Test-Time Regret Minimization in Meta Reinforcement Learning

Mirco Mutti [1] **Aviv Tamar** [1]

Abstract

Meta reinforcement learning sets a distribution over a set of tasks on which the agent can *train* at will, then is asked to learn an optimal policy for any *test* task efficiently. In this paper, we consider a *finite* set of tasks modeled through Markov decision processes with various dynamics. We assume to have endured a long training phase, from which the set of tasks is perfectly recovered, and we focus on *regret minimization* against the optimal policy in the unknown test task. Under a separation condition that states the existence of a state-action pair revealing a task against another, Chen et al. (2022) show that $\mathcal{O}(M^2 \log(H))$ regret can be achieved, where M, H are the number of tasks in the set and test episodes, respectively. In our first contribution, we demonstrate that the latter rate is nearly optimal by developing a novel *lower bound* for test-time regret minimization under separation, showing that a linear dependence with M is unavoidable. Then, we present a family of stronger yet reasonable assumptions beyond separation, which we call *strong identifiability*, enabling algorithms achieving fast rates $\log(H)$ and sublinear dependence with M simultaneously. Our paper provides a new understanding of the statistical barriers of test-time regret minimization and when fast rates can be achieved.

1. Introduction

Reinforcement Learning (RL, Sutton & Barto, 2018) is a popular tool for learning an optimal decision policy through sampled interactions with a Markov Decision Process (MDP), a general framework encompassing countless applications, ranging from robotics (Xu et al., 2023; Kaufmann et al., 2023) to algorithms design (Fawzi et al., 2022), conversational agents (Stiennon et al., 2020), and others.

[1]Technion – Israel Institute of Technology, Haifa, Israel. Correspondence to: Mirco Mutti <mirco.m@technion.ac.il>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

Although powerful, the efficiency of RL is a long-standing issue. The theory says that the *regret* of a RL algorithm, i.e., the difference between the value of the deployed policy and the optimal policy in hindsight, inescapably scales with $\sqrt{H}$ in the worst case (Jaksch et al., 2010; Osband & Van Roy, 2016), H being the total number of episodes of interactions with the MDP. Even if the real world is arguably better behaved than the worst-case MDP, the most successful algorithms (Schulman et al., 2015; Mnih et al., 2015) still take thousands of interaction episodes to learn a competitive policy in simulation, which draws pessimism for RL to be applied for learning in the real world.

A promising direction to improve RL efficiency is *meta RL* (Ghavamzadeh et al., 2015), in which a distribution over the set of tasks we can face is considered. In meta RL, we first have a *training* phase on some tasks sampled from the latter distribution, for which the learning efficiency is less of an issue (e.g., a simulator is available). Then, we exploit the collected knowledge to achieve faster learning in a *test* task, which is assumed to come from the same distribution.

Much of the previous work in meta RL focuses on algorithms for the training stage (Duan et al., 2016; Finn et al., 2017; Rakelly et al., 2019; Zintgraf et al., 2019; 2021), or analyse generalization of the trained model to the test task (Simchowitz et al., 2021; Tamar et al., 2022; Rimon et al., 2022; Zhao et al., 2022; Zisselman et al., 2023).

Here we study meta RL from a different perspective. We assume to have spent infinite time in the training phase, such that the task distribution can be recovered (we mean the full specifications of all the MDPs in the set, not just the task distribution itself), and we aim to minimize the regret against the optimal policy in the test task. Especially,

Does perfect meta RL training provably improve the learning efficiency on the test task against standard RL?

We believe that a positive answer is an essential theoretical ground for motivating meta RL, as there is little incentive to undergo a costly training (at least in terms of computation) without guarantees of improved efficiency on the test task.

Even in simple settings, in which the distribution is supported on a finite set of M tasks, meta RL provides little hope, as the regret still scales with $\sqrt{H}$ in the worst case, with only marginal gains in the statistical efficiency w.r.t.

STRUCTURE	REACHABILITY	REGRET UPPER BOUND	REGRET LOWER BOUND
Separation (Ass. 1)	Reachable (Ass. 2)	$\mathcal{O}\left(\frac{TM^2\log(MH)}{\lambda^4}\right)$ (Thr. 3.1)	$\Omega\left(\frac{TM\log(H)}{\lambda}\log\left(\frac{1}{\delta}\right)\right)$ (Thr. 4.1)*
Clustering (Ass. 3)	Cluster Reachable (Ass. 4)	$\mathcal{O}\left(\frac{T(K^2+N^2)\log(NH)}{\lambda^4}\right)$ (Thr. 5.1)*	
Tree Structure (Ass. 6)	Strongly Reachable (Ass. 5)	$\mathcal{O}\left(\frac{Td\log(dH)}{\lambda^4}\right)$ (Thr. 5.2)*	
Revealing Policies (Ass. 7)	Reachable (Ass. 2)	$\mathcal{O}\left(\frac{TI\log(MH)}{\lambda^4}\right)$ (Thr. 5.3)*	

Table 1. Overview of the theoretical results. T is the horizon of an episode, M the number of tasks, H the number of episodes, λ a separation parameter, δ a confidence, K the number of clusters with size N, $d = \log_{1/\beta} M$ the depth of a binary tree with entropy factor β, I the number of revealing policies. Note that the rates are simplified to highlight relevant factors. Novel results are marked with *.

standard RL (Chen et al., 2022; Ye et al., 2023). Nonetheless, under a common *separation* assumption on set of tasks (Chen et al., 2022; Kwon et al., 2021b), i.e., there exists at least one reachable state-action pair that *reveals* one task against the others, the prospects of meta RL become brighter. Chen et al. (2022) show that $\mathcal{O}(M^2\log(MH))$ regret can be achieved by first identifying the test task (with high probability) and then deploying the best policy for the latter. Their approach is somewhat wasteful in the identification, as the algorithm performs a sequence of *one-vs-one* tests on candidate tasks, which induces the M^2 factor. However, it is unclear if the latter is necessary or better algorithms could be developed.

In this paper, we provide a nuanced study of the statistical barriers of test-time regret minimization in meta RL. First, we provide a lower bound $\Omega(TM\log(H))$ for test-time regret minimization under separation, where T is the horizon of an interaction episode with the test MDP. Our lower bound demonstrates that the "wasteful" algorithm by Chen et al. (2022) is nearly optimal and one factor M is unavoidable under separation alone. The way the lower bound is derived philosophically confirms that probing the MDP to first identify the test task and then exploit the collected information is not just reasonable but also optimal. Nevertheless, we note that a linear dependence with M is not desirable if we aim to scale meta RL to large task distributions.

Thus, we formulate stronger yet reasonable requirements under which the latter limitation may be overcome. To formalize our desiderata, we call *strong identifiability* a family of assumptions that allow for fast rates in H and M simultaneously, for which we provide three instances.

The first assumes the tasks can be partitioned in coherent *clusters*, such that the algorithm can efficiently identify the cluster to which the test task belongs, and then identify the latter within that cluster down the hierarchy. Clusters are fairly natural in practice, e.g., in a meta RL problem applied to movie recommendation systems, in which the set of tasks consists of different users with specific tastes. We could first provide recommendations to probe whether the test user especially likes a movie genre, and then provide fine-grained recommendations within the genre catalogue.

The second assumes that the test task can be identified following a *tree structure*. This means that the algorithm can *split* a subset from the current set of candidate tasks by collecting information on a single state-action pair, which allows to eliminate a large chunk of candidates at every iteration. This may happen in practice whenever the test task can be identified testing a sequence of revealing characteristics.

The third strong identifiability assumption we analyse is admitting the presence of a small number of *revealing policies*, which allow for collecting highly informative data irrespective of the test task, then to perform the identification without further interactions. To provide intuition on such policies, let us think of a meta RL problem in which the set of tasks consists of morphologies of terrains we aim to explore without daylight. A revealing policy would light a torch to probe the surroundings before taking the right direction for the test morphology.

Contributions. Our main contributions are:

- We revise an analysis of domain randomization (Chen et al., 2022) through the lenses of meta RL, adapting their result on sim-to-real gap into a regret upper bound $\mathcal{O}(M^2\log(MH))$ for our setting (Section 3);
- We derive a *lower bound* $\Omega(TM\log(H))$ to test-time regret minimization under separation (Section 4) through original techniques that formally link our problem to Best Policy Identification (BPI, Fiechter, 1994). The proof requires a *tailored* lower bound to the sample complexity of BPI, which can be of independent interest (Appendix B);
- We present structural assumptions beyond separation, called *strong identifiability* (Section 5). Those include: A *clustering* assumption that leads to a regret upper bound of order $\mathcal{O}(T(K^2+N^2)\log(NH))$, where K is the number of clusters and N is the size of the largest cluster; A *tree structure* assumption that leads to a regret of order $\mathcal{O}(Td\log(dH))$, where $d = \log_{1/\beta} M$ is the depth of the tree and β is the splitting factor; A *revealing policies* assumption that leads to a regret of order $\mathcal{O}(TI\log(MH))$, where I is the number of revealing policies;
- We provide additional sharp rates to the test-time regret for meta learning in bandits (Appendix C).

Complete proofs of the theorems are in Appendix A.

2. Problem Formulation

We first present the necessary background on MDPs and meta RL (Sections 2.1, 2.2) before formulating the learning problem we will address in the paper (Section 2.3).

Notation. Let $\mathcal{A}$ a space of size $|\mathcal{A}|$ with elements $a \in \mathcal{A}$. Then, $\mathcal{P}(\mathcal{A}) := (p \in [0,1]^{|\mathcal{A}|} | \sum_{a\in\mathcal{A}} p(a) = 1)$ is the simplex for a finite $\mathcal{A}$. Let $p, q \in \mathcal{P}(\mathcal{A})$, their $\ell 1$-distance is $\|p - q\|_1 = \sum_{a\in\mathcal{A}} |p(a) - q(a)|$, their total variation is $\mathrm{TV}(p, q) = \sup_{a\in\mathcal{A}} |p(a) - q(a)|$, their Kullback-Leibler divergence is $\mathrm{KL}(p, q) = \sum_{a\in\mathcal{A}} p(a) \log(p(a)/q(a))$ and $\mathrm{KL}_{p|q} := \mathrm{KL}(p, q) + \mathrm{KL}(q, p)$. We will denote sets and sequences as $(a_i)_{i\in[I]} := (a_1, \dots, a_I)$, where $[I] := (1, \dots, I)$ for some constant $I \in \mathbb{N}$.

2.1. Markov Decision Processes and RL

A finite-horizon time-homogeneous Markov Decision Process (MDP, Puterman, 2014) is defined by a tuple[1] $\mathcal{M}_i := (\mathcal{S}, \mathcal{A}, p_i, r_i, s_1, T)$ where $\mathcal{S}$ is a finite set of states ($S = |\mathcal{S}|$), $\mathcal{A}$ is a finite set of actions ($A = |\mathcal{A}|$), $p_i : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ is a transition model such that $p_i(s'|s, a)$ denotes the conditional probability of transitioning to s' taking action a in state s, $r_i : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is a reward function such that $r_i(s, a)$ is the reward collected by taking action a in s, s_1 is the initial state,[2] and $T < \infty$ is the horizon of an episode.

An episode of interaction between an agent and the MDP $\mathcal{M}_i$ goes as follows. At each step $t \in [T]$, the agent observes $s_t \in \mathcal{S}$ and takes $a_t \in \mathcal{A}$. The environment transitions to $s_{t+1} \sim p_i(\cdot|s_t, a_t)$ and the agent collects $r_i(s_t, a_t)$. Hence, an episode can be summarized through a sequence $\tau = (s_t, a_t)_{t\in[T]}$ called a *trajectory*.

The agent selects its actions by means of a non-stationary Markovian *policy* $\pi := (\pi_t : \mathcal{S} \to \mathcal{P}(\mathcal{A}))_{t\in[T]} \in \Pi$ where $\pi_t(a|s)$ denotes the conditional probability of action a in state s at time step t, and Π is the set of policies. We define the *value* at step t of playing policy π in state s of $\mathcal{M}_i$ as

$$V^{\pi}_{it}(s) := \mathop{\mathbb{E}}_{\pi,\mathcal{M}_i} \left[\sum_{t'=t}^{T} r_i(s_{t'}, a_{t'}) \;\middle|\; s_t = s \right],$$

where the expectation is on all the sources of randomness, i.e., the action selection induced by π and the state transitions induced by p_i, which may be stochastic. We further denote $V_i(\pi) := V_{i1}(s_1)$ the value of the policy in the initial state. The objective function of the agent in $\mathcal{M}_i$ can then be written as $\max_{\pi\in\Pi} V_i(\pi)$, where we denote as π^* the policy attaining the maximum and $V^*_i := V_i(\pi^*)$. RL (Sutton & Barto, 2018) is a paradigm for learning an (approximately) optimal policy π, such that $V^*_i - V_i(\pi) \le \epsilon$ for some $\epsilon > 0$, from sampled interactions with an *unknown* MDP $\mathcal{M}_i$.

[1] The meaning of the subscripts will become clear later.

[2] Note that unique initial state is without loss of generality, as we can accommodate an initial state distribution $\mu \in \mathcal{P}(\mathcal{S})$ through a fictitious state s_0 such that $p(s|s_0, a) = \mu(s), \forall a \in \mathcal{A}$.

2.2. Meta Reinforcement Learning

Meta RL (Duan et al., 2016), initially introduced by Schmidhuber (1987), extends the RL paradigm to a set of M MDPs $\mathcal{M} := (\mathcal{M}_i)_{i\in[M]} = (\mathcal{S}, \mathcal{A}, p_i, r_i, s_1, T)_{i\in[M]}$ having the same $\mathcal{S}, \mathcal{A}, s_1, T$, but possibly different transition model p_i and reward r_i. Just like in RL, the latter MDPs are typically unknown to the agent. In a process called *training*, the agent can collect interactions with a number of tasks[3] drawn from $\mathcal{M}$ according to a *task distribution* $P \in \mathcal{P}(\mathcal{M})$ such that the probability of drawing $\mathcal{M}_i$ is $P(\mathcal{M}_i)$. In training, the agent collects information into a *prior* model, e.g., a policy, a model of transitions, or an algorithm, that is then used to address a RL problem on a *test* task $\mathcal{M}_i$ assumed to be drawn from the same task distribution P.

Bayesian RL (Ghavamzadeh et al., 2015) formulates the target of the training as learning a *Bayes-optimal* policy[4] $\pi_{\text{BO}} \in \arg\max_{\pi\in\Pi} \mathbb{E}_{\mathcal{M}_i\sim P}[V_i(\pi)]$ under the distribution P. As we describe below, here we study meta RL from a *frequentist* perspective rather than a Bayesian formulation.

2.3. Test-Time Regret Minimization

In this paper, instead of focusing on the training phase of meta RL, we assume perfect knowledge of the set of tasks $\mathcal{M}$, such that every transition model p_i and reward r_i are fully known to the agent. With this prior knowledge, we aim to minimize the *test-time regret* over H episodes

$$R_H(\mathcal{M}_i, \mathbb{A}) := \mathbb{E}\left[\sum_{h=1}^{H} V^*_i - V_i(\pi_h) \right] \tag{1}$$

where $\mathcal{M}_i \in \mathcal{M}$ is *any* test task, π_h is the policy deployed in episode h by algorithm $\mathbb{A}$, and the expectation is over realizations of the episodes $1, \dots, H$ taken from $\mathcal{M}_i$. The motivation for this objective is that comparing against the optimal policy *for* the test task, instead of an optimal policy *on average* over the task distribution (Ye et al., 2023), gives a regret measure that is robust to the worst-case task, arguably a minimal requirement given the perfect training assumption. We see the latter as a necessary first step towards a more realistic setting with *approximate* knowledge of $\mathcal{M}$ only. If we cannot succeed with the former, the latter is hopeless.

Two other important observations are in order. First, in this paper we study the regret of *adaptive* algorithms $\mathbb{A}$, that is, the deployed policy is Markovian within an episode, but can change from episode to episode. This is only slightly restrictive as the policy π_h is computed

[3] We are going to use the term *task* and *MDP* interchangeably.

[4] Note that the Bayes-optimal policy is history-dependent in general (Ghavamzadeh et al., 2015).

having the full history of realizations in previous episodes $\mathcal{H}_h = ((s_{t,h}, a_{t,h}, r_{t,h})_{t\in[T]})_{h'\in[h]}$, which means an algorithm $\mathbb{A}$ corresponds to a non-Markovian policy with low switching cost (Bai et al., 2019). Second, the expression in (1) is different from the notion of *Bayesian regret* that is common Bayesian RL (Ghavamzadeh et al., 2015), in which the regret is taken in expectation over the task distribution P instead of taking the worst-case task. As a result, the optimal algorithm $\mathbb{A}$ for the test-time regret does not correspond to the Bayes-optimal policy in general, although it holds $R_H(\mathcal{M}_i, \pi_{\text{BO}}) = \mathcal{O}(M \cdot R_H(\mathcal{M}_i, \mathbb{A}))$ for any algorithm $\mathbb{A}$ from (Chen et al., 2022, Lemma 1).

3. Previous Fast Rates for Test-Time Regret

In this section, we discuss the known results for the test-time regret minimization objective we described above. In this paper we especially care for *fast rates*, i.e., those settings in which the knowledge of the set of tasks and its structure allow to overcome the statistical barrier for regret minimization in finite-horizon RL, which we know is of order $\Theta(\mathsf{poly}(T, S, A)\sqrt{H})$ from lower bounds and minimax algorithms (Osband & Van Roy, 2016; Azar et al., 2017).

Chen et al. (2022) address a regret minimization problem that is very close to our test-time regret formulation, although their narrative is centered around domain randomization rather than meta RL. When the set of tasks is finite, they provide a lower bound of order $\Omega(\sqrt{DMH})$, in which D is the diameter of a communicating infinite-horizon MDP (see Assumption 1 and Theorem 3 in Chen et al., 2022).[5] The latter result demonstrates that additional assumptions are needed to break the $\sqrt{H}$ barrier of RL.

To this end, Chen et al. (2022) introduce a separation condition within the set of tasks $\mathcal{M}$. Formally,

Assumption 1 (λ-separation (Chen et al., 2022))**.** *For any $\mathcal{M}_i, \mathcal{M}_j \in \mathcal{M}$, there exists $(s, a) \in \mathcal{S} \times \mathcal{A}$ such that $\|(p_i - p_j)(\cdot|s, a)\|_1 \geq \lambda$.*

The latter assumption guarantees the existence of a *revealing* state-action pair to tell apart a task from another. This allows to design an algorithm which repeatedly tests that revealing state-action pair to identify the test task efficiently. First, we need to further make sure that the revealing state-action can be reached with meaningful probability.[6]

Definition 1. *Let $X(s|\mathcal{M}_i, \pi)$ denote the random variable of the first time step in which the state $s \in \mathcal{S}$ is reached by playing policy π in the MDP $\mathcal{M}_i$. Let $X(s, a|\mathcal{M}_i, \pi)$ be the analogous for state-action pairs $(s, a) \in \mathcal{S} \times \mathcal{A}$.*

[5]Note that the results in Chen et al. (2022) are given for a slightly different setting (detailed comparisons are in Section 6). We will explicitly adapt to our setting the most relevant results.

[6]This is the technical adaptation of the communicating MDP assumption in Chen et al. (2022) for the finite-horizon setting.

Algorithm 1 Identify-then-Commit (Chen et al., 2022)

1: **input** set of MDPs $\mathcal{D}$ and visitation count n
2: **while** $|\mathcal{D}| > 1$ **do**
3: Draw $\mathcal{M}_1, \mathcal{M}_2$ from $\mathcal{D}$ randomly
4: Let $(\bar{s}, \bar{a}) \in \arg\max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \|(p_1 - p_2)(\cdot|s, a)\|_1$
5: Call Algorithm 2 with $\mathcal{D}, (\bar{s}, \bar{a}), n$ to collect $\mathcal{X}$
6: **if** $\exists s' \in \mathcal{X} : p_2(s'|\bar{s}, \bar{a}) = 0$ or $\prod_{s'\in\mathcal{X}} \frac{p_1(s'|\bar{s},\bar{a})}{p_2(s'|\bar{s},\bar{a})} \geq 1$ **then**
7: Eliminate $\mathcal{M}_2$ from $\mathcal{D}$
8: **else**
9: Eliminate $\mathcal{M}_1$ from $\mathcal{D}$
10: **end if**
11: **end while**
12: Take $\hat{\mathcal{M}} \in \mathcal{D}$ and run $\hat{\pi} \in \arg\max_{\pi\in\Pi} V_{\hat{\mathcal{M}}}(\pi)$ for the remaining episodes

Algorithm 2 Sampling Routine

1: **input** set of MDPs $\mathcal{D}$, state-action pair $(\bar{s}, \bar{a})$, and visitation count n
2: Initialize count $N_{\bar{s}\bar{a}} = 0$ and set $\mathcal{X} = \emptyset$
3: **while** $N_{\bar{s}\bar{a}} < n$ **do**
4: **for** $\mathcal{M}_i \in \mathcal{D}$ **do**
5: Run the policy $\pi_i \in \arg\min_{\pi\in\Pi} \mathbb{E}[X(\bar{s}|\mathcal{M}_i, \pi)]$ for two episodes
6: **if** $\bar{s}$ is reached **then**
7: Take action $\bar{a}$ and collect the next state s'
8: Update $N_{\bar{s}\bar{a}} = N_{\bar{s}\bar{a}} + 1, \mathcal{X} = \mathcal{X} \bigcup (s')$
9: **end if**
10: **end for**
11: **end while**
12: **output** the set $\mathcal{X} = (s'_1, \ldots, s'_n)$

Assumption 2 (Reachable MDPs)**.** *An MDP $\mathcal{M}_i$ is reachable if it holds $\min_{\pi\in\Pi} \mathbb{E}[X(s|\mathcal{M}_i, \pi)] \leq T/2, \forall s \in \mathcal{S}$.*

With the combination of the latter assumptions, we can directly adapt the algorithmic solution in Chen et al. (2022) to the finite-horizon setting.[7] We report the pseudocode of the resulting procedure in Algorithm 1.

The procedure consists of two stages: An "Identify" stage aiming at identifying the test task with high probability (lines 2-11) and a "Commit" stage in which the collected information is exploited (line 12). The "Identify" stage works as follows. At each iteration, a pair of MDPs are drawn from the set of potential test tasks (line 3). A sampling routine (Algorithm 2) is invoked (line 5) to collect samples from the state-action pair where the transition models of the two tasks differ the most (see line 4). The collected information is used to eliminate the task that is less likely to be the test task within the drawn pair (lines 6-10). The "Identify" stage ends when the set of potential tasks $\mathcal{D}$ is reduced to a single candidate. The "Commit" stage then runs the optimal policy of the candidate task for the remaining episodes.

[7]We refer to Algorithm 1 in Chen et al. (2022), which we name here the "Identify-then-Commit" algorithm.

We can provide the following regret upper bound for Algorithm 1 by adapting (Chen et al., 2022, Theorem 1).

Theorem 3.1 (Chen et al. 2022)**.** *Let $\mathcal{M}$ be a set of MDPs for which Assumption 1, 2 hold. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{TM^2 \log(MH) \log^2(SMH/\lambda)}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 1 with inputs $\mathcal{D} = \mathcal{M}$ and $n = \frac{c \log^2(SMH/\lambda) \log(MH)}{\lambda^4}$ for a sufficiently large constant c.

The latter result is promising as it provides a fast rate with only logarithmic dependencies on H, but it also scales superlinearly with the size of the set $\mathcal{M}$, which is less than ideal for larger task sets. A natural question that arises is whether this is the best we can achieve under the considered separation condition. The latter is arguably a very important question as the separation condition is the minimal structural assumption that makes test-time regret minimization "interesting", statistically separating the problem from RL. In the next section, we provide an answer through a lower bound specifically designed for this setting.

4. A Lower Bound for Test-Time Regret Minimization under Separation

In this section, we analyze the statistical barrier for test-time regret minimization under separation (Assumption 1) by providing a novel lower bound. Formally,

Theorem 4.1 (Lower bound)**.** *Let $\mathcal{M}$ be a set of MDPs for which Assumptions 1, 2 hold. Let $T > M$ and $M - 1 \leq H \leq C$ for some constant $C < \infty$. For any $\mathcal{M}_i \in \mathcal{M}$, algorithm $\mathbb{A}$, and confidence $\delta \in (0, 1)$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \Omega\left(\frac{TM \log(H)}{\lambda} \log\left(\frac{1}{\delta}\right)\right)$$

with probability at least $1 - \delta$.

Interesting observations come through the lenses of the result above. The lower bound shows that the regret rate of Algorithm 1 (Chen et al., 2022) matches the optimal dependencies in H, T factors, while it nearly matches the dependencies with λ and the size of the set of tasks M. The factor of $1/\lambda$ is not surprising, tying the complexity of the problem to how hard it is to distinguish one task from another. The result has a fairly negative flavor on the dependency with M instead. It demonstrates that the regret of any algorithm achieving fast rate $\log(H)$ has to scale at least linearly with M under separation, essentially implying that those algorithms are unfit for large sets of tasks.

Unfortunately, the latter settings in which the size of $\mathcal{M}$ may grow exponentially with the size of the tasks, or even be infinite (e.g., when tasks are continuous), are extremely relevant in practice, and it is arguably where the promises of meta RL for improved efficiency are the most enticing.

An open question remains on whether there exist relevant meta RL settings in which the structure of the problem can be further exploited to achieve fast rates $\log(H)$ while avoiding the dependence with M. In the next section, we discuss structural assumptions that go beyond the separation condition and allow to obtain the most efficient algorithms for test-time regret minimization.

Before that, we briefly sketch the main components of the proof of Theorem 4.1, which make use of original techniques and constructions that may be of independent interest. We defer thorough derivations to Appendix A.2. Note that a gentler introduction to the lower bound in the meta bandit setting is reported in Appendix C.

4.1. Proof Sketch

The key to our proof is to design a hard instance that links test-time regret minimization to the problem of best policy identification (Fiechter, 1994), and then invokes a lower bound to the sample complexity of the latter to derive the result. While instance-dependent lower bounds of this kind exist in the literature (e.g., Al Marjani et al., 2021; Wagenmaker et al., 2022; Al-Marjani et al., 2023), as a preliminary step we derive a result that is specifically tailored to our setting. Here we do not report derivations, which are non-trivial adaptations from a lower bound for BPI in infinite-horizon MDPs (see Al Marjani et al., 2021, Proposition 2). We leave a detailed description to Appendix B.

In Figure 1, we report a visualization of the instance constructed to derive the lower bound, which consists of M MDPs having $2M + 3$ states and 2 actions each, a high-rewarding state s_H, and an absorbing state s_L with zero reward. It is easy to see that the optimal policy for $\mathcal{M}_i$ goes to the state s_i and then takes action a_1, which gives the highest probability to visit s_H. However, taking action a_1 in every other state $s_j \in (s_1, \dots, s_M) \setminus s_i$ is only slightly sub-optimal, meaning that regret minimization is hard. Instead, it is easier to identify the test task $\mathcal{M}_i$ first, by playing action a_1 in the states $s_j \in (s_{M+1}, \dots, s_{2M})$, for which at least one (s_j, a_1) pair is guaranteed to reveal the test task against any other, and then to play the optimal policy thereafter. To formalize the latter intuition, we note that identifying the test task is equivalent to a BPI problem for this instance and we center the proof around the event

$$\mathcal{E} := \{\text{"best policy is identified within } H \text{ episodes"}\}.$$

Then, we show that the regret is lower bounded by $\Omega(\sqrt{H})$ when $\mathcal{E}$ does not hold, which implies that solving the BPI problem is necessary to obtain the best rate.[8] At this point,

[8]Note that this does not prescribe how the algorithm collect

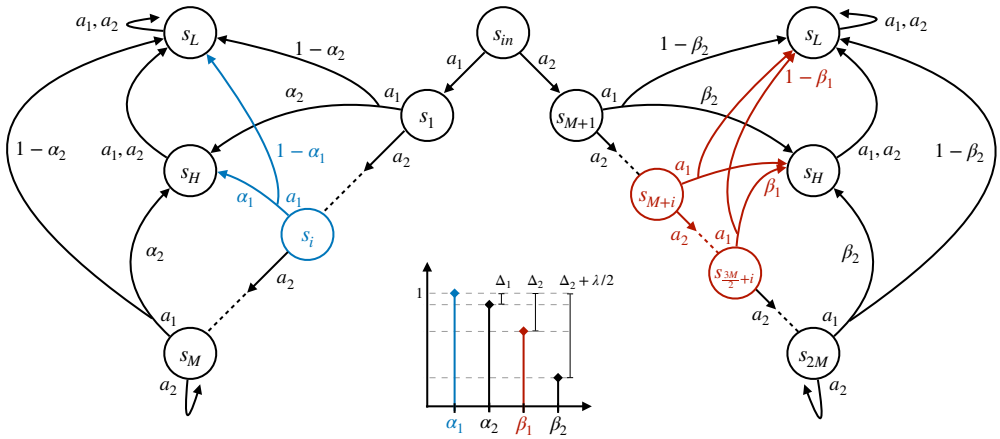

Figure 1. Visualization of the MDP $\mathcal{M}_i$ in the lower bound instance. Note that the role of state s_i and $s_{M+i}, \dots, s_{\frac{3M}{2}+i}$ change for every MDP in $\mathcal{M}$. Also note that s_L, s_H on the left and right refer to the same pair of states, which are reported twice only to ease inspection. The bottom chart report the specification of the transition probabilities. The values of Δ_1, Δ_2 are designed to be small enough to make the optimal policy hard to identify playing only slightly sub-optimal policies and large enough to penalize easy identification, respectively.

we invoke the BPI lower bound in Lemma B.1, which simultaneously guarantees that $\mathcal{E}$ holds with probability at least $1-\delta$ and that the regret is lower bounded by

$$R_H(\mathcal{M}_i, \mathbb{A}) \geq \mathbb{E}[\tau]\Delta_2$$

where $\mathbb{E}[\tau] \geq TM/\lambda^2 \log(1/2.4\delta)$ is the sample complexity of the BPI problem on the constructed instance and $\Delta_2 := V_i^* - V(\pi) = \log(H)/\sqrt{H}$ is the value gap of playing a sub-optimal policy to identify the task. Theorem 4.1 is obtained through additional algebraic manipulations.

5. Strong Identifiability: Beyond Separation for Faster Rates

In the previous section, we have settled the statistical complexity of test-time regret minimization in meta RL with a finite set of tasks under a common separation condition (Assumption 1 has been previously considered in Chen et al. 2022 as is, while a stronger version of the condition has been used in Kwon et al. 2021b).

Especially, we provided a lower bound to the test-time regret of order $\Omega(TM\log(H)/\lambda)$ and we showed that an "Identify-then-Commit" strategy (Chen et al., 2022) leads to a nearly matching upper bound $\mathcal{O}(TM^2\log(MH)/\lambda^4)$. Most importantly, our analysis shows that a linear dependence with the size M of the set of tasks cannot be avoided under the considered separation condition, at least if we aim for a fast rate in H. In this section, we present stronger yet reasonable assumptions that allow for faster rates of the test-time regret. These assumptions implicitly discriminate the problem instances, which we call *strongly identifiable*, that are truly worth framing into a meta RL paradigm, as the latter provides undeniable statistical benefits over RL.

Definition 2 (Strong identifiability)**.** *Let $\mathcal{M}$ be a set of MDPs for which Assumption 1 hold. $\mathcal{M}$ is* strongly identifiable *if, for all $\mathcal{M}_i \in \mathcal{M}$, we can identify $\mathcal{M}_i$ with high probability by only playing policies from a restricted policy class Π^{id} where $|\Pi^{id}| \leq \gamma M$ for some factor $\gamma \in (0,1)$.*

Next, we provide three instances of strong identifiability, with specialized algorithms and corresponding analyses, while investigating a broader range of strongly identifiable problems is an interesting directions for future works.

5.1. Meta RL with Clustering

Let us assume that the tasks in $\mathcal{M}$ can be grouped in coherent *clusters* admitting a peculiar and testable property that allows to efficiently discriminate one cluster from the others. Formally,

Assumption 3 ($KN\lambda$-clustering)**.** *Let $\mathcal{C} = (\mathcal{C}_k)_{k\in[K]}$ be a partition of $\mathcal{M}$ such that $|\mathcal{C}_k| \leq N$ for all $k \in [K]$ and $N > K$. For all $\mathcal{C}_k \in \mathcal{C}$, there exists $(s,a) \in \mathcal{S}\times\mathcal{A}$ such that $\min_{\mathcal{M}_i\in\mathcal{C}_k} \min_{\mathcal{M}_j\in\mathcal{M}\setminus\mathcal{C}_k} \|(p_i - p_j)(\cdot|s,a)\|_1 \geq \lambda$.*

Further, we consider a slightly stronger reachability condition within a cluster, which essentially assures that a state can be reached throughout the cluster with the same policy.

Assumption 4 (Cluster reachability)**.** *Let $C_k \in \mathcal{C}$ be a cluster of reachable MDPs. For all $\mathcal{M}_i, \mathcal{M}_j \in \mathcal{C}_k$ and $s \in \mathcal{S}$, it holds $\mathbb{E}[X(s|\mathcal{M}_j, \pi_i)] \leq T/2$ for $\pi_i \in \arg\min_{\pi\in\Pi} \mathbb{E}[X(s|\mathcal{M}_i, \pi)]$.*

The combination of latter assumptions allows to take a different algorithmic approach w.r.t. the standard separation condition (Assumption 1). Essentially, we can split the

samples. It tells that, whatever the sampling strategy, the BPI problem has to be solved within H episodes.

Algorithm 3 Double-Identify-then-Commit

input set of MDPs $\mathcal{M}$, clusters $\mathcal{C}$, visitation count $n_{\mathcal{C}}$
Initialize $\mathcal{D} = (\mathcal{M}_k)_{k\in[K]}$ with $\mathcal{M}_k$ from $\mathcal{C}_k$ randomly
while $|\mathcal{C}| > 1$ **do**
  Draw $\mathcal{C}_k$ from $\mathcal{C}$ randomly and let $(\mathcal{M}_1, \mathcal{M}_2, \bar{s}, \bar{a}) \in \arg\min_{\mathcal{M}_i\in\mathcal{C}_k, \mathcal{M}_j\in\mathcal{M}\setminus\mathcal{C}_k} \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \|(p_i - p_j)(\cdot|s,a)\|_1$
  Call Algorithm 2 with $\mathcal{D}, (\bar{s}, \bar{a}), n_{\mathcal{C}}$ to collect $\mathcal{X}$
  if $\exists s' \in \mathcal{X} : p_1(s'|\bar{s},\bar{a}) = 0$ or $\prod_{s'\in\mathcal{X}} \frac{p_1(s'|\bar{s},\bar{a})}{p_2(s'|\bar{s},\bar{a})} \geq 1$ **then**
    Update $\mathcal{C} = \mathcal{C}_k$
  else
    Update $\mathcal{C} = \mathcal{C} \setminus \mathcal{C}_k$
  end if
end while
Take $\hat{\mathcal{C}} \in \mathcal{C}$ and run Algorithm 1 with $\mathcal{D} = \hat{\mathcal{C}}, n = \frac{c\log^2(SNH/\lambda)\log(NH)}{\lambda^4}$ for the remaining episodes

"Identify" stage of Algorithm 1 into two separate phases: First, we identify the cluster to which the test task belongs with high probability, then, we call Algorithm 1 to identify the test task within the cluster selected in the previous step. We call this procedure "Double-Identify-then-Commit" to underline the two-phase structure of the "Identify" stage. We report the pseudocode of the procedure in Algorithm 3.

Since both the number of clusters K and the size of each cluster are smaller than the size M of the set of MDPs $\mathcal{M}$ under Assumption 3, the set of MDPs is strongly identifiable. Thus, we can expect the "Identify" stage of Algorithm 3 to be more efficient than the same stage of Algorithm 1 taking $\mathcal{M}$ as input, which is confirmed by the result below.

Theorem 5.1. *Let $\mathcal{M}$ be a set of MDPs for which Assumption 2, 3, 4 hold. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{T(K^2+N^2)\log(NH)\log^2(\frac{SNH}{\lambda})}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 3 with inputs $\mathcal{M}, \mathcal{C}$, and $n_{\mathcal{C}} = \frac{c\log^2(SKH/\lambda)\log(KH)}{\lambda^4}$ for a sufficiently large constant c.

While the latter proves a significant speed-up of the "Identify" stage, which can be relevant in practical settings, the regret does not essentially escape the quadratic dependency on the number of tasks M. Indeed, if we look into the rate we get $K^2 + N^2 \approx K^2 + M^2/K^2$ as opposed to M^2 with the separation condition (Assumption 1) alone. At this point, it is interesting to ask which kind of structure can lead to a regret rate that is truly sublinear in the number of tasks M. In the next section, we achieve that by considering a tree structure on the set of MDPs $\mathcal{M}$.

5.2. Meta RL with a Tree Structure

Let us suppose that for any set of candidate test MDPs we extract from the original set there always exist a state-action pair from which we can collect information to *split* the candidates in two subsets. For instance, we can think of a set of indoor physical domains with peculiar configurations, e.g., some with a window on the right-hand side and some without. If those states not only exist, but they are also easy to reach, we can then sequentially split the original set of MDPs to increasingly smaller subsets, until a single candidate remains. To formalize this intuition, we first define what we mean for a state-action to be *easy-to-reach*.

Algorithm 4 Tree-Identify-then-Commit

input set of MDPs $\mathcal{M}$ and visitation count n
Initialize $\mathcal{D} = \mathcal{M}$
while $|\mathcal{D}| > 1$ **do**
  Compute $\mathcal{M}_1, \mathcal{M}_2, \mathcal{D}^+, \mathcal{D}^-, (\bar{s}, \bar{a})$ by solving

$$\min_{\mathcal{M}_i\in\mathcal{D}^+, \mathcal{M}_j\in\mathcal{D}^-} \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \|(p_i - p_j)(\cdot|s,a)\|_1$$
$$\text{s. t. } \mathcal{D}^+ \cup \mathcal{D}^- = \mathcal{D}, (s,a) \text{ is strongly reachable}$$

  Initialize $N_{\bar{s}\bar{a}} = 0$ and $\mathcal{X} = \emptyset$
  while $N_{\bar{s}\bar{a}} < n$ **do**
    Run the policy $\pi \in \arg\min_{\pi\in\Pi} \mathbb{E}[X(\bar{s}|\mathcal{M}_1, \pi)]$ for two episodes
    if $\bar{s}$ is reached **then**
      Take action $\bar{a}$ and collect the next state s'
      Update $N_{\bar{s}\bar{a}} = N_{\bar{s}\bar{a}} + 1, \mathcal{X} = \mathcal{X} \bigcup (s')$
    end if
  end while
  if $\exists s' \in \mathcal{X} : p_1(s'|\bar{s},\bar{a}) = 0$ or $\prod_{s'\in\mathcal{X}} \frac{p_1(s'|\bar{s},\bar{a})}{p_2(s'|\bar{s},\bar{a})} \geq 1$ **then**
    Update $\mathcal{D} = \mathcal{D}^+$
  else
    Update $\mathcal{D} = \mathcal{D}^-$
  end if
end while
Take $\hat{\mathcal{M}} \in \mathcal{D}$ and run $\hat{\pi} \in \arg\max_{\pi\in\Pi} V_{\hat{\mathcal{M}}}(\pi)$ for the remaining episodes

Assumption 5 (Strong reachability)**.** *Let $\mathcal{D} \subseteq \mathcal{M}$ a set of MDPs. We say that $(s,a) \in \mathcal{S}\times\mathcal{A}$ is* strongly reachable *in $\mathcal{D}$ if, for all $\mathcal{M}_i, \mathcal{M}_j \in \mathcal{D}$, it holds $\mathbb{E}[X(s,a|\mathcal{M}_j, \pi_i)] \leq T/2$ where $\pi_i \in \arg\min_{\pi\in\Pi} \mathbb{E}[X(s,a|\mathcal{M}_i, \pi)]$.*

Then, we can formally define our structural assumption.

Assumption 6 (β-tree)**.** *Let $\mathcal{M}$ be a set of MDPs and let $\beta \in (1/2, 1)$. For all $\mathcal{D} \subseteq \mathcal{M}$, there exist a strongly reachable $(s,a) \in \mathcal{S}\times\mathcal{A}$ and a partition $(\mathcal{D}^+, \mathcal{D}^-)$ of $\mathcal{D}$ such that $\min_{\mathcal{M}_i\in\mathcal{D}^+} \min_{\mathcal{M}_j\in\mathcal{D}^-} \|(p_i - p_j)(\cdot|s,a)\|_1 \geq \lambda$ where $\max(|\mathcal{D}^+|, |\mathcal{D}^-|)/|\mathcal{D}| \leq \beta$.*

The latter induces a tree structure on the set of MDPs, such that we can render the identification problem as traversing a (binary) decision tree. The routine works as follows, we start with the initial set of MDPs $\mathcal{M}$, then we look for a state-action pair inducing a large enough split, which we visit several times to understand whether the test task belongs to $\mathcal{D}^+$ or $\mathcal{D}^-$. We can iterate these steps again on the resulting set, i.e., $\mathcal{D}^+$ or $\mathcal{D}^-$, so that the set of candidate

MDPs iteratively shrinks towards one single MDP, which is our candidate test task for the remaining episodes. The pseudocode for the procedure is reported in Algorithm 4.

Clearly, the cost of the "Identify" stage of Algorithm 4, i.e., the number of times the while loop between lines 2-12 has to be executed, is tied to the depth of the tree structure, which is $\log_{1/\beta}(M)$ in the worst case. With this consideration, we can derive the following regret upper bound.

Theorem 5.2. *Let $\mathcal{M}$ be a set of MDPs for which Assumption 5, 6 hold and let $d = \log_{1/\beta}(M)$. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{Td\log(dH)\log^2(SdH/\lambda)}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 4 with inputs $\mathcal{M}$ and $n = \frac{c\log^2(SdH/\lambda)\log(dH)}{\lambda^4}$ for a sufficiently large constant c.

5.3. Meta RL with a few Revealing Policies

What happens if we assume that we can extract from $\mathcal{M}$ a small set of *revealing* policies that allow to traverse all of the revealing state-action pairs in expectation? We formalize this intuition in the following assumption. Then, we provide an algorithm and corresponding regret analysis that shows the quadratic dependence with M can be escaped.

Before stating the assumption, we define the set of revealing state-action pairs $\overline{\mathcal{SA}} \subseteq \mathcal{S} \times \mathcal{A}$ such that $\forall \mathcal{M}_i, \mathcal{M}_j \in \mathcal{M}, \exists (s,a) \in \overline{\mathcal{SA}} : \|(p_i - p_j)(\cdot|s,a)\|_1 \geq \lambda$. Then,

Assumption 7 (Revealing policy set)**.** *Let $\mathcal{M}$ be a set of MDPs for which Assumption 1, 2 hold. There exists a set of policies Π_I of size $|\Pi_I| \leq I$ such that $\forall \mathcal{M}_i \in \mathcal{M}$ it holds $\max_{\pi \in \Pi_I} \min_{(s,a)\in\overline{\mathcal{SA}}} P(X(s,a|\mathcal{M}_i,\pi) \leq T) \geq 1/2$.*

In Algorithm 5, we report the pseudocode of a procedure that exploits Assumption 7 to increase the efficiency of the data collection for the "Identify" stage. First, the policies in Π_I are repeatedly deployed with the goal of collecting n samples for each revealing state-action pair in $\overline{\mathcal{SA}}$ within an "Explore" stage (lines 3-11). Then, differently from previous approaches, the "Identify" stage (lines 13-21) is carried out offline with the previously collected data, until the set of tasks $\mathcal{D}$ is reduced to one candidate. Finally, in the "Commit" stage (line 22), the optimal policy $\hat{\pi}$ in the identified test task $\hat{\mathcal{M}}$ is deployed for the remaining episodes.

Now, we provide an upper bound to its test-time regret.

Theorem 5.3. *Let $\mathcal{M}$ be a set of MDPs for which Assumption 7 holds. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{TI\log(MH)\log^2(SMH/\lambda)}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 5 with inputs $\mathcal{M}, \Pi_I, \overline{\mathcal{SA}}$, $n = \frac{c\log^2(SMH/\lambda)\log(MH)}{\lambda^4}$ for a sufficiently large constant c.

Algorithm 5 Explore-Identify-then-Commit

1: **input** set of MDPs $\mathcal{M}$, set of policies Π_I, set of state-action pairs $\overline{\mathcal{SA}}$, visitation count n
2: Initialize $N_{\bar{s}\bar{a}} = 0$ and $\mathcal{X}_{\bar{s}\bar{a}} = \emptyset$ for all $(\bar{s},\bar{a}) \in \overline{\mathcal{SA}}$
3: **while** $\min(N_{\bar{s}\bar{a}}) < n$ **do**
4: **for** $\pi_i \in \Pi_I$ **do**
5: Run the policy π_i for two episodes
6: **if** $\bar{s}$ such that $(\bar{s},\cdot) \in \overline{\mathcal{SA}}$ is reached **then**
7: Take action $\bar{a}$ and collect the next state s'
8: Update $N_{\bar{s}\bar{a}} = N_{\bar{s}\bar{a}} + 1$ and $\mathcal{X}_{\bar{s}\bar{a}} = \mathcal{X}_{\bar{s}\bar{a}} \bigcup (s')$
9: **end if**
10: **end for**
11: **end while**
12: Initialize $\mathcal{D} = \mathcal{M}$
13: **while** $|\mathcal{D}| > 1$ **do**
14: Draw $\mathcal{M}_1, \mathcal{M}_2$ from $\mathcal{D}$ randomly
15: Let $(\bar{s},\bar{a}) \in \arg\max_{(s,a)\in\overline{\mathcal{SA}}} \|(p_1 - p_2)(\cdot|s,a)\|_1$
16: **if** $\exists s' \in \mathcal{X}_{\bar{s}\bar{a}} : p_2(s'|\bar{s},\bar{a}) = 0$ or $\prod_{s'\in\mathcal{X}_{\bar{s}\bar{a}}} \frac{p_1(s'|\bar{s},\bar{a})}{p_2(s'|\bar{s},\bar{a})} \geq 1$ **then**
17: Eliminate $\mathcal{M}_2$ from $\mathcal{D}$
18: **else**
19: Eliminate $\mathcal{M}_1$ from $\mathcal{D}$
20: **end if**
21: **end while**
22: Take $\hat{\mathcal{M}} \in \mathcal{D}$ and run $\hat{\pi} \in \arg\max_{\pi\in\Pi} V_{\hat{\mathcal{M}}}(\pi)$ for the remaining episodes

The latter result provides a fast rate for the test-time regret when $I < M^2$, i.e., the problem is strongly identifiable. However, Algorithm 5 takes a set of revealing policies as input, leaving to the training phase the burden of providing it. Here we discuss briefly how this requirement can be avoided, while we leave as future work a thorough investigation.

Sampling from revealing policies. Instead of pre-computing a set of revealing policies, one may try to replicate the sampling of the "Explore" stage (lines 3-11) while interacting with the test task. In Algorithm 6 we describe an adaptive sampling procedure for this purpose. The idea is to iteratively compute a policy that maximizes the (expected) visitation of the revealing state-action pairs left uncovered. The latter objective can be encoded into a trajectory reward defined as in line 4. Once such a policy is computed (line 5), it can be deployed in the test task to collect a trajectory (line 6) and to update the set of the remaining state-action pairs to be covered (line 7). The process repeats until of the revealing state-action pairs have been visited (lines 3-9).

To implement Algorithm 6, two technical hurdles have to be overcome. First, the trajectory reward in line 4 is non-standard. A naive solution is to encode in the state space the necessary information to define the reward on a state level or, alternatively, to adapt algorithms to deal with trajectory rewards (Efroni et al., 2021; Chatterji et al., 2021). Secondly, the problem in line 5 is akin to a robust MDP, which is known to be intractable in general (Wiesemann et al., 2013). To address the latter, a policy gradient method (Kumar et al.,

Algorithm 6 Revealing Policies Sampling

1: **input** set of MDPs $\mathcal{M}$, set of state-action pairs $\overline{\mathcal{SA}}$
2: Initialize $h \leftarrow 1$ and the sets $\overline{\mathcal{SA}}_1 = \overline{\mathcal{SA}}$, $\mathcal{X}_{\bar{s}\bar{a}} = \emptyset$
3: **while** $\overline{\mathcal{SA}}_h$ is not empty **do**
4: Define $\widetilde{r}(\tau) := \sum_{(s,a)\in\overline{\mathcal{SA}}_h} \mathbb{1}\big(\exists (s_t, a_t) \in \tau : (s_t, a_t) = (s, a)\big)$
5: Solve $\pi_h \in \arg\max_{\pi\in\Pi} \min_{\mathcal{M}_i\in\mathcal{M}} \mathbb{E}_{\tau\sim\pi,\mathcal{M}_i}[\widetilde{r}(\tau)]$
6: Take trajectory τ_h with π_h and populate $\mathcal{X}_{\bar{s}\bar{a}}$
7: Compute $\overline{\mathcal{SA}}_{h+1} = \overline{\mathcal{SA}}_h \setminus \big((s, a) \in \overline{\mathcal{SA}}_h : (s, a) \in \tau_h\big)$
8: Increment $h \leftarrow h + 1$
9: **end while**
10: **output** the sets $\mathcal{X}_{\bar{s}\bar{a}}$

2024) can be used to get an approximate solution efficiently, although this may slow down convergence of the while loop (lines 3-9) as opposed to deploying (in line 6) the exact solution of the problem.

6. Related Works

While we are not aware of any previous work explicitly addressing test-time regret minimization under perfect training within the meta RL paradigm, slight variations of our problem setting have been considered in different domains.

As we extensively reported in the previous sections, Chen et al. (2022) provides theoretical results on the sim-to-real gap in domain randomization that can be transferred to our setting almost verbatim. Their *sim* stage coarsely correspond to our training, while their *real* stage is our test task, for which they study a notion of regret against the optimal policy specific to the task. Differently from our setting, they consider infinite-horizon MDPs and they analyse the regret rate of a Bayes-optimal policy instead of our adaptive algorithms. Notably, they assume to have recovered an exact Bayes-optimal policy from simulations, which is similar in nature to our perfect training assumption. In their setting of interest, they provide regret guarantees of order $\mathcal{O}(M^2 \log(MH))$ for finite set of tasks under separation, $\mathcal{O}(M^2\sqrt{H})$ for finite set of tasks without separation, and $\mathcal{O}(\sqrt{d_E H})$ for infinite set of tasks with function approximation, where d_E is the eluder dimension of the function class. Finally, they provide a lower bound $\Omega(H)$ for finite sets without separation. We fill the gaps in their analysis providing a lower bound specialized for the separation condition and assumptions beyond separation for faster rates.

The work by Ye et al. (2023) studies generalization guarantees of pre-training in RL, in terms of Bayesian and frequentist regret, zero-shot generalization or with additional test-time interaction. The latter setting in the frequentist regret formulation is the analogous to our test-time regret minimization, for which they provide a policy-collection elimination algorithm with regret of order $\mathcal{O}(\sqrt{\mathbb{C}(P)H})$, where $\mathbb{C}(P)$ is a measure of complexity of the task distribution. Although the complexity $\mathbb{C}(P)$ can capture structured finite or infinite set of tasks, it does not allow for escaping the $\sqrt{H}$ regret of standard RL.

Kwon et al. (2021b) address regret minimization in Latent MDPs (LMDPs). In their setting, at every episode a task is drawn from a set of finite but unknown tasks, for which the agent tries to minimize the regret against an optimal policy for the specific task. Essentially, LMDPs formalism can be seen as a variation of our setting in which the test task is not persistent but changes at every episode, and the agent does not have full knowledge of the transition dynamics of the tasks, which have to be estimated from samples. In its full generality, LMDPs are statistically intractable. Analogously to our work, they consider a (stronger) version of the separation condition to achieve a regret rate of order $\mathcal{O}(\sqrt{MH})$ for their inherently harder setting. Further variations of LMDPs have been studied, including reward-mixing MDPs (Kwon et al., 2021a; 2023a), analogous of LMDPs with fixed dynamics but changing rewards, LMDPs with side information partially revealing the current task (Kwon et al., 2023b), and mixture of MDPs (Kausik et al., 2023).

7. Conclusion

In this paper, we provided a formal study on the statistical barriers of test-time regret minimization under strong structural assumptions, shedding light on when meta RL can be expected to provide significant benefits over standard RL.

First, we settled the complexity of test-time regret minimization under separation deriving a lower bound specialized for the assumption, for which only upper bounds were known in the literature. Then, to overcome the (super)linear dependency with the size of the set of tasks, we studied a family of structural assumptions beyond separation, i.e., the set of tasks can be grouped in coherent clusters, the test task can be identified following a tree structure, or a small set of revealing policies can be deployed to identify the test task.

Future works may extend our results in various directions. Additional structural assumptions fitting in the broad family of strong identifiability may be investigated, as well as separation conditions at the level of trajectory generation processes rather than single state-action pairs. Understanding the impact of approximate training, i.e., only imperfect estimates of the tasks' transition dynamics are available, on our test-time regret minimization results is important to bring our analysis in a more realistic setting. Whether there exists minimal assumptions that allow for fast rates of order $\log(H)$ for infinite set of tasks is also a question worth investigating.

Finally, we hope that our theoretical study can bring inspiration to design practical algorithms for improved efficiency of test-time learning in meta RL.

Acknowledgments

This research was Funded by the European Union (ERC, Bayes-RL, 101041250). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency (ERCEA). Neither the European Union nor the granting authority can be held responsible for them.

The authors thank Shie Mannor for insightful discussions on an early draft of this work.

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

Agarwal, A., Kakade, S., and Yang, L. F. Model-based reinforcement learning with a generative model is minimax optimal. In *Conference on Learning Theory*, 2020.

Al Marjani, A. and Proutiere, A. Adaptive sampling for best policy identification in Markov decision processes. In *International Conference on Machine Learning*, 2021.

Al Marjani, A., Garivier, A., and Proutiere, A. Navigating to the best policy in Markov decision processes. In *Advances in Neural Information Processing Systems*, 2021.

Al-Marjani, A., Tirinzoni, A., and Kaufmann, E. Towards instance-optimality in online pac reinforcement learning. *arXiv preprint arXiv:2311.05638*, 2023.

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: 325–349, 2013.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, 2017.

Bai, Y., Xie, T., Jiang, N., and Wang, Y.-X. Provably efficient q-learning with low switching cost. In *Advances in Neural Information Processing Systems*, 2019.

Chatterji, N., Pacchiano, A., Bartlett, P., and Jordan, M. On the theory of reinforcement learning with once-per-episode feedback. In *Advances in Neural Information Processing Systems*, 2021.

Chen, X., Hu, J., Jin, C., Li, L., and Wang, L. Understanding domain randomization for sim-to-real transfer. In *International Conference on Learning Representations*, 2022.

Dann, C. and Brunskill, E. Sample complexity of episodic fixed-horizon reinforcement learning. In *Advances in Neural Information Processing Systems*, 2015.

Dann, C., Li, L., Wei, W., and Brunskill, E. Policy certificates: Towards accountable reinforcement learning. In *International Conference on Machine Learning*, 2019.

Duan, Y., Schulman, J., Chen, X., Bartlett, P. L., Sutskever, I., and Abbeel, P. Rl^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016.

Efroni, Y., Merlis, N., and Mannor, S. Reinforcement learning with trajectory feedback. In *AAAI Conference on Artificial Intelligence*, 2021.

Fawzi, A., Balog, M., Huang, A., Hubert, T., Romera-Paredes, B., Barekatain, M., Novikov, A., R Ruiz, F. J., Schrittwieser, J., Swirszcz, G., et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610(7930):47–53, 2022.

Fiechter, C.-N. Efficient reinforcement learning. In *Conference on Learning Theory*, 1994.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017.

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

Ghavamzadeh, M., Mannor, S., Pineau, J., and Tamar, A. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.

Jaksch, T., Ortner, R., and Auer, P. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11:1563–1600, 2010.

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

Kaufmann, E., Ménard, P., Domingues, O. D., Jonsson, A., Leurent, E., and Valko, M. Adaptive reward-free exploration. In *Algorithmic Learning Theory*, 2021.

Kaufmann, E., Bauersfeld, L., Loquercio, A., Müller, M., Koltun, V., and Scaramuzza, D. Champion-level drone racing using deep reinforcement learning. *Nature*, 620 (7976):982–987, 2023.

Kausik, C., Tan, K., and Tewari, A. Learning mixtures of markov chains and mdps. In *International Conference on Machine Learning*, 2023.

Kumar, N., Derman, E., Geist, M., Levy, K. Y., and Mannor, S. Policy gradient for rectangular robust markov decision processes. In *Advances in Neural Information Processing Systems*, 2024.

Kwon, J., Efroni, Y., Caramanis, C., and Mannor, S. Reinforcement learning in reward-mixing mdps. In *Advances in Neural Information Processing Systems*, 2021a.

Kwon, J., Efroni, Y., Caramanis, C., and Mannor, S. Rl for latent mdps: Regret guarantees and a lower bound. In *Advances in Neural Information Processing Systems*, 2021b.

Kwon, J., Efroni, Y., Caramanis, C., and Mannor, S. Reward-mixing mdps with few latent contexts are learnable. In *International Conference on Machine Learning*, 2023a.

Kwon, J., Efroni, Y., Mannor, S., and Caramanis, C. Prospective side information for latent mdps. *arXiv preprint arXiv:2310.07596*, 2023b.

Lattimore, T. and Szepesvári, C. *Bandit algorithms*. Cambridge University Press, 2020.

Ménard, P., Domingues, O. D., Jonsson, A., Kaufmann, E., Leurent, E., and Valko, M. Fast active learning for pure exploration in reinforcement learning. In *International Conference on Machine Learning*, 2021.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533, 2015.

Osband, I. and Van Roy, B. On lower bounds for regret in reinforcement learning. *arXiv preprint arXiv:1608.02732*, 2016.

Puterman, M. L. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Rakelly, K., Zhou, A., Finn, C., Levine, S., and Quillen, D. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *International Conference on Machine Learning*, 2019.

Rimon, Z., Tamar, A., and Adler, G. Meta reinforcement learning with finite training tasks-a density estimation approach. In *Advances in Neural Information Processing Systems*, 2022.

Schmidhuber, J. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International conference on machine learning*, 2015.

Simchowitz, M., Tosh, C., Krishnamurthy, A., Hsu, D. J., Lykouris, T., Dudik, M., and Schapire, R. E. Bayesian decision-making under misspecified priors with applications to meta-learning. In *Advances in Neural Information Processing Systems*, 2021.

Stiennon, N., Ouyang, L., Wu, J., Ziegler, D., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano, P. F. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, 2020.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Tamar, A., Soudry, D., and Zisselman, E. Regularization guarantees generalization in bayesian reinforcement learning through algorithmic stability. In *AAAI Conference on Artificial Intelligence*, 2022.

Tirinzoni, A., Al Marjani, A., and Kaufmann, E. Near instance-optimal pac reinforcement learning for deterministic mdps. In *Advances in Neural Information Processing Systems*, 2022.

Tirinzoni, A., Al-Marjani, A., and Kaufmann, E. Optimistic pac reinforcement learning: the instance-dependent view. In *Algorithmic Learning Theory*, 2023.

Wagenmaker, A. J., Simchowitz, M., and Jamieson, K. Beyond no regret: Instance-dependent pac reinforcement learning. In *Conference on Learning Theory*, 2022.

Wiesemann, W., Kuhn, D., and Rustem, B. Robust markov decision processes. *Mathematics of Operations Research*, 2013.

Xu, K., Hu, Z., Doshi, R., Rovinsky, A., Kumar, V., Gupta, A., and Levine, S. Dexterous manipulation from images: Autonomous real-world rl via substep guidance. In *IEEE International Conference on Robotics and Automation*, 2023.

Ye, H., Chen, X., Wang, L., and Du, S. S. On the power of pre-training for generalization in RL: Provable benefits and hardness. In *International Conference on Machine Learning*, 2023.

Zhao, M., Abbeel, P., and James, S. On the effectiveness of fine-tuning versus meta-reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022.

Zintgraf, L., Shiarlis, K., Igl, M., Schulze, S., Gal, Y., Hofmann, K., and Whiteson, S. Varibad: A very good method for bayes-adaptive deep rl via meta-learning. In *International Conference on Learning Representations*, 2019.

Zintgraf, L., Schulze, S., Lu, C., Feng, L., Igl, M., Shiarlis, K., Gal, Y., Hofmann, K., and Whiteson, S. Varibad: Variational bayes-adaptive deep rl via meta-learning. *The Journal of Machine Learning Research*, 22(1):13198–13236, 2021.

Zisselman, E., Lavie, I., Soudry, D., and Tamar, A. Explore to generalize in zero-shot rl. In *Advances in Neural Information Processing Systems*, 2023.

A. Missing Proofs

In this section, we provide complete derivations to prove the theoretical results presented in the paper.

A.1. Proof of Theorem 3.1

Here we prove the upper bound to the test-time regret under separation of Algorithm 1, which is a straightforward adaptation of the derivations in Chen et al. (2022, Theorem 5) to the finite-horizon setting.

Theorem 3.1 (Chen et al. 2022)**.** *Let $\mathcal{M}$ be a set of MDPs for which Assumption 1, 2 hold. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{TM^2 \log(MH) \log^2(SMH/\lambda)}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 1 with inputs $\mathcal{D} = \mathcal{M}$ and $n = \frac{c \log^2(SMH/\lambda)\log(MH)}{\lambda^4}$ for a sufficiently large constant c.

Proof. Analogously as in Chen et al. (2022, Theorem 5), the proof is based on showing that the true MDP $\mathcal{M}_i$ will not be eliminated from $\mathcal{D}$ (lines 6-10 in Algorithm 1) with probability at least $1 - 1/H$ (Chen et al., 2022, Lemma 4). Especially, we can write

$$\begin{aligned} &P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D}\text{”}\Big) \\ &= P\left(\bigcup_{m=1}^{M-1} \text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D} \text{ at iteration } m\text{”}\right) \\ &\leq \sum_{m=1}^{M-1} P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D} \text{ at iteration } m\text{”}\Big) \end{aligned}$$

by noting that the loop in lines 2-11 of Algorithm 1 is executed for $M - 1$ iterations and then applying a union bound. Next, we call Lemma A.1 to prove that the event “$\mathcal{M}_i$ is eliminated from $\mathcal{D}$ at iteration m” holds with probability less than $1/MH$.[9]

Then, we just need to prove that those n samples rquired by Lemma A.1 can be collected efficiently through the sampling routine in Algorithm 2, which is where our approach differs from Chen et al. (2022). In Lemma A.2, we provide a quick adaptation of their infinite-horizon communicating MDP setting to our finite-horizon reachable MDP setting.

Now let H_0 the number of episodes needed to collect n samples through Algorithm 2 for every time is called from Algorithm 1 (line 5), which is $M - 1$ times in total. We have

$$\mathbb{E}[H_0] = h_0 \leq 2(M-1)Mn \leq \frac{cM^2 \log^2(SMH/\lambda)\log(MH)}{\lambda^4}.$$

Finally, we can write

$$R_H(\mathcal{M}_i, \pi) = \mathbb{E}\left[\sum_{h=1}^{h_0} V_i^* - V_i(\pi_h)\right] + \mathbb{E}\left[\sum_{h=h_0}^{H} V_i^* - V_i(\hat{\pi})\right] \leq \frac{cTM^2 \log^2\left(\frac{SMH}{\lambda}\right)\log(MH)}{\lambda^4}$$

by noting that $\mathbb{E}[\sum_{h=1}^{h_0} V_i^* - V_i(\pi_h)] \leq \mathbb{E}[H_0]T$ through $r_i(s,a) \in [0,1], \forall \mathcal{M}_i \in \mathcal{M}$, and that $\mathbb{E}[\sum_{h=h_0}^{H} V_i^* - V_i(\hat{\pi})] \leq T$, as it is $V_i^* - V_i(\hat{\pi}) = 0$ with probability at least $1 - 1/H$ and $V_i^* - V_i(\hat{\pi}) \leq TH$ with probability at most $1/H$. □

Lemma A.1 (Chen et al. 2022)**.** *Let $\mathcal{X} = (s'_1, \dots, s'_n)$ be a set of $n = \frac{c \log^2(SMH/\lambda)\log(MH)}{\lambda^4}$ independent samples from $p_i(\cdot|\bar{s}, \bar{a})$ for a large enough constant c and let $\mathcal{M}_1$ be an MDP such that $\|(p_i - p_1)(\cdot|\bar{s}, \bar{a})\|_1 \geq \lambda$. Then, it holds*

$$\prod_{s' \in \mathcal{X}} \frac{p_i(s'|\bar{s}, \bar{a})}{p_1(s'|\bar{s}, \bar{a})} > 1$$

with probability at least $1 - 1/MH$.

[9] Note that the derivations in the corresponding (Chen et al., 2022, Lemma 5) apply verbatim as there is no assumption on how samples in $\mathcal{X}$ are collected, and whether they are coming from a finite-horizon or an infinite-horizon MDP.

Lemma A.2. *Let $\mathcal{M}_i \in \mathcal{D}$ an MDP and let H_0 a random variable denoting the number of episodes needed by Algorithm 2 to collect n samples from $p_i(\cdot|\bar{s},\bar{a})$ in $\mathcal{M}_i$. We can upper bound the expected number of episodes as $h_0 := \mathbb{E}[H_0] \leq 2Mn$.*

Proof. We can follow similar steps as in Chen et al. (2022, Lemma 7). From Assumption 2 we have that $\mathbb{E}(X(\bar{s}|\mathcal{M}_i,\pi_i)) \leq T/2$ for $\pi_i \in \arg\min_{\pi\in\Pi} \mathbb{E}[X(\bar{s}|\mathcal{M}_i,\pi)]$. Then, by applying the Markov's inequality $P(X(\bar{s}|\mathcal{M}_i,\pi_i) \geq T) \leq \mathbb{E}[X(\bar{s}|\mathcal{M}_i,\pi_i)]/T$ we get

$$X(\bar{s}|\mathcal{M}_i,\pi_i) \leq T \tag{2}$$

with probability at least $1/2$. Let Y be the random variable denoting the number of episodes needed to reach state $\bar{s}$. From (2), we have that $P(Y = k) \leq 1/2^k$, which gives $\mathbb{E}[Y] \leq \sum_{k=1}^{\infty} k/2^k = 2$. Since Algorithm 2 deploys $\pi_j \in \arg\min_{\pi\in\Pi} \mathbb{E}[X(\bar{s}|\mathcal{M}_j,\pi)]$ for all $\mathcal{M}_j \in \mathcal{D}$ (lines 4-9), then also π_i is deployed. □

A.2. Proof of Theorem 4.1

Here we provide complete derivations for the proof of the lower bound to the test-time regret under separation, which we briefly sketched in Section 4. In the proof, we will refer to the hard instance depicted in Figure 1 of the main paper.

Theorem 4.1 (Lower bound)**.** *Let $\mathcal{M}$ be a set of MDPs for which Assumptions 1, 2 hold. Let $T > M$ and $M-1 \leq H \leq C$ for some constant $C < \infty$. For any $\mathcal{M}_i \in \mathcal{M}$, algorithm $\mathbb{A}$, and confidence $\delta \in (0,1)$, we have*

$$R_H(\mathcal{M}_i,\mathbb{A}) = \Omega\left(\frac{TM\log(H)}{\lambda}\log\left(\frac{1}{\delta}\right)\right)$$

with probability at least $1-\delta$.

Proof. The key idea behind this proof is to construct an instance of the problem in which it is hard to minimize the regret without knowing the MDP, whereas it is easy to identify the MDP playing sub-optimal policies.

Hard instance. The instance consists of M MDPs having $2M+3$ states and 2 actions each. Figure 1 depicts the sample MDP $\mathcal{M}_i \in \mathcal{M}$, but all of the MDPs in the instance are similarly constructed. For any $\mathcal{M}_i \in \mathcal{M}$, the state s_{in} is the initial state such that $p(s_1|s_{in},a_1) = p(s_{M+1}|s_{in},a_2) = 1$, s_L is an absorbing state with $p(s_L|s_L,a_1) = p(s_L|s_L,a_2) = 1$, s_H is an high-reward state such that $p(s_L|s_H,a_1) = p(s_L|s_H,a_2) = 1$ and $r(s_H,a_1) = r(s_H,a_2) = 1$. For all the other states $s \in \mathcal{S}\setminus(s_H)$ we have $r(s,a_1) = r(s,a_2) = 0$. The states $s \in \mathcal{S}\setminus(s_H,s_L,s_{in})$ are arranged in two different chains: $(s_1,\dots,s_i,\dots,s_M)$ on the left and $(s_{M+1},\dots,s_{M+i},\dots,s_{2M})$ on the right, respectively. In every state of those chains, the action a_2 gives a deterministic transition to the next state in their respective chain, i.e., $p(s_{j+1}|s_j,a_2) = 1, \forall s_j \in (s_1,\dots,s_{M-1}) \cup (s_{M+1},\dots,s_{2M-1})$ and self-loops $p(s_M|s_M,a_2) = p(s_{2M}|s_{2M},a_2) = 1$ for the closing end of the chains. For any $\mathcal{M}_i \in \mathcal{M}$, the state s_i is the one leading to the state s_H with higher probability than all of the other states $p_i(s_H|s_i,a_1) = 1$. For all of the other states in the left chain, i.e., $s \in (s_1,\dots,s_M)\setminus(s_i)$, the transition to s_H has probability $p_i(s_H|s,a_1) = 1-\Delta_1 = 1-1/\sqrt{H}$. In the right chain, the states are divided in two groups of $M/2$ states, which are $\mathcal{G}_1 := (s_{M+i},\dots,s_{\frac{3M}{2}+i})$ and $\mathcal{G}_2 := (s_{M+1},\dots,s_{M+i-1}) \cup (s_{\frac{3M}{2}+i},\dots,s_{2M})$.[10] The transition model is equivalent within the two groups, which is $p_i(s_H|s\in\mathcal{G}_1,a_1) = 1-\Delta_2 = 1-\log(H)/\sqrt{H}$ and $p_i(s_H|s\in\mathcal{G}_2,a_1) = 1-\Delta_2-\lambda/2 = 1-\log(H)/\sqrt{H}-\lambda/2$. Thanks to the construction of $\mathcal{G}_1$ and $\mathcal{G}_2$, for every pair $\mathcal{M}_i,\mathcal{M}_j \in \mathcal{M}$ there is at least one state-action pair for which the λ-separation holds (Assumption 1). To ease the visual inspection of the sample MDP $\mathcal{M}_i$ in Figure 1, the state s_i and related transitions are blue colored, the states in $\mathcal{G}_1$ and related transitions are red colored.

Event. Together with the described instance, we use terminology from best policy identification (see Appendix B for details) to define a convenient event around which the analysis is centered. We consider a class of stopping rules τ such that $\mathbb{E}_{\mathcal{M}_i}[\tau] \leq H$, and we define:

$$\mathcal{E} = \left\{\hat{\pi}_\tau \in \arg\max_{\pi\in\Pi} V_i(\pi) \ : \ \text{“best policy is identified within } H \text{ steps”}\right\}.$$

To derive the lower bound, we consider the two cases in which $\mathcal{E}$ hold or does not hold with high probability, respectively.

[10] Here we consider $i \leq M/2$ for the sake of clarity. If $i > M/2$ some of the indices of $\mathcal{G}_1$ will exceed $2M$, so that the exceeding states are to be taken from the start of the right chain (s_{M+i}, s_{M+u}), where u is the number of exceeding indices.

Bad Event. If the event $\mathcal{E}$ does not hold with high probability, i.e., $\mathbb{P}(\mathcal{E}) < 1 - \delta$, then we can show that the regret scales with $\Omega(\sqrt{H})$.

Let us consider any triplet $(\pi, \tau, \hat{\pi}_\tau)$. Without loss of generality, we take $\mathbb{E}_{\mathcal{M}_i}[\tau] = h$, from which we have

$$\mathcal{R}_H(\mathcal{M}_i, \pi) = \mathcal{R}_h(\mathcal{M}_i, \pi) + \sum_{\bar{h}=h}^{H} V_i^* - V_i(\hat{\pi}_\tau) \geq \mathcal{R}_h(\mathcal{M}_i, \pi) + (H - h)\Delta_1 \tag{3}$$

with probability at least δ. The latter inequality is obtained by noting that the set of policies going to the left in the initial state $\pi_1(a_1|s_{in}) = 1$, then taking action a_1 at some state $s_j \in (s_1, \dots, s_M)$, and then taking the same action until the episode ends (formally, $\pi_t(a_2|s_t) = 1$ for all $t < j$, $\pi_j(a_1|s_j) = 1$, and $\pi_t(a_1|s_H) = \pi_t(a_1|s_L) = 1$ for all $t > j$) include an optimal policy for every $\mathcal{M}_i \in \mathcal{M}$. We denote this *sufficient* set of policies as $\widetilde{\Pi}$. For any MDP $\mathcal{M}_i \in \mathcal{M}$, it holds $V_i(\pi) = \Delta_1$ for all $\pi \in \widetilde{\Pi} \setminus (\pi_i^*)$ and $V_i^* = 1$, which gives the above.

Then, we lower bound the term $\mathcal{R}_h(\mathcal{M}_i, \pi)$ in (3) through regret minimization. Due to how the instance is constructed, there is no incentive to take action a_2 in s_{in} since the best policy identification fails in the bad event. Thus, we restrict the set of policies to $\widetilde{\Pi}$ again. Notably, this set of policies is finite, having size $|\widetilde{\Pi}| = M$. We can cast the regret minimization problem over this set of policies as a bandit with M actions with parameters $(\mu_j = V_i(\pi_j))_{j\in[M]}$ for some arbitrary ordering of the policies in $\widetilde{\Pi}$. It is easy to see that the regret of the original MDP problem cannot be smaller than the regret of the latter bandit reformulation, which we can lower bound through the techniques in the proof of Theorem C.1. We have

$$\mathcal{R}_h(\mathcal{M}_i, \pi) \geq \frac{h\Delta_1}{8} \exp\left(-\frac{h(\Delta_1)^2}{2}\right). \tag{4}$$

Finally, substituting (4) into (3) we get

$$\mathcal{R}_H(\mathcal{M}_i, \pi) \geq \frac{h\Delta_1}{8} \exp\left(-\frac{h(\Delta_1)^2}{2}\right) + (H - h)\Delta_1 \geq \frac{\sqrt{H}}{8} \exp\left(-\frac{1}{2}\right)$$

with probability at least δ, where the last inequality is obtained by taking $\Delta_1 = 1/\sqrt{H}$ and noting that the left-hand side is minimized for $\mathbb{E}_{\mathcal{M}_i}[\tau] = h = H$.

Good Event. The previous result states that the regret is at least $\Omega(\sqrt{H})$ when the event $\mathcal{E}$ does not hold with high probability. This hints that solving the best policy identification problem is necessary to minimize the regret. To derive the lower bound, we instantiate a proper best policy identification problem on the considered instance $\mathcal{M}$ and we derive the corresponding sample complexity through Lemma B.1. We have $\mathbb{E}_{\mathcal{M}_i}[\tau] \geq T^*(\mathcal{M}_i)^{-1} \log(1/2.4\delta)$ where

$$T^*(\mathcal{M}_i) = \sup_{\omega \in \Sigma(\mathcal{M}_i)} \inf_{\mathcal{M}_j \in \mathcal{M}_{-i}} \sum_{s,a} \omega(s,a) \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a).$$

From Assumption 1 and the Pinsker's inequality we have

$$\mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq 4\mathrm{TV}^2(p_i(\cdot|s,a), p_j(\cdot|s,a)) \geq \|(p_i - p_j)(\cdot|s,a)\|_1^2 \geq \lambda^2.$$

By staring at the instance, it can be seen that the allocation vector attaining the supremum is the one assigning even probabilities to all the pairs $(s_{M+x}, a_1)_{x\in[M]}$, as it guarantees $\omega(s,a) = 1/TM$ for at least two revealing state-action pairs against any MDP $\mathcal{M}_j \in \mathcal{M}_{-i}$, such that $\sum_{s,a} \omega(s,a) \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq \frac{2\lambda^2}{TM}$, while any other allocation can be hacked by the infimum over $\mathcal{M}_j \in \mathcal{M}_{-i}$ to a lesser value.[11]

We just need to show that the desired allocation can be obtained and does not violate the flow constraints of the MDP (see the statement of Lemma B.1). We set $\omega(s_1, a_2, 1) = 1$, which implies $\sum_a \omega(s_{M+x}, a) \leq 1/T, \forall x \in [M]$, since the states of the right chain cannot be visited more than once in an episode. Then, we set $\omega(s_{M+x}, a_1) = 1/TM, \forall x \in [M]$ from the

[11] There are actually other allocation vectors that have equivalent value of $T^*(\mathcal{M}_i)$, which is the one assigning even probabilities to all the pairs $(s, a_1)_{s\in\mathcal{G}_1}$ or $(s, a_1)_{s\in\mathcal{G}_2}$. For the sake of the proof, we use the most convenient to algebraic manipulations.

desired allocation, which gives $\omega(s_{M+x}, a_1, x+1) = 1/M, \forall x \in [M]$. We have

$$\omega(s_{M+1}, a_1, 2) = \frac{1}{M} \text{ and } \omega(s_{M+1}, a_2, 2) = \frac{M-1}{M}$$
$$\text{from } \sum_a \omega(s_{M+1}, a, 2) = \sum_{s',a'} p_i(s_{M+1}|s',a')\omega(s',a',1) = \omega(s_1, a_2, 1) = 1,$$
$$\omega(s_{M+2}, a_1, 3) = \frac{1}{M} \text{ and } \omega(s_{M+2}, a_2, 3) = \frac{M-2}{M}$$
$$\text{from } \sum_a \omega(s_{M+2}, a, 3) = \sum_{s',a'} p_i(s_{M+2}|s',a')\omega(s',a',2) = \omega(s_{M+1}, a_2, 2) = \frac{M-1}{M},$$
$$\omega(s_{M+3}, a_1, 4) = \frac{1}{M} \text{ and } \omega(s_{M+3}, a_2, 4) = \frac{M-3}{M}$$
$$\text{from } \sum_a \omega(s_{M+3}, a, 4) = \sum_{s',a'} p_i(s_{M+3}|s',a')\omega(s',a',3) = \omega(s_{M+2}, a_2, 3) = \frac{M-2}{M},$$
$$\dots$$
$$\omega(s_{2M}, a_1, M+1) = \frac{1}{M} \text{ and } \omega(s_{2M}, a_2, M+1) = 0$$
$$\text{from } \sum_a \omega(s_{2M}, a, M+1) = \sum_{s',a'} p_i(s_{2M}|s',a')\omega(s',a',M) = \omega(s_{2M-1}, a_2, M) = \frac{1}{M},$$

which gives $\omega(s_{M+x}, a_2, x) = \frac{M-x}{M}, \forall x \in [M]$, while all of the additional probability to have $\omega(s,a,t) \in \mathcal{P}(\mathcal{S} \times \mathcal{A}), \forall t \in [T]$ is absorbed by s_L and s_H.

Having proved that the desired allocation complies to the flow constraints, we proceed as

$$\mathop{\mathbb{E}}_{\mathcal{M}_i}[\tau] \geq \frac{TM}{2\lambda^2} \log\left(\frac{1}{2.4\delta}\right).$$

Finally, we can derive the lower bound through

$$\begin{aligned}\mathcal{R}_H(\mathcal{M}_i, \pi) \geq \mathbb{E}_{\mathcal{M}_i}[\tau]\Delta_2 &= \frac{TM}{2\lambda^2}\frac{\log(H)}{\sqrt{H}} \log\left(\frac{1}{2.4\delta}\right) \\ &\geq \frac{1}{4(\sqrt{C} - \log(C))}\frac{TM\log(H)}{\lambda} \log\left(\frac{1}{2.4\delta}\right)\end{aligned}$$

where the last inequality is obtained by exploiting $H \leq C$ and that transition probabilities are in $[0,1]$ to write $\frac{\lambda}{2} + \frac{\log(H)}{\sqrt{H}} \leq 1$, which gives $\lambda \leq \frac{2(\sqrt{C} - \log(C))}{\sqrt{H}}$. □

A.3. Proofs of Section 5

Here we report the proofs for the test-time regret upper bounds provided in Theorem 5.1, 5.2, 5.3, respectively.

Theorem 5.1. *Let $\mathcal{M}$ be a set of MDPs for which Assumption 2, 3, 4 hold. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{T(K^2+N^2)\log(NH)\log^2(\frac{SNH}{\lambda})}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 3 with inputs $\mathcal{M}, \mathcal{C}$, and $n_{\mathcal{C}} = \frac{c\log^2(SKH/\lambda)\log(KH)}{\lambda^4}$ for a sufficiently large constant c.

Proof. The derivations are straightforward following the steps in the proof of Theorem 3.1 and being careful to count the expected number of episodes for both the "Identify" stages.

For the first "Identify" stage (lines 3-12 in Algorithm 3), we want to make sure that the cluster $\mathcal{C}_k^i$ to which the test task belongs, i.e., $\mathcal{M}_i \in \mathcal{C}_k^i$, is not eliminated from the set $\mathcal{C}$ with probability at least $1 - 1/2H$. We write

$$P\Big(\text{“}\mathcal{C}_k^i \text{ is eliminated from } \mathcal{C}\text{”}\Big) \leq \sum_{u=1}^{K-1} P\Big(\text{“}\mathcal{C}_k^i \text{ is eliminated from } \mathcal{C} \text{ at iteration } u\text{”}\Big)$$

by noting that the loop is executed $K-1$ times and through a union bound on the iterations. Next, we call Lemma A.1 on the MDPs $\mathcal{M}_1, \mathcal{M}_2$ selected at line 4 (Algorithm 3) on the set $\mathcal{X}$ of samples collected at line 5 (Algorithm 3) to prove that the event "$\mathcal{C}_k^i$ is eliminated from $\mathcal{C}$ at iteration u" holds with probability less than $1/2KH$, and we further invoke Lemma A.2 to prove that $\mathcal{X}$ can be collected efficiently from the sampling routine (Algorithm 2). Let us denote H_0 the random variable associated to the number of episodes spent in the first "Identify" stage. We have

$$\mathbb{E}[H_0] = h_0 \leq 2(K-1)Kn_{\mathcal{C}} \leq \frac{c_0 K^2 \log^2(SKH/\lambda)\log(KH)}{\lambda^4}. \tag{5}$$

Then, we look at the second "Identify" stage (lines 3-11 in Algorithm 1), which is called at line 12 of Algorithm 3 on the cluster $\hat{\mathcal{C}}$ resulting from the previous stage. Just like before, we want to make sure that the MDP $\mathcal{M}_i$ is not eliminated from the set $\hat{\mathcal{C}}$ with probability at least $1 - 1/2H$. We write

$$P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \hat{\mathcal{C}}\text{”}\Big) \leq \sum_{u=1}^{N-1} P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \hat{\mathcal{C}} \text{ at iteration } u\text{”}\Big)$$

by noting that the loop is executed at most $N-1$ times and through a union bound on the iterations. This follows verbatim the proof of Theorem 3.1 on the cluster $\hat{\mathcal{C}}$ insetad of $\mathcal{M}$, for which we can call Lemma A.1 and Lemma A.2 to get similar results. We denote as H_1 the random variable associated to the number of episodes spent in the second "Identify" stage. We have

$$\mathbb{E}[H_1] = h_1 \leq 2(N-1)Nn \leq \frac{c_1 N^2 \log^2(SNH/\lambda)\log(NH)}{\lambda^4}. \tag{6}$$

Then, we can call yet another union bound on the events defined for the two "Identify" stages and, with similar considerations as in Theorem 3.1, we write

$$\begin{aligned}
R_H(\mathcal{M}_i, \pi) &= \mathbb{E}\left[\sum_{h=1}^{h_0+h_1} V_i^* - V_i(\pi_h)\right] + \mathbb{E}\left[\sum_{h=h_0+h_1}^{H} V_i^* - V_i(\hat{\pi})\right] \\
&\leq \frac{c_0 T K^2 \log^2(SKH/\lambda)\log(KH)}{\lambda^4} + \frac{c_1 T N^2 \log^2(SNH/\lambda)\log(NH)}{\lambda^4} \\
&\leq \frac{cT(K^2+N^2)\log^2(SNH/\lambda)\log(NH)}{\lambda^4}
\end{aligned}$$

where we plugged (5) and (6) together with $\mathbb{E}[\sum_{h=1}^{h_0+h_1} V_i^* - V_i(\pi_h)] \leq \mathbb{E}[H_0 + H_1]T$ to get the first inequality, $c = \max(c_0, c_1)$ and $N > K$ to obtain the second inequality. □

Theorem 5.2. *Let $\mathcal{M}$ be a set of MDPs for which Assumption 5, 6 hold and let $d = \log_{1/\beta}(M)$. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{Td\log(dH)\log^2(SdH/\lambda)}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 4 with inputs $\mathcal{M}$ and $n = \frac{c\log^2(SdH/\lambda)\log(dH)}{\lambda^4}$ for a sufficiently large constant c.

Proof. The proof follows derivations of Theorem 3.1 with some slight yet crucial modifications. Just as in Theorem 3.1, we want to show that the true MDP $\mathcal{M}_i$ will not be eliminated from $\mathcal{D}$ (lines 2-18 in Algorithm 4) with probability at least $1 - 1/H$. However, the number of times the loop in lines 3-18 of Algorithm 4 is executed is not M, as in Theorem 3.1, but

depends on the depth of the tree structure given by Assumption 6. Since the size of $\mathcal{D}$ is reduced at every iteration by a factor β at least, we have

$$M\beta^{\text{depth}} \leq 1 \quad \implies \quad \text{depth} \leq \frac{\log(1/M)}{\log(\beta)} = \log_{1/\beta}(M) =: d.$$

With the latter, we can write

$$P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D}\text{”}\Big) \leq \sum_{m=1}^{d} P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D} \text{ at iteration } m\text{”}\Big)$$

through a union bound on the iterations. Then, we call Lemma A.1 (with care of substituting M with d the result holds verbatim) to prove that each event “$\mathcal{M}_i$ is eliminated from $\mathcal{D}$ at iteration m” in the summation holds with probability less than $1/dH$.

For each iteration of the outer loop (lines 3-18), we want to prove that the n samples from $(\bar{s}, \bar{a})$ can be collected efficiently through the inner loop (lines 6-12). We follow similar steps as in Lemma A.2. Through Assumption 5, we have that $\mathbb{E}[X(\bar{s}, \bar{a}|\mathcal{M}_i, \pi)] \leq T/2$. By applying the Markov's inequality we get $X(\bar{s}, \bar{a}|\mathcal{M}_i, \pi) \leq T$ with probability at least $1/2$. Let Y the random variable denoting the number of episodes needed to reach state $(\bar{s}, \bar{a})$ in the test MDP $\mathcal{M}_i$. We have that $P(Y = k) \leq 1/2^k$, which gives $\mathbb{E}[Y] \leq \sum_{k=1}^{\infty} k/2^k = 2$. Thus, the expected number of episodes to collect the n samples is upper bounded by $2n$.

Let H_0 denote the random variable associated to the number of episodes spent in the “Identify” stage as a whole (lines 2-18). From the considerations above, we have

$$\mathbb{E}[H_0] = h_0 \leq 2dn \leq \frac{cd\log^2(SdH/\lambda)\log(dH)}{\lambda^4}. \tag{7}$$

Then, we can write

$$R_H(\mathcal{M}_i, \pi) = \mathbb{E}\left[\sum_{h=1}^{h_0} V_i^* - V_i(\pi_h)\right] + \mathbb{E}\left[\sum_{h=h_0}^{H} V_i^* - V_i(\hat{\pi})\right] \leq \frac{cTd\log^2\left(\frac{SdH}{\lambda}\right)\log(dH)}{\lambda^4}$$

where we plugged (7) together with $\mathbb{E}[\sum_{h=1}^{h_0} V_i^* - V_i(\pi_h)] \leq \mathbb{E}[H_0]T$ to get the inequality. □

Theorem 5.3. *Let $\mathcal{M}$ be a set of MDPs for which Assumption 7 holds. For any $\mathcal{M}_i \in \mathcal{M}$, we have*

$$R_H(\mathcal{M}_i, \mathbb{A}) = \mathcal{O}\left(\frac{TI\log(MH)\log^2(SMH/\lambda)}{\lambda^4}\right)$$

where $\mathbb{A}$ is Algorithm 5 with inputs $\mathcal{M}, \Pi_I, \overline{\mathcal{SA}}$, $n = \frac{c\log^2(SMH/\lambda)\log(MH)}{\lambda^4}$ for a sufficiently large constant c.

Proof. The proof follows the derivation of Theorem 3.1 verbatim, with the only difference that the samples $\mathcal{X}$ for each iteration of the loop in the “Identify” stage (lines 13-21 of Algorithm 5) are entirely collected during the “Explore” stage (lines 3-11). Analogously as before, we show that the true MDP $\mathcal{M}_i$ will not be eliminated from $\mathcal{D}$ (lines 6-10 in Algorithm 1) with probability at least $1 - 1/H$. Especially, we can write

$$P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D}\text{”}\Big) \leq \sum_{m=1}^{M-1} P\Big(\text{“}\mathcal{M}_i \text{ is eliminated from } \mathcal{D} \text{ at iteration } m\text{”}\Big)$$

by noting that the loop in the “Identify” stage is executed for $M - 1$ iterations and then applying a union bound. Just like before, we call Lemma A.1 to prove that the event “$\mathcal{M}_i$ is eliminated from $\mathcal{D}$ at iteration m” holds with probability less than $1/MH$, and then we invoke Lemma A.2 to prove that the “Explore” stage can collect the desired number of samples efficiently. Let H_0 denote the random variable associated to the number of episodes spent in the “Explore” stage. Following similar considerations as in the proof of Theorem 3.1, we have

$$\mathbb{E}[H_0] = h_0 \leq 2In \leq \frac{cI\log^2(SMH/\lambda)\log(MH)}{\lambda^4}. \tag{8}$$

Then, we can write

$$R_H(\mathcal{M}_i, \pi) = \mathbb{E}\left[\sum_{h=1}^{h_0} V_i^* - V_i(\pi_h)\right] + \mathbb{E}\left[\sum_{h=h_0}^{H} V_i^* - V_i(\hat{\pi})\right] \leq \frac{cTI\log^2\left(\frac{SMH}{\lambda}\right)\log(MH)}{\lambda^4}$$

where we plugged (8) together with $\mathbb{E}[\sum_{h=1}^{h_0} V_i^* - V_i(\pi_h)] \leq \mathbb{E}[H_0]T$ to get the inequality. □

B. Best Policy Identification in Finite-Horizon MDPs: A Tailored Lower Bound

In Best Policy Identification (BPI, Fiechter, 1994), the learner interacts with an unknown MDP $\mathcal{M}_i$ with the goal of minimizing the expected number of samples to be taken in order to tell an optimal policy $\pi^* \in \arg\max_{\pi\in\Pi} V_i(\pi)$ for $\mathcal{M}_i$ with probability at least $1-\delta$, where $\delta \in (0,1)$ is a fixed confidence.

The literature provides theoretical guarantees on the latter expected number of samples, called *sample complexity*, in a variety of settings ranging from *worst-case* results for discounted (Azar et al., 2013; Agarwal et al., 2020) and finite-horizon MDPs (Dann & Brunskill, 2015; Dann et al., 2019; Kaufmann et al., 2021; Ménard et al., 2021) to *instance-dependent* analyses (Al Marjani & Proutiere, 2021; Al Marjani et al., 2021; Wagenmaker et al., 2022; Tirinzoni et al., 2022; 2023; Al-Marjani et al., 2023).

For the purpose of deriving a lower bound for test-time regret minimization (Theorem 4.1), we use, as a building block, an instance-dependent, non-asymptotic lower bound to the sample complexity of any δ-PC (Probably Correct) BPI algorithm in finite-horizon MDPs.[12] To the best of our knowledge, the only result of this kind is given in Al-Marjani et al. (2023, Theorem 2). Here we derive an alternative result that is tailored to our setting, i.e., in which the set of possible MDPs is restricted to a finite set $\mathcal{M}$ fulfilling the λ-separation (Assumption 1).

Additional Notation. Let $\mathcal{H}_h := (s_t, a_t, r_t)_{t\in[T]}$ be a trajectory collected by executing a policy π_h at the episode h. We denote $\mathcal{F}_h := \sigma((\mathcal{H}_{h'})_{h'\in[h]})$ the sigma algebra of the trajectories up to episode h, such that $(\mathcal{F}_{h'})_{h'\in[h]}$ is the corresponding filtration. We define

- $(\pi_{h'} : \mathcal{F}_{h'-1} \to \mathcal{P}(\mathcal{A}))_{h'\in[h]}$ a *sampling rule* that determines the policy to be run at each episode given the past observations;
- τ a *stopping rule* that gives the time at which the sampling process is stopped given past observations;
- $\hat{\pi}_\tau \in \Pi$ a *decision rule*, which is the policy selected when τ is triggered, i.e., the best guess on the optimal policy given past observations.

We denote as $\mathbb{E}[\tau]$ the sample complexity of the BPI problem. Notably, the identification can span several episodes of our finite-horizon MDP setting, which means that at any step h' such that $mod(k', T) = 0$ the process will be reset to state s_1. To simplify the analysis, we assume that whenever the stopping rule τ is triggered, the process proceeds until the end of the episode, which means the sample complexity is a multiple of T.

Now, we have all of the elements to derive our tailored lower bound. Specifically, we adapt to our BPI problem of interest the result (Al Marjani et al., 2021, Proposition 2), which was originally derived for the infinite-horizon and δ-asymptotic setting. We obtain the following.

Lemma B.1 (Best policy identification)**.** *Let assume all the $\mathcal{M}_j \in \mathcal{M}$ admit unique optimal policies. For $\mathcal{M}_i \in \mathcal{M}$, let us define the set of allocation vectors*

$$\begin{aligned}\Sigma(\mathcal{M}_i) = \Big\{\omega \in \mathcal{P}(\mathcal{S}\times\mathcal{A}) : \quad & \omega(s,a) = \frac{1}{T}\sum_{t\in[T]}\omega(s,a,t),\\ & \omega(\cdot,\cdot,1) \in \mathcal{P}(\mathcal{S}\times\mathcal{A}), \quad \sum_{a\in\mathcal{A}}\omega(s_1,a,1) = 1,\\ & \forall (s,t) \in \mathcal{S}\times(2,\dots,T), \quad \sum_{a\in\mathcal{A}}\omega(s,a,t) = \sum_{s',a'} p_i(s|s',a')\omega(s',a',t-1)\Big\}.\end{aligned}$$

[12] A δ-PC algorithm (e.g., Al Marjani et al., 2021) is an algorithm that is guaranteed to output an optimal policy π^* with probability at least $1-\delta$ with a finite sample complexity.

Let $\mathcal{M}_{-i} := \mathcal{M} \subseteq \mathcal{M}_i$. For $\delta \in (0,1)$, any δ-PC BPI algorithm has sample complexity

$$\mathop{\mathbb{E}}_{\mathcal{M}_i}[\tau] \geq T^*(\mathcal{M}_i)^{-1} \log(1/2.4\delta) \quad \textit{where} \quad T^*(\mathcal{M}_i) = \sup_{\omega \in \Sigma(\mathcal{M}_i)} \inf_{\mathcal{M}_j \in \mathcal{M}_{-i}} \sum_{s,a} \omega(s,a) \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a).$$

Proof. The derivations are adapted from the proof of Proposition 2 in Al Marjani et al. (2021). First, we report a sample complexity result on best policy identification with a generative model (Al Marjani & Proutiere, 2021), which, for any $\mathcal{M}_j \in \mathcal{M}_{-i}$, states that

$$\sum_{s,a} \mathop{\mathbb{E}}_{\mathcal{M}_i}[N_\tau(s,a)] \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq \mathrm{kl}(\delta, 1-\delta) \tag{9}$$

where $N_\tau(s,a)$ is the number of samples of the (s,a) pair collected within $\tau \in \mathbb{N}$ steps and $\mathrm{kl}(x,y)$ denotes the Kullback Leibler divergence between Bernoulli distributions with parameters x, y respectively. Differently from the generative model setting, we have to enforce MDP constraints on $\mathbb{E}_{\mathcal{M}_i}[N_\tau(s,a)]$, which gives the recursive expression

$$\text{if } mod(\tau, T) \neq 0 \qquad \sum_a \mathop{\mathbb{E}}_{\mathcal{M}_i}[N_\tau(s,a)] = \sum_{s',a'} p_{\mathcal{M}_i}(s|s',a') \left(\mathop{\mathbb{E}}_{\mathcal{M}_i}[N_{\tau-1}(s',a')] + 1 \right), \quad \forall (s,a) \in \mathcal{S} \times \mathcal{A}$$

$$\text{else} \qquad \sum_a \mathop{\mathbb{E}}_{\mathcal{M}_i}[N_\tau(s_1,a)] = \sum_a \mathop{\mathbb{E}}_{\mathcal{M}_i}[N_{\tau-1}(s_1,a)] + 1 \quad \text{and} \quad \mathop{\mathbb{E}}_{\mathcal{M}_i}[N_\tau(s,a)] = \mathop{\mathbb{E}}_{\mathcal{M}_i}[N_{\tau-1}(s,a)], \quad \forall s \neq s_1.$$

Then, we can combine the latter constraints with (9) to write the following optimization problem

$$\inf_{n \geq 0} \sum_{s,a,t} n_{sat} \tag{10}$$

$$\text{subject to} \sum_{s,a,t} n_{sat} \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq \mathrm{kl}(\delta, 1-\delta) \qquad \forall \mathcal{M}_j \in \mathcal{M}_{-i} \tag{11}$$

$$\sum_a n_{sat} = \sum_{s',a'} p_{\mathcal{M}_i}(s|s',a')(n_{s'a't-1} + 1) \qquad \forall s, \forall t : mod(t,T) \neq 0 \tag{12}$$

$$\sum_a n_{s_1at} = \sum_a n_{s_1at-1} + 1 \qquad \forall t : mod(t,T) = 0 \tag{13}$$

$$n_{sat} = n_{sat-1} \qquad \forall s \neq s_1, \forall t : mod(t,T) = 0 \tag{14}$$

To prove the result, let us take the constraint (11). Since it has to hold for every $\mathcal{M}_j \in \mathcal{M}_{-i}$, we can write

$$\inf_{\mathcal{M}_j \in \mathcal{M}_{-i}} \sum_{s,a,t} n_{sat} \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq \mathrm{kl}(\delta, 1-\delta).$$

Let us denote N^* the value of (10), we write

$$\inf_{\mathcal{M}_j \in \mathcal{M}_{-i}} \sum_{s,a,t} \frac{n_{sat}}{N^*} \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq \frac{\mathrm{kl}(\delta, 1-\delta)}{N^*}.$$

Through constraints (12-14) and the definition of N^*, we have that $(n_{sat}/N^*) \in \Sigma(\mathcal{M}_i)$, where the latter is the set of allocation vectors. Hence, we can write

$$\sup_{\omega \in \Sigma(\mathcal{M}_i)} \inf_{\mathcal{M}_j \in \mathcal{M}_{-i}} \sum_{s,a} \omega(s,a) \, \mathrm{KL}_{\mathcal{M}_i|\mathcal{M}_j}(s,a) \geq \frac{\mathrm{kl}(\delta, 1-\delta)}{N^*}.$$

Finally, we know from (Al Marjani et al., 2021, Proposition 10) that $\mathbb{E}_{\mathcal{M}_i}[\tau] \geq N^*$, which together with $\mathrm{kl}(\delta, 1-\delta) \geq \log(1/2.4\delta)$ (see Kaufmann et al., 2016) gives the result. □

C. Meta Learning in Bandits

In this section, we analyze a simplified bandit version of the test-time regret minimization problem described in the paper. The aim of this study is to serve both as a gentle introduction to the more advanced results and techniques presented in the paper, which come more naturally in the bandit setting, as well as a standalone analysis that may be of independent interest.

We consider a class $\mathcal{M} = (\mathcal{M}_i)_{i\in[M]}$ of bandits (Lattimore & Szepesvári, 2020), each of them having a set of actions $\mathcal{A} = (a_j)_{j\in[A]}$ with corresponding reward distributions $R_i(a_j)$ for all $i \in [M], j \in [A]$ with bounded mean $\mu_{ij} \in [0, 1]$. First, we rephrase the separation condition presented in the paper (Assumption 1) as follows,

Assumption 8 (λ-separation (bandit))**.** *For any $\mathcal{M}_i, \mathcal{M}_j \in \mathcal{M}$, there exists $a \in \mathcal{A}$ such that $\|R_i(a) - R_j(a)\|_1 \geq \lambda$.*

Just like in the MDP setting, we assume the reward distribution of all bandits $\mathcal{M}_i \in \mathcal{M}$, as well as the set $\mathcal{M}$ itself, to be fully known to the agent, who faces a test task (i.e., bandit) that is instead unknown but belonging to $\mathcal{M}$. To evaluate the agent's performance, we redefine the H-steps test-time regret for the task $\mathcal{M}_i \in \mathcal{M}$ as

$$\mathcal{R}_H(\mathcal{M}_i, \pi) = \mathbb{E}\left[\sum_{h=1}^{H} R_i(a^*) - R_i(a_h)\right] = \mathbb{E}\left[\sum_{h=1}^{H} \mu^* - \mu_h\right]$$

where $a^* \in \arg\max_{a\in\mathcal{A}} \mu(a)$ is the optimal action in the bandit $\mathcal{M}_i$, $a_h \in \mathcal{A}$ is the action played by policy π at step h, and μ^*, μ_h are the mean of their reward distribution, respectively. Just as we did in the paper for the more general MDP setting, we provide both a lower bound and a nearly minimax optimal algorithm for the test-time regret minimization in bandits.

C.1. Lower Bound

We now prove a lower bound to the test-time regret suffered by any algorithm in the introduced meta learning in bandits setting under the above separation condition (Assumption 8). Formally,

Theorem C.1 (Lower bound)**.** *Let $\mathcal{M}$ be a set of bandits for which Assumption 8 holds. Let $C < \infty$ a constant, and let $\delta \in (0, 1)$. For any horizon $M - 1 \leq H \leq C$, it holds*

$$R_H(\mathcal{M}_i, \pi) = \Omega\left(\frac{M\log(H)}{\lambda}\log\left(\frac{1}{\delta}\right)\right)$$

with probability at least $1 - \delta$.

Proof. To prove the lower bound, we first construct a convenient instance in which it is hard to minimize the regret without knowing the task, while it is easy to identify the task playing sub-optimal actions. Then, we derive the lower bound on the regret suffered by any algorithm leveraging minimax lower bounds for standard bandits (Lattimore & Szepesvári, 2020) and best arm identification results (Garivier & Kaufmann, 2016).

Let $\mathcal{M} = (\mathcal{M}_i)_{i\in[M]}$ a problem instance in which every $\mathcal{M}_i$ has $|\mathcal{A}| = 2M$ actions and Gaussian reward distributions $R_i(a_j) = \mathcal{N}(\mu_{ij}, 1)$. For each $\mathcal{M}_i$, we specify the first set of M actions $(a_j)_{j=1}^{M}$ as follows: The action a_i is the optimal action with mean $\mu_i = \mu^*$, while all of the other actions are slightly sub-optimal $\mu^* - \mu_j = \Delta_1 = 1/\sqrt{H}$. The second set of actions $(a_j)_{j=M+1}^{2M}$ is specified as follows: The actions $\mathcal{A}_1 := (a_{M+i}, \dots, a_{\frac{3M}{2}+i})$ have mean reward such that $\mu^* - \mu_a = \Delta_2 = \log(H)/\sqrt{H}, \forall a \in \mathcal{A}_1$, and all of the other actions $\mathcal{A}_2 := (a_{M+1}, \dots, a_{M+i-1}) \cup (a_{\frac{3M}{2}+i+1}, \dots, 2M)$ have mean reward $\mu_a < \mu_{M+i}$ such that $\|R(a_{M+i}) - R(a)\|_1 \geq \lambda, \forall a \in \mathcal{A}_2$, fulfilling λ-separation. The instance is depicted in Figure 2.[13]

In general, there are two ways to approach the described instance. Since it is known that the second set of M actions is sub-optimal in every $\mathcal{M}_i$, we can minimize the regret playing only the first set of actions. Otherwise, we can exploit the separation condition on the second set of arms to identify the task and then playing the optimal arm. To formalize this intuition, we borrow notation from best arm identification literature (e.g. Garivier & Kaufmann, 2016) similarly as we did in Appendix B.

[13]Note that, for describing the instance, we conveniently consider $i \leq M/2$, but it is straightforward to understand how it works for $i > M/2$ by substituting the exceeding indices in $\mathcal{A}_1$ back with the first arms of the sequence $a_{M+1}, a_{M+2}, \dots$ until $\mathcal{A}_1$ consists of $M/2$ arms.

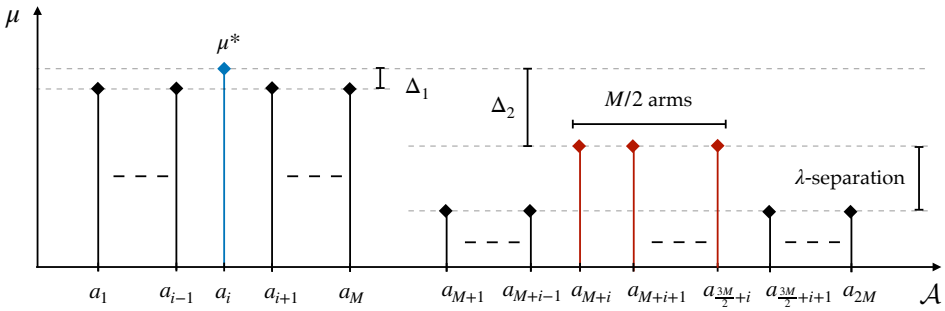

Figure 2. Visualization of the $\mathcal{M}_i$ bandit in the problem instance designed to derive the lower bound. The optimal action a_i and the identifying actions $a \in \mathcal{A}_1 \cup \mathcal{A}_2$ change for every $\mathcal{M}_i$.

Additional Notation. Let $\mathcal{H}_h := (a_{h'}, r_{h'})_{h' \in [h]}$ be a trajectory collected by executing a policy π. We denote $\mathcal{F}_{h \geq 1}$ the corresponding filtration on $\mathcal{H}_h$. We define

- $(\pi_{h'})_{h'=1}^{h}$ a *sampling rule* over $\mathcal{A}$ that determines the next action to play given past observations;
- τ a *stopping rule* that gives the stopping time w.r.t. $\mathcal{H}_h$;
- $\hat{a}_\tau \in \mathcal{A}$ a *decision rule*, which is the action selected when τ is triggered, i.e., the best guess on the optimal arm given past observations.

We denote as $\mathbb{E}[\tau]$ the sample complexity of the best arm identification problem. Further, we restrict τ to the class of stopping rules such that $\mathbb{E}_{\mathcal{M}_i}[\tau] \leq H$, and we define the following event:

$$\mathcal{E} = \left\{ \hat{a}_\tau \in \arg\max_{a \in \mathcal{A}} \mu(a) \ : \ \text{“best arm is identified within } H \text{ steps”} \right\}.$$

To derive the lower bound, we consider the two cases in which $\mathcal{E}$ hold or does not hold with high probability, respectively.

Bad Event. If the event $\mathcal{E}$ does not hold with high probability, i.e., $\mathbb{P}(\mathcal{E}) < 1 - \delta$, we can show that the regret scales with $\Omega(\sqrt{H})$.

Let us consider any triplet $(\pi, \tau, \hat{a}_\tau)$. Without loss of generality, we take $\mathbb{E}_{\mathcal{M}_i}[\tau] = h$, from which we have

$$\mathcal{R}_H(\mathcal{M}_i, \pi) = \mathcal{R}_h(\mathcal{M}_i, \pi) + (H - h)\Delta_1 \tag{15}$$

with probability at least δ.

First, we lower bound the term $\mathcal{R}_h(\mathcal{M}_i, \pi)$ through regret minimization. We can restrict the action set to $\tilde{\mathcal{A}} = (a_j)_{j=1}^{M}$ as there is no incentive to play surely sub-optimal actions a_j for $j > M$ when minimizing the regret. We take a policy π inducing pulls $(a_{h'})_{h'=1}^{h}$ and corresponding counts $T_j(h) := \sum_{h'=1}^{h} \mathbf{1}(a_j = a_{h'})$ over the actions $\tilde{\mathcal{A}}$ of $\mathcal{M}_1$. Then, we select $\mathcal{M}_i \in \mathcal{M}$ such that $i = \arg\min_{j \in [M]} \mathbb{E}_{\mathcal{M}_1}[T_j(h)]$. We have that

$$\max_{\mathcal{M}^* \in \mathcal{M}} \mathcal{R}_h(\mathcal{M}^*, \pi) \geq \max\left\{\mathcal{R}_h(\mathcal{M}_1, \pi), \mathcal{R}_h(\mathcal{M}_i, \pi)\right\} \geq \frac{\mathcal{R}_h(\mathcal{M}_1, \pi) + \mathcal{R}_h(\mathcal{M}_i, \pi)}{2}.$$

We can further expand the terms on the right hand-side by noting that

$$\mathcal{R}_h(\mathcal{M}_1, \pi) \geq P_{\mathcal{M}_1}(T_1(h) \leq h/2)\frac{h\Delta_1}{2} \ \text{ and } \ \mathcal{R}_h(\mathcal{M}_i, \pi) \geq P_{\mathcal{M}_i}(T_1(h) > h/2)\frac{h\Delta_1}{2}$$

from which we can write

$$\begin{aligned}\mathcal{R}_h(\mathcal{M}_1, \pi) + \mathcal{R}_h(\mathcal{M}_i, \pi) &> \frac{h\Delta_1}{2}\left(P_{\mathcal{M}_1}(T_1(h) \leq h/2) + P_{\mathcal{M}_i}(T_1(h) > h/2)\right) \\ &\geq \frac{h\Delta_1}{4}\exp\left(-\mathrm{KL}(\mathbb{P}_{\mathcal{M}_1}, \mathbb{P}_{\mathcal{M}_i})\right)\end{aligned}$$

where the latter is obtained through the Bretagnolle-Huber inequality. Then, we can upper bound the KL divergence as

$$\begin{aligned}
&\mathrm{KL}(\mathbb{P}_{\mathcal{M}_1}, \mathbb{P}_{\mathcal{M}_i}) \\
&\quad = \mathop{\mathbb{E}}_{\mathcal{M}_1}[T_1(h)]\mathrm{KL}(\mathcal{N}(\mu^*, 1), \mathcal{N}(\mu - \Delta_1, 1)) + \mathop{\mathbb{E}}_{\mathcal{M}_1}[T_i(h)]\mathrm{KL}(\mathcal{N}(\mu - \Delta_1, 1), \mathcal{N}(\mu^*, 1)) \\
&\quad \leq \frac{(\Delta_1)^2}{2}\left(\mathop{\mathbb{E}}_{\mathcal{M}_1}[T_1(h)] + \mathop{\mathbb{E}}_{\mathcal{M}_1}[T_i(h)]\right) \leq \frac{h(\Delta_1)^2}{2}
\end{aligned}$$

from which we derive

$$\max_{\mathcal{M}^* \in \mathcal{M}} \mathcal{R}_h(\mathcal{M}^*, \pi) \geq \frac{h\Delta_1}{8} \exp\left(-\frac{h(\Delta_1)^2}{2}\right).$$

Finally, we substitute the latter in (15) to get

$$\mathcal{R}_H(\mathcal{M}_i, \pi) \geq \frac{h\Delta_1}{8} \exp\left(-\frac{h(\Delta_1)^2}{2}\right) + (H - h)\Delta_1 \geq \frac{\sqrt{H}}{8} \exp\left(-\frac{1}{2}\right)$$

with probability at least δ, where the last inequality is obtained by taking $\Delta_1 = 1/\sqrt{H}$ and noting that the left-hand side is minimized for $\mathbb{E}_{\mathcal{M}_i}[\tau] = h = H$.

Good Event. The previous result states that the regret is at least $\Omega(\sqrt{H})$ when the event $\mathcal{E}$ does not hold with high probability. This hints that solving the best arm identification problem is necessary to minimize the regret. To derive the lower bound, we instantiate a proper best arm identification problem on the considered instance $\mathcal{M}$.

Since the separation condition is fulfilled in the second set of actions, we can restrict our best arm identification problem to the action set $\hat{\mathcal{A}} = (a_j)_{j=M+1}^{2M}$. From Theorem 1 in (Garivier & Kaufmann, 2016), for any confidence $\delta \in (0, 1)$, we have that

$$\mathbb{E}_{\mathcal{M}_i}[\tau] \geq T^*(\mathcal{M}_i)^{-1} \log(1/2.4\delta)$$

where

$$T^*(\mathcal{M}_i)^{-1} := \sup_{\omega \in \mathcal{P}(\hat{\mathcal{A}})} \inf_{\mathcal{M}_j \in \mathcal{M} \setminus (\mathcal{M}_i)} \left(\sum_{k=M+1}^{2M} \omega_k \mathrm{KL}(R_i(a_k), R_j(a_k))\right) \tag{16}$$

holds with probability $1 - \delta$. From the separation condition (Assumption 8) and the Pinsker's inequality we have

$$\mathrm{KL}(R_i(a), R_j(a)) \geq 2\mathrm{TV}^2(R_i(a), R_j(a)) \geq \frac{1}{2}\|R_i(a) - R_j(a)\|_1^2 \geq \frac{\lambda^2}{2}$$

for every $i \neq j$. By noting that the supremum in (16) is attained by $\omega = (1/M, \ldots, 1/M)$ we get

$$\mathbb{E}_{\mathcal{M}_i}[\tau] \geq \frac{2M}{\lambda^2} \log\left(\frac{1}{2.4\delta}\right).$$

Finally, we can derive the lower bound through

$$\mathcal{R}_H(\mathcal{M}_i, \pi) \geq \mathbb{E}_{\mathcal{M}_i}[\tau]\Delta_2 = \frac{2M}{\lambda^2}\frac{\log(H)}{\sqrt{H}} \log\left(\frac{1}{2.4\delta}\right) \geq \frac{2}{\sqrt{C} - \log(C)}\frac{M\log(H)}{\lambda} \log\left(\frac{1}{2.4\delta}\right)$$

where the last inequality is obtained by exploiting $H \leq C$ and that the mean of the reward distribution is bounded in $[0, 1]$ to write $\lambda + \frac{\log(H)}{\sqrt{H}} \leq 1$, which gives $\lambda \leq \frac{\sqrt{C} - \log(C)}{\sqrt{H}}$. □

C.2. Upper Bound

In this section, we provide a simple algorithm, which is practically a direct adaptation to the bandit setting of Algorithm 1, in turn inspired by Chen et al. (2022), that nearly matches the lower bound presented in the previous section.

The idea of the algorithm is to exploit the knowledge of the class $\mathcal{M}$ to quickly identify the test task $\mathcal{M}^*$ and then commit to the optimal action a^* for $\mathcal{M}^*$. The pseudocode of this simple procedure is provided in Algorithm 7.

Algorithm 7 Identify-then-Commit for Bandits

1: Initialize $\mathcal{D} = \mathcal{M}$ and $n = \frac{2\log(2MH)}{\lambda^4}$
2: **while** $|\mathcal{D}| > 1$ **do**
3: Draw $\mathcal{M}_1, \mathcal{M}_2$ from $\mathcal{D}$ randomly
4: Take $\tilde{a} \in \arg\max_{a\in\mathcal{A}} \|R_1(a) - R_2(a)\|_1$
5: Collect n samples $\mathcal{X} = (x_1, \dots, x_n)$ pulling action $\tilde{a}$
6: **if** $\prod_{x_h\in\mathcal{X}} \frac{R_1(x_h|\tilde{a})}{R_2(x_h|\tilde{a})} \geq 1$ **then**
7: Eliminate $\mathcal{M}_2$ from $\mathcal{D}$
8: **else**
9: Eliminate $\mathcal{M}_1$ from $\mathcal{D}$
10: **end if**
11: **end while**
12: Take $\hat{\mathcal{M}} \in \mathcal{D}$ and pull the action $\hat{a} \in \arg\max_{a\in\mathcal{A}} \hat{\mu}(a)$ for the remaining steps

The upper bound to the test-time regret suffered by Algorithm 7 is provided by the following result.

Theorem C.2 (Upper bound)**.** *Let $\mathcal{M}$ be a set of bandits for which Assumption 8 holds. For any $H \geq M-1$, we have*

$$R_H(\mathcal{M}_i, \hat{\pi}) = \mathcal{O}\left(\frac{M\log(MH)}{\lambda^4}\right)$$

where $\hat{\pi}$ is the sampling rule induced by Algorithm 7.

Proof. The scheme of the proof follows closely the one of Chen et al. (2022, Theorem 1), which is simplified and adapted to the bandit setting we care about here.

First, we note that the Algorithm 7 is made of two stages: An "Identify" stage (lines 2-11) in which we seek to find out the test task $\mathcal{M}_i$ irrespective of the regret, a "Commit" stage (line 12) in which we exploit the gathered information to minimize the regret in the remaining steps. Notably, at every iteration of the while loop (lines 2-11) a potential task is eliminated from the set $|\mathcal{D}|$, which means the "Identify" stage consists of exactly $h_0 := (M-1)n$ steps, and the "Commit" stage takes the remaining $H - h_0$ steps. Thus, we can decompose the regret as

$$R_H(\mathcal{M}_i, \hat{\pi}) = \mathbb{E}\left[\sum_{h=1}^{h_0} R_i(a^*) - R_i(\tilde{a})\right] + \mathbb{E}\left[\sum_{h=h_0}^{H} R_i(a^*) - R_i(\hat{a})\right]. \tag{17}$$

Now, we just need to upper bound the term on the left with h_0 through $R_i(a^*) - R_i(\tilde{a}) \leq 1$ and to show that the second term is zero with high probability to prove the result.

Since $\hat{a}$ is the optimal action of the remaining task in the set $\mathcal{D}$, to prove that it holds $\mathbb{E}[\sum_{h=h_0}^{H} R_i(a^*) - R_i(\hat{a})] = 0$ with high probability, we have to show that the test task $\mathcal{M}_i$ is not eliminated from $\mathcal{D}$ with high probability. Especially, for some confidence $\delta \in (0,1)$ we need

$$\mathbb{P}\Big(\text{“}M_i \text{ is eliminated from } \mathcal{D}\text{”}\Big) = \mathbb{P}\left(\prod_{x_h\in\mathcal{X}} \frac{R_i(x_h|\tilde{a})}{R_j(x_h|\tilde{a})} < 1\right) \leq \frac{\delta}{M}$$

where the right-hand side is obtained from a union bound over the event that the test task $\mathcal{M}_i$ is eliminated in each iteration of the while loop (lines 2-11). Equivalently, we need

$$\log\left(\prod_{x_h\in\mathcal{X}} \frac{R_i(x_h|\tilde{a})}{R_j(x_h|\tilde{a})}\right) = \sum_{x_h\in\mathcal{X}} \log\left(\frac{R_i(x_h|\tilde{a})}{R_j(x_h|\tilde{a})}\right) > 0$$

to hold with probability at least $1 - \frac{\delta}{M}$. First, we note that

$$\mathop{\mathbb{E}}_{x\sim R_i(\tilde{a})}\left[\sum_{x_h\in\mathcal{X}} \log\left(\frac{R_i(x|\tilde{a})}{R_j(x|\tilde{a})}\right)\right] = \sum_{x_h\in\mathcal{X}} \mathop{\mathbb{E}}_{x\sim R_i(\tilde{a})}\left[\log\left(\frac{R_i(x|\tilde{a})}{R_j(x|\tilde{a})}\right)\right] = n\mathrm{KL}(R_i(\tilde{a}), R_j(\tilde{a})) \leq \frac{n\lambda^2}{2}$$

where the last inequality is obtained from the separation condition (Assumption 8) and the Pinsker's inequality. Then, we have

$$\sum_{x_h \in \mathcal{X}} \log\left(\frac{R_i(x_h|\tilde{a})}{R_j(x_h|\tilde{a})}\right) \geq \mathop{\mathbb{E}}_{x \sim R_i(\tilde{a})}\left[\sum_{x_h \in \mathcal{X}} \log\left(\frac{R_i(x|\tilde{a})}{R_j(x|\tilde{a})}\right)\right] - \sqrt{\frac{n}{2}\log\left(\frac{2M}{\delta}\right)} \geq \frac{n\lambda^2}{2} - \sqrt{\frac{n}{2}\log\left(\frac{2M}{\delta}\right)}$$

with probability $1 - \frac{\delta}{M}$ through the Hoeffding's inequality. Now, we need to set n such that the right-hand side is greater than zero, which gives $n = \frac{2\log(\frac{2M}{\delta})}{\lambda^4}$ and $h_0 = \frac{2(M-1)\log(\frac{2M}{\delta})}{\lambda^4}$.

Finally, we set $\delta = \frac{1}{H}$ and we plug the expression into (17). Noting that, in the bad event occurring with probability less than $1/H$ the right-hand side of (17) is still less than H, we have $\mathbb{E}[\sum_{h=h_0}^{H} R_i(a^*) - R_i(\hat{a})] \leq 1$ from which we get

$$R_H(\mathcal{M}_i, \hat{\pi}) = \mathcal{O}\left(\frac{M\log(MH)}{\lambda^4}\right).$$

□