Identifying the Best Transition Law

Mehrasa Ahmadipour, Elise Crépon, Aurélien Garivier

Identifying the Best Transition Law

Mehrasa Ahmadipour, **élise Crepon**, **Aurélien Garivier**
UMPA, ENS de Lyon, Lyon, France

1. Abstract

Motivated by recursive learning in Markov Decision Processes, this paper studies best-arm identification in bandit problems where each arm's reward is drawn from a multinomial distribution with a known support. We compare the performance reached by strategies including notably LUCB without and with use of this knowledge. In the first case, we use classical non-parametric approaches for the confidence intervals. In the second case, where a probability distribution is to be estimated, we first use classical deviation bounds (Hoeffding and Bernstein) on each dimension independently, and then the Empirical Likelihood method (EL-LUCB) on the joint probability vector. The effectiveness of these methods is demonstrated through simulations on scenarios with varying levels of structural complexity.

2. Introduction

The importance of interactions between entities such as human-computer interfaces, complex decision-making in autonomous systems like self-driving cars (Chen et al., 2024), or dynamic difficulty adjustment (DDA) in online gaming (Lopes & Lopes, 2022) has fostered the development of Reinforcement Learning (RL) as a model for dynamical systems where responsible agents aim for optimal decisions in an uncertain environment. Markov Decision Processes (MDP) proved able to capture many interesting features of these scenarios, while providing a rich toolbox of computationally efficient and mathematically founded algorithms(Puterman, 1994; Bertsekas, 2005; Sutton, 2018; Moerland et al., 2023a).

An MDP is defined as a tuple consisting of the state space $\mathcal{S}$, the action set $\mathcal{A}$, the transition probability kernel $\mathcal{P}$, and the reward function $\mathcal{R}$. At time t, a (deterministic) policy is a mapping $\pi_t : \mathcal{S} \to \mathcal{A}$ that specifies which action to choose according to the current state. To determine the optimal policy in an MDP, *Bellman equations* provide a recursive formulation for the value function $V_t(s)$, the expected cumulative reward starting from state s at time t. In the case of a finite horizon T, the optimal policy π^* satisfies:

$$V_t^{\pi^*}(s) = \max_{\pi \in \Pi} \left[\mathcal{R}(s,a) + \sum_{s' \in \mathcal{S}} \mathcal{P}(s' \mid s,a) V_{t+1}^{\pi^*}(s') \right]. \tag{1}$$

When the transition probabilities are fully known in the problem, the agent can compute an optimal policy without interacting with the environment – a process known as *Planning* (see (Moerland et al., 2023b)). Otherwise, the process of *learning* requires the estimation of expected future returns. This paper focuses on a particular learning sub-task: the choice of the policy at time t with known value function V_{t+1}[1]. The player, in state s, can transition to one of d possible next states s', each associated with a certain value $V_{t+1}(s')$. She chooses an action $a \in \mathcal{A}$, and transitions to a next state S' according to the transition probability vector P_a (see Figure 1). Each destination state has an associated expected value, and the goal is to find the best action that maximizes this expected value.

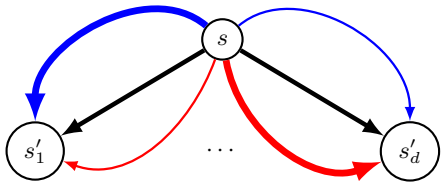

Figure 1. A learner at s selects a color and transitions to S'. Each color represents a probability vector, meaning that the likelihood of arriving at a destination varies depending on the chosen color.

This decision-making problem is very reminiscent of a Multi-Armed Bandit (MAB) problem, as the outcome of each action a depends only on the current state s and transition probabilities P_a, independently of past decisions. The considered task is usually called Best Arm Identification (BAI): identify the arm yielding the highest expected value with as few samples as possible.

But the considered bandit problem has a specific feature:

[1] the use of this sub-task in a backward induction for a step-by-step solution to the dynamic programming formulation is left for future work.

the support of the arms, $\{V_{s'} : s' \in \mathcal{S}\}$, is known for all arms. This feature can be used or not by the agent: in the *non-structured setting*, the agent ignores these values and simply adopts a classical bandit strategy. She can then rely on established algorithms such as LUCB (Kalyanakrishnan et al., 2012) and Track&Stop (Garivier & Kaufmann, 2016). In the *structured setting*, each arm is modeled as multinomial distribution on a known support – see also (Agrawal et al., 2020). For this case, we propose Structured-LUCB, a modified version of the LUCB algorithm, and an Empirical Likelihood approach (EL-LUCB) inspired by (Filippi et al., 2010). The goal of this paper is to investigate in which way the structured approach outperforms the non-structured one in specific scenarios on the support of the distribution, while in others, the non-structured method can perform better.

3. State-of-the-art and connections

In this work, we focus on the fixed-confidence PAC (Probably Approximately Correct) setting for MAB problems. While the fixed-budget setting remains somewhat mysterious (see e.g., (Bubeck et al., 2012)), the fixed-confidence setting is better understood since its sample complexity was identified in (Garivier & Kaufmann, 2016) thanks to change-of-measure techniques. Different structures for bandit arms have been considered: multi-modal (Saber & Maillard, 2024), linear (Abbasi-Yadkori et al., 2011), contextual(Lu et al., 2010), kernel-based models(Neu et al., 2024), etc. However, these models do not address bandit problems with multinomial reward distributions. We propose novel adaptations of LUCB, —*Structured-LUCB* using Bernstein and Hoeffding inequalities— and an *EL-LUCB* algorithm.

In the Structured-LUCB, we use empirical Bernstein bounds which have been investigated in UGapE (Gabillon et al., 2012) and in the context of Racing algorithms (Mnih et al., 2008; Heidrich-Meisner & Igel, 2009), where Bernstein-based methods were used to design efficient stopping times. Bernstein-based concentration is also used in regret minimization in (Audibert & Bubeck, 2010). The empirical likelihood method seems particularly well suited for multinomial distributions. In the EL-LUCB algorithm, we employ Kullback–Leibler (KL)-divergence-based confidence regions with LUCB, inspired by the regret minimization algorithms of (Filippi et al., 2010), (Kaufmann & Kalyanakrishnan, 2013) or (Cappé et al., 2013). These structured methods aim to improve performance by incorporating additional information about the underlying probability vector and its constraints on a simplex.

Our study compares these two methodologies under various scenarios. We employ the Top-two (leader-challenger) sampling rule (Russo, 2016; Jourdan et al., 2022; You et al., 2023), which has shown robust performance in both Bayesian and frequentist settings, to guide our sampling strategies. Recent work by (Jourdan et al., 2022) extends this method to bounded distributions, and (You et al., 2023) presents a further enhancement with theoretical guarantees.

By contrasting Structured and Non-Structured algorithms, we explore whether leveraging known structures can yield significant benefits in terms of sample efficiency and decision accuracy. To the best of our knowledge, no prior research has systematically examined shifting perspectives between the two approaches. Our contribution lies in this comparative analysis, providing insights into when and how structural assumptions can be beneficial.

The paper is organized as follows. We begin by formally explaining the model and our assumptions. We then develop the non-structured approach before the structured cases. We finally present the numerical experiments that we compare and discuss on different algorithms.

4. System Model

We consider K multinomial distributions $P_1, \dots, P_K$. Each distribution has d mutually exclusive outcomes, associated with values $V = [v_1, \dots, v_d] \in \mathbb{R}^d$. An outcome v_i represents the value obtained at the next state S'. We describe each distribution as a vector $P_a = [p_{a,1}, \dots, p_{a,d}]$ lies on the simplex $\Delta^d := \left\{\theta \in \mathbb{R}^{d+1} \;\middle|\; \sum_{i=1}^{d+1} \theta_i = 1,\ \theta_i \geq 0 \text{ for all } i = 1, \dots, d\right\}$, i.e., $p_{a,i} \geq 0$ and $\sum_{i=1}^{d} p_{a,i} = 1$.

At discrete time intervals $t = 1, 2, \dots$, the learner selects an action $A_t \in \mathcal{A}$ and receives an independent sample $Z_{A_t} = [Z_{A_t,1}, \cdots, Z_{A_t,d}]$ where we assume Z_{A_t} is a one-hot vector indicating the next state S'. Specifically, Z_{A_t} is drawn such that $\mathbb{P}[Z_{A_t} = e_i] = p_{A_t,i}$, where e_i is the i-th standard basis vector in $\mathbb{R}^d$. Denoting by $p \cdot v$ the scalar product of two vectors p and v, the expected value of the reward V under the probability vector P_a is expressed as $\mathbb{E}_{P_a}[V] = \sum_{i=1}^{d} p_{a,i} v_i = P_a \cdot v$. Without loss of generality, we can assume the following order:

$$\mathbb{E}_{P_1}[V] > \mathbb{E}_{P_2}[V] > \cdots > \mathbb{E}_{P_K}[V]. \tag{2}$$

The learner empirically constructs $\hat{P}_a$ and attempts to find the action a^* that maximizes the expected reward as soon as possible:

$$\max_{a \in \mathcal{A}} P_a \cdot V \quad \text{s.t. } \mathrm{Dist}(\hat{P}_a, P_a) \leq \epsilon, \tag{3}$$

where $\mathrm{Dist}(\cdot, \cdot)$ quantifies the "distance" between the estimated transition probabilities $\hat{P}_a$ and the optimistic transition probabilities P_a. We define this distance more precisely later, first using an L-norm, then using the KL-divergence.

We operate in the *fixed-confidence* regime, where a maximal risk parameter $\delta \in (0, 1)$ is fixed.

Definition 1. *A strategy is called δ-PAC (Probably Approximately Correct) if, for every tuple of distributions $\mathcal{P} = (P_1, \ldots, P_K)$, it satisfies $\mathbb{P}_{\mathcal{P}}[\tau < \infty] = 1$ and $\mathbb{P}_{\mathcal{P}}[\hat{a}^* \neq a^*] \leq \delta$.*

Definition 2 (Sample Complexity)**.** *Given a bandit model with K arms, the fixed-confidence sample complexity κ is defined as the minimum expected number of samples needed by a δ-PAC algorithm:*

$$\kappa := \inf_{\text{PAC algorithms}} \limsup_{\delta \to 0} \frac{\mathbb{E}[\tau]}{\log \frac{1}{\delta}}. \tag{4}$$

To achieve the goal of identifying the best arm, the learner must employ a strategy denoted by the triple $\left((A_t), \tau, \hat{a}^*\right)$, which includes a sampling rule A_t determining the chosen arm based on past actions and rewards, a stopping rule τ, and a recommendation rule $\hat{a}^*$ that suggests the best action at termination. The learner's strategy is crucial for efficiently identifying the best arm, requiring a careful balance between exploration and exploitation, while considering the observed outcomes to make informed decisions about arm selection and termination of the sampling process.

For simplicity, we assume that the rewards are bounded:

Assumption 1. *Assume that $v_i \in [0, 1]$ for all $i \in \{1, \ldots, d\}$.*

Since the deviations of any $[0, 1]$ random variables from its expectation are bounded by those of a Bernoulli distribution with the same mean, this assumption allows us to model the expected reward using a Bernoulli distribution with parameter $\mu_a := \mathbb{E}_{P_a}[V]$. Our initial approach to the problem employs the known BAI methods within the MAB for Single Parameter Exponential Family(SPEF) distributions.

5. Non-Structured Approach

We assume that rewards are i.i.d. and follow a Bernoulli distribution with parameter $\mu_a := \mathbb{E}_{P_a}[V]$ under Assumption 1. This setting leads to a MAB problem with distributions belong to the SPEF as described in (Garivier & Kaufmann, 2016). The concept of distinguishability is employed to characterize the lower bounds for the sample complexity in (Garivier & Kaufmann, 2016). This notion is quantified using the KL-divergence, denoted as $\mathrm{KL}(x, y) := x \log(\frac{x}{y}) + (1 - x) \log(\frac{(1-x)}{(1-y)})$. Denote by $\mathcal{S}$ a set of SPEF bandit models such that each bandit model $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$ in $\mathcal{S}$ has a unique optimal arm $a^*(\boldsymbol{\mu})$. For each $\boldsymbol{\mu} \in \mathcal{S}$, there exists an arm $a^*(\boldsymbol{\mu})$ s.t. They use the notion of the alternative set

$$\mathrm{Alt}(\boldsymbol{\mu}) := \{\Lambda \in \mathcal{S} : a^*(\Lambda) \neq a^*(\boldsymbol{\mu})\},$$

which is the set of problems Λ for which the optimal arm $a^*(\lambda)$ differs from the optimal arm $a^*(\mu)$ of the reference distribution $\boldsymbol{\mu}$. They characterize a lower bound for any δ-PAC strategy and any bandit model $\boldsymbol{\mu} \in \mathcal{S}$ under a given risk level $\delta \in (0, 1)$:

$$\mathbb{E}[\tau_\delta] \geq T^*(\boldsymbol{\mu}) \mathrm{KL}(\delta, 1 - \delta), \tag{5}$$

where

$$T^*(\boldsymbol{\mu})^{-1} := \sup_{\boldsymbol{\omega} \in \Sigma_K} \inf_{Q \in \mathrm{Alt}(\boldsymbol{\mu})} \sum_{a=1}^{K} \omega_a \mathrm{KL}(\mu_a, Q_a). \tag{6}$$

Here, the set of proportions for pulling arms is defined as $\Sigma_K = \{\omega \in \mathbb{R}_+^K : \sum_{i=1}^K \omega_i = 1\}$.

The assumption of SPEF on bandits' distributions allows the inner minimization of (6) to be solved and to obtain an explicit formulation for the optimal weights ω. The same authors introduced an asymptotic optimal algorithm matching this lower bound, called Track&Stop (T&S). But its computational complexity often motivates the use of more practical alternatives such as the LUCB (Lower and Upper Confidence Bound) algorithm.

5.1. Non-structured LUCB

The LUCB algorithm is a standard approach for the PAC problem in stochastic multi-armed bandits, specifically for BAI. The main idea is to construct and iteratively refine upper and lower confidence bounds around each arm's empirical mean (Kalyanakrishnan et al., 2012) until they are separated.

At each round t, for each arm a, we compute a confidence bonus (CB) $\beta_a^{\text{No-St}}(n, t)$. Over time, $\beta_a^{\text{No-St}}(n, t)$ shrinks according to a concentration inequality (e.g., Hoeffding or Bernstein). Let $\hat{\mu}_a(t) := \hat{P}_a(t) \cdot V$ be the empirical estimate of arm a's expectation at time t. For the current best arm a and the current second best arm b, we construct lower confidence bound and upper confidence bound respectively. The algorithm stops when

$$\hat{\mu}_a(t) - \hat{\mu}_b(t) - \left[\beta_a^{\text{No-St}}(n, t) + \beta_b^{\text{No-St}}(n, t)\right] \geq \epsilon, \tag{7}$$

where $\beta_i^{\text{No-St}}\left(n_i^t, t\right) = \sqrt{\frac{\log\left(\frac{2Kt}{\delta}\right)}{2\, n_i(t)}}$ is derived from Hoeffding's inequality. We call this approach Non-structured because the algorithm ignores the vector V.

6. Structured System Model

We now consider the model introduced in Section 4 as a K-action bandit problem where rewards are drawn i.i.d. from *multinomial* distributions $P_a, a \in \mathcal{A}$ and $V = [v_1, v_2, \ldots, v_d]$ is the support. We rely here directly on estimates of all d components of the probability vector P_a. We construct the empirical distribution as a vector

$\hat{P}_{A_t} = \frac{1}{n_{A_t}} \sum_{i=1}^{n_{A_t}} \delta_{Z_i}$ where each component is given by $\hat{P}_{A_t,i} = \frac{1}{n_{A_t}} \sum_{k=1}^{n_{A_t}} \mathbb{I}(Z_{A_t,k} = e_i)$ with $\mathbb{I}$ being the indicator function. We keep the Assumption 1 to facilitate the comparison of previously non-structured approach with the proposed structured cases. We address a more general case involving bounded support probabilities and heavy-tail distributions, as introduced and analyzed in (Agrawal et al., 2020). While SPEF distributions in (Garivier & Kaufmann, 2016) allow inner minimization of (6) in Euclidean space, introducing a probability vector confines the problem to the simplex. Agrawal et al. (Agrawal et al., 2020) address this by using functions $\mathrm{KL}_{\inf}^{\mathrm{L}}$ and $\mathrm{KL}_{\inf}^{\mathrm{U}}$, which measure the minimal KL-divergence required to distinguish a distribution η from alternatives with means below or above a threshold x. Specifically, for distributions κ_1, κ_2 over a finite set with mean $m(\kappa)$, define

$$\mathrm{KL}_{\inf}^{\mathrm{U}}(\eta, x) := \min_{\substack{\kappa \in \mathcal{L} \\ m(\kappa) \geq x}} \mathrm{KL}(\eta, \kappa),$$

with a similar definition for $\mathrm{KL}_{\inf}^{\mathrm{L}}(\eta, x)$. Inspired by (Honda & Takemura, 2010) in regret minimization setup, Agrawal et al. propose a Lagrangian dual problem to overcome the technical challenge of lower bounding sample complexity in this simplex-based model and proved (5) with new definition of inner minimization of $T^*(\mathcal{P})$ using $\mathrm{KL}_{\inf}^{\mathrm{L}}$ and $\mathrm{KL}_{\inf}^{\mathrm{U}}$. In (Agrawal et al., 2020), a modified version of T&S is proposed.

The computational complexity issue is further exacerbated in the modified T&S algorithm, where the complexity increases due to the need to solve an optimization problem involving two $\mathrm{KL}_{\inf}$ terms. Additionally, incorporating the effect of V when transitioning from Track-and-Stop to modified Track-and-Stop is not straightforward. In the latter, the support vector independently influences the Lagrangian problem, making it difficult to unify and clearly reflect the impact of V. These challenges motivate us to explore the Structured-LUCB algorithm practically, where the influence of V on the algorithm becomes more transparent to analyze.

6.1. Structured-LUCB

The Structured-LUCB algorithm constructs confidence bounds for each d component of the probability vectors and combines them to compute confidence intervals for expected rewards. Each $p_{a,i}$ is estimated independently. The algorithm applies larger thresholds in decision-making for components with v_i that are more likely to occur, prioritizing areas of higher uncertainty. Algorithm 1 provides the schema for the Structured-LUCB algorithm. At time t, let $\hat{p}_{k,i}(t)$ denote the empirical estimate for the probability of arm k producing outcome v_i, and let $\beta_{k,i}^{\mathrm{St}}(n,t)$ quantify the uncertainty of this estimate based on n samples. Following the LUCB framework, we require a lower bound for the best arm a and an upper bound for the current second-best arm b on the i-th outcome:

$$p_{a,i} \geq \hat{p}_{a,i}(t) - \beta_{a,i}^{\mathrm{St}}(n,t), \tag{8}$$

$$p_{b,i} \leq \hat{p}_{b,i}(t) + \beta_{b,i}^{\mathrm{St}}(n,t), \tag{9}$$

where the CB $\beta_{k,i}^{\mathrm{St}}(n,t)$ is derived using either Hoeffding's inequality:

$$\beta_{k,i}^{\text{Str-H}}(n,t) = \sqrt{\frac{\log(\frac{2dKt}{\delta})}{2n_k(t)}}, \tag{10}$$

or Bernstein's inequality, which incorporates the variance for tighter bounds:

$$\beta_{k,i}^{\text{ST-B}}(t) = \sqrt{2\hat{\sigma}_{k,i,t}^2} \sqrt{\frac{\ln\left(\frac{2dKt}{\delta}\right)}{2n_k}} + \frac{\ln\left(\frac{2dKt}{\delta}\right)}{3n_k}, \tag{11}$$

where $\hat{\sigma}_{k,i,t}^2$ denotes the variance at time t , calculated as $\hat{\sigma}_{k,i,t}^2 = \hat{p}_{k,i}(t)(1 - \hat{p}_{k,i}(t))$, providing confidence bonuses (CBs) that depend more closely on the observed variance.

The lower and upper confidence bounds for the expected rewards of arms a and b are given by :

$$\mathrm{LCB}_a^{\mathrm{str}}(t) = \sum_{i=1}^{d} \left(\hat{p}_{a,i}(t) - \beta_{a,i}^{\mathrm{St}}(n,t)\right) v_i, \tag{12}$$

$$\mathrm{UCB}_b^{\mathrm{str}}(t) = \sum_{i=1}^{d} \left(\hat{p}_{b,i}(t) + \beta_{b,i}^{\mathrm{St}}(n,t)\right) v_i. \tag{13}$$

The algorithm stops when $\mathrm{LCB}^{\mathrm{str}}$ exceeds $\mathrm{UCB}^{\mathrm{str}}$, ensuring that the expected reward of arm a is sufficiently higher than that of arm b with high confidence. Since the confidence bounds are uniform across all components of each probability vector, the stopping condition is simplified to:

$$\left(\hat{P}_a(t) - \hat{P}_b(t)\right) \cdot V - \left(\beta_{a,i}^{\mathrm{St}}(n,t) + \beta_{b,i}^{\mathrm{St}}(n,t)\right) \cdot |V| \geq \epsilon, \tag{14}$$

where $|V|$ is L_1-norm of vector V. Next in this section, we look at a joint estimation of d components of the probability vectors.

6.2. Structured Model 2: EL-LUCB Method

The problem introduced in (3) can be interpreted as a maximization of an unknown probability vector P_a in the direction of a known vector V. While the Structured-LUCB algorithm considers this problem as an estimation of all component independently, we suggest here to think of a joint estimation within a KL-*ball*, that reminiscent of the approach in (Filippi et al., 2010) for a regret minimization problem. In previous Structured-LUCB, the $\mathrm{Dist}(\cdot,\cdot)$ was

Algorithm 1 Structured-LUCB Algorithm with Leader-Challenger Sampling Strategy

1: **Input:** V , δ, $\alpha \in [0,1]$
2: **Output:** Optimal arm a^*
3: **Initialization:** $\forall a \in \mathcal{A}$, $n_a \leftarrow 1$, $\forall a \in [K]$, observing the reward Z_a and $\hat{P}_a \leftarrow Z_a$
4: **while** $\text{LCB}_l^{\text{str}} - \text{UCB}_c^{\text{str}}(t) < \epsilon$ **do**
5: **for** $a \in [K]$ **do**
6: Choose $\beta = \min(\beta_{k,i}^{\text{Str-H}}, \beta_{k,i}^{\text{ST-B}})$ based on (10)-(11)
7: Construct $(\text{LCB}_a^{\text{str}}, \text{UCB}_a^{\text{str}})$, $\forall a \in \mathcal{A}$
8: **end for**
9: $l \leftarrow \arg\max_a \text{LCB}_a^{\text{str}}$
10: $c \leftarrow \arg\max_{a \neq a_{\text{leader}}} \text{UCB}_a^{\text{str}}$
11: Assign $a_t \leftarrow \begin{cases} l & \text{if } X = 1 \sim \text{Bernoulli}(\alpha), \\ c & \text{otherwise.} \end{cases}$
12: **Observe:** Reward Z_{a_t} from pulling arm a_t
13: **update the parameters**
14: **end while**

defined as an L-norm while here we use $\text{Dist}(\hat{P}_a, P_a) := \text{KL}(\hat{P}_a, P_a) \leq \epsilon$.

The primary modification, compared to Algorithm 1, appears on the seventh line, where the construction of the LCBs and UCBs is specified. At each iteration, given the updated empirical distributions of the leader arm $\hat{P}_a$ and the challenger arm $\hat{P}_a$, we apply Algorithm 2 from (Filippi et al., 2010) to obtain respectively a lower bound on $\hat{P}_a(t) \cdot V$ and an upper bound on $\hat{P}_b(t) \cdot V$.

At each iteration of the loop, the updated KL−ball, or more specifically the KL-ellipses, around the estimates shrinks until there is no overlap, indicating that the estimates have reached a desired precision. The Algorithm 2As of (Filippi et al., 2010) is solved using a Lagrangian multiplier approach.

7. Experiments and Discussions

In this section, we explore our proposed algorithms for different cases and explain the different results obtained. We focus on three algorithms: Non-structured LUCB (or simply LUCB with Assumption 1), Structured-LUCB presented in Alg 1 and the EL-LUCB algorithm explained in Subsection 6.2.

The results are averaged on 100 trials. The confidence parameter is set to $\delta = 0.05$, the sampling probabilities of leader-challenger are initialized at $[0.5, 0.5]$. The reward of each arm is drawn according to a row of the matrix P. Its columns contain the probabilities of each outcome of V. We evaluate their performance on different support vectors V.

7.1. Structured-LUCB vs. Non-structured-LUCB

In the situations where the outcome probabilities are highly concentrated on a single outcome of V for each arm, the structured algorithm does not offer significant advantages over the non-structured algorithm. Both algorithms will perform similarly and non-structured algorithm can quickly and accurately estimate the expected rewards based on observed averages with less complex implementation effort. We hence consider the following cases to determine the suitability of each algorithm under varying conditions:

$$P^{test1} = \begin{bmatrix} 0.5 & 0.3 & 0.2 \\ 0.4 & 0.3 & 0.3 \\ 0.3 & 0.2 & 0.5 \end{bmatrix}, \qquad V^{\text{test1}} = [0.5, 0.1, 0].$$

Figure 2 shows that for V^{test1}, the Structured-LUCB algorithm significantly outperforms Non-structured-LUCB.

For the same P, we change the support to $V_{\text{test2}} = [0.9, 0.6, 0.4]$, as shown in Figure 3, Structured-LUCB demonstrates less efficient performance compared to Non-structured-LUCB.

The reasoning lies in how the stopping time and the effect of V are introduced in the algorithms. In Non-structured-

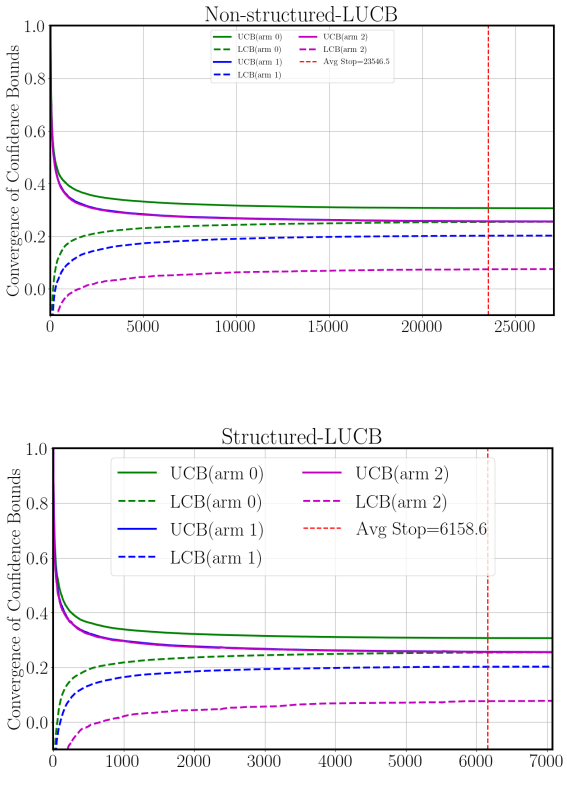

Figure 2. Comparing the stopping times of two algorithms on V^{test1}

LUCB, the summation of CBs for Bernoulli distributions in (7) is not effected by V. However, in the stopping condition of the Structured-LUCB algorithm (14), V plays a significant role. While the Hoeffding's CB is scaled directly by V, the Bernstein's CB is more affected by the features of V. Here, we provide the summation of Bernstein's CBs for two arms in terms of Hoeffding's CBs:

$$\Delta^{\text{St-B}} := \Big(\frac{\ln\left(\frac{2dKt}{\delta}\right)}{3n_a^t} + \frac{\ln\left(\frac{2dKt}{\delta}\right)}{3n_b^t}\Big).(\sum_{i=1}^{d} v_i) + \sqrt{2\sum_{i=1}^{d}\left(\sigma_{a,i}\beta_{a,i}^{\text{ST-Hf}} + \sigma_{b,i}\beta_{b,i}^{\text{ST-Hf}}\right)^2} \cdot ||V||. \quad (15)$$

In the structured case, the vector V is integrated into the stopping condition, making the algorithm sensitive to both individual outcomes v_i and the l_1-norm of V. This sensitivity is particularly evident when using Hoeffding's bound and becomes more nuanced with Bernstein's bound. By comparing the structured CB to the non-structured CB, we can draw the following insights. When $|V| \leq 1$, the Structured-LUCB stops earlier because its CBs shrink more rapidly.

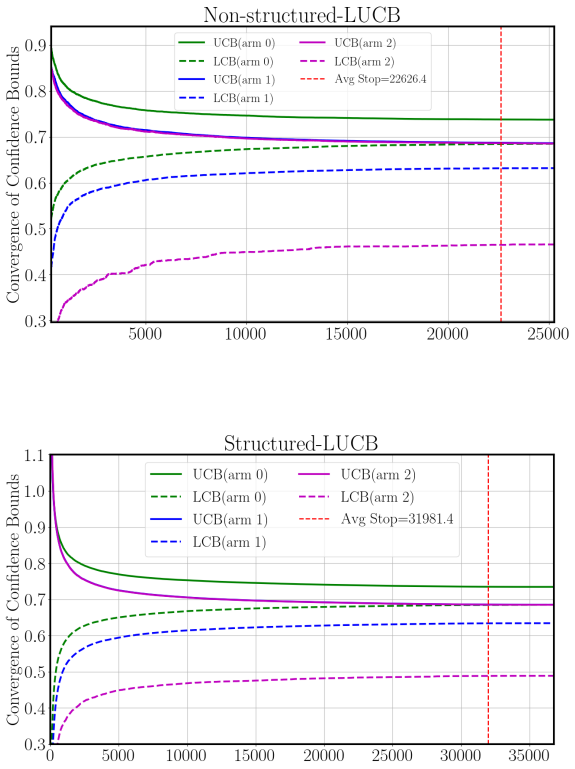

Figure 3. Comparing the stopping times of two algorithms on V^{test2}.

This accelerated reduction is due to the CB being scaled by $|V|$. In contrast, when $|V| \geq 1$ the Non-structured-LUCB algorithm becomes the preferred choice. However, if the number of states d is very large ($d \gg 1$), it may offset the advantage of having $|V| \leq 1$ in the structured scenario.

The EL-LUCB Method LUCB-based algorithms have two main procedures: updating empirical distributions and constructing CBs. Their dependence on V is trackable because each step's contribution can be isolated. In contrast, the EL-LUCB method directly builds upper and lower bounds at each iteration rather than separately constructing CBs, making the role of V less transparent. We illustrate V's impact through various examples.

First, we run the EL-LUCB (EL) algorithm for previous cases. Interestingly, it remains robust across both supports, showing consistent performance despite changes in the distribution's support. It outperforms both Structured-LUCB and Non-structured-LUCB:

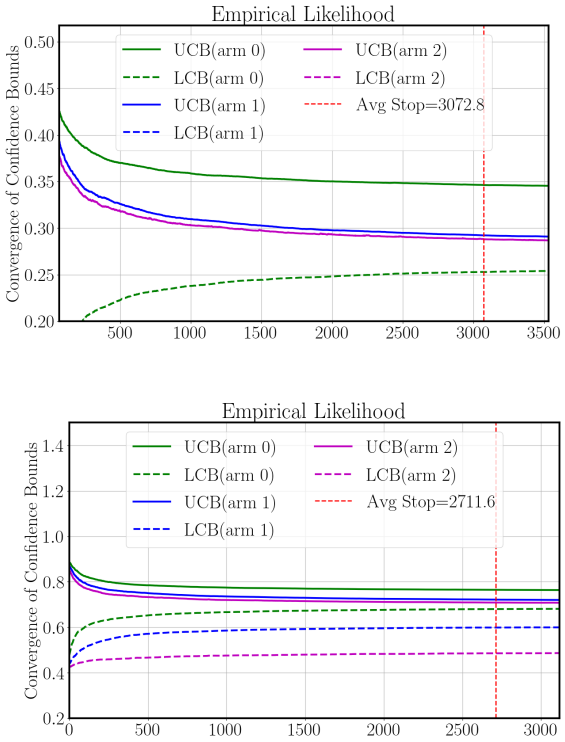

Figure 4. Comparing the stopping times of EL-LUCB algorithm on V^{test1} (above) and V^{test2} (below).

In these examples, EL-LUCB works on a same range $\text{Range}(V) = \max(V) - \min(V)$ on the support and seems the condition $\text{KL}(\cdot,\cdot) \leq \epsilon$ as we can see does not directly by each component of V. Consequently, we propose two cases

where the distinguishability of two arms remains close and we show how the performance is affected by the range of V. We consider

$$P^{\text{test2}} = \begin{bmatrix} 0.142 & 0.311 & 0.153 & 0.391 \\ 0.386 & 0.114 & 0.154 & 0.344 \end{bmatrix},$$

$$V^{\text{test3}} = \left[0.144, 0.152, 0.505, 0.984\right],$$

$$V^{\text{test4}} = \left[0.573, 0.518, 0.409, 0.505\right],$$

where the range of V changes from $\Delta V^{\text{test3}} = 0.84$ to $\Delta V^{\text{test4}} = 0.164$.

We tailored these two cases so that the expected rewards of two arms remain close, yielding $\mathbb{E}_{P[0]}[V^{\text{test3}}] - \mathbb{E}_{P[1]}[V^{\text{test3}}] = 0.04$ and $\mathbb{E}_{P[0]}[V^{\text{test4}}] - \mathbb{E}_{P[1]}[V^{\text{test4}}] = 0.014$.

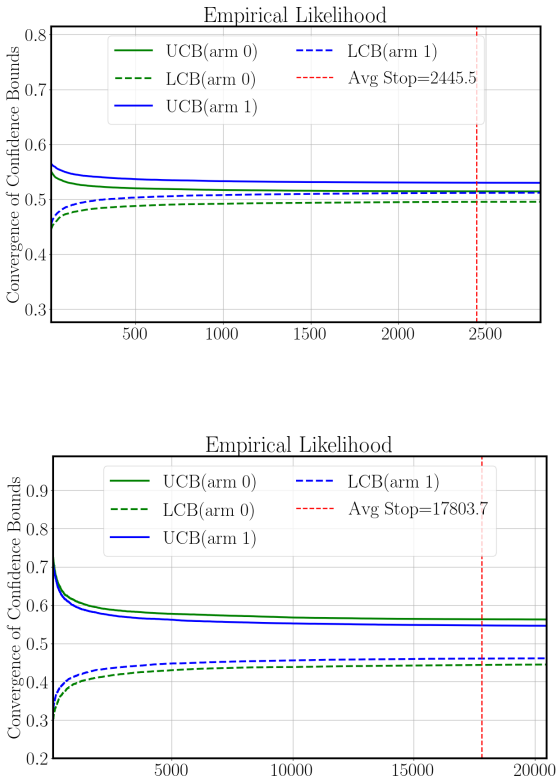

Figure 5. Comparing the stopping times of EL-LUCB algorithm on V^{test3} with low range (above) and with V^{test4} high range (below)

It appears that EL-LUCB 's stopping time is lower than the previous approaches in both cases, at the price of a somewhat higher computational complexity for solving several Lagrangian problems.

8. Conclusion

We presented and compared two multi-armed bandit (MAB) frameworks to tackle the choice of a transition in a finite horizon Markov decision process with known future values. Experimental results compared stopping times and the impact of leveraging distributional support across various scenarios, illustrating that incorporating structure – when available – may significantly improve performance and decision-making efficiency. The quantitative amplitude of this gain remains to be better understood from a theoretical point of view.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Agrawal, S., Juneja, S., and Glynn, P. Optimal δ-correct best-arm selection for heavy-tailed distributions. In *Algorithmic Learning Theory*, pp. 61–110. PMLR, 2020.

Audibert, J.-Y. and Bubeck, S. Best arm identification in multi-armed bandits. In *COLT-23th Conference on learning theory-2010*, pp. 13–p, 2010.

Bertsekas, D. P. *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, MA, 3rd edition, 2005. ISBN 978-1886529267.

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

Cappé, O., Garivier, A., Maillard, O.-A., Munos, R., and Stoltz, G. Kullback-leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics*, pp. 1516–1541, 2013.

Chen, S., Hu, X., Zhao, J., Wang, R., and Qiao, M. A review of decision-making and planning for autonomous vehicles in intersection environments. *World Electric Vehicle Journal*, 15(3), 2024. ISSN 2032-6653. doi: 10.3390/wevj15030099. URL https://www.mdpi.com/2032-6653/15/3/99.

Filippi, S., Cappé, O., and Garivier, A. Optimism in reinforcement learning and kullback-leibler divergence. In

2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 115–122, 2010. doi: 10.1109/ALLERTON.2010.5706896.

Gabillon, V., Ghavamzadeh, M., and Lazaric, A. Best arm identification: A unified approach to fixed budget and fixed confidence. *Advances in Neural Information Processing Systems*, 25, 2012.

Garivier, A. and Kaufmann, E. Optimal best arm identification with fixed confidence. In Feldman, V., Rakhlin, A., and Shamir, O. (eds.), *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pp. 998–1027, Columbia University, New York, New York, USA, June 2016. PMLR. URL https://proceedings.mlr.press/v49/garivier16a.html.

Heidrich-Meisner, V. and Igel, C. Hoeffding and bernstein races for selecting policies in evolutionary direct policy search. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 401–408, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553426. URL https://doi.org/10.1145/1553374.1553426.

Honda, J. and Takemura, A. An asymptotically optimal bandit algorithm for bounded support models. In *Annual Conference Computational Learning Theory*, 2010. URL https://api.semanticscholar.org/CorpusID:120162138.

Jourdan, M., Degenne, R., Baudry, D., de Heide, R., and Kaufmann, E. Top two algorithms revisited. *Advances in Neural Information Processing Systems*, 35:26791–26803, 2022.

Kalyanakrishnan, S., Tewari, A., Auer, P., and Stone, P. Pac subset selection in stochastic multi-armed bandits. In *International Conference on Machine Learning*, 2012. URL https://api.semanticscholar.org/CorpusID:1635758.

Kaufmann, E. and Kalyanakrishnan, S. Information complexity in bandit subset selection. In *Conference on Learning Theory*, pp. 228–251. PMLR, 2013.

Lopes, J. C. and Lopes, R. P. A review of dynamic difficulty adjustment methods for serious games. In Pereira, A. I., Košir, A., Fernandes, F. P., Pacheco, M. F., Teixeira, J. P., and Lopes, R. P. (eds.), *Optimization, Learning Algorithms and Applications*, pp. 144–159, Cham, 2022. Springer International Publishing. ISBN 978-3-031-23236-7.

Lu, T., Pál, D., and Pál, M. Contextual multi-armed bandits. In *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, pp. 485–492. JMLR Workshop and Conference Proceedings, 2010.

Mnih, V., Szepesvári, C., and Audibert, J.-Y. Empirical bernstein stopping. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pp. 672–679, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390241. URL https://doi.org/10.1145/1390156.1390241.

Moerland, T. M., Broekens, J., Plaat, A., and Jonker, C. M. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1):1–118, 2023a. ISSN 1935-8237. doi: 10.1561/2200000086. URL http://dx.doi.org/10.1561/2200000086.

Moerland, T. M., Broekens, J., Plaat, A., Jonker, C. M., et al. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1): 1–118, 2023b.

Neu, G., Olkhovskaya, J., and Vakili, S. Adversarial contextual bandits go kernelized. In Vernade, C. and Hsu, D. (eds.), *Proceedings of The 35th International Conference on Algorithmic Learning Theory*, volume 237 of *Proceedings of Machine Learning Research*, pp. 907–929. PMLR, 25–28 Feb 2024. URL https://proceedings.mlr.press/v237/neu24a.html.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New York, 1994. ISBN 978-0471727828.

Russo, D. Simple bayesian algorithms for best arm identification. In *Conference on Learning Theory*, pp. 1417–1418. PMLR, 2016.

Saber, H. and Maillard, O.-A. Bandits with multimodal structure. In *Reinforcement Learning Conference*, volume 1, pp. 39, 2024.

Sutton, R. S. Reinforcement learning: An introduction. *A Bradford Book*, 2018.

You, W., Qin, C., Wang, Z., and Yang, S. Information-directed selection for top-two algorithms. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 2850–2851. PMLR, 2023.