2008; Abbasi-Yadkori et al., 2011), or a combinatorial one (Cesa-Bianchi & Lugosi, 2012; Kveton et al., 2015). All these assumptions have been regrouped under the name "structured bandit" (Combes et al., 2017). As was the case in that last work, we aim at providing a family of algorithms that adapt to the known structure.

Bandit strategies realize a trade-off between exploitation (pulling the apparently best arm) and exploration. In the related problem of bandit pure exploration, in which no reward is gained but the goal is to answer a query, algorithms that adapt to any structure (and query) have been recently developed, first with asymptotic (Garivier & Kaufmann, 2016a), then with non-asymptotic guarantees (Degenne et al., 2019). Our approach here will be to develop these techniques further, and obtain results for reward maximisation.

### 1.1. Structured stochastic bandits

A finite number $K$ of arms have reward distributions $(\nu_k)_{k\in[K]}$ in a known sub-Gaussian canonical exponential family with one parameter (the mean of the distribution) in an open interval $\Theta \subseteq \mathbb{R}$. The vector of means of these distributions is denoted by $\boldsymbol{\mu} \in \mathbb{R}^K$. The arm with highest mean for vector $\boldsymbol{\mu}$ is denoted by $i^*(\boldsymbol{\mu})$ and we suppose that it is unique. We write $\mu^*$ for the value of that highest mean. An arm with mean below $\mu^*$ is called sub-optimal. For $x, y \in \Theta$, we denote by $d(x, y)$ the Kullback-Leibler divergence from the distribution parametrised by $x$ to that parametrised by $y$.

The mean vector $\boldsymbol{\mu}$ is known to belong to a set $\mathcal{M} \subseteq \Theta^K$, which represents the structure of the problem: the structural knowledge restricts the set of admissible mean parameters. For example, Lipschitz bandits prescribe that the means

[1] INRIA - DIENS - PSL Research University, Paris, France [2] Toyota Technological Institute at Chicago [3] Centrum Wiskunde & Informatica. Correspondence to: Rémy Degenne <remydegenne@gmail.com>.

*Proceedings of the $37^{th}$ International Conference on Machine Learning*, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

of successive arms cannot be far apart. We make the assumption that there exists a compact set $\mathcal{C} \subseteq \Theta^K$ such that $\mathcal{M} \subseteq \mathcal{C}$, which is convenient for the proof, but most likely not necessary. We also restrict our attention to what we call *regular* structures: those verifying Assumption 7 in appendix C. These include all examples of structures we found in the literature. The simplified Assumption 1 below presents the same idea, is verified by all structures we consider but the Sparse one and is presented here instead of the more detailed one to preserve the brevity of the exposition. For $j \in [K]$ we define $\neg j = \mathrm{cl}(\{\boldsymbol{\mu}{\in}\mathcal{M} | i^*(\boldsymbol{\mu}){\neq}j\})$ the closure of the set of alternative structured problems where the best arm is not $j$. Let also $\neg_x i = \{\boldsymbol{\lambda} \in \mathbb{R}^K \ : \ \lambda^i = x\} \cap \neg i$.

**Assumption 1** (Regularity, simplified)**.** There exists $c_{\mathcal{M}}$ such that for all $\boldsymbol{\mu} \in \mathcal{M}$, denoting $i^*(\boldsymbol{\mu}) = *$, if $\neg_{\mu^*} * \neq \emptyset$ then for all $\boldsymbol{\lambda} \in \neg *$ there exists $\boldsymbol{\xi} \in \neg_{\mu^*} *$ such that for all $k \in [K]$, $|\xi^k - \lambda^k| \leq c_{\mathcal{M}} |\mu^* - \lambda^*|$ .

The interaction between the algorithm and its environment is the following: at stage $t \geq 1$, **(1)** the algorithm pulls arm $k_t \in [K]$, **(2)** it observes a reward sample $X_t^{k_t} \sim \nu_{k_t}$, **(3)** its total reward is accrued by the mean reward $\mu^{k_t}$ (which is unobserved). An algorithm is a sequence of functions, one for each time $t \in \mathbb{N}$, that take $(k_1, X_1^{k_1}, \ldots, k_{t-1}, X_{t-1}^{k_{t-1}})$ as input and return $k_t \in [K]$.

We define the gap of an arm $k \in [K]$ as $\Delta^k = \mu^* - \mu^k$. Let $N_T^k$ be the number of pulls of arm $k$ up to time $T$. The goal of a bandit algorithm is to maximise its cumulated expected reward. Subtracting obtained reward from achievable reward, we arrive at the standard evaluation metric of expected regret,

$$\mathbb{E} \, R_T = T\mu^* - \sum_{t=1}^{T} \mathbb{E} \, \mu^{k_t} = \mathbb{E} \sum_{t=1}^{T} \Delta^{k_t} = \sum_{k=1}^{K} \Delta^k \, \mathbb{E} \, N_T^k \, .$$

### 1.2. Contributions

An algorithm is said to be asymptotically optimal if its regret verifies that for all $\boldsymbol{\mu} \in \mathcal{M}$, $\mathbb{E}_{\boldsymbol{\mu}}[R_t]/\ln(t)$ converges to a constant $V^{\mathcal{M}}(\boldsymbol{\mu})$, which matches the constant prescribed by the corresponding lower bound (see Section 2.1).

- We introduce a family of algorithms that are asymptotically optimal for all structures $\mathcal{M}$, while having explicit non-asymptotic regret bounds.
- We exhibits members of that family with computational complexity much lower than that of earlier structure-adaptive algorithms for many structures, since they never solve fully the lower bound minimax problem.
- On experiments, we verify that the proposed algorithms adapt to the structure. Their regret is close to that of the UCB algorithm up to a time depending on the complexity $V^{\mathcal{M}}(\boldsymbol{\mu})$ of the problem instance, after which the structural information is successfully exploited and the regret is of order $V^{\mathcal{M}}(\boldsymbol{\mu}) \ln T$.

### 1.3. Related work on structure adaptive methods

Lower bounds for the regret of asymptotically consistent algorithms (i.e. with regret $o(T)$ for all $\boldsymbol{\mu} \in \mathcal{M}$) take the form of a constrained minimisation problem (Lai & Robbins, 1985; Graves & Lai, 1997). The OSSB algorithm (Combes et al., 2017) uses a test to decide whether to exploit or explore (we will use a similar mechanism). If it explores, it solves a plug-in estimate of the lower bound problem, and pulls arms in order to approach the corresponding optimal sampling behaviour. Its analysis is based on the continuity of the optimiser, seen as a function of the bandit instance $\boldsymbol{\mu}$: if the plug-in estimate is close enough to the true mean vector, the estimated pulling proportions are close to optimal. In order to make sure that the empirical means converge to the true means, it pulls all arms $o(\ln T)$ times, a technique that is called "forced exploration". The OSSB algorithm is claimed to be asymptotically optimal up to a multiplicative factor, which can be as close to $1$ as wanted, with the drawback that the asymptotic regime is delayed.

A similar idea of tracking a plug-in estimate of the optimal pulling proportions was used before in the setting of fixed-confidence pure exploration, with the Track-and-Stop algorithm of Garivier & Kaufmann (2016a) (generalised to combinatorial settings by Chen et al. 2017). Track-and-Stop achieves asymptotic optimality for any structure under the above continuity assumption (Kaufmann & Koolen, 2018), and without it (Degenne & Koolen, 2019). In recent work, a game-based algorithm was developed which gets non-asymptotic guarantees for any structure in the pure exploration setting (Degenne et al., 2019). We take inspiration from that line of work.

Our algorithm uses an explore or exploit test as is done in OSSB. If it explores, it treats the lower bound as the value of a zero-sum game between two players. We implement two regret-minimising algorithms, one for each player, and their interaction ensures convergence to the value of the game.

## 2. Algorithm and Results

Our goal is to design efficient algorithms for the problem of reward maximisation in structured stochastic multi-armed bandit models. We specifically aim to incur little regret. To calibrate our expectations about what can be achieved (at least in principle), we start by reviewing the classic asymptotic lower bound of (Graves & Lai, 1997).

### 2.1. Asymptotic Lower Bound

Consider any algorithm, and assume that it is reasonable in the sense of *asymptotic consistency*, meaning that for any $\boldsymbol{\mu} \in \mathcal{M}$, any sub-optimal arm $k \notin \mathrm{argmax}_i \, \mu^i$ is sampled only sub-polynomially often, i.e. $\mathbb{E}_{\boldsymbol{\mu}}[N_T^k] \in \cap_{a>0} o(T^\alpha)$.

**Theorem 1** (Graves & Lai 1997)**.** *An algorithm asymptotically consistent for bandit structure $\mathcal{M}$ must incur regret*

$$\liminf_{T\to\infty} \frac{\mathbb{E}_{\boldsymbol{\mu}}[R_T]}{\ln T} \geq V^{\mathcal{M}}(\boldsymbol{\mu}) \quad \textit{for any instance } \boldsymbol{\mu} \in \mathcal{M}$$

*where the instance-dependent asymptotic regret rate $V^{\mathcal{M}}(\boldsymbol{\mu})$ is the value of the optimisation problem (here $\Lambda := \neg i^*(\boldsymbol{\mu})$)*

$$\min_{\boldsymbol{N}\geq \boldsymbol{0}} \sum_k N^k \Delta^k \quad \textit{s.t.} \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_k N^k d(\mu^k, \lambda^k) \geq 1. \qquad (1)$$

Given this insurmountable limit, we take as our goal to construct algorithms that satisfy $\mathbb{E}_{\boldsymbol{\mu}}[R_T] \leq V^{\mathcal{M}}(\boldsymbol{\mu}) \ln T + o(\ln T)$, with explicit finite-time control over the lower-order term.

### 2.2. Perturbed Game and Saddle Point Problems

In a multi-armed bandit, the optimal arm has zero gap, $\Delta^* = 0$. This creates several technical complications later on, which we choose to avoid by picking a small $\varepsilon > 0$ and defining the *$\varepsilon$-perturbed gaps* $\Delta^k_\varepsilon = \Delta^k \vee \varepsilon$. We call $V^{\mathcal{M}}_\varepsilon(\boldsymbol{\mu})$ the instance-dependent regret rate (1) with these perturbed gaps substituted. We find three useful saddle point expressions.

**Lemma 1.** *The reciprocal $D^{\mathcal{M}}_\varepsilon(\boldsymbol{\mu}) := 1/V^{\mathcal{M}}_\varepsilon(\boldsymbol{\mu})$ satisfies*

$$\begin{aligned} D^{\mathcal{M}}_\varepsilon(\boldsymbol{\mu}) &= \max_{\boldsymbol{w}\in\triangle} \inf_{\boldsymbol{\lambda}\in\Lambda} \frac{\sum_k w^k d(\mu^k, \lambda^k)}{\sum_k w^k \Delta^k_\varepsilon} && \text{(2a)} \\ &= \max_{\tilde{\boldsymbol{w}}\in\triangle} \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_k \tilde{w}^k \frac{d(\mu^k, \lambda^k)}{\Delta^k_\varepsilon} && \text{(2b)} \\ &= \inf_{\boldsymbol{q}\in\triangle(\Lambda)} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}}\left[d(\mu^k, \lambda^k)\right]}{\Delta^k_\varepsilon} && \text{(2c)} \end{aligned}$$

Compared to (1), problem (2a) is parametrised by the **fraction of rounds** $w^k \propto N^k$ spent pulling each arm $k$. The objective here is quasi-concave in $\boldsymbol{w}$ but not concave. The rewrite (2b) uses the **fraction of regret** $\tilde{w}^k \propto N^k \Delta^k_\varepsilon$ incurred by pulling arm $k$. This objective is linear (hence concave) in $\tilde{\boldsymbol{w}}$. From either form, randomising $\boldsymbol{\lambda} \sim \boldsymbol{q}$ licenses the min-max swap resulting in (2c).

These expressions correspond to a zero-sum game in which a pure strategy for the learner selects an arm $k \in [K]$, while for the opponent it picks a confusing instance $\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})$. The resulting payoff is then the information-regret ratio $\frac{d(\mu^k,\lambda^k)}{\Delta^k_\varepsilon}$, which quantifies the rate of progress in satisfying the information constraint in (1) per unit of objective value (i.e. regret) invested. In (2) the moves are ordered, upon which the outermost player needs to employ randomisation.

We quantify the cost of perturbation (Proof in Appendix E)

**Theorem 2.** *Under Assumption 1, there is a $c > 0$ such that for small $\varepsilon$ we have $D^{\mathcal{M}}_\varepsilon \geq D^{\mathcal{M}} - c\sqrt{\varepsilon D^{\mathcal{M}}}$ and hence $V^{\mathcal{M}}_\varepsilon \leq V^{\mathcal{M}} + c\sqrt{\varepsilon V^{\mathcal{M}}}$.*

### 2.3. Noise-Free Case

In this section we assume that we know the bandit model $\boldsymbol{\mu} \in \mathcal{M}$, and our goal is to compute the perturbed lower-bound value $V^{\mathcal{M}}_\varepsilon(\boldsymbol{\mu})$ and a matching strategy $\boldsymbol{N}$ from (1) or $\boldsymbol{w}$, $\tilde{\boldsymbol{w}}$ or $\boldsymbol{q}$ from (2). Our approach will be to pick either form (2b) or (2c), and run an online learner for the outer player against best response for the inner player. We will run this interaction for a carefully selected number of rounds $n$ to get our result.

In the following we will call the maximising player, controlling $k$, the $k$-player, even if this player randomises $k \sim \tilde{\boldsymbol{w}}$. Similarly we will talk about the minimising $\boldsymbol{\lambda}$-player. To treat the structure in a modular way, we will make the following computational assumption:

**Assumption 2.** We are given an *alt-min* oracle computing

$$(\boldsymbol{\mu}, \boldsymbol{w}, j, k) \mapsto \operatorname*{argmin}_{\boldsymbol{\lambda}\in\mathcal{M}:\lambda_k\geq\lambda j} \sum_i w^i d(\mu^i, \lambda^i) \qquad (3)$$

for any vector $\boldsymbol{\mu} \in \Theta^K$, non-negative weights $\boldsymbol{w}$ and arms $j \neq k$. This implies tractability of argmin over $\mathcal{M}$ and $\neg j$.

This assumption is satisfied (either by a closed-form expression, a binary search or full-blown numerical convex optimisation) for all structures we use for the experiments in Section 3 (unconstrained, sparse, linear, concave, unimodal and categorised).

#### 2.3.1. $\boldsymbol{\lambda}$-LEARNER

In this section we will target the saddle point (2c) using an online learner for $\boldsymbol{\lambda}$. Each round $t$, this $\boldsymbol{\lambda}$-learner outputs a distribution $\boldsymbol{q}_t$ on $\Lambda$. Given $\boldsymbol{q}_t$, we define the $k$-opponent to play best response, i.e.

$$k_t \in \operatorname*{argmin}_k \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}_t}\left[d(\mu^k, \lambda^k)\right]}{\Delta^k_\varepsilon}. \qquad (4)$$

We then have the $\boldsymbol{\lambda}$-learner update based on the linear loss function $\ell_t(\boldsymbol{q}) := \mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}}\left[d(\mu^{k_t}, \lambda^{k_t})\right]$. A $\boldsymbol{\lambda}$-learner regret bound of $\mathcal{B}^{\boldsymbol{\lambda}}_n$ for $n$ rounds of interaction gives us

$$\begin{aligned} \sum_{t=1}^n \ell_t(\boldsymbol{q}_t) &\leq \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_{t=1}^n d(\mu^{k_t}, \lambda^{k_t}) + \mathcal{B}^{\boldsymbol{\lambda}}_n \\ &= \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_k N^k_n d(\mu^k, \lambda^k) + \mathcal{B}^{\boldsymbol{\lambda}}_n. \qquad (5) \end{aligned}$$

By definition of $k_t$,

$$\begin{aligned}\sum_{t=1}^n \ell_t(\boldsymbol{q}_t) &= \sum_{t=1}^n \Delta_\varepsilon^{k_t} \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}_t}\left[d(\mu^{k_t},\lambda^{k_t})\right]}{\Delta_\varepsilon^{k_t}} \\ &\overset{(4)}{=} \sum_{t=1}^n \Delta_\varepsilon^{k_t} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}_t}\left[d(\mu^{k},\lambda^{k})\right]}{\Delta_\varepsilon^{k}} .\end{aligned}$$

We then have, abbreviating $R_n^\varepsilon = \sum_{t=1}^n \Delta_\varepsilon^{k_t}$,

$$\begin{aligned}\sum_{t=1}^n \ell_t(\boldsymbol{q}_t) &\geq \max_k \sum_{t=1}^n \Delta_\varepsilon^{k_t} \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}_t}\left[d(\mu^{k},\lambda^{k})\right]}{\Delta_\varepsilon^{k}} \\ &= R_n^\varepsilon \max_k \sum_{t=1}^n \frac{\Delta_\varepsilon^{k_t}}{R_n^\varepsilon} \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}_t}\left[d(\mu^{k},\lambda^{k})\right]}{\Delta_\varepsilon^{k}} \\ &\geq R_n^\varepsilon \inf_{\boldsymbol{q}} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda}\sim\boldsymbol{q}}\left[d(\mu^{k},\lambda^{k})\right]}{\Delta_\varepsilon^{k}} \\ &\overset{(2c)}{=} R_n^\varepsilon D_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}). \end{aligned} \tag{6}$$

We will choose $n$ adaptively, running the algorithm as long as $\inf_{\boldsymbol{\lambda}\in\Lambda} \sum_k N_n^k d(\mu^k,\lambda^k) \leq \ln T$. Chaining (5) and (6) then yield $R_n^\varepsilon D_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) \leq \ln T + \mathcal{B}_n^{\boldsymbol{\lambda}}$, so that

$$R_n := \sum_{t=1}^n \Delta^{k_t} \leq R_n^\varepsilon \leq V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) \left(\ln T + \mathcal{B}_n^{\boldsymbol{\lambda}}\right).$$

At this point we need $\mathcal{B}_n^{\boldsymbol{\lambda}}$ to be $o(\ln T)$ for $V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu})$ to be the dominant term. This will be feasible, as $n = \mathcal{O}(\ln T)$ and $\mathcal{B}_n^{\boldsymbol{\lambda}}$ can be taken to be $\mathcal{O}(\sqrt{n})$.

In particular, because $n\varepsilon \leq R_n^\varepsilon \leq V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) \left(\ln T + c\sqrt{n}\right)$,

$$\sqrt{n} \leq \frac{cV_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) + \sqrt{c^2 V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu})^2 + 4\varepsilon V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) \ln T}}{2\varepsilon}. \tag{7}$$

So that indeed $\sqrt{n} = o(\ln T)$. Moreover, these asymptotics start kicking in once $\ln T \gg \frac{c^2 V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu})}{4\varepsilon}$. Conversely, to ensure asymptotic optimality we need to pick $\varepsilon = \omega(1/\ln T)$. In our next section we will develop an anytime version using a time-decaying $\varepsilon_t$.

The practicality of implementing a $\boldsymbol{\lambda}$-learner depends on the geometry of the sets $\neg j$ and the makeup of the loss function. One may always write $\neg j$ as a union of $K-1$ cells, which are intersections of $\mathcal{M}$ with one linear inequality

$$\neg j = \bigcup_{k\neq j} \left\{\boldsymbol{\lambda} \in \mathcal{M} | \lambda_k \geq \lambda_j\right\}.$$

For the structures considered in the experiments section (except the sparse structure), we find that these cells are in fact convex sets (while $\neg j$ is not). Moreover, for Gaussian and Bernoulli bandits the relative entropy function $d(\cdot,\cdot)$ is strongly convex in its second argument. Hence we can instantiate a sub-learner on each cell. An interesting choice is Follow-the-Leader, which is tractable by Assumption 2 and incurs $\mathcal{O}(\ln t)$ regret. On the meta-level we aggregate the sub-learner iterates using an experts algorithm (we use AdaHedge by De Rooij et al. (2014) for $\mathcal{O}(\sqrt{t})$ regret).

### 2.3.2. $k$-LEARNER

In this section we will target the saddle point (2b). We will work with a $k$-learner that in each round $t$ outputs an action $\tilde{\boldsymbol{w}}_t$ (recall these are the desired per-arm fractions of regret, not of rounds). Given $\tilde{\boldsymbol{w}}_t$ from the $k$-learner, we define time proportions $\boldsymbol{w}_t$ and the best-response opponent $\boldsymbol{\lambda}_t \in \Lambda$ by

$$w_t^k \propto \tilde{w}_t^k / \Delta_\varepsilon^k, \tag{8}$$

$$\boldsymbol{\lambda}_t \in \operatorname*{argmin}_{\boldsymbol{\lambda}\in\Lambda} \sum_k w_t^k d(\mu^k, \lambda^k). \tag{9}$$

The $k$-learner then updates using the linear gain function

$$g_t(\tilde{\boldsymbol{w}}) = \left(\sum_k w_t^k \Delta_\varepsilon^k\right) \sum_k \tilde{w}^k \frac{d(\mu^k, \lambda_t^k)}{\Delta_\varepsilon^k}. \tag{10}$$

Note that the $w_t^k$ in its leading factor do *not* vary with the argument $\tilde{\boldsymbol{w}}$; they are computed from $\tilde{\boldsymbol{w}}_t$ using (8).

A $k$-learner regret bound of $\mathcal{B}_n^k$ provides

$$\sum_{t=1}^n g_t(\tilde{\boldsymbol{w}}_t) \geq \max_k \sum_{t=1}^n \left(\sum_j w_t^j \Delta_\varepsilon^j\right) \frac{d(\mu^k, \lambda_t^k)}{\Delta_\varepsilon^k} - \mathcal{B}_n^k. \tag{11}$$

Moreover, the total gain satisfies

$$\begin{aligned}\sum_{t=1}^n g_t(\tilde{\boldsymbol{w}}_t) &= \sum_{t=1}^n \left(\sum_k w_t^k \Delta_\varepsilon^k\right) \sum_k \tilde{w}_t^k \frac{d(\mu^k, \lambda_t^k)}{\Delta_\varepsilon^k} \\ &\overset{(8)}{=} \sum_{t=1}^n \sum_k w_t^k d(\mu^k, \lambda_t^k) \\ &\overset{(9)}{=} \sum_{t=1}^n \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_k w_t^k d(\mu^k, \lambda^k) \\ &\leq \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_{t=1}^n \sum_k w_t^k d(\mu^k, \lambda^k). \end{aligned} \tag{12}$$

Running as long as $\inf_{\boldsymbol{\lambda}\in\Lambda} \sum_{t=1}^n \sum_k w_t^k d(\mu^k, \lambda^k) \leq \ln T$ and chaining (11) with (12) results in

$$\begin{aligned}\ln T &\geq \max_k \sum_{t=1}^n \left(\sum_j w_t^j \Delta_\varepsilon^j\right) \frac{d(\mu^k, \lambda_t^k)}{\Delta_\varepsilon^k} - \mathcal{B}_n^k \\ &= R_n^\varepsilon \max_k \sum_{t=1}^n \frac{\left(\sum_j w_t^j \Delta_\varepsilon^j\right)}{R_n^\varepsilon} \frac{d(\mu^k, \lambda_t^k)}{\Delta_\varepsilon^k} - \mathcal{B}_n^k \end{aligned} \tag{13}$$

where we abbreviated $R_n^\varepsilon = \sum_{t=1}^n \sum_k w_t^k \Delta_\varepsilon^k$. Minimizing over $\boldsymbol{q} \in \triangle(\Lambda)$,

$$\begin{aligned} \ln T \;&\geq\; R_n^\varepsilon \inf_{\boldsymbol{q} \in \triangle(\Lambda)} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}}\left[d(\mu^k, \lambda^k)\right]}{\Delta_\varepsilon^k} - \mathcal{B}_n^k \\ &= R_n^\varepsilon D_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) - \mathcal{B}_n^k, \end{aligned} \tag{14}$$

All in all we showed

$$R_n \;:=\; \sum_{t=1}^n \sum_k w_t^k \Delta^k \;\leq\; R_n^\varepsilon \;\leq\; V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) \left(\ln T + \mathcal{B}_n^k\right)$$

and again we need to set things up so that $\mathcal{B}_n^k = o(\ln T)$. Now this may work, as we will have $n = \mathcal{O}(\ln T)$ and $\mathcal{B}_n^k = \mathcal{O}(\sqrt{n})$ so that, in total, $\mathcal{B}_n^k = \mathcal{O}(\sqrt{\ln T})$.

Tuning $\varepsilon$ is a bit trickier than for (7), since the range of the gain function (10) scales with $1/\varepsilon$ and so hence will $\mathcal{B}_n^k$. Still $n = \mathcal{O}(V_\varepsilon/\varepsilon \ln T)$ and hence the asymptotics kick in for $\ln T \gg (V_\varepsilon/\varepsilon)^3$, or equivalently $\varepsilon \gg V_\varepsilon/(\ln T)^{1/3}$. We expect that power of $T$ is artificial; the range of practically observed gains (10) may well be constant.

A bandit algorithm cannot pull any $\boldsymbol{w}_t$ in the simplex, as would be prescribed by the $k$-learner. It has to pull one arm at each time. We circumvent that difficulty by using a so-called tracking procedure, which ensures that for all times $\boldsymbol{N}_t \approx \sum_{s=1}^t \boldsymbol{w}_s$. We will use $k_t \in \operatorname{argmin}_{k \in [K]} N_{t-1}^k - \sum_{s=1}^t w_s^k$ (breaking ties arbitrarily). A similar rule was analysed in Garivier & Kaufmann (2016a). Our analysis reveals that it ensures that for all $t \in \mathbb{N}$ and $k \in [K]$,

$$-\ln(K) \leq N_t^k - \sum_{s=1}^t w_s^k \leq 1\,.$$

The previous result (Garivier & Kaufmann, 2016a, Lemma 15) has $K-1$ instead of our $\ln K$. Our proof makes use of a subtle invariant; it can be found in Appendix B.

In terms of implementation, we need to supply two things. First, a learner for linear losses defined on the simplex. This is a standard experts problem, for which we use AdaHedge (De Rooij et al., 2014). Second, we need to compute the best response $\boldsymbol{\lambda}_t \in \neg i^*(\boldsymbol{\mu})$. This is where the structure dependence of the approach manifests; this step is tractable by Assumption 2. The computational complexity of either the $k$-learner or $\boldsymbol{\lambda}$-learner based iterative saddle point approach is dominated by $K-1$ alt-min oracle calls every round.

### 2.4. Saddle Point-Based MAB Reward Maximisation

In this section we build atop the noise-free saddle point computation presented above. We obtain algorithms for regret minimisation in structured bandit problems that have finite-time bounds which convey, in particular, asymptotic optimality. The k-learner variant is displayed as Algorithm 1.

**Algorithm 1** SP$_k$ Learner

```
Require: Exploration threshold $f(t)$
Require: Learner $\mathcal{A}$ for linear losses on the simplex
Start an instance $\mathcal{A}_j$ for each arm $j$.
Pull each arm once and get $\hat{\boldsymbol{\mu}}_K$.
Initialise $N_t^k = 1$ and $n_t^k = 0$ for all $k \in [K]$.
for $t = K+1, \cdots, T$ do
    if there is an exploit-worthy $i$ for which (15) then
    $k_t = i$ (pick any if there are several suitable).
    else
    Estimate best arm: $j_t = \operatorname{argmax}_k \hat{\mu}_{t-1}^k$.
    if $n_t^{j_t}$ is even then $k_t = j_t$. else
    Compute confidence interval $[\underline{\mu_{t-1}^k}, \overline{\mu_{t-1}^k}]$ for every
      arm $k \in [K]$ using (16).
    Estimate gaps: $\tilde{\Delta}_{\varepsilon_t}^k = \max\{\varepsilon_t, \overline{\mu_{t-1}^{j_t}} - \overline{\mu_{t-1}^k}\}$.
    Get $\tilde{\boldsymbol{w}}_t$ from $\mathcal{A}_{j_t}$, compute $w_t^k \propto \tilde{w}_t^k / \tilde{\Delta}_{\varepsilon_t}^k$.
    Find the best response in $\neg j_t$ to $\boldsymbol{w}_t$ given $\hat{\boldsymbol{\mu}}_{t-1}$:

        $\boldsymbol{\lambda}_t \;=\; \operatorname{argmin}_{\boldsymbol{\lambda} \in \neg j_t} \sum_k w_t^k d(\hat{\mu}_{t-1}^k, \lambda^k).$

    Compute estimates $\mathrm{UCB}_t^k$ as in (17)
    Feed $g_t(\tilde{\boldsymbol{w}}) = \left(\sum_k w_t^k \tilde{\Delta}_{\varepsilon_t}^k\right) \sum_k \tilde{w}^k \,\mathrm{UCB}_t^k$ to $\mathcal{A}_{j_t}$.
    Pull $k_t = \operatorname{argmin}_{k \in [K]} N_{t-1}^k - \sum_{s=1}^t w_s^k$.
    end if
    Increase $n_t^{j_t}$ by 1.
    end if
    Access reward $X_t^{k_t}$, update $\hat{\boldsymbol{\mu}}_t$ and $\boldsymbol{N}_t$.
end for
```

The main challenge is that we do not know $\boldsymbol{\mu}$ (nor anything derived, including the gaps $\boldsymbol{\Delta}_\varepsilon$ and the index of the best arm $i^*(\boldsymbol{\mu})$), and that we hence need to estimate these live.

We sketch an outline of the construction. The main distinction made by our algorithm is whether to *explore* or *exploit*. Exploitation occurs when we are certain enough that we have the right best arm for our bandit $\boldsymbol{\mu}$, that is when

$$\exists i \in [K],\ \min_{\boldsymbol{\lambda} \in \neg i} \sum_k N_{t-1}^k d(\hat{\mu}_{t-1}^k, \lambda^k) \;>\; f(t-1), \tag{15}$$

where we will use a threshold $f(t)$ ($\approx \ln t$ plus lower order) that is high enough so that the cumulative contribution to the regret of rounds where (15) fails is bounded by a constant. Note that we cannot afford a failure probability $\geq \frac{1}{t}$, for then the contribution of the failure cases would be of order $\geq \ln T$, voiding asymptotic optimality.

During the other rounds, our algorithm explores. The main idea here is to pick an online learner for either the $k$ or $\boldsymbol{\lambda}$ side, and adapt the corresponding noise-free pipeline (13) and (14) or (5) and (6). These rounds all happen under the complement of (15), which now fulfils the role of the stopping condition "running as long as …" in Section 2.3.

We now go over the major upgrades one by one, relating back to our Algorithm 1 as we go along.

**Estimating $i^*(\boldsymbol{\mu})$.** First, in Section 2.3 we assumed we know the best arm $i^*(\boldsymbol{\mu})$. Here that is no longer the case, and instead we have to rely on estimates, which can be wrong. Our approach to deal with this is to run not one but $K$ many saddle point computations, one for every candidate best arm. In each exploration round, we produce an estimate $j_t$ of the best arm, and only advance the saddle point interaction corresponding to that estimate. This way, the saddle point iteration for arm $i^*(\boldsymbol{\mu})$ will simulate a pure trajectory as per Section 2.3. Using concentration of measure and invoking our learners' regret bounds, we show (in Appendix C.7.2) that the other instances corresponding to incorrect estimates of the best arm will only make a lower-order $\mathcal{O}(\ln\ln T)$ contribution to the regret. With that out of the way, we can reason about exploration rounds in which $j_t = i^*(\boldsymbol{\mu})$.

**Estimating $\boldsymbol{\mu}$.** Our second problem is then that we need to estimate the bandit instance $\boldsymbol{\mu}$ and its perturbed sub-optimality gaps $\Delta^k_\varepsilon$. We will be guided by the following idea. We want to look at the noise-free chain (13) and (14), and relate all occurrences of estimates $\hat{\boldsymbol{\mu}}$, $\tilde{\Delta}^k_{\varepsilon_t}$, to their unknown truth $\boldsymbol{\mu}$ and $\Delta^k_{\varepsilon_t}$. We will take inspiration from the classic UCB analysis, and use concentration to bound the instantaneous estimation error after $n$ exploration rounds by $\sqrt{\ln(n)/n}$. These errors then accumulate over $n$ exploration rounds to an order $\sqrt{n\ln n}$ overhead. This is a lower-order term as long as the number $n$ of exploration steps is $o(\ln T)^2$.

More precisely, let $E^*_s$ be the event that we explore at stage $s$ and have the right guess for $i^*(\boldsymbol{\mu})$ and define $n^*_t = \sum_{s=1}^t \mathbb{1}\{E^*_s\}$. With estimates $\hat{\mu}_t$, we have similarly to equation (12) that if we explore at state $t+1$,

$$\ln t \geq \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_{s\leq t: E^*_s} \sum_{k=1}^K w^k_s d(\hat{\mu}^k_t, \lambda^k).$$

Our algorithm cannot play at time $s < t$ based on $\hat{\boldsymbol{\mu}}_t$, since it does not know it yet. It will use the available estimate $\hat{\boldsymbol{\mu}}_{s-1}$. We then quantify how far these estimated means are from $\boldsymbol{\mu}$ with a concentration event $\mathcal{E}^{exp}_n$, which states that for all $k \in [K]$ and all times $t$,

$$\mu^k \in [\underline{\mu^k_t}(n), \overline{\mu^k_t}(n)], \text{ with } \underline{\mu^k_t}(n), \overline{\mu^k_t}(n) \text{ solution to}$$
$$N^k_t d(\hat{\mu}^k_t, x) = \ln(n) + \mathcal{O}(\ln\ln t)\,. \qquad (16)$$

That event is such that apart from a small number of rounds, when we explore event $\mathcal{E}^{exp}_{n^*_t}$ happens. If we can ensure that $n^*_t$ is less than a power of $\ln(t)$, then these intervals are of order $\ln\ln t$. We get

$$\ln t \geq \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_{s\leq t: E^*_s\cap\mathcal{E}^{exp}_{n^*_t}} \sum_{k=1}^K w^k_s d(\hat{\mu}^k_{s-1}, \lambda^k) - \mathcal{O}(\sqrt{n^*_t})\,.$$

In order to continue along computations (12), (13) and (14), we need estimates for the gaps. Based on the small confidence intervals $[\underline{\mu^k_t}, \overline{\mu^k_t}]$ (where $\overline{\mu^k_t} = \overline{\mu^k_t}(n^*_t)$), we define $\tilde{\Delta}^k_t = \max\{\varepsilon, \overline{\mu^{j_t}_t} - \overline{\mu^k_t}\}$. Then up to a $\mathcal{O}(\sqrt{n^*_t})$ term

$$\ln t \geq \inf_{\boldsymbol{\lambda}\in\Lambda} \sum_{s\leq t: E^*_s\cap\mathcal{E}^{exp}_{n^*_t}} (\sum_{j=1}^K w^j_s \tilde{\Delta}^j_s) \sum_{k=1}^K \tilde{w}^k_s \frac{d(\hat{\mu}^k_{s-1}, \lambda^k)}{\tilde{\Delta}^k_s}$$
$$\geq \sum_{s\leq t: E^*_s\cap\mathcal{E}^{exp}_{n^*_t}} (\sum_{j=1}^K w^j_s \tilde{\Delta}^j_s) \sum_{k=1}^K \tilde{w}^k_s \frac{d(\hat{\mu}^k_{s-1}, \lambda^k_s)}{\tilde{\Delta}^k_s}\,.$$

**Optimism.** Using the estimated gains directly could lead to high regret, in the same way as the Follow-The-Leader strategy might accrue linear regret in stochastic bandits. We use the now widely employed Optimism in Face of Uncertainty principle (Agrawal, 1995) and replace the gains by an optimistic estimate. In the hypothetical noiseless case, we would like at time $s$ to feed to the $k$-learner the gain vector $k \mapsto (\sum_{i=1}^K w^i_s \Delta^i_{\varepsilon_s}) \frac{d(\mu^k, \lambda^k_s)}{\Delta^k_{\varepsilon_s}}$ (see (10)). We define

$$\mathrm{UCB}^k_t = \max_{\xi\in[\underline{\mu^k_{t-1}}, \overline{\mu^k_{t-1}}]} \frac{d(\xi, \lambda^k_s)}{\max\{\varepsilon_t, \mathbb{1}\{k \neq j_t\}(\overline{\mu^{j_t}_{t-1}} - \xi)\}}. \qquad (17)$$

Under the event that $\mu^k \in [\underline{\mu^k_{t-1}}, \overline{\mu^k_{t-1}}]$, this is an upper bound for the factor depending on $k$ in the gain of the $k$-learner, up to the difference between $\overline{\mu^{j_t}_{t-1}}$ and $\mu^{j_t}$, which will be small since we ensure that $j_t$ is pulled every other round in the exploration phase. Again under that event, $\tilde{\Delta}^k_{t-1}(\mathrm{UCB}^k_t - d(\hat{\mu}^k_{t-1}, \lambda^k_s)/\tilde{\Delta}^k_{t-1})$ is upper bounded, such that the total cost of the introduction of those upper confidence bounds is $\mathcal{O}(\sqrt{n^*_t})$. We can now finish the computations of equations (13) and (14): up to $\mathcal{O}(\sqrt{n^*_t})$,

$$\ln t \geq \sum_{s\leq t: E^*_s\cap\mathcal{E}^{exp}_{n^*_t}} (\sum_{j=1}^K w^j_s \tilde{\Delta}^j_s) \sum_{k=1}^K \tilde{w}^k_s \,\mathrm{UCB}^k_s$$
$$\geq \max_{k\in K} \sum_{s\leq t: E^*_s\cap\mathcal{E}^{exp}_{n^*_t}} (\sum_{j=1}^K w^j_s \tilde{\Delta}^j_s)\, \mathrm{UCB}^k_s$$
$$\geq \max_{k\in K} \sum_{s\leq t: E^*_s\cap\mathcal{E}^{exp}_{n^*_t}} (\sum_{j=1}^K w^j_s \tilde{\Delta}^j_s) \frac{d(\mu^k, \lambda^k_s)}{\Delta^k_{\varepsilon_s}}$$
$$\geq R^{\varepsilon,*}_t D^{\mathcal{M}}_\varepsilon(\boldsymbol{\mu}) - \mathcal{O}(\sqrt{n^*_t})\,.$$

Finally, since $n_t^* \le R_t^{\varepsilon,*}/\varepsilon$ this is an equation which asymptotically gives $R_T^{\varepsilon,*} \le V_\varepsilon^{\mathcal{M}}(\boldsymbol{\mu}) \ln T$ as desired. Since this is the only term in the regret which is not $o(\ln T)$, we have asymptotic optimality if we have $\varepsilon \to 0$ at a suitable rate.

**Effect of the confidence intervals on the sampling behaviour.** During the first rounds, the algorithm does not exploit yet. Hence the number of exploration rounds $n_t$ is $t$. If the best arm is pulled enough to be well estimated, our gap estimates are $\tilde{\Delta}_t^k \approx \mu^{j_t} - \mu^k - \sqrt{2\ln(n_t)/N_t^k}$ in the Gaussian case. For these estimates to be representative of the true gaps $\mu^{j_t} - \mu^k$, we need $N_t^k \propto \ln(n_t)/(\Delta^k)^2$. This is of order $\ln \ln t$ once $n_t$ is logarithmic in $t$, but it is $\ln(t)/(\Delta^k)^2$ at the beginning. We indeed verify experimentally that during the first rounds our algorithm pulls similarly to vanilla UCB. Afterwards, the exploitation regime starts and the structure adaptive sampling begins.

**How our algorithm relates to previous techniques.** Our algorithm employs a explore/exploit test based on a log-likelihood ratio, as is done in the OSSB algorithm (Combes et al., 2017). While the exploration phase of OSSB is akin to the exploration algorithm Track-and-Stop (Garivier & Kaufmann, 2016b), our exploration phase is inspired from the game point of view of (Degenne et al., 2019). The strategy presented above generalizes that exploration algorithm in several key aspects (the pure exploration case is recovered by forcing all gaps $\Delta^k$ to 1).

- addressing the regret saddle-point problem with online learning algorithms is non-trivial even for known gaps, and requires several innovations (Section 2.3). For example, one of the players has to play regret proportions instead of pull proportions, as in (8), and gap-weighted regret bounds need to be employed, as in (10).
- the unknown gaps need to be estimated appropriately (line 11 of Alg. 1),
- one of the gaps is 0, which is a problem since we divide by the gaps: we introduce the epsilon-perturbed problem to solve that issue,
- we need to relate the values of the perturbed and original games (Theorem 2).

### 2.5. Main Result

**Theorem 3.** *For any $k$-learner or $\lambda$-learner with regret bound of order $\mathcal{O}(\sqrt{n})$ after $n$ steps, the expected regret of our structure adaptive algorithm verifies for all $\boldsymbol{\mu} \in \mathcal{M}$,*

$$\lim_{T \to +\infty} \frac{\mathbb{E}_{\boldsymbol{\mu}}[R_T]}{\ln T} \le V^{\mathcal{M}}(\boldsymbol{\mu}) \,.$$

The arguments presented above in fact lead to a finite time bound, but it contains many $o(\ln T)$ terms that we choose not to detail here. Proof in Appendix C.

## 3. Experiments

As mentioned in Section 2.3, the asymptotics kick in when $\ln T \gg \frac{c^2 V_{\varepsilon_T}^{\mathcal{M}}(\boldsymbol{\mu})}{4\varepsilon_T}$ for the $\boldsymbol{\lambda}$-learner and $\ln T \gg \left(V_{\varepsilon_T}^{\mathcal{M}}(\boldsymbol{\mu})/\varepsilon_T\right)^3$ for the $k$-learner. This indicates that for small time horizons, the asymptotic theoretical guarantees become meaningless. It is worth noting that this is ubiquitous in the asymptotic literature. We still investigate empirically whether and when these algorithms start exploiting the structures and how they perform in small horizons. Implementation details are in Appendix G.

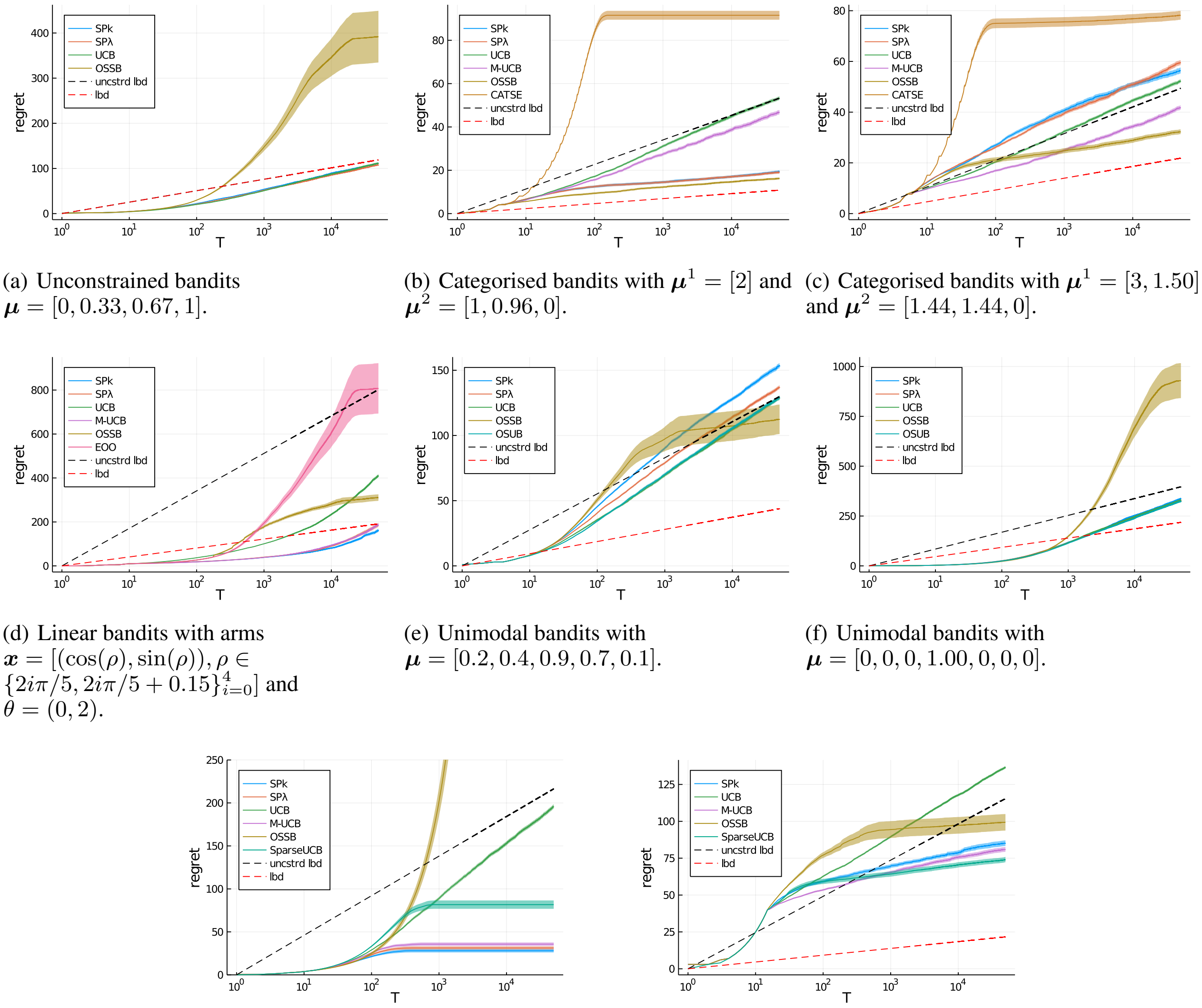
We perform four series of experiments. In all cases, rewards are Gaussian with variance 1. The structures are:

- Figure 1(a): unconstrained. This is the usual stochastic bandit setting, without additional structure.
- Figures 1(b) and 1(c): categorised bandits with strong dominance (Jedor et al., 2019). The arms belong to one of two categories (and this information on each arm is known), and all arms of one category have means higher than all arms of the other (but which category is the "good" one is unknown).
- Figure 1(d) : linear (Auer, 2002). There exists an unknown parameter $\theta \in \mathbb{R}^d$ with $d < K$ and known vectors $x_1, \ldots, x_K \in \mathbb{R}^d$ such that $\mu^k = x_k^\intercal \theta$.
- Figures 1(e) and 1(f): unimodal (Combes & Proutiere, 2014). The mean function $k \mapsto \mu^k$ is unimodal, i.e. $\mu^1 \le \mu^2 \le \ldots \le \mu^* \ge \ldots \ge \mu^{K-1} \ge \mu^K$.
- Figures 1(g) and 1(h): sparse (Kwon et al., 2017). Given sparsity $s$ and value $\gamma$, $s$ of $K$ arms have means above $\gamma$ while others have means $\gamma$.

We mainly compare the following four algorithms,

- SP$_k$, our saddle point algorithm based on a $k$-learner. The learner used is AdaHedge (De Rooij et al., 2014).
- SP$_\lambda$, our algorithm based on a $\lambda$-learner. In all experiments except the sparse structure, the minimisation over the alternative sets is equivalent to minimising on the union of a small number of convex sets. For each such convex set, we implement one expert running Follow-The-Leader, and the $\lambda$-learner aggregates these experts using AdaHedge. For the sparse setting, there is still a decomposition into a finite number of convex sets, but that number is combinatorially large and we did not implement SP$_\lambda$ in that case.
- The OSSB algorithm of (Combes et al., 2017), which solves a plug-in estimate of optimisation problem (1) with empirical estimates and tracks its solution.
- The vanilla UCB algorithm of (Auer et al., 2002; Garivier & Cappé, 2011), which selects the arm with the highest upper confidence bound calculated without additional structural information.

The implementation of our algorithms departs from the



(a) Unconstrained bandits $\boldsymbol{\mu} = [0, 0.33, 0.67, 1]$.

(b) Categorised bandits with $\boldsymbol{\mu}^1 = [2]$ and $\boldsymbol{\mu}^2 = [1, 0.96, 0]$.

(c) Categorised bandits with $\boldsymbol{\mu}^1 = [3, 1.50]$ and $\boldsymbol{\mu}^2 = [1.44, 1.44, 0]$.

(d) Linear bandits with arms $\boldsymbol{x} = [(\cos(\rho), \sin(\rho)), \rho \in \{2i\pi/5, 2i\pi/5 + 0.15\}_{i=0}^4]$ and $\theta = (0, 2)$.

(e) Unimodal bandits with $\boldsymbol{\mu} = [0.2, 0.4, 0.9, 0.7, 0.1]$.

(f) Unimodal bandits with $\boldsymbol{\mu} = [0, 0, 0, 1.00, 0, 0, 0]$.

(g) Sparse bandits with $s = 1, \gamma = 0.3$ and $\mu_1 = 0.8$.

(h) Sparse bandits with $s = 2, \gamma = 0$ and $[\mu_1, \mu_2] = [3.00, 2.00]$.

*Figure 1.* Regret of algorithms.

theory in two ways. First, we set $\varepsilon_t \propto 1/n_t^{j_t}$, where $n_t^{j_t}$ is the number of exploration steps with the current estimated best arm. This is a faster decrease than allowed by our proof. Second, the width of the small confidence intervals (16) is smaller: $\ln(n_t)$ instead of $\ln(n_t) + \mathcal{O}(\ln \ln t)$.

We also compare with structure specific algorithms:

- The structural UCB algorithm, denoted by $\mathcal{M}$-UCB, which looks for the highest upper confidence bound over the intersection of the confidence region and the structure. In the linear case, it uses an adapted estimation of the means, such that it is the same as LinUCB (Auer, 2002).
- The CATSE algorithm of (Jedor et al., 2019) for categorised instances, which eliminates all arms in the "bad" category once we are confident to tell which category is "bad".
- The "End of Optimism" algorithm of (Lattimore & Szepesvári, 2017) (called "EOO" on figures) for linear instances, which is similar to OSSB with different forced exploration.
- The OSUB algorithm of (Combes & Proutiere, 2014) for unimodal instances, which pulls among the neighbourhood of the empirical optimal arm.
- The SparseUCB algorithm of (Kwon et al., 2017) for sparse instances, which constructs a set of $\geq s$ arms which are estimated to have means larger than $\gamma$ and then play UCB among this set.

Figure 1 reports the mean regret of these algorithms over 200 repetitions. The shaded areas show an interval around the mean of width twice its empirical standard deviation. We contrast the empirical regret of the algorithms with the unconstrained lower bound (called "uncstrd lbd" in the figures) $\sum_{k \neq i^*(\boldsymbol{\mu})} \Delta_k / d(\mu^k, \mu^*) \ln(T)$ and the structural lower bound (called "lbd") $V^{\mathcal{M}}(\boldsymbol{\mu}) \ln(T)$.

We investigate when the theoretically motivated $\mathrm{SP}_k$, $\mathrm{SP}_\lambda$ and OSSB also benefit from good empirical performances. OSSB adapts to the structure in more experiments (the slope of its regret is close to the slope of the structural lower bound, see Figures 1(a), 1(c), 1(e), 1(h)), but it incurs a high initial regret in cases 1(a), 1(b), 1(c), 1(f). In Figure 1(g), OSSB performs poorly and is equivalent to Follow-The-Leader, since its exploration test never triggers on that structure. In contrast, our algorithms are safer, in the sense that their regret is at worst comparable to that of UCB, but they adapt to the structure often only later, sometimes after the chosen horizon. We verify that adaptation to the structure indeed happens, notably in Figures 1(b) and 1(h).

We are not able to provide a characterisation of the problems on which these algorithms manage to adapt to the structure for small horizons. Obtaining such a characterization, or methods that would adapt to the structure for small times, is the most obvious open problem for structured bandits.

## 4. Open Questions in Structured Bandits

While our algorithms are asymptotically optimal, the dependence of the non-asymptotic bound on problem parameters like the number of arms is definitely not the best possible. Some structures would allow an algorithm to face a problem with a huge number of arms and not suffer from that multiplicity, while our bounds have terms linear in $K$. Broadly, our algorithm is suited to large times, while adaptivity to structure in a small horizon regime remains to be explored.

On the topic of computational complexity, we remarked that our algorithm never solves the lower bound problem completely, but for example in the case of the $k$-learner only computes the best-response over an alternative set. If the structure set $\mathcal{M}$ is complicated, that computation can still be expensive, or even infeasible. However one could argue that it does not make sense to compute exactly the best response to a noisy problem, and indeed our algorithm remains asymptotically optimal if that minimization is approximate, with an error at the $n^{\text{th}}$ exploration step of order $1/\sqrt{n}$.

A weakness of the analysis of algorithms based on an explore/exploit test like ours is the concentration inequality used to bound the number of times that the exploitation phase is wrongly entered. A concentration result gives a threshold $\beta(t, \delta)$ such that with probability $1-\delta$, a deviation is lower than $\beta(t, \delta)$ for all times. That threshold is linear in the number of arms $K$, while it could be smaller for many particular structures. For example, in the unconstrained case, it can be made to depend only on $\ln(K)$. (see also the discussion on the Pure Exploration Problem Rank by Kaufmann & Koolen (2018)). Adapting the concentration inequality to the structure is in general an open question.

Finally, the behaviour of our algorithms in experiments has an initial phase where the structure information cannot yet be used. The length of this phase depends on the instance complexity (our bounds currently admit an exponential dependence here), and in some of our experiments we have not decidedly left the initial phase. To shorten this phase without changing the high-level approach, one could work on improving confidence regions for the gradients. Our current choice, $\mathrm{UCB}_t^k$ in (17), is possibly quite conservative.

## Acknowledgements

This work was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute). H. Shao is supported in part by the National Science Foundation under grant CCF-181501. H. Shao would like to thank Emilie Kaufmann for her early discussion on unimodal bandits.

## References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 2312–2320, 2011.

Agrawal, R. Sample mean based index policies by o (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27(4):1054–1078, 1995.

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

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

Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521833787.

Bubeck, S. and Cesa-Bianchi, N. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

Cesa-Bianchi, N. and Lugosi, G. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5):1404–1422, 2012.

Chen, L., Gupta, A., Li, J., Qiao, M., and Wang, R. Nearly optimal sampling algorithms for combinatorial pure exploration. In *Proceedings of the 2017 Conference on Learning Theory*, volume 65, pp. 482–534. PMLR, July 2017.

Combes, R. and Proutiere, A. Unimodal bandits: Regret lower bounds and optimal algorithms. In *International Conference on Machine Learning*, pp. 521–529, 2014.

Combes, R., Magureanu, S., and Proutiere, A. Minimal exploration in structured stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 1763–1771, 2017.

Dani, V., Hayes, T. P., and Kakade, S. M. Stochastic linear optimization under bandit feedback. In *21st Annual Conference on Learning Theory - COLT 2008, Helsinki, Finland, July 9-12, 2008*, pp. 355–366. Omnipress, 2008.

De Rooij, S., Van Erven, T., Grünwald, P. D., and Koolen, W. M. Follow the leader if you can, hedge if you must. *The Journal of Machine Learning Research*, 15(1):1281–1316, 2014.

Degenne, R. and Koolen, W. M. Pure exploration with multiple correct answers. In *Advances in Neural Information Processing Systems (NeurIPS) 32*, pp. 14564–14573. Curran Associates, Inc., December 2019.

Degenne, R., Koolen, W. M., and Ménard, P. Non-asymptotic pure exploration by solving games. In *Advances in Neural Information Processing Systems*, 2019.

Garivier, A. and Cappé, O. The KL-UCB algorithm for bounded stochastic bandits and beyond. In *Proceedings of the 24th annual conference on learning theory*, pp. 359–376, 2011.

Garivier, A. and Kaufmann, E. Optimal best arm identification with fixed confidence. In *Proceedings of the 29th Conference On Learning Theory (COLT)*, 2016a.

Garivier, A. and Kaufmann, E. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*, pp. 998–1027, 2016b.

Graves, T. L. and Lai, T. L. Asymptotically efficient adaptive choice of control laws in controlled Markov chains. *SIAM Journal on Control and Optimization*, 35(3):715–743, 1997.

Jedor, M., Perchet, V., and Louedec, J. Categorized bandits. In *Advances in Neural Information Processing Systems*, pp. 14399–14409, 2019.

Kaufmann, E. and Koolen, W. M. Mixture martingales revisited with applications to sequential tests and confidence intervals. Preprint, October 2018.

Kveton, B., Wen, Z., Ashkan, A., and Szepesvári, C. Tight regret bounds for stochastic combinatorial semi-bandits. In *Artificial Intelligence and Statistics*, pp. 535–543, 2015.

Kwon, J., Perchet, V., and Vernade, C. Sparse stochastic bandits. In *Conference On Learning Theory*, 2017.

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

Lattimore, T. and Szepesvári, C. The end of optimism? an asymptotic analysis of finite-armed linear bandits. In *Artificial Intelligence and Statistics*, pp. 728–737, 2017.

Lattimore, T. and Szepesvári, C. Bandit algorithms. *Cambridge University Press*, 2019.

Magureanu, S., Combes, R., and Proutiere, A. Lipschitz bandits: Regret lower bound and optimal algorithms. In *Conference on Learning Theory*, pp. 975–999, 2014.

Thompson, W. R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.