Adaptive Sampling for Best Policy Identification in Markov Decision Processes

Aymen Al Marjani [1] **Alexandre Proutiere** [2]

Abstract

We investigate the problem of best-policy identification in discounted Markov Decision Processes (MDPs) when the learner has access to a generative model. The objective is to devise a learning algorithm returning the best policy as early as possible. We first derive a problem-specific lower bound of the sample complexity satisfied by any learning algorithm. This lower bound corresponds to an optimal sample allocation that solves a non-convex program, and hence, is hard to exploit in the design of efficient algorithms. We then provide a simple and tight upper bound of the sample complexity lower bound, whose corresponding nearly-optimal sample allocation becomes explicit. The upper bound depends on specific functionals of the MDP such as the sub-optimality gaps and the variance of the next-state value function, and thus really captures the hardness of the MDP. Finally, we devise KLB-TS (KL Ball Track-and-Stop), an algorithm tracking this nearly-optimal allocation, and provide asymptotic guarantees for its sample complexity (both almost surely and in expectation). The advantages of KLB-TS against state-of-the-art algorithms are discussed and illustrated numerically.

1. Introduction

Reinforcement Learning (RL) algorithms are designed to interact with an unknown stochastic dynamical system, and through this interaction, to identify, as fast as possible, an optimal control policy. The efficiency of these algorithms is usually measured through their *sample complexity*, defined as the number of samples (the number of times the algorithm interacts with the system) required to identify an optimal policy with some prescribed levels of accuracy and certainty. This paper, as most related work in this field, focuses on systems and control objectives that are modelled as a standard discounted Markov Decision Processes (MDPs) with finite state and action spaces. Various interaction models have been investigated, but sample complexity analyses have been mainly conducted under the so-called *generative model*, where in each step, the algorithm may sample a transition and a reward from any given (state, action) pair. We also restrict our attention to this model.

We investigate the design of RL algorithms with minimal sample complexity. This problem has attracted a lot of attention over the last two decades. Most studies follow a minimax approach. For example, it is known (Azar et al., 2013) that for the worst possible MDP, identifying an ε-optimal policy with probability $1-\delta$ requires at least $\frac{SA}{\varepsilon^2(1-\gamma)^3}\log(\frac{SA}{\delta})$ samples, where S and A are the number of states and actions, respectively, and γ is the discount factor. Note that to obtain this sample complexity lower bound, one needs to design a very specific worst-case MDP (in particular, its transition probabilities must depend on ε and γ). Since the aforementioned minimax lower bound appeared, most researchers have been aiming at devising algorithms matching this bound. In contrast, we are interested in analyzing the minimal *problem-specific* sample complexity. Specifically, we seek to understand the dependence of the sample complexity on the MDP that has to be learnt. Problem-specific performance metrics are much more informative than their minimax counterparts, because they encode and express the inherent hardness of the MDP. Minimax metrics just represent the hardness of the worst MDP. In particular, establishing that the sample complexity of an algorithm does not exceed the minimax lower bound just reveals that the algorithm performs well for this worst MDP. However, it does not indicate whether the algorithm *adapts* to the hardness of the MDP, i.e., whether the optimal policy of a very easy MDP would be learnt very quickly. As a matter of fact, an algorithm with sample complexity matching the minimax lower bound just consists in sampling (state, action) pairs uniformly at random, and is not adapting to the MDP.

The problem-specific sample complexity of identifying the best arm in stochastic Multi-Armed Bandit (MAB) prob-

[1] UMPA, ENS Lyon. This work was done while Aymen Al Marjani was at KTH. [2] KTH Royal institute of Technology. Correspondence to: Aymen Al Marjani <aymen.al_marjani@ens-lyon.fr>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

lems is now well understood (Garivier & Kaufmann, 2016). In this work, we explore whether the methodology used in (Garivier & Kaufmann, 2016) for MAB problems can be extended to RL problems. This methodology consists in first deriving a problem-specific sample complexity lower bound which should reveal the sample allocation leading to the minimal sample complexity. One may then devise a *track-and-stop* algorithm that (i) tracks the optimal sample allocation identified in the lower bound, and (ii) stops when the information gathered is judged sufficient to get the desired PAC guarantees. As it turns out, extending this methodology to RL problems raises fundamental issues, mainly due to the difficulty of computing the sample allocation leading to the minimal problem-specific sample complexity. We propose a set of tools to solve these issues. Our contributions are as follows:

1. We derive a problem-specific sample complexity lower bound for identifying an optimal policy in a given MDP ϕ. This bound is expressed as $T^*(\phi)\log(1/\delta)$, where the *characteristic time* $T^*(\phi)$ encodes the hardness of the MDP ϕ. $T^*(\phi)$ is the value of a complex non-convex optimization problem. This complexity makes the design of a track-and-stop algorithm similar to that proposed in (Garivier & Kaufmann, 2016) and achieving the sample complexity lower bound elusive. To circumvent this difficulty, we derive an explicit upper bound $U(\phi)$ of $T^*(\phi)$. The advantage of $U(\phi)$ is two-fold: (i) $U(\phi)$ remains problem-specific, and explicitly depends on functionals of the MDP characterizing its hardness. (ii) $U(\phi)$ corresponds to an explicit and simple sample allocation. This allows us to devise a procedure that tracks this allocation.

2. Based on our upper bound analysis, we devise KLB-TS (KL Ball Track-and-Stop), an algorithm whose sample complexity is at most $U(\phi)\log(1/\delta)$. Our algorithm relies on a procedure tracking the sample allocation leading to $U(\phi)$, and a stopping rule that we refer to as KL Ball Stopping rule because of its analogy to the way we derive the upper bound $U(\phi)$.

3. We highlight the differences of our design approach compared to that leading to BESPOKE (Zanette et al., 2019), a recently proposed adaptive algorithm. As it turns out, the adaptive part of BESPOKE is very limited in practice (see related work and Appendix H for details), and KLB-TS exhibits a much better performance numerically.

2. Related Work

Most work on the best policy identification in MDPs adopt a minimax approach (Kearns & Singh, 1999), (Kakade, 2003), (Even-Dar et al., 2006), (Azar et al., 2013), (Sidford et al., 2018), (Agarwal et al., 2020), (Li et al., 2020). In the most recent of these papers (Li et al., 2020), the authors propose an algorithm whose sample complexity achieves the minimax lower bound of (Azar et al., 2013) for a wide range of values of ε, namely for $\varepsilon \in (0, \frac{1}{1-\gamma}]$. Refer to the appendix for a detailed account on the minimax framework.

As far as we are aware, the only paper attempting to propose a problem-specific analysis of the best policy identification in MDPs with a generative model is (Zanette et al., 2019). There, the authors proposed BESPOKE, an adaptive algorithm designed to find ε-optimal policies. BESPOKE starts by allocating an extremely large number of samples $n_{\min} = \frac{2\times 625^2\times\gamma^2\times S\times\log(1/\delta)}{(1-\gamma)^2}$ to each (state, action) pair. Then, at each iteration, BESPOKE solves a convex program whose objective is an upper-bound of the sub-optimality gap (in terms of the ℓ_∞-norm of the value function) of the empirical optimal policy. The solution of this program corresponds to the sampling strategy that the algorithm uses to halve the sub-optimality gap of the empirical policy in the next iteration. Interestingly, BESPOKE is the first algorithm with a problem-dependent sample complexity upper-bound. Note however that BESPOKE has not been tested numerically in (Zanette et al., 2019); we fill this gap in this paper. Because of its very long initialization phase, it turns out that the part where BESPOKE actually adapts its sample allocation is negligible in comparison of its total sample complexity. In Appendix H, we provide a more detailed discussion on BESPOKE, and further compare the sample complexity upper bounds of KLB-TS and BESPOKE. Experiments in Section 7 show that KLB-TS significantly outperforms BESPOKE numerically.

3. Preliminaries and Notation

3.1. Discounted MDPs

We investigate the optimal control of dynamical systems modelled as an infinite time-horizon MDP with finite state space $\mathcal{S}$ and finite action spaces $\mathcal{A}_s$ for any $s \in \mathcal{S}$. Let $\mathcal{A} = \cup_{s\in\mathcal{S}}\mathcal{A}_s$. The MDP is defined by its kernels: $\phi = (p_\phi, q_\phi)$, where p_ϕ captures the system dynamics and q_ϕ the random collected rewards. Specifically, $p_\phi(s'|s,a)$ denotes the probability of the system to be in state s' after taking the action $a \in \mathcal{A}_s$ in state s. Let $p_\phi(s,a) = (p_\phi(s'|s,a))_{s'}$. $q_\phi(\cdot|s,a)$ or simply $q_\phi(s,a)$ is the density of the distribution of the reward collected in state s when action a is selected, w.r.t. some positive measure λ with support included in $[0,1]$. Let $r_\phi(s,a)$ denote the expected reward collected in state s when action a is selected, $r_\phi(s,a) = \int_0^1 Rq_\phi(R|s,a)\lambda(dR)$.

The objective is to identify a control policy $\pi : \mathcal{S} \to \mathcal{A}$ maximizing the long-term discounted reward $\mathbb{E}_\phi[\sum_{t=0}^\infty \gamma^t r_\phi(s^\pi(t), \pi(s^\pi(t)))]$, where $s^\pi(t)$ is the state of the system at time t under the policy π and $\mathbb{E}_\phi[\cdot]$ represents the expectation taken w.r.t. to the randomness induced by

(p_ϕ, q_ϕ).

We denote by V_ϕ^π the value function of the MDP ϕ when the control policy is π: for any s, $V_\phi^\pi(s) = \mathbb{E}_\phi[\sum_{t=0}^\infty \gamma^t r_\phi(s^\pi(t), \pi(s^\pi(t))|s^\pi(0) = s]$. $V_\phi^\star$ corresponds to the value function when the policy π is optimal. Note that since the rewards are lower and upper bounded by 0 and 1, respectively, we have for any s, $V_\phi^\star(s) \in [0, \frac{1}{1-\gamma}]$. Similarly, the Q-function is denoted by Q_ϕ^π, and $Q_\phi^\star$ when π is optimal. The sub-optimality gap of action a in state s is defined as $\Delta_{sa} = V_\phi^\star(s) - Q_\phi^\star(s,a)$. Finally, denote by $\Pi_\phi^\star$ the set of optimal policies for ϕ.

Assumption 1. To simplify notation and the analysis, we assume that ϕ admits a unique optimal control policy denoted by $\pi_\phi^\star$. This means that $\phi \in \Phi$, where Φ is defined as $\Phi = \{\phi : |\Pi_\phi^\star| = 1\}$.

3.2. Best-policy identification

We aim at devising an algorithm identifying $\pi_\phi^\star$ as quickly as possible in the fixed-confidence setting: when the algorithm stops and returns an estimated optimal policy $\hat{\pi}$, we should have $\mathbb{P}_\phi[\hat{\pi} \neq \pi_\phi^\star] \leq \delta$, for some pre-defined confidence parameter $\delta > 0$. Such an algorithm consists of a sampling rule, a stopping rule, and a decision rule. An algorithm χ gathers information sequentially, and we denote by $\mathcal{F}_t^\chi$ the σ-algebra generated by all observations made under χ up to and including round t.

Sampling rule. In round t, the algorithm χ selects a (state, action) pair (s_t, a_t) to explore, depending on past observations. (s_t, a_t) is $\mathcal{F}_{t-1}^\chi$-measurable. χ observes the next state denoted by s_t' and a random reward R_t. Note that any admissible (state, action) pair may be selected (we consider a generative model).

Stopping and decision rules. After gathering enough information, χ may decide to stop sampling and to return an estimated best policy. The algorithm stops after collecting τ samples, and τ is a stopping time w.r.t. the filtration $(\mathcal{F}_t^\chi)_{t\geq 1}$. The estimated best policy $\hat{\pi}$ is then $\mathcal{F}_\tau^\chi$-measurable. τ is referred to as the sample complexity of χ.

δ-PC algorithms. An algorithm is δ-Probably Correct (δ-PC) if it satisfies the two following conditions: for any MDP $\phi \in \Phi$, (i) it stops in finite time almost surely, $\mathbb{P}_\phi[\tau < \infty] = 1$, and (ii) $\mathbb{P}_\phi[\hat{\pi} \neq \pi_\phi^\star] \leq \delta$.

3.3. Additional notation

$\mathbb{1}(s)$ denotes the canonical base vector in $\mathbb{R}^\mathcal{S}$ whose only non-zero entry is at index s. $\Sigma = \{\omega \in [0,1]^{S\times A} : \sum\limits_{s,a} w_{sa} = 1\}$ denotes the simplex in $\mathbb{R}^{S\times A}$. The Kullback-Leibler divergence between two probability distributions P and Q on some discrete space $\mathcal{S}$ is defined as: $KL(P\|Q) = \sum_{s\in\mathcal{S}} P(s)\log(\frac{P(s)}{Q(s)})$. For Bernoulli distributions of respective means p and q, the KL divergence is denoted by $\mathrm{kl}(p,q)$. For distributions over $\mathbb{R}$ defined through their densities p and q w.r.t. some positive measure λ, the KL divergence is: $KL(p\|q) = \int_{-\infty}^{\infty} p(x)\log\left(\frac{p(x)}{q(x)}\right)\lambda(dx)$. For two MDPs ϕ and ψ, we say that $\phi \ll \psi$ if for all (s,a), $p_\phi(\cdot|s,a) \ll p_\psi(\cdot|s,a)$ and $q_\phi(\cdot|s,a) \ll q_\psi(\cdot|s,a)$. In that case, we define $\mathrm{KL}_{\phi|\psi}(s,a)$ as the KL divergence between the distributions of the random observations made for the (state, action) pair (s,a) under ϕ and ψ:

$$\begin{aligned}\mathrm{KL}_{\phi|\psi}(s,a) = {} & KL(p_\phi(s,a)\|p_\psi(s,a)) \\ & + KL(q_\phi(s,a)\|q_\psi(s,a)).\end{aligned}$$

4. Problem-Specific Sample Complexity Lower Bound

To derive a problem-specific sample complexity lower bound, we use classical change-of-measure arguments as those leveraged towards regret and sample complexity lower bounds (Lai & Robbins, 1985; Garivier & Kaufmann, 2016) in bandit problems. These arguments lead to constraints on the expected numbers of times each (state, action) pair should be explored under any δ-PC algorithm.

Definition 1. *The set of alternative MDPs is defined as:* $\mathrm{Alt}(\phi) = \{\psi \text{ MDP} : \phi \ll \psi$ *and* $\Pi_\phi^\star \cap \Pi_\psi^\star = \emptyset\}$.

Let $\psi \in \mathrm{Alt}(\phi)$ be an alternative MDP and consider a δ-PC algorithm. We denote by O_τ the set of observations made under the algorithm until it stops. Further consider L_τ the log-likelihood ratio of O_τ under the MDPs ϕ and ψ. Using similar techniques as those used in the proof of Wald's first lemma, we get (all proofs are detailed in the appendix):

Lemma 1. *Let $n_t(s,a)$ be the number of times (s,a) has been explored up to and including step t. For any $\phi \in \Phi$,* $\mathbb{E}_\phi[L_\tau] = \sum_{s,a}\mathbb{E}_\phi[n_\tau(s,a)]\,\mathrm{KL}_{\phi|\psi}(s,a)$.

From the above lemma, and using the same arguments as in (Kaufmann et al., 2016), one may derive the following data processing inequality, valid for any $\mathcal{F}_\tau$-measurable event E:

$$\sum_{s,a}\mathbb{E}_\phi[n_\tau(s,a)]\,\mathrm{KL}_{\phi|\psi}(s,a) \geq \mathrm{kl}(\mathbb{P}_\phi[E], \mathbb{P}_\psi[E]).$$

Next, we select the event E as $\{\hat{\pi} \notin \Pi^\star(\phi)\}$. Since the algorithm is δ-PC, and since $\psi \in \mathrm{Alt}(\phi)$, we have: $\mathbb{P}_\phi[E] \leq \delta$ and $\mathbb{P}_\psi[E] \geq \mathbb{P}_\psi[\hat{\pi} \in \Pi^\star(\psi)] \geq 1-\delta$. Using the monotonicity of the KL divergence, we deduce that $\mathrm{kl}(\mathbb{P}_\phi[E], \mathbb{P}_\psi[E]) \geq \mathrm{kl}(\delta, 1-\delta)$. We have established that under any δ-PC algorithm, the numbers of times

$(n_\tau(s,a))_{s,a}$ the different (state, action) pairs are explored satisfy: for any MDP $\psi \in \mathrm{Alt}(\phi)$,

$$\sum_{s,a} \mathbb{E}_\phi[n_\tau(s,a)]\, \mathrm{KL}_{\phi|\psi}(s,a) \geq \mathrm{kl}(\delta, 1-\delta). \quad (1)$$

Combining the above constraints with the fact that $\tau = \sum_{s,a} n_\tau(s,a)$, we obtain the following sample complexity lower bound.

Proposition 1. *The sample complexity of any δ-PC algorithm satisfies: for any $\phi \in \Phi$,*

$$\mathbb{E}_\phi[\tau] \geq T^*(\phi)\mathrm{kl}(\delta, 1-\delta), \quad (2)$$

$$\textit{where } T^*(\phi)^{-1} = \sup_{\omega\in\Sigma} \inf_{\psi\in\mathrm{Alt}(\phi)} \sum_{s,a} \omega_{sa} KL_{\phi|\psi}(s,a). \quad (3)$$

In the above proposition, $\omega_{sa}\mathrm{kl}(\delta, 1-\delta)$ can be interpreted as the expected proportion of times the pair (s,a) is explored under the algorithm. Taking the supremum over ω then corresponds to selecting an optimal sampling rule. In the following, ω is referred to as the allocation vector.

Remark. *In (Ok et al., 2018), the authors identify a similar optimization problem as (3) leading to a problem-specific regret lower bound satisfied by online learning algorithms in generic structured MDP. Interestingly, we note that this optimization problem is simpler than (3). This is due to fact that when minimizing regret, confusing MDPs have the same transitions and rewards as in the original MDP at optimal (state, action) pairs. This considerably simplifies the analysis, and explains why, in general, deriving problem-specific sample complexity lower bound is much harder than obtaining regret lower bounds.*

4.1. Properties of the problem (3)

We now provide useful properties of the optimization problem (3). Additional properties of the problem are presented in Appendix B.

(i) The set of alternative MDPs. To simplify the notation we use $\pi^\star$ instead of $\pi^\star_\phi$. Our first result concerns the set $\mathrm{Alt}(\phi)$ of *alternative* MDPs:

Lemma 2. $\mathrm{Alt}(\phi) = \bigcup_{s,a\neq\pi^\star(s)} \mathrm{Alt}_{sa}(\phi)$ *where*

$$\mathrm{Alt}_{sa}(\phi) = \{\psi : Q^{\pi^\star}_\psi(s,a) > V^{\pi^\star}_\psi(s)\}.$$

The above lemma states that an alternative MDP ψ is such that $\pi^\star$, the optimal policy of ϕ, can be improved under ψ locally at some state s, by selecting in s some previously sub-optimal action a, instead of $\pi^\star(s)$. Using this lemma, we can simplify the expression of the characteristic time appearing in Proposition 1. Indeed, (3) is equivalent to:

$$\sup_{\omega\in\Sigma} \min_{s,a\neq\pi^\star(s)} \inf_{\psi\in\mathrm{Alt}_{sa}(\phi)} \sum_{s',a'} \omega_{s',a'} \mathrm{KL}_{\phi|\psi}(s',a'). \quad (4)$$

Next, we rewrite the problem in an analytic manner. To this aim, we parametrize ψ by its transition probabilities and rewards $u = (q_\psi(s,a), p_\psi(s,a))_{s,a\in\mathcal{S}\times\mathcal{A}}$ and introduce the following notations: for all (s,a), $dr(s,a) = (r_\psi - r_\phi)(s,a)$ and $dp(s,a) = (p_\psi - p_\phi)(s,a)$. Further define $dV^{\pi^\star} = \Big([V^{\pi^\star}_\psi - V^{\pi^\star}_\phi](s)\Big)_{s\in\mathcal{S}}$.

Combining the condition : $Q^{\pi^\star}_\psi(s,a) > V^{\pi^\star}_\psi(s)$ with the fact that $Q^{\pi^\star}_\phi(s,a) + \Delta_{sa} = V^{\pi^\star}_\phi(s)$ we obtain that $\psi \in \mathrm{Alt}_{sa}(\phi)$ if and only if:

$$\begin{aligned}\Delta_{sa} < dr(s,a) + \gamma dp(s,a)^\top V^{\pi^\star}_\phi \\ + [\gamma p_\psi(s,a) - \mathbb{1}(s)]^\top dV^{\pi^\star}. \quad (5)\end{aligned}$$

The above inequality states that for ψ to be in $\mathrm{Alt}_{sa}(\phi)$, the changes in the rewards and transitions between ϕ and ψ should be greater than the sub-optimality gap of action a in state s. Defining $\mathcal{U}_{sa} = \{u : (5) \text{ holds}\}$, we conclude that both the optimization problems (3) and (4) are equivalent to:

$$\sup_{\omega\in\Sigma} \min_{s,a\neq\pi^\star(s)} \inf_{u\in\mathcal{U}_{sa}} \sum_{s',a'} \omega_{s',a'} \mathrm{KL}_{\phi|\psi}(s',a'). \quad (6)$$

(ii) Non-convexity of the problem (3). The characteristic time $T^*(\phi)$, as well as the optimal sampling rule are characterized by the solution of (3) or that of (4). If we think of a track-and-stop algorithm to identify the best policy (as proposed in (Garivier & Kaufmann, 2016) for the simple MAB problem), one would need to repeatedly solve these optimization problems. It is then important to be able to do it in a computationally efficient way. Unfortunately, these problems are probably very hard to solve. This is well illustrated by the fact that the following sub-problem is not convex:

$$T(\phi,\omega)^{-1} = \inf_{\psi\in\mathrm{Alt}(\phi)} \sum_{s,a} \omega_{sa} \mathrm{KL}_{\phi|\psi}(s,a). \quad (7)$$

Actually, in the example presented in Fig. 1, we can specify ϕ such that the sets $\mathrm{Alt}(\phi)$ and $\mathrm{Alt}_{sa}(\phi)$ are not convex.

Consider $\phi, \psi, \overline{\psi}$ belonging to the class of MDPs specified in Fig. 1, each defined by the vector (r_2, r_1, p_1) (all other parameters values are fixed as in the figure):

$$\begin{cases}\psi = (r_2 = 0.25,\ r_1 = 0.93,\ p_1 = 0.7) \\ \overline{\psi} = (r_2 = 0.1,\ r_1 = 0.47,\ p_1 = 0.6) \\ \phi = \frac{\psi+\overline{\psi}}{2} = (r_2 = 0.175,\ r_1 = 0.6925,\ p_1 = 0.65)\end{cases}$$

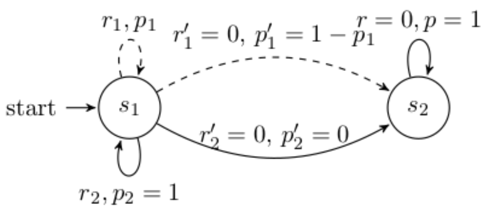

Figure 1. A class of two-state MDPs, with $\gamma = 0.9$. Actions a_1 and a_2 are available in state s_1. State s_2 is absorbing. Dashed (resp. full) arrows indicate the transitions when action a_1 (resp. a_2) is chosen. Numbers above each arrow indicate the transition probability and the average reward, e.g. $p'_2 = \mathbb{P}[s_2|s_1, a_2]$.

Then a simple calculation shows that the pair (s_1, a_1) is optimal : $\frac{r_1}{1-\gamma p_1} > \frac{r_2}{1-\gamma p_2}$ for both ψ and $\overline{\psi}$, while it is sub-optimal : $\frac{r_1}{1-\gamma p_1} < \frac{r_2}{1-\gamma p_2}$ for ϕ. In other words, both ψ and $\overline{\psi}$ are in $\mathrm{Alt}(\phi)$ and $\mathrm{Alt}_{s_1 a_1}(\phi)$ but their average is not: $\frac{\psi+\overline{\psi}}{2} = \phi \notin \mathrm{Alt}(\phi)$. Therefore the sets $\mathrm{Alt}(\phi)$ and $\mathrm{Alt}_{s_1 a_1}(\phi)$ are not convex. Observe that this non-convexity does not arise in simple MAB problems. Indeed, there, the set of parameters (e.g., the average reward vectors $\mu = (\mu_1, \dots, \mu_K)$) such that a given arm is optimal is always convex, i.e., $\{\mu : \mu_k > \max_{j \neq k} \mu_j\}$ is convex.

Remark. *Note that if we have access to an optimization oracle that solves the problem (3) then we can simply apply the classical Track-and-Stop algorithm and achieve asymptotically optimal sample complexity. In the absence of such an oracle, we will devise an upper bound of $T^*(\phi)$, which we will use in our sampling rule as a proxy for the characteristic time.*

4.2. Upper bound of $T^*(\phi)$

We use the analytic version (6) of the optimization problem that defines the sample complexity lower bound to derive a simple (but still problem-specific) upper bound of the characteristic time $T^*(\phi)$. The upper bound actually corresponds to a sampling rule that is explicit, i.e., we do not need to solve any optimization problem to get it. Using this upper bound and the corresponding sampling rule, we will be able to devise a simple track-and-stop algorithm with provable performance guarantees. In addition, the upper bound has the right dependence in the sub-optimality gaps, and we also prove that it remains smaller than existing minimax sample complexity lower bounds.

Before we state the main result leading to our upper bound, we introduce additional notations.

- $\Delta_{\min} = \min\limits_{s, a \neq \pi^\star(s)} \Delta_{sa}$ denotes the minimum sub-optimality gap in ϕ.
- $\mathrm{Var}_{p_\phi(s,a)}[V^\star_\phi] = \mathrm{Var}_{s' \sim p_\phi(.|s,a)}[V^\star_\phi(s')]$ is the variance of the next-state value after taking state-action pair (s, a). Similarly $\mathrm{Var}^\star_{\max}[V^\star_\phi] = \max\limits_{s} \mathrm{Var}_{p_\phi(s, \pi^\star(s))}[V^\star_\phi]$ is the maximum variance of the next-state value after taking an optimal action.
- $\mathrm{sp}[V^\star_\phi] = \max\limits_{s,s'} V^\star_\phi(s') - V^\star_\phi(s)$ is the span of the value function.

Theorem 1. *We have for all vectors ω in the simplex $T(\phi, \omega) \leq U(\phi, \omega)$ where,*

$$U(\phi, \omega) \triangleq \max_{s, a \neq \pi^\star(s)} \frac{T_1(s,a;\phi) + T_2(s,a;\phi)}{\omega_{sa}} + \frac{T_3(\phi) + T_4(\phi)}{\min\limits_{s} \omega_{s,\pi^\star(s)}}, \tag{8}$$

and

$$T_1(s,a;\phi) \triangleq \frac{2}{\Delta_{sa}^2},$$

$$T_2(s,a;\phi) \triangleq \max\left(\frac{16\mathrm{Var}_{p_\phi(s,a)}[V^\star_\phi]}{\Delta_{sa}^2}, \frac{6\mathrm{sp}[V^\star_\phi]^{4/3}}{\Delta_{sa}^{4/3}}\right),$$

$$T_3(\phi) \triangleq \frac{2}{[\Delta_{\min}(1-\gamma)]^2},$$

and

$$T_4(\phi) \triangleq \min\Bigg(\frac{27}{\Delta_{\min}^2(1-\gamma)^3}, \max\left(\frac{16\mathrm{Var}^\star_{\max}[V^\star_\phi]}{\Delta_{\min}^2(1-\gamma)^2}, \frac{6\mathrm{sp}[V^\star_\phi]^{4/3}}{\Delta_{\min}^{4/3}(1-\gamma)^{4/3}}\right)\Bigg).$$

The proof of the theorem relies on writing each of the difference terms $dr(s,a)$, $dp(s,a)$, $dr^{\pi^\star}$ and $dp^{\pi^\star}$ involved in the constraint (5) as a proportion of the sub-optimality gap Δ_{sa}. Then, using classical f-divergences inequalities, as well as a variance inequality from (Azar et al., 2013), we relate each difference term to the KL divergences appearing in the objective function of the problem (6). With this perspective in mind, the terms $T_1(s,a;\phi)$ and $T_2(s,a;\phi)$ can be interpreted as the sample complexity costs to learn the reward of (state,action) pair (s,a) and the corresponding transition probabilities, respectively. Similarly, the terms $T_3(\phi)$ and $T_4(\phi)$ are interpreted as the sample complexity costs to estimate the future rewards collected from the next state and the transitions from the next state.

Corollary 1. *Let $H_{sa} \triangleq T_1(s,a;\phi) + T_2(s,a;\phi)$ and $H^\star \triangleq S(T_3(\phi) + T_4(\phi))$. Then the solution of the problem $\inf\limits_{\omega \in \Sigma} U(\phi, \omega)$ is given by the unique allocation vector $\overline{\omega} \in \Sigma$ defined by ($\sim$ means proportional to): for all $s \in \mathcal{S}$,*

$$\begin{cases} \overline{\omega}_{s,\pi^\star(s)} \sim \frac{1}{S}\sqrt{H^\star(\sum_{s,a \neq \pi^\star(s)} H_{sa})}, \\ \overline{\omega}_{sa} \sim H_{sa}, \quad \textit{for } s, a \neq \pi^\star(s). \end{cases} \tag{9}$$

This allocation yields the following upper bound:

$$T^*(\phi) \le U(\phi) \triangleq 2(H^\star + \sum_{s,a\neq\pi^\star(s)} H_{sa}). \tag{10}$$

In the previous corollary, $\overline{\omega}_{sa}$ is the optimal proportion of times (s,a) should be sampled, and hence for $s,a \neq \pi^\star(s)$, H_{sa} corresponds to the *hardness* of learning that (s,a) is sub-optimal. It scales as the inverse of the square of the gap Δ_{sa} and is proportional to the variance of future rewards after taking (s,a).

Further observe that since the rewards are normalized, we always have: for all (s,a), $\mathrm{Var}_{p_\phi(s,a)}[V_\phi^\star] \le \frac{1}{(1-\gamma)^2}$ and $\mathrm{sp}[V_\phi^\star] \le \frac{1}{(1-\gamma)}$. In addition, we show in Lemma 7 (see Appendix E) that $\Delta_{\min}$ is always smaller than 1. These observations allow us to upper bound $T_1(s,a;\phi)$, $T_2(s,a;\phi)$, $T_3(\phi)$ and $T_4(\phi)$, and to prove the following corollary.

Corollary 2. *We have:* $U(\phi) = \mathcal{O}\left(\frac{SA}{\Delta_{\min}^2(1-\gamma)^3}\right)$.

The above result is obtained by plugging the uniform allocation $\omega_{sa} = 1/SA$ in (8). Hence this naive uniform allocation yields an upper bound scaling as the known minimax sample complexity lower bound $\frac{SA}{\Delta_{\min}^2(1-\gamma)^3}$. This result also implies that a track-and-stop algorithm sampling the pairs (s,a) according to $\overline{\omega}$ will perform better than the minimax bound. This algorithm will become strictly better when $\mathrm{Var}_{\max}^\star[V_\phi^\star] = o(1/(1-\gamma))$, i.e., when the variance of the next-state value after taking the optimal action is small.

5. Algorithm

In this section, we present KLB-TS (KL-Ball Track-and-Stop), an algorithm that selects the successive (state, action) pairs so as to track the allocation $\overline{\omega}$, the problem-specific allocation (9) that leads to the upper bound (10). The algorithm is a track-and-stop, whose stopping rule does not follow a generic Generalized Likelihood Ratio Test as that used (Garivier & Kaufmann, 2016) for MAB problems (refer to Subsection 5.2 for detail).

The algorithm takes as input the confidence parameter δ and any black-box planner MDP-SOLVER. The latter takes as input an MDP ϕ, and returns an optimal policy $\pi_\phi^\star \in \Pi_\phi^\star$. For practical implementations, we use the Policy Iteration algorithm.

KLB-TS starts exploring each (state, action) pair once, to construct an initial estimate $\widehat{\phi}$ of the true MDP ϕ. The algorithm maintains, after t collected observations, an estimate $\widehat{\phi}_t$ of the true MDP. Based on this estimate, KLB-TS computes an estimate of the allocation $\overline{\omega}$, and selects the next (state, action) pair to track it. After each observation, the estimated MDP $\widehat{\phi}_t$ is updated. Finally, the algorithm checks if a stopping condition is satisfied, in which case the algorithm stops and returns the empirical optimal policy $\widehat{\pi}_\tau^\star$. The stopping condition is referred to as the *KL-Ball stopping rule* since it is inspired by the derivation of the upper bound of $T^*(\phi)$. There, the various terms involved in the exploration constraints are upper bounded by KL divergences, i.e., are in a KL ball.

The pseudo-code of KLB-TS is presented in Algorithm 1. Its sampling and stopping rule are described in detail in the next two sub-sections.

Algorithm 1 KLB-TS

```
input Black-box planner MDP-SOLVER(), Confidence parameter $\delta$.
  Collect one sample from each (s,a) in $\mathcal{S} \times \mathcal{A}$.
  Set $t \leftarrow SA$ and $n_t(s,a) \leftarrow 1$, for all (s,a).
  Initialize empirical estimate $\widehat{\phi}_t$ of $\phi$.
  $\widehat{\pi}_t^\star \leftarrow$ MDP-SOLVER($\widehat{\phi}_t$).
  while Stopping condition (14) is not satisfied do
    Compute allocation vector $\overline{\omega}(\widehat{\phi}_t)$ of equation (9).
    Sample from $(s_{t+1}, a_{t+1})$ determined by equation (11).
    For all (s,a) set:
      $n_{t+1}(s,a) \leftarrow \begin{cases} n_t(s,a) + 1 \text{ if } (s,a) = (s_{t+1}, a_{t+1}) \\ n_t(s,a) \text{ Otherwise} \end{cases}$
    $t \leftarrow t + 1$.
    Update empirical estimate $\widehat{\phi}_t$ of $\phi$.
    $\widehat{\pi}_t^\star \leftarrow$ MDP-SOLVER($\widehat{\phi}_t$).
  end while
output Empirical optimal policy $\widehat{\pi}_\tau^\star$
```

5.1. Sampling rule

To build an algorithm with sample complexity matching the upper-bound of Corollary 1, the sampling proportions of (state,action) pairs should be as close as possible to the near-optimal weights defined in (9). To this aim, we simply use the C-tracking rule defined in (Garivier & Kaufmann, 2016), which we recall below.

Define $\overline{\omega}^\varepsilon(\phi)$ as the L^∞ projection of $\overline{\omega}(\phi)$ onto $\Sigma^\varepsilon = \{\omega \in [\varepsilon, 1]^{SA} : \sum_{s,a} \omega_{sa} = 1\}$. Further define $\varepsilon_t = (S^2A^2 + t)^{-1/2}/2$. Then the (state, action) pair to be sampled in round $t+1$ is defined as:

$$(s_{t+1}, a_{t+1}) \in \underset{(s,a)\in\mathcal{S}\times\mathcal{A}}{\arg\max} \sum_{k=1}^{t} \overline{\omega}_{sa}^{\varepsilon_k}(\widehat{\phi}_k) - n_t(s,a) \tag{11}$$

with ties broken arbitrarily. The projection onto Σ^ε forces a minimal amount of exploration so that no pair is left under-explored because of bad initial estimates. The same analysis of the sampling rule given in (Garivier & Kaufmann, 2016) holds in the MDP case and guarantees that:

$$\mathbb{P}_\phi\left(\forall (s,a)\in\mathcal{S}\times\mathcal{A},\quad \lim_{t\to\infty}\frac{n_t(s,a)}{t}=\overline{\omega}_{sa}(\phi)\right)=1.$$

5.2. Stopping rule

It is first worth noting that the proposed stopping condition constitutes the first stopping rule for best-policy identification in the MDP setting. Previous stopping rules in the literature are designed to identify ε-optimal policies. Unless we have access to an oracle that reveals the minimal gap between the best policy and a sub-optimal policy (in which case we can set ε smaller than this gap), we cannot identify the best-policy using these rules.

A *good* stopping rule determines when the set of samples collected so far is *just* enough to declare that $\widehat{\pi}^\star_t=\pi^\star$ with probability $1-\delta$. The design of our stopping rule is inspired by the proof of the upper-bound $U(\phi)$, which uses the following fact (refer to the inequalities (19)-(20)-(21)-(24)-(23) in the appendix): For all $\psi\in\mathrm{Alt}(\phi)$, there exists $s,a\neq\pi^\star(s)$ and a vector α in the simplex of $\mathbb{R}^4$ (which we denote Σ_4) such that the four following conditions are verified:

$$\begin{cases}\frac{\alpha_1^2}{T_1(s,a;\phi)}\leq \mathrm{kl}\left(r_\phi(s,a),r_\psi(s,a)\right),\\ \frac{\alpha_2^2}{T_2(s,a;\phi)}\leq KL\left(p_\phi(s,a)\|p_\psi(s,a)\right),\\ \frac{\alpha_3^2}{T_3(\phi)}\leq \max\limits_{s\in\mathcal{S}}\mathrm{kl}\left(r_\phi(s,\pi^\star_\phi(s)),r_\psi(s,\pi^\star_\phi(s))\right),\\ \frac{\alpha_4^2}{T_4(\phi)}\leq \max\limits_{s\in\mathcal{S}}KL\left(p_\phi(s,\pi^\star_\phi(s))\|p_\psi(s,\pi^\star_\phi(s))\right).\end{cases}\tag{12}$$

Then defining the quantities

$$\begin{cases}\rho_1(\phi,\psi)(s,a)=T_1(s,a;\phi)\mathrm{kl}\left(r_\phi(s,a),r_\psi(s,a)\right),\\ \rho_2(\phi,\psi)(s,a)=T_2(s,a;\phi)KL\left(p_\phi(s,a)\|p_\psi(s,a)\right),\\ \rho_3(\phi,\psi)=\max\limits_{s\in\mathcal{S}}T_3(\phi)\mathrm{kl}(r_\phi(s,\pi^\star_\phi(s)),r_\psi(s,\pi^\star_\phi(s))),\\ \rho_4(\phi,\psi)=\max\limits_{s\in\mathcal{S}}T_4(\phi)KL(p_\phi(s,\pi^\star_\phi(s))\|p_\psi(s,\pi^\star_\phi(s))),\end{cases}\tag{13}$$

(12) suggests that to design a PAC stopping condition, it is sufficient to check that the event

$$\mathcal{E}=\Big(\forall\alpha\in\Sigma_4\;\forall s,a\neq\widehat{\pi}^\star_t(s),\;\rho_1(\widehat{\phi}_t,\phi)(s,a)<\alpha_1^2\text{ or }\rho_2(\widehat{\phi}_t,\phi)(s,a)<\alpha_2^2\text{ or }\rho_3(\widehat{\phi}_t,\phi)<\alpha_3^2\text{ or }\rho_4(\widehat{\phi}_t,\phi)<\alpha_4^2\Big)$$

or equivalently[1]:

$$\mathcal{E}=\Big(\forall s,a\neq\widehat{\pi}^\star_t(s),\;\sqrt{\rho_1(\widehat{\phi}_t,\phi)(s,a)}+\sqrt{\rho_2(\widehat{\phi}_t,\phi)(s,a)}+\sqrt{\rho_3(\widehat{\phi}_t,\phi)}+\sqrt{\rho_4(\widehat{\phi}_t,\phi)}<1\Big)$$

holds with probability $1-\delta$. Indeed, if $\mathcal{E}$ holds, then by contraposition of (12), we have $\phi\notin\mathrm{Alt}(\widehat{\phi}_t)$, which means that $\widehat{\pi}^\star_t=\pi^\star$. To define our stopping rule, we further introduce the threshold function:

$$x(\delta,n,m)=\log(1/\delta)+(m-1)[1+\log\left(1+n/(m-1)\right)].$$

We finally define $\widehat{T}_1(s,a)=T_1(s,a;\widehat{\phi}_t)$, $\widehat{T}_2(s,a)=T_2(s,a;\widehat{\phi}_t)$, $\widehat{T}_3=T_3(\widehat{\phi}_t)$, $\widehat{T}_4=T_4(\widehat{\phi}_t)$ and $\delta'=\frac{\delta}{4S^3A}$. The KL-Ball stopping condition, which guarantees that the event $\mathcal{E}$ above holds with probability $1-\delta$, is:

$$\max_{s,a\neq\widehat{\pi}^\star_t(s)}\frac{\sqrt{\widehat{T_1}(s,a)x(\delta',n_t(s,a),2)}+\sqrt{\widehat{T_2}(s,a)x(\delta',n_t(s,a),S)}}{\sqrt{n_t(s,a)}}+\max_{s\in\mathcal{S}}\frac{\sqrt{\widehat{T_3}x(\delta',n_t(s,\widehat{\pi}^\star_t(s)),2)}+\sqrt{\widehat{T_4}x(\delta',n_t(s,\widehat{\pi}^\star_t(s)),S)}}{\sqrt{n_t(s,\widehat{\pi}^\star_t(s))}}\leq 1\tag{14}$$

More precisely: $\tau_\delta=\inf\{t\in\mathbb{N}:(14)\text{ holds}\}$.

Theorem 2. *Under the KL-Ball stopping rule, we have:* $\mathbb{P}_\phi(\tau_\delta<\infty,\widehat{\pi}^\star_{\tau_\delta}\neq\pi^\star_\phi)\leq\delta$.

6. Sample Complexity Analysis

6.1. Main Results

Our main results take the form of asymptotic (when δ goes to 0) upper bounds on the sample complexity of KLB-TS. These bounds are proved as follows. First, the use of the C-tracking rule makes it possible to establish the convergence of the vector $(n_t(s,a))_{s,a}/t$ (the (state, action) pair visit frequencies) to the nearly-optimal allocation vector $\overline{\omega}$, as well as the convergence of the empirical MDP $\widehat{\phi}_t$ to the true MDP ϕ. Then, plugging these convergence results in the definition of the stopping rule (14), and combining the obtained results with the asymptotic shape of the threshold function $x(\delta',n,m)\underset{\delta\to0}{\sim}\log(1/\delta)$, we obtain (refer to Appendix G for a detailed description of these arguments):

$$\tau_\delta\underset{\delta\to0}{\sim}\inf\left\{t\in\mathbb{N}:\sqrt{\log(1/\delta)}\left(\max_{s,a\neq\pi^\star(s)}\frac{\sqrt{T_1(s,a;\phi)}+\sqrt{T_2(s,a;\phi)}}{\sqrt{t\times\overline{\omega}_{sa}}}+\max_{s\in\mathcal{S}}\frac{\sqrt{T_3(\phi)}+\sqrt{T_4(\phi)}}{\sqrt{t\times\overline{\omega}_{s,\pi^\star(s)}}}\right)\leq1\right\}.$$

Finally, we show that the condition in the 'inf' above holds as soon as $t\geq 4U(\phi)\log(1/\delta)$ (see Lemma 11). The above arguments lead to an upper bound of the sample complexity of KLB-TS, valid almost surely (Proposition 2) and in expectation (Theorem 3).

Proposition 2. *The KL-Ball stopping rule, coupled with any sampling rule ensuring that for every state-action pair (s,a), $n_t(s,a)/t$ converges almost surely to the nearly-optimal allocations $\overline{\omega}_{sa}$ of Corollary 1, yields a sample complexity τ_δ satisfying for all $\delta\in(0,1)$: $\mathbb{P}_\phi(\tau_\delta<\infty)=1$ and* $\mathbb{P}_\phi\left(\limsup_{\delta\to0}\frac{\tau_\delta}{\log(1/\delta)}\leq 4U(\phi)\right)=1$.

[1] Hence the name KL-Ball stopping rule.

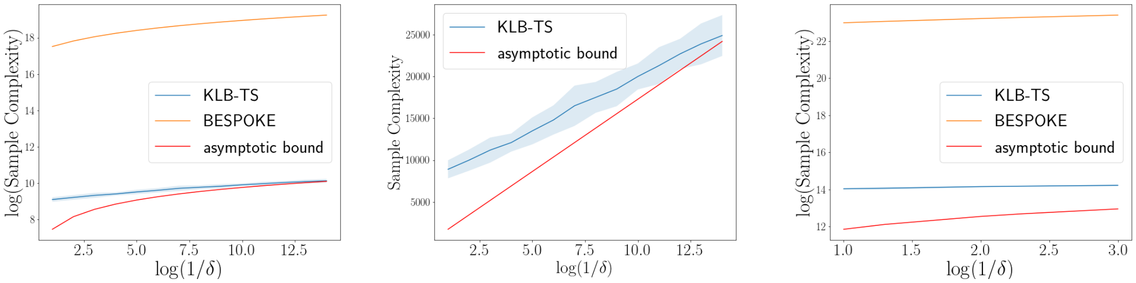

Figure 2. KLB-TS vs. BESPOKE. Left and center: S=A=2, $\gamma = 0.5$, right: $S = 5, A = 10, \gamma = 0.7$.

Theorem 3. *The KL-Ball stopping rule, coupled with the C-tracking rule defined in (11), yields a sample complexity τ_δ satisfying: for all $\delta \in (0, 1)$, $\mathbb{E}_\phi[\tau_\delta]$ is finite and* $\limsup_{\delta\to 0} \frac{\mathbb{E}_\phi[\tau_\delta]}{\log(1/\delta)} \leq 4U(\phi)$.

The proof of the theorem above is similar to that of Theorem 14 in (Garivier & Kaufmann, 2016) with a few notable differences. First, we defined a distance on MDPs through the L^∞-norm of their reward and transition kernels. Then, we adapted Lemma 19 from (Garivier & Kaufmann, 2016), which gives a concentration inequality of the empirical average-rewards in the MAB setting, to include the concentration of transition probabilities of the empirical MDP.

6.2. Interpretation of the bound $U(\phi)$

The bound $U(\phi)$ is the sum of two interpretable components. The first term $U_1 = 2\sum_{s,a\neq\pi^\star(s)} H_{sa}$ represents the number of samples needed to estimate and identify sub-optimal pairs (s, a). It is proportional to the variance of the value function $\mathrm{Var}_{p(s,a)}[V^\star_\phi]$ and inversely proportional to the squared gap Δ^2_{sa}:

$$U_1 = \mathcal{O}\Bigg(\sum_{s,a\neq\pi^\star(s)} \max\left\{\frac{1 + \mathrm{Var}_{p(s,a)}[V^\star_\phi]}{\Delta^2_{sa}}, \frac{\mathrm{sp}[V^\star_\phi]^{4/3}}{\Delta^{4/3}_{sa}}\right\}\Bigg).$$

The second term $2H^\star$ represents the samples needed to estimate and identify the optimal (state, action) pairs. It is proportional to $\mathrm{Var}^*_{max}[V^\star_\phi]$ the maximum variance of the value function across the trajectory of the optimal policy and inversely proportional to the squared minimum gap $\Delta^2_{\min}$ and to $(1-\gamma)^3$:

$$\begin{aligned} H^\star =& \mathcal{O}\Bigg(\frac{S}{(1-\gamma)^2\Delta^2_{\min}} + \min\left\{\frac{S}{(1-\gamma)^3\Delta^2_{\min}},\right.\\ &\left.\max\left\{\frac{S\mathrm{Var}^*_{max}[V^\star_\phi]}{(1-\gamma)^2\Delta^2_{\min}}, \frac{S\mathrm{sp}[V^\star_\phi]^{4/3}}{(1-\gamma)^{4/3}\Delta^{4/3}_{\min}}\right\}\right\}\Bigg). \end{aligned}$$

These dependencies were intuitively expected: The larger the variances are, the more samples we need to accurately estimate the value function. Similarly, the smaller the gaps are, the harder it is to distinguish optimal (state, action) pairs from sub-optimal ones. Finally, as γ gets closer to one, it is natural to require more samples, as a small error in estimating rewards or transitions can induce a large change in the total discounted reward, thereby modifying the optimal policy.

7. Experiments

In this section, we run numerical experiments to compare the performances of KLB-TS and BESPOKE (these are so far the two algorithms with problem-specific sample complexity guarantees). We refer the reader to Appendix H for a detailed description of the differences between KLB-TS and BESPOKE, as well as a comparison of their theoretical guarantees. To compare the two algorithms, we generated two MDPs randomly: a first small MDP with two states and two actions, and a second larger and more realistic MDP with five states and ten actions per state. We used BESPOKE with an accuracy parameter $\epsilon = 0.9\Delta_{\min}$ (note that $\Delta_{\min}$ is revealed to BESPOKE). For each value of the confidence level δ, we run 10 simulations for the first MDP under both algorithms. To save computation time in the case of the second MDP, we run 5 simulations for each δ and only compare KLB-TS's sample complexity with BESPOKE's initial number of samples $n_{\min}$ which, as noted in Appendix H, contributed for more than 99% of its sample complexity.

Figure 2 shows the mean sample complexity along with its 2-standard-deviations interval (which seems very small due to the use of a log-scale). The red curve (referred to as 'asymptotic bound') shows the upper bound $4U(\phi)\log(1/\delta)$ guaranteed by Theorem 3. Note that KLB-TS sample complexity is greater than $4U(\phi)\log(1/\delta)$ for moderate values of δ and only matches it for $\delta = 10^{-14}$. For both MDPs, KLB-TS clearly outperforms BESPOKE.

8. Conclusion

In this work, we have investigated the design of RL algorithms with *minimal problem-specific* sample complexity. To this aim, we first derived the information-theoretical sample complexity limit (a lower bound on the sample complexity satisfied by any algorithm) and the corresponding optimal sample allocation. Our hope was that, as for the MAB problem, this allocation would be easy to compute and could then lead to a simple and optimal track-and-stop algorithm. Unfortunately, for RL problems, it turns out that the optimal allocation solves an involved non-convex program. Approaching the fundamental sample complexity limit seems possible only if one could solve this program. To circumvent this issue, we derived a tight upper bound of the characteristic time. Remarkably, this bound corresponds to a sample allocation that is explicit, and hence can be easily plugged in into a track-and-stop algorithm. Based on this upper bound, we proposed KLB-TS, an algorithm whose sample complexity matches this upper bound.

This work opens up interesting research directions. First, the computational complexity of the sample complexity lower bound strongly suggests the existence of a fundamental trade-off between sample and computational complexities. Investigating this trade-off is intriguing. Then, we restricted our attention to the generative model, where one can sample any (state, action) pair at any step. In most practical cases however, one needs to learn an optimal policy by observing a single trajectory of the system. Hence, the numbers of times one observes the various (state, action) pairs are correlated, inducing some additional constraints in the optimization problem leading to the sample complexity lower bound. It is worth studying the impact of these navigation constraints on the sample complexity. Finally, we plan to extend our results to the framework of RL with function approximation.

Acknowledgments: The authors would like to thank the reviewers whose comments and questions helped improve the clarity of the paper.

References

Agarwal, A., Kakade, S., and Yang, L. F. Model-based reinforcement learning with a generative model is minimax optimal. volume 125 of *Proceedings of Machine Learning Research*, pp. 67–83. PMLR, 09–12 Jul 2020. URL http://proceedings.mlr.press/v125/agarwal20b.html.

Azar, M. G., Munos, R., and Kappen, H. J. Minimax pac bounds on the sample complexity of reinforcement learning with a generative model. *Machine learning*, 91(3): 325–349, 2013.

Even-Dar, E., Mannor, S., and Mansour, Y. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(Jun):1079–1105, 2006.

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, 23–26 Jun 2016. PMLR. URL http://proceedings.mlr.press/v49/garivier16a.html.

Kakade, S. M. *On the sample complexity of reinforcement learning*. PhD thesis, University of London, England, 2003.

Kaufmann, E., Cappé, O., and Garivier, A. On the complexity of best-arm identification in multi-armed bandit models. *The Journal of Machine Learning Research*, 17 (1):1–42, 2016.

Kearns, M. and Singh, S. Finite-sample convergence rates for q-learning and indirect algorithms. *Advances in Neural Information Processing*, 11, 04 1999.

Lai, T. and Robbins, H. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1): 4–2, 1985.

Li, G., Wei, Y., Chi, Y., Gu, Y., and Chen, Y. Breaking the sample size barrier in model-based reinforcement learning with a generative model. *arXiv preprint*, arXiv:2005.12900, 2020.

Ok, J., Proutiere, A., and Tranos, D. Exploration in structured reinforcement learning. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/d693d554e0ede0d75f7d2873b015f228-Paper.pdf.

Sidford, A., Wang, M., Wu, X., Yang, L., and Ye, Y. Near-optimal time and sample complexities for solving markov decision processes with a generative model. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 5186–5196. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7765-near-optimal-time-and-sample-complexities-for-solving-markov-decision-processes-with-a-generative-model.pdf.

Zanette, A., Kochenderfer, M. J., and Brunskill, E. Almost horizon-free structure-aware best policy identification with a generative model. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 5625–5634. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/8800-almost-horizon-free-structure-aware-best-policy-identification-with-a-generative-model.pdf.