Logarithmic regret in communicating MDPs: Leveraging known dynamics with bandits

Hassan Saber HASSAN.SABER@INRIA.FR
Fabien Pesquerel FABIEN.PESQUEREL@INRIA.FR
Odalric-Ambrym Maillard ODALRIC.MAILLLARD@INRIA.FR
Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 – CRIStAL, F-59000, Lille, France

Mohammad Sadegh Talebi M.SHAHI@DI.KU.DK
Universitetsparken 1, DK-2100 Copenhagen Ø, Denmark

Editors: Berrin Yanıkoğlu and Wray Buntine

Abstract

We study regret minimization in an average-reward and communicating Markov Decision Process (MDP) with known dynamics, but unknown reward function. Although learning in such MDPs is a priori easier than in fully unknown ones, they are still largely challenging as they include as special cases large classes of problems such as combinatorial semi-bandits. Leveraging the knowledge on transition function in regret minimization, in a statistically efficient way, appears largely unexplored. As it is conjectured that achieving exact optimality in generic MDPs is NP-hard, even with known transitions, we focus on a computationally efficient relaxation, at the cost of achieving order-optimal logarithmic regret instead of exact optimality. We contribute to filling this gap by introducing a novel algorithm based on the popular Indexed Minimum Empirical Divergence strategy for bandits. A key component of the proposed algorithm is a carefully designed stopping criterion leveraging the recurrent classes induced by stationary policies. We derive a non-asymptotic, problem-dependent, and logarithmic regret bound for this algorithm, which relies on a novel regret decomposition leveraging the structure. We further provide an efficient implementation and experiments illustrating its promising empirical performance.

Keywords: Average-reward Markov decision process, regret minimization, logarithmic regret, Markov chain, recurrent classes

1. Introduction

In Reinforcement learning (RL), a learning agent (henceforth, learner) interacts with an environment that is often modeled using a Markov Decision Process (MDP), and her goal is to optimize a notion of reward (Puterman, 1994; Sutton and Barto, 2018). The learner does not fully know the underlying MDP, and tries to learn a near-optimal behavior quickly from the experience collected via interaction. In the average-reward setting, the learner's performance is often measured in terms of regret, which compares her cumulative reward to that of an optimal policy (Jaksch et al., 2010); equivalently, the learner's goal is to minimize regret. A standard assumption in most settings in RL is that the environment's dynamics is unknown, while the reward function may be known. This assumption is justified since state dynamics are not controlled by the learner, but is also in line with the argument that the main challenge in RL stems from unknown dynamics rather than unknown rewards.

Consequently, the vast majority of existing regret minimizing algorithms have some key ingredient, in design or analysis, to tackle unknown transition probabilities. In model-based algorithms (e.g., Jaksch et al. (2010); Filippi et al. (2010); Burnetas and Katehakis (1997)), this is featured in the form of confidence sets around the empirical transition distributions.

In contrast, in some applications of RL, the learner has some prior knowledge on the transition function; for example, she may know the associated support sets, some transition probabilities, or even the entire transition function up to some small deviation error. This could arise, for example, when the learner has access to an accurate estimate of the transition function via data collected while performing another task on the same environment (but with a different reward function). For instance, in the context of personalized recommendation, where the rewards are given by a user based on her internal evaluation of the recommendations, and where the task (hence transitions) is fixed across users, it is natural to assume that based on previous interactions, the transitions of the system are perfectly known, but the rewards associated to the current user are unknown. Note that although rewards are provided by a user, this does not mean they are known, as evaluation at a point in time can be subjective and noisy. Another scenario could arise in learning tasks where the dynamics are governed by some physical phenomena that are perfectly known to the learner. In such scenarios, the following question arises naturally: *What is the most statistically efficient way to perform exploration when the dynamics are known?*

While any form of prior knowledge on the transition function do not appear directly advantageous to model-free algorithms, which is in line with their design principle, model-based algorithms can benefit directly from it. In the case of perfectly known dynamics, most off-the-shelf algorithms can simply remove the relevant confidence sets, which would lead to improved exploration, and hence, smaller regret bounds.[1] Despite such straightforward modifications of model-based algorithms, it still remains open as to what the best way is to incorporate such prior knowledge into algorithm design in a non-trivial manner, and whether it could lead to instance-dependent (and logarithmic) regret bounds. To our best knowledge, existing literature on learning in MDPs, albeit rich, fails to provide algorithmic ideas to leverage such prior knowledge in a statistically efficient way, and the potential gains thereof in terms of regret or sample complexity remain largely unexplored.

Contributions We focus on regret minimization in communicating MDPs with known dynamics but unknown reward functions, and introduce a class of strategies called *rarely-switching algorithms*, which provide a principled way to leverage the connectivity structure in the MDP through viewing the problem as a multi-policy Multi-Armed Bandit (MAB), thanks to the prior knowledge on the dynamics. The novel design of these strategies considers recurrent classes induced by stationary policies as well as a carefully designed stopping criterion based on the said classes. For these strategies, we present a generic regret bound, which relies on a novel regret decomposition leveraging the structure, which could be of independent interest for learning in MDPs in general. Then, we instantiate a specific rarely-switching algorithm called `IMED-KD`, which uses the popular Indexed Minimum Empirical Divergence (IMED) strategy for MABs (Honda and Takemura, 2015). IMED

[1] For example, it is straightforward to show that UCRL2 (Jaksch et al., 2010), when equipped with the knowledge on dynamics, attains a regret bound of $\widetilde{\mathcal{O}}(\sqrt{(SA+D)T})$ with high probability, in any communicating MDP with S states, A actions, and diameter D, where $\widetilde{\mathcal{O}}(\cdot)$ hides $\log(T)$ terms and universal constants. In contrast, without prior knowledge, UCRL2 achieves a regret of $\widetilde{\mathcal{O}}(DS\sqrt{AT})$.

offers an interesting alternative to optimistic strategies such as UCB or KL-UCB, and to Bayesian strategies such as Thompson sampling. Owing to its form directly inspired by the constraints of the optimization problem appearing in asymptotic regret lower bounds, it has been shown to yield optimal regret performance, like KL-UCB or Thompson Sampling. We stress that a key departure from existing IMED-style algorithms for MDPs (e.g., IMED-RL (Pesquerel and Maillard, 2022)) is to exploit the intrinsic structure of the problem via use of a rarely-switching algorithm. Under some standard assumption on the reward function and MDP regularity, as well as a mild assumption on the involved hitting times (Assumption 5), we derive a non-asymptotic, problem-dependent, and logarithmic regret bound for `IMED-KD`, whose proof relies on the generic properties of rarely-switching algorithms as well as proof machinery of IMED-style indices adapted to MDPs. We further provide an efficient implementation and experiments illustrating its promising empirical performance. To the best of our knowledge, `IMED-KD` is the first algorithm specifically designed to leverage the structure in MDPs with known dynamics.

Related work There is a rich literature on regret minimization in average-reward MDPs. Early papers like (Burnetas and Katehakis, 1997; Graves and Lai, 1997) mostly presented regret bounds for ergodic MDPs and with an asymptotic flavour, whereas more recent literature, e.g., (Jaksch et al., 2010; Filippi et al., 2010; Talebi and Maillard, 2018; Fruit et al., 2018; Zhang and Ji, 2019; Wei et al., 2020; Bourel et al., 2020; Pesquerel and Maillard, 2022), reported non-asymptotic regret guarantees and, often, for the bigger of class of (weakly) communicating MDPs. The majority of recent literature on learning in MDPs, following Jaksch et al. (2010), report worst-case regret bounds growing as $\widetilde{\mathcal{O}}(\sqrt{T})$ after T steps. In contrast, comparatively there exists little work that present logarithmic and instance-dependent regret bounds for average-reward MDPs. The most notable exceptions include (Jaksch et al., 2010), which reports a logarithmic regret bound for UCRL2 (albeit with a large mixing-time related additive term), and more recent papers (Gopalan and Mannor, 2015; Pesquerel and Maillard, 2022), which only consider ergodic MDPs. We also mention the logarithmic regret bounds derived in (Ortner, 2009; Tranos and Proutiere, 2021) for the much simpler setting of MDPs with deterministic transitions.

We also mention that some studies consider regret minimization in MDPs in the *episodic* setting, with a fixed and known horizon; see, e.g., Osband et al. (2013); Azar et al. (2017); Simchowitz and Jamieson (2019), where the latter work presents a problem-dependent, logarithmic regret bound. However, the proof machinery used in episodic RL often fails to work in average-reward RL due to relying on the fixed episode length and resetting of the state. Finally, it is worth remarking that an MDP with known transitions but unknown rewards may be viewed as a MAB instance with highly structured actions (one action corresponding to a policy), in a way which is reminiscent of combinatorial MABs (Chen et al., 2013; Combes et al., 2015). Despite such resemblance, the problem is more challenging as the learner is traversing an MDP without a resetting device. As a result, algorithmic ideas for combinatorial MABs or those with generic structure (Combes et al., 2017; Saber et al., 2020) do not directly carry over to MDPs with known dynamics.

Notations For an integer $n \in \mathbb{N} \cup \{0\}$, we denote $[n] = \{0, \dots, n\}$. For a Boolean event A, $\mathbb{I}\{A\} \in \{0,1\}$ denotes the indicator function of A. For a sequence $(h_t)_{t\in\mathbb{N}}$, and $t_1 < t_2 \in \mathbb{N}$, $h_{t_1:t_2} := (h_t)_{t\in\{t_1,\dots,t_2\}}$ denotes the sub-sequence of elements indexed in between t_1 and t_2. Last, for a set A, $\mathcal{P}(A)$ denotes the set of probability distributions over A.

2. Problem formulation

We consider an average-reward Markov Decision Process $\mathbf{M} = (\mathcal{S}, \mathcal{A}, \mathbf{p}, \mathbf{r})$, where $\mathcal{S}$ is the set of states with cardinality S, and $\mathcal{A} = (\mathcal{A}_s)_{s\in\mathcal{S}}$, where $\mathcal{A}_s$ specifies the set of actions available in $s \in \mathcal{S}$. For convenience, we introduce the set of pairs $\mathcal{C} = \{(s, a) : s \in \mathcal{S}, a \in \mathcal{A}_s\}$. Further, $\mathbf{p} : \mathcal{C} \to \mathcal{P}(\mathcal{S})$ denotes the transition function, and $\mathbf{r} : \mathcal{C} \to \mathcal{P}(\mathbb{R})$ the reward function. We denote the corresponding mean reward function by $\mathbf{m} : \mathcal{C} \to \mathbb{R}$.

Policies Each stationary policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ acting on $\mathbf{M}$ induces a Markov chain on $\mathcal{C}$, with corresponding transition probability $\mathbf{p}_\pi : \mathcal{C}^2 \to \mathcal{P}(\mathcal{C})$, defined by $\mathbf{p}_\pi(s, a)(s', a') = \mathbf{p}(s'|s, a)\pi(a'|s')$. We denote by $\overline{\mathbf{p}}_\pi : \mathcal{C}^2 \to \mathcal{P}(\mathcal{C})$ the Cesaro-average of $\mathbf{p}_\pi$; formally, $\overline{\mathbf{p}}_\pi = \lim\limits_{T\to\infty} \frac{1}{T} \sum\limits_{t=1}^{T} \mathbf{p}_\pi^{t-1}$, where $\overline{\mathbf{p}}_\pi(c_1, c)$ captures the frequency of reaching the pair $c \in \mathcal{C}$ under policy π starting in pair $c_1 \in \mathcal{C}$. This enables to introduce the gain of policy π, when starting from state-action pair $c_1 = (s_1, a_1)$, defined by $\mathbf{g}_{c_1,\pi} := (\overline{\mathbf{p}}_\pi \mathbf{m})(c_1) = \sum_{c\in\mathcal{C}} \overline{\mathbf{p}}_\pi(c_1, c) \cdot \mathbf{m}(c)$, where we recall that $\mathbf{m}(c)$ is the mean reward of pair c. Given a *finite* set of stationary policies Π, we define $\mathbf{g}^\star_c = \max\limits_{\pi\in\Pi} \mathbf{g}_{c,\pi}$ the optimal gain starting from c, and $\Pi^\star_c = \{\pi \in \Pi : \mathbf{g}_{c,\pi} = \mathbf{g}^\star_c\}$ the set of policies achieving the optimal gain.

Cycles The set of *(positive) recurrent* state-action pairs (i.e., pairs with finite return times) under π is defined as $\mathcal{C}^+_\pi = \{c \in \mathcal{C} : \overline{\mathbf{p}}_\pi(c)(c) > 0\}$. Further, the relation $\sim_\pi$ such that $c \sim_\pi c' \Leftrightarrow \overline{\mathbf{p}}_\pi(c)(c') \cdot \overline{\mathbf{p}}_\pi(c')(c) > 0$, is an equivalence relation on $\mathcal{C}^+_\pi$. Denoting $[c]_\pi$ the class of $c \in \mathcal{C}^+_\pi$ for relation $\sim_\pi$, the asymptotic *cycles* under policy π are defined as $\mathcal{X}_\pi = \mathcal{C}^+_\pi / \sim_\pi = \left\{[c]_\pi :\ c \in \mathcal{C}^+_\pi\right\}$. Distinct elements of $\mathcal{X}_\pi$ correspond to disjoint cycles. A policy π with $|\mathcal{X}_\pi| = 1$ is called a *unichain* policy.

Remark 1 *A remarkable property is that for a* unichain policy π *and recurrent* $c' \in \mathcal{C}^+_\pi$, $\overline{\mathbf{p}}_\pi(c)(c')$ *is independent of the starting pair* c *and equals* $1/\boldsymbol{\tau}_\pi(c', c')$, *where* $\boldsymbol{\tau}_\pi(c', c')$ *is the expected hitting time of* c' *when starting from* c' *and following policy* π*; see (Puterman, 1994). As a consequence,* $\mathbf{g}_{c,\pi}$ *also does not depend on* c*.*

We consider the two following assumptions on MDP regularity and the reward function:

Assumption 1 (MDP) $\mathbf{M}$ *is communicating, that is,* $\forall c, c', \exists \pi, t \in \mathbb{N} : \mathbf{p}^t_\pi(c)(c') > 0$*. Also,* Π *is proper, that is, the Cesaro-average* $\overline{\mathbf{p}}_\pi$ *of* $\mathbf{p}_\pi$ *exists for each* $\pi \in \Pi$*. There is a unique gain-optimal policy* $\pi^\star \in \bigcap\limits_{c\in\mathcal{C}} \Pi^\star_c$ *that is* unichain *(i.e., it has a unique asymptotic cycle).*

Assumption 2 (Reward function) *For each* $c \in \mathcal{C}$*, the reward distribution* $\mathbf{r}(c)$ *is supported on* $[0, 1]$ *(in particular, it is 1/2-sub-Gaussian), with bounded mean* $\mathbf{m}(c) \in [0, 1)$*.*

In particular, under Assumption 2, $\forall c \in \mathcal{C}, \pi \in \Pi$, the gain is bounded: $\mathbf{g}_{c,\pi} \in [0, 1)$. To gain insight into the motivations behind this assumption, we refer to discussion in Appendix F.

Local monotony Finally, a key property of *unichain* MDPs is that there always exists a modification of a sub-optimal policy in a single state having (stricly) larger gain (Puterman, 1994). We generalize this useful monotony property to *larger neighborhoods* as follows:

Assumption 3 (Policy-improving neighborhood) $\forall c \in \mathcal{C}, \pi \notin \Pi^\star_c$, $\exists \pi' \in \Pi$, $\hbar(\pi, \pi') \leqslant k$, *such that* $\mathbf{g}_{c,\pi'} > \mathbf{g}_{c,\pi}$*, where* $\hbar$ *denotes the Hamming distance between two policies.*

Here, k is a given constant. Note that as k increases from 1 to S, Assumption 3 interpolates between (at least) all unichain MDPs, when $k = 1$, and all discrete MDPs, when $k = S$.

Remark 2 *In a communicating MDP, for Π consisting of all stationary policies, the set of optimal policies does not depend on the starting pair and is simply denoted by $\Pi^\star$. Moreover, an optimal policy is also unichain in this case. The gain of a unichain policy π does not depend on the starting state-action pair, and is simply denoted by $\mathbf{g}_\pi$ (and $\mathbf{g}^\star$ for an optimal policy); hence, we denote $\Pi^\star_c = \Pi^\star$ and $\mathbf{g}^\star_c = \mathbf{g}^\star$ for all c. Note, however, that for sub-optimal policies $\pi \in \Pi$ that are not unichain, $\mathbf{g}_{c,\pi}$ may still depend on the initial state-action $c \in \mathcal{C}$.*

The online learning problem The learner interacts with MDP $\mathbf{M}$ for T time steps, starting in an initial state-action pair $(s_1, a_1) \in \mathcal{C}$ chosen by Nature. At each time $t \geq 2$, she is in state $s_t \in \mathcal{S}$ and chooses an action $a_t \in \mathcal{A}_{s_t}$ according to a stationary policy $\pi_t \in \Pi$, that is $a_t \sim \pi_t(s_t)$. The stationary policy π_t is selected based on the learner's observations so far. Then, (i) she receives a reward $r_t \in [0, 1]$, where $r_t \sim \mathbf{r}(s_t, a_t)$; and (ii) Nature decides a next state $s_{t+1} \in \mathcal{S}$, where $s_{t+1} \sim \mathbf{p}(\cdot | s_t, a_t)$. The sequence of chosen policies is denoted by $(\pi_t)_{t \geqslant 1}$, and simply by (π) when for all time step $t \geqslant 1$, $\pi_t = \pi$. Further, we denote by $c_t = (s_t, a_t)$ the state-action pair at time step t. We assume that the learner *does not know* the reward function $\mathbf{r}$, but *knows* the transition function $\mathbf{p}$, and can thus compute $\overline{\mathbf{p}}_\pi$ for each $\pi \in \Pi$. Her performance is measured through the notion of (expected) regret, as defined next. Let $V_\mathbf{M}(\mathbb{A}, T)$ denote the cumulative reward of an algorithm $\mathbb{A}$ following a policy sequence $(\pi_t)_{t \leqslant T}$ up to time T:

$$V_\mathbf{M}(\mathbb{A}, T) = \mathbb{E}_{(\pi_t)}\left[\sum_{t=1}^{T} r_t\right].$$

For a policy sequence (π), it is simply denoted by $V_\mathbf{M}((\pi), T)$. The (expected) regret with respect to playing a gain-optimal policy sequence $(\pi^\star)$, up to time T, is defined as:

$$\mathcal{R}_\mathbf{M}(\mathbb{A}, T) = V_\mathbf{M}((\pi^\star), T) - V_\mathbf{M}(\mathbb{A}, T). \tag{1}$$

Remark 3 (Pseudo-regret) *For each given T, the quantity $V^\star_\mathbf{M}(T) = \max_{\pi \in \Pi} V_\mathbf{M}((\pi), T)$ and the set $\operatorname*{Argmax}_{\pi \in \Pi} V_\mathbf{M}((\pi), T)$ differ a priori from the cumulative reward of gain-optimal policies $(V_\mathbf{M}((\pi^\star), T))_{\pi^\star \in \Pi^\star}$ and the set $\Pi^\star$, respectively. However, it is easily checked that $\lim_{T\to\infty} V^\star_\mathbf{M}(T)/T = \lim_{T\to\infty} V_\mathbf{M}((\pi^\star), T)/T = \mathbf{g}^\star$, for all gain-optimal stationary policy $\pi^\star \in \Pi^\star$. That is, the asymptotic maximal average reward coincides both with the asymptotic average reward of gain-optimal policies and the optimal gain. Since the set of considered stationary policies Π is finite, this further implies that $\Pi^\star_T \subset \Pi^\star$ when horizon T is large enough, which also implies $V^\star_\mathbf{M}(T) - V_\mathbf{M}((\pi^\star), T) \overset{T\to\infty}{=} O(1)$.*

3. Rarely-switching Algorithms

Rarely-switching learners We choose to restrict the learner to follow a *rarely-switching* strategy, which forces the learner to keep playing the same policy until some criterion — to be introduced momentarily — is met. The T time steps are divided into episodes of random durations, where episode $k \in \{1, 2, \dots\}$ starts at random time $\tau_{k-1} + 1$ and ends at random time τ_k (with $\tau_0 = 0$). We gather in the sequence $\mathcal{T} = (\tau_k)_{k\in\mathbb{N}}$ the last time step before starting each new episode. Hence, for $\tau \in \mathcal{T}$, the learner starts at time step $\tau+1$ a new episode (to which we refer as "episode τ"), and after pulling state-action pair $c_{\tau+1} = (s_{\tau+1}, a_{\tau+1})$, she follows the same policy $\pi = \pi_{\tau+1}$ until the event `Event` is triggered (and the episode ends). `Event` is a generic function of the current policy π and the history $h_{\tau+1:t}$ of all observations and decisions made from the beginning of the episode until the current time. We resume the generic structure of rarely-switching learners in Algorithm 1.

Algorithm 1 Rarely-switching learner

```
input: $(\mathbf{p}_\pi)_{\pi\in\Pi}$, $(s_1, \pi_1)$ and Event function
Start a new episode $\tau \leftarrow 0$, $\pi \leftarrow \pi_1$
Pull action $a_1 \sim \pi(s_1)$
for time step $t \geqslant 1$ do
    Receive reward $r_t$, update history $h_t = (s_t, \pi, a_t, r_t)$
    if $\neg$Event$(\pi, h_{\tau+1:t})$ then
        Keep the same policy $\pi \leftarrow \pi_{\tau+1}$
        Pull action $a_{t+1} \sim \pi(s_{t+1})$
    else
        Start a new episode $\tau \leftarrow t$
        Compute a new policy $\pi_{\tau+1}$ and update $\pi \leftarrow \pi_{\tau+1}$.
        Pull action $a_{t+1} \sim \pi(s_{t+1})$
    end if
end for
```

Counters For a rarely-switching learner, let $N^{\text{ini}}_{c,\pi}(0:T) = \sum\limits_{\tau\in\mathcal{T}\cap[T]} \mathbb{I}\{\pi_{\tau+1} = \pi, c_{\tau+1} = c\}$ denote the number of times when an episode starts in pair c and follows policy π until time T. This quantity should not be confused with the (possibly much larger) number of visits $N_{c,\pi}(T) = \sum_{t\in[T]} \mathbb{I}\{\pi_t = \pi, c_t = c\}$ of pair c by policy π until time T. In view of the introduction of `Event`, it is also convenient to introduce $N_c(h) = \sum_{(s,\pi,a,r)\in h} \mathbb{I}\{(s,a) = c\}$ that counts the number of visits of pair c on the piece of history h.

Owing to the fact that the criterion used to stop an episode is *independent* of the rewards accumulated during the episode, and using properties of the expectation, we can show the following decomposition lemma, somewhat reminiscent of bandit analyses.

Assumption 4 (Whole number of episodes) *We assume that $T \in \mathcal{T}$, that is horizon time T coincides with the last time step of an episode. We abusively conserve the notation $\mathbb{E}_{(\pi_t)}[Z]$ instead of $\mathbb{E}_{(\pi_t)}[Z|T \in \mathcal{T}]$ to compute the expectation of any random variable Z.*

Lemma 4 (Cumulative reward and regret decomposition) *Under Assumption 4, the cumulative reward of a rarely-switching algorithm $\mathbb{A}$ satisfies*

$$V_{\mathbf{M}}(\mathbb{A}, T) = \sum_{c\in\mathcal{C}} \sum_{\pi\in\Pi} \mathbb{E}_{(\pi_t)}\left[N^{ini}_{c,\pi}(0\!:\!T)\right] \cdot \mathbb{E}[\ell_{c,\pi}] \cdot G_{c,\pi}\,,$$

where $\ell_{c,\pi} = \min\{t > 0 : \mathtt{Event}(\pi, h_{1:t}), c_1 = c\}$ denotes the (random) length of the episode, and where $G_{c,\pi} = \mathbb{E}_{(\pi_t)}\left[\frac{1}{\ell_{c,\pi}} \sum\limits_{t=1}^{\ell_{c,\pi}} r_t \middle| (s_1, a_1) = c\right]$ denotes the expected average reward of an episode starting in pair c and following policy π. When `Event` further ensures an episode *running policy π always stops in a same reference pair $c_\pi \in \mathcal{C}$, then writing $G^\star = G_{c_{\pi^\star},\pi^\star}$, it holds*

$$\begin{aligned}\mathcal{R}_{\mathbf{M}}(\mathbb{A},T) \;&=\; \sum_{\substack{c\in\mathcal{C}\\ \pi\neq\pi^\star}} \mathbb{E}_{(\pi_t)}\left[N^{ini}_{c,\pi}(0\!:\!T)\right]\cdot\mathbb{E}[\ell_{c,\pi}]\cdot(G^\star\!-G_{c,\pi})\\ &\quad+\sum_{c\neq c_{\pi^\star}} \mathbb{E}_{(\pi_t)}\left[N^{ini}_{c,\pi^\star}(1\!:\!T)\right]\cdot\mathbb{E}[\ell_{c,\pi^\star}]\cdot(G^\star\!-G_{c,\pi^\star})\,. \end{aligned} \tag{2}$$

Note that $N^{\mathrm{ini}}_{c,\pi^\star}(1\!:\!T)$ excludes the first episode. Furthermore, we stress that $G^\star$ is defined using the stopping time induced by $\pi^\star$, and $G_{c,\pi}$ by the one induced by π.

The proof of Lemma 4 is provided in Appendix A. To give some intuition, Lemma 4 decomposes the cumulative reward of a rarely-switching learner according to *each configuration* when the policy being played is π and the initial pair in this episode is c. Thus, it makes appear the number of times such a configuration happens, $\mathbb{E}_{(\pi_t)}\left[N^{\mathrm{ini}}_{c,\pi}(0\!:\!T)\right]$, as well as the reward accumulated in that episode. A similar decomposition can be written for the optimal policy, and using a reference state ensures that $R_{\mathbf{M}}(\pi^\star,T)\simeq TG^\star$, up to the contribution of the first episode in which the episode may not start from $c_{\pi^\star}$. Combining the two cumulative reward decompositions yields the convenient form in Equation (2).

Further, the product form term $\mathbb{E}[\ell_{c,\pi}]\cdot G_{c,\pi}$ reveals that Lemma 4 offers a decoupling between the expected number $\mathbb{E}[\ell_{c,\pi}]$ of steps of an episode starting in c with policy π, and its average reward $G_{c,\pi}$ received during that episode. It is worth mentioning that the decoupling between the gain and the length of an episode holds by virtue of the Markov property and since we consider a decomposition in expectation.

Remark 5 (Simplifications) *Note that $\mathbb{E}_{(\pi_t)}\left[N^{ini}_{c,\pi}(0\!:\!T)\right]=0$ for policies π not explored by a rarely-switching algorithm. Typically, a learning algorithm will progressively focus on a few policies, and hence, the sum over all stationary policies π should effectively involve much fewer terms than A^S (i.e., the number of all stationary deterministic policies). Interestingly, in the case of bandits, there is a unique state, and hence, Equation (2) simplifies to the classical regret decomposition, in which case the second term disappears:*

$$\sum_{c\neq c_{\pi^\star}} \mathbb{E}_{(\pi_t)}\left[N^{ini}_{c,\pi^\star}(1\!:\!T)\right]\cdot\mathbb{E}[\ell_{c,\pi^\star}]\cdot(G^\star\!-G_{c,\pi^\star})=0\,.$$

Gain One may wonder about the link between $G_{c,\pi}$ and the gain $\mathbf{g}_{c,\pi}$: $G_{c,\pi}$ can be seen as a proxy for the gain $\mathbf{g}_{c,\pi}$ of the policy, since $\mathbf{g}_{c,\pi}=\lim\limits_{T\to\infty}\mathbb{E}_{(\pi_t)}\left[\frac{1}{T}\sum\limits_{t=1}^{T} r_t\middle| c_1=c,\pi_1=\pi\right]$, that is, as $\ell_{c,\pi}\to\infty$, then $G_{c,\pi}$ indeed approaches $\mathbf{g}_{c,\pi}$. This interpretation is however valid only when $\ell_{c,\pi}$ is sufficiently large. Luckily, thanks to the regenerating properties of the chain, if we start and stop an episode in the same *recurrent* pair c_π, hence "completing a loop", then the average of the rewards received during that episode must, in expectation, equal that of infinitely many such loops. More formally:

Lemma 6 (Regeneration property) *For any unichain policy π, any recurrent reference pair $c_\pi\in\mathcal{C}^+_\pi$, and any function `Event` ensuring that an episode always stops in c_π when we play π, then $G_{c_\pi,\pi}=\mathbf{g}_{c_\pi,\pi}$, that is the expected average reward received during an episode starting and ending at pair c_π is equal to the gain of the policy.*

A proof of Lemma 6 is provided in Appendix A.1. This motivates us to introduce for each π a reference pair $c_\pi \in \mathop{\mathrm{Argmax}}_{c\in\mathcal{C}} \overline{\mathbf{p}}_\pi(c)(c)$ (which belongs to $\mathcal{C}_\pi^+$ by construction), and define $\texttt{Event}(\pi, h_{\tau+1:t})$ to ensure that $(s_t, a_t) = c_\pi$. Indeed, this choice of c_π also minimizes $\boldsymbol{\tau}_\pi(c,c)$ over c, hence tends to reduce $\mathbb{E}[\ell_{c_\pi,\pi}]$. This construction of events further yields the following useful control on the regret:

Proposition 7 (Rarely-switching learners with reference pair) *Under Assumption 1, if the rarely-switching learner $\mathbb{A}$ specifies for each π to stop the episode starting with π in the same reference pair $c_\pi \in \mathcal{C}_\pi^+$, then the following bound holds almost surely:*

$$\sum_{c\neq c_{\pi^\star}} N^{ini}_{c,\pi^\star}(1\!:\!T) \leqslant \sum_{c\in\mathcal{C}} \sum_{\pi\neq\pi^\star} N^{ini}_{c,\pi}(0\!:\!T)\,.$$

Moreover, the cumulative regret of any such rarely-switching algorithm $\mathbb{A}$ with respect to the unique optimal policy $\pi^\star$, up to the end T of any episode, is upper-bounded by

$$\mathcal{R}_{\mathbf{M}}(\mathbb{A},T) \;\leqslant\; \mathbb{E}_{(\pi_t)}\Bigg[\sum_{c\in\mathcal{C},\pi\neq\pi^\star} N^{ini}_{c,\pi}(0\!:\!T)\Bigg]\cdot\Bigg(\max_{(c,\pi)\neq(c_{\pi^\star},\pi^\star)} \mathbb{E}[\ell_{c,\pi}](G^\star - G_{c,\pi}) + \mathbf{B}_\star\Bigg)\,,$$

where $\mathbf{B}_\star := \max_{c\neq c_{\pi^\star}} \mathbb{E}_{(\pi_t)}[\ell_{c,\pi^\star}](G^\star - G_{c,\pi^\star})$ is a problem-dependent quantity.

Remark 8 *It holds $\mathbf{B}_\star \leqslant \max_{(c,\pi)\neq(c_{\pi^\star},\pi^\star)} \mathbb{E}[\ell_{c,\pi}](G^\star - G_{c,\pi})$. Further, $\mathbf{B}_\star = 0$ for bandits.*

Estimation and covering time Before we specify the algorithm, let us remind that since the transitions are known, only the mean rewards need to be estimated. Since $\mathbf{g}_{c,\pi} = \sum_{c'\in\mathcal{C}_\pi^+} \overline{\mathbf{p}}_\pi(c)(c')\mathbf{m}(c')$, where $\mathbf{m}$ is unknown, it is natural to collect observations of pairs $c' \in \mathcal{C}_\pi^+$ to estimate the corresponding $\mathbf{m}(c')$, and hence the gain $\mathbf{g}_\pi$. A natural way to ensure the estimation error reduces in each episode is to stop an episode when all pairs in $\mathcal{C}_\pi^+$ have been visited at least once: Formally, $\min_{c'\in\mathcal{C}_\pi^+} N_{c'}(h_{\tau+1:t}) > 0$, that is after *covering* the set $\mathcal{C}_\pi^+$. In order to control the resulting episode length, unfortunately, there is in general no simple control of the cover time by a policy π of its recurrent pairs. The policy could be diffusive or lazy (see Appendix B), yielding an arbitrarily large cover time. Formally, given $C \subset \mathcal{C}$ and $c \in \mathcal{C}$, we denote by $\pi_c^H(C)$ a policy that minimizes over policies π the expected time $\boldsymbol{\tau}^H_{c,\pi}(C)$ to reach *any element* of C starting from c and following π. In a similar manner, we let $\overline{\pi}_c(C)$ denote a policy minimizing over π the expected time $\overline{\boldsymbol{\tau}}_{c,\pi}(C)$ to cover *all elements* of C starting from c and following π. Letting $D_{\mathbf{M}}$ denote the diameter of $\mathbf{M}$,[2] it holds: $\min_\pi \boldsymbol{\tau}^H_{c,\pi}(C) \leqslant D_{\mathbf{M}}$ and $\min_\pi \overline{\boldsymbol{\tau}}_{c,\pi}(C) \leqslant |C| D_{\mathbf{M}}$ for all c and C. In contrast, $\overline{\boldsymbol{\tau}}_{c,\pi}(\mathcal{C}_\pi^+)$ could be *arbitrarily large*, even for a gain-optimal policy π.

[2] The diameter of a finite MDP $\mathbf{M}$ is defined as $D_{\mathbf{M}} = \max_{s\neq s'} \min_\pi \mathbb{E}[T^\pi(s,s')]$, where $T^\pi(s,s')$ denotes the number of steps it takes to reach s' starting from s and following policy π (Jaksch et al., 2010).

Frequently recurrent pairs and restricted gain This motivates us to discard states with *too small return frequency.* To formalize this, we introduce a notion of gain, which we call η*-restricted gain*, defined using a parameter $\eta \in \mathbb{R}^+$. Formally, for a constant $\eta \in \mathbb{R}^+$, define the set of frequently recurrent pairs of a stationary policy π:

$$\text{(Frequently recurrent pairs)} \qquad \mathcal{C}^+_{c,\pi}(\eta) := \{c' \in \mathcal{C} : \overline{\mathbf{p}}_\pi(c)(c') > \eta\}\,,$$

which leads to defining the corresponding η-restricted gain function:

$$(\eta\text{-restricted gain}) \qquad \mathbf{g}_{c,\pi}(\eta) := \sum_{c' \in \mathcal{C}^+_{c,\pi}(\eta)} \overline{\mathbf{p}}_\pi(c)(c') \cdot \mathbf{m}(c') / \sum_{c' \in \mathcal{C}^+_{c,\pi}(\eta)} \overline{\mathbf{p}}_\pi(c)(c')\,.$$

We further naturally introduce $\mathbf{g}^\star_c(\eta) = \max_{\pi\in\Pi} \mathbf{g}_{c,\pi}(\eta)$ and $\Pi^\star_c(\eta) = \operatorname{Argmax}_{\pi\in\Pi} \mathbf{g}_{c,\pi}(\eta)$. Note that for $\eta = 0$, we recover the usual definitions (e.g., $\mathbf{g}_{c,\pi}(0) = \mathbf{g}_{c,\pi}$). More generally:

Lemma 9 (Restricted-gain approximation)

$$\forall \pi, c, \eta, \quad \mathbf{g}_{c,\pi} - \mathbf{g}_{c,\pi}(\eta) \quad \leqslant \quad \eta \mathbf{m}_{\max} |\mathcal{C}^+_{c,\pi} \setminus \mathcal{C}^+_{c,\pi}(\eta)|\,,$$

where $\mathbf{m}_{\max} = \max_{c\in\mathcal{C}} \mathbf{m}(c)$ *is the maximal state-action pair mean.*

This lemma is proven in Appendix C. In particular, for a given ε, choosing $\eta \leqslant \frac{\varepsilon}{\mathbf{m}_{\max}|\mathcal{C}^+_{c,\pi}\setminus\mathcal{C}^+_{c,\pi}(\eta)|}$ (for instance, $\eta = \varepsilon/(\mathbf{m}_{\max} S)$) ensures that the gain is still well-approximated by the η-restricted gain up to the desired precision ε. Hence, we can restrict to cover $C = \mathcal{C}^+_{c,\pi}(\eta)$ instead of $\mathcal{C}^+_{c,\pi}$ and define $\texttt{Event}(\pi, h_{\tau+1:t})$ accordingly. Unfortunately, $\overline{\boldsymbol{\tau}}_{c,\pi}(\mathcal{C}^+_\pi(\eta))$ can still be arbitrary in general. This motivates us to introduce:

Definition 10 (Laziness) *A chain induced by* π *is* (B,η)*-lazy if* $\max_{c'\in\mathcal{C}^+_{c,\pi}(\eta)} \overline{\boldsymbol{\tau}}_{c',\pi}(\mathcal{C}^+_{c,\pi}(\eta)) > B$.

We assume (the laziness constant B may be unknown to the learner, or computed offline):

Assumption 5 (No-laziness) $\mathbf{M}$ *has no* (B,η)*-lazy chain, where* $\eta \in [0,1]$ *is given.*

Structure of policies We conclude this section by showing that choosing this specific form of event further enables us to revisit the decomposition of regret to better exploit structure of the policies. Indeed, while $\mathbb{E}_{(\pi_t)}\big[N^{\text{ini}}_{c,\pi}(0\!:\!T)\big] = 0$ for policies π not explored by a rarely-switching learner, there is more: policies are structured, in the sense that visiting one state-action pair (s,a) is not only informative about the actual policy π playing a in state s, but all such ones as well. Using Proposition 7 and the form of stopping event introduced in Lemma 12 (in the next section), we derive the following result, showing, remarkably, that the **sum over all policies can be removed in favor of a maximum**.

Theorem 11 (Rarely-switching learners exploiting recurrence structure) *Let* $\mathbb{A}$ *be a rarely-switching algorithm using stopping event* $\texttt{Event}(\pi, h_{\tau+1:t}) = \{\min_{c'\in C} N_{c'}(h_{\tau+1:t}) > 0 \text{ and } (s_t, a_t) = c_\pi\}$ *where* $C = \mathcal{C}^+_{c,\pi}(\eta)$ *is parameterized by* η. *Then,*

$$\sum_{c\in\mathcal{C},\ \pi\neq\pi^\star} N^{ini}_{c,\pi}(0\!:\!T) \leqslant |\mathcal{C}| \max_{\substack{c\in\mathcal{C}\\ \pi\neq\pi^\star}} \mathbf{N}^\eta_{c,\pi}(T)\,,$$

where we introduced $\mathbf{N}^{\eta}_{c,\pi}(t) = \min_{c' \in \mathcal{C}^{+}_{c,\pi}(\eta)} N_{c'}(h_{1:t})$. *In particular, using Remark 8,*

$$\frac{\mathcal{R}_{\mathbf{M}}(\mathbb{A},T)}{\mathbb{E}_{(\pi_t)}[\max_{c\in\mathcal{C},\pi\neq\pi^\star}\mathbf{N}^{\eta}_{c,\pi}(T)]} \leqslant |\mathcal{C}|\Bigg(\underbrace{\max_{(c,\pi)\neq(c_{\pi^\star},\pi^\star)}\mathbb{E}[\ell_{c,\pi}]}_{=:\mathbf{L}}\underbrace{(G^\star - G_{c,\pi})}_{\in[-1,1]} + \mathbf{B}_\star\Bigg) \leqslant 2|\mathcal{C}|\mathbf{L}\,. \tag{3}$$

4. The IMED-KD strategy

In this section, we present `IMED-KD` (Indexed Minimum Empirical Divergence for MDPs with Known Dynamics), which is a rarely-switching algorithm that uses an IMED-type index together with the knowledge of $\mathbf{p}$ to attain a logarithmic regret in communicating MDPs. The IMED strategy (Honda and Takemura, 2015) has been proven asymptotically optimal in stochastic MABs and is computationally appealing when compared with the optimistic KL-UCB or the Bayesian Thompson sampling (TS) strategy that require, at each step, solving an optimization problem or sampling from a posterior, respectively. Although posterior sampling can be made efficient for some parametric distributions such as Gaussians, current extensions of TS to MDPs require introducing a forced optimism mechanism (Agrawal and Jia, 2017), which makes it less appealing both from theory and computational perspectives.

High-level description At a high level, the algorithm computes at the beginning of each episode τ an empirical best candidate policy $\widehat{\pi}^\star_\tau$, as well as a best informative policy $\widehat{\pi}^I_\tau$. The algorithm considers the stopping event targeting $C = \mathcal{C}^{+}_{c_\tau,\pi}(\eta)$ and final pair $c_0 = c_\pi$ for the policy $\pi = \widehat{\pi}^I_\tau$. It runs the episode using $\widehat{\pi}^H_\tau$ until hitting C, followed by policy π (so if $c_\tau \in C$, this reduces to running π). We now detail the computation of $\widehat{\pi}^\star_\tau, \widehat{\pi}^I_\tau$, and $\widehat{\pi}^H_\tau$.

a. Empirical best policy $\widehat{\pi}^\star_\tau$ is computed via classical value (or policy) iteration algorithms in the MDP $\widehat{\mathbf{M}}_\tau = (\mathcal{S},\mathcal{A},\mathbf{p},\widehat{\mathbf{r}}_\tau)$ where for each $c \in \mathcal{C}$, we introduce $\widehat{\mathbf{r}}_\tau(c) = \mathcal{N}(\widehat{\mathbf{m}}_\tau(c),\sigma^2)$ with $\widehat{\mathbf{m}}_\tau(c) = \frac{1}{N_c(h_\tau)}\sum_{t'=1}^{\tau} r_{t'}\mathbb{I}\{c_{t'} = c\}$ being the classical empirical estimate of the mean $\mathbf{m}(c)$ computed on observations received until time τ.

b. Informative policy To compute $\widehat{\pi}^I_\tau$, we first form $\widehat{\mathbf{g}}^\star_{c,\tau}(\eta) = \widehat{\mathbf{g}}_{c,\widehat{\pi}^\star_\tau,\tau}(\eta)$, where for each policy π, we introduced its *η-restricted gain estimate* defined by

$$\widehat{\mathbf{g}}_{c,\pi,\tau}(\eta) = \frac{\sum_{c'\in\mathcal{C}^{+}_{c,\pi}(\eta)}\overline{\mathbf{p}}_\pi(c)(c')\widehat{\mathbf{m}}_\tau(c')}{\sum_{c'\in\mathcal{C}^{+}_{c,\pi}(\eta)}\overline{\mathbf{p}}_\pi(c)(c')}.$$

We further introduce for each policy the notation $\mathbf{N}_\pi(\tau) = \mathbf{N}^{\eta}_{c_\tau,\pi}(\tau)$ and the IMED-type index, inspired by (Honda and Takemura, 2015) for MABs,

$$I_\tau(\pi) = \mathbf{N}_\pi(\tau)\mathtt{d}\big(\widehat{\mathbf{g}}_{c_\tau,\pi,\tau}(\eta)\big|\widehat{\mathbf{g}}^\star_{c_\tau,\tau}(\eta)\big) + \log(\mathbf{N}_\pi(\tau)),$$

where $\mathtt{d}(x|y) = \frac{(x-y)^2}{2\sigma^2} = 2(x-y)^2$ denotes the Kullback-Leibler divergence between Gaussian distributions with respective means x and y, and identical standard deviation $\sigma = 1/2$. This is justified since under Assumption 2, all gains fall in $[0,1]$, and hence can be considered 1/2-sub-Gaussians. Finally, we let $\widehat{\pi}^I_\tau$ (also written $\tilde{\pi}_{\tau+1}$) be a policy minimizing I_τ over a subset of policies $\Pi_\tau \subset \Pi$ containing $\widehat{\pi}^\star_\tau$. Following Assumption 3, we introduce $\mathcal{V}_{\widehat{\pi}^\star_\tau}(k) = \{\pi : \mathtt{h}(\widehat{\pi}^\star_\tau,\pi) \leqslant k\}$, and define Π_τ such that $\mathcal{V}_{\widehat{\pi}^\star_\tau} \subset \Pi_\tau$. We discuss choices of Π_τ in Section 6.

c. Exploratory policy To compute the fast hitting policy $\widehat{\pi}^H_\tau = \pi^H_c(C)$ that tries to reach $C = \mathcal{C}^+_{c_\tau,\widehat{\pi}^I_\tau}(\eta)$ as fast as possible starting from $c = c_\tau$, we introduce a specific MDP $\mathbf{M}^H_\tau = (\mathcal{S}, \mathcal{A}, \mathbf{p}, \mathbf{r}^H_\tau)$ with modified reward function $\mathbf{r}^H_\tau(c) = \begin{cases} 1 & \text{if } c \in C \\ 0 & \text{else} \end{cases}$. We compute an optimal policy for this MDP, under the average reward criterion, using value iteration. This policy is used to reach the set C and ensures the hitting time is always finite.

Strategy Finally, we define `IMED-KD` to be the rarely-switching algorithm with update rule (line 11 of Algorithm 1) given by choosing at each new episode $\tau \in \mathcal{T}$ the policy

$$\pi_{\tau+1} = \pi^H_{c_\tau}(\mathcal{C}^+_{c_\tau,\widehat{\pi}^I_\tau}(\eta)) \text{ followed by } \widehat{\pi}^I_\tau\,,$$

with stopping event $\texttt{Event}(\pi_{\tau+1}, h_{\tau+1:t}) = \left\{ \min_{c' \in \mathcal{C}^+_{c_\tau,\widehat{\pi}^I_\tau}(\eta)} N_{c'}(h_{\tau+1:t}) > 0 \,\text{and}\, (s_t, a_t) = c_{\widehat{\pi}^I_\tau} \right\}$.

We provide the following control on the length of episodes run with `IMED-KD`, whose proof is given in Appendix C (together with a complementary control for generic learners).

Lemma 12 (Bound on episode lengths) *Assuming* $\mathbf{M}$ *has diameter* $D_\mathbf{M}$ *and no* (B, η)*-lazy chain, the expected length of an episode of* `IMED-KD` *started at* τ *satisfies*

$$\mathbb{E}[\ell_{c,\pi} | h_{1:\tau}, c_\tau = c] \leqslant D_\mathbf{M} + 2B.$$

5. Regret performances

In this section, we provide performance bounds of the `IMED-KD` strategy, starting with a non-asymptotic control on the number of visits of sub-optimal policies. We stress that the existing lower performance bounds from, e.g., (Burnetas and Katehakis, 1997) are explicit only for ergodic MDPs, and presumably NP-hard to compute in general. Hence, we allow for deviating from this and derive an upper-bound involving a different problem-dependent term. Closing the gap is an interesting challenge (both computationally and theoretically).

Theorem 13 (Performance bound of `IMED-KD`) *For an MDP* $\mathbf{M}$ *with diameter* $D_\mathbf{M}$ *and satisfying Assumptions 1–5, the* `IMED-KD` *strategy ensures, provided that* $\eta < \frac{\varepsilon_\mathbf{M}(0)}{2\mathbf{m}_{\max} S}$ *where* $\varepsilon_\mathbf{M}(\eta) = \min_{\substack{c \in \mathcal{C} \\ \pi \notin \Pi^\star}} \left\{ \max_{\pi' \in \mathcal{V}_\pi} \mathbf{g}_{c,\pi'}(\eta) - \mathbf{g}_{c,\pi}(\eta) \right\}$, *the following*

$$\mathbb{E}_{(\pi_t)}\left[\max_{\substack{c \in \mathcal{C} \\ \pi \neq \pi^\star}} \mathbf{N}^\eta_{c,\pi}(T) \right] \leqslant \max_{\substack{c \in \mathcal{C} \\ \pi \neq \pi^\star}} \frac{(1 + \alpha_\mathbf{M}(\varepsilon)) \log(T)}{\mathsf{d}\big(\mathbf{g}_{c,\pi}(\eta) \big| \mathbf{g}^\star_c(\eta)\big)} + K_T(\varepsilon, \eta)(D_\mathbf{M} + 2B)\,,$$

for all accuracy $0 < \varepsilon < \frac{\varepsilon_\mathbf{M}(\eta)}{2}$, *where* $\lim_{\varepsilon \to 0} \alpha_\mathbf{M}(\varepsilon) = 0$ *and*

$$K_T(\varepsilon, \eta) \leqslant \frac{5\,|\mathcal{C}|\, e^{2\varepsilon^2}}{2\varepsilon^2} + |\mathcal{C}| \left(1 + c^{-1}_{\varepsilon_\mathbf{M}(\eta)} + 2C_{\varepsilon_\mathbf{M}(\eta)} \sqrt{\log(c_{\varepsilon_\mathbf{M}(\eta)} T)} \right).$$

with C_ε *and* c_ε *being constants independent of* $\mathbf{M}$ *and* T.

We combine this result together with Equation (3) and the fact that $G_{c,\pi} \leqslant 1$ to obtain

$$\mathcal{R}_\mathbf{M}(\mathbb{A}, T) \leqslant \left[\max_{\substack{c \in \mathcal{C} \\ \pi \neq \pi^\star}} \frac{(1 + \alpha_\mathbf{M}(\varepsilon)) \log(T)}{\mathsf{d}\big(\mathbf{g}_{c,\pi}(\eta) \big| \mathbf{g}^\star_c(\eta)\big)} + K_T(\varepsilon, \eta)(D_\mathbf{M} + 2B) \right] \cdot 2(D_\mathbf{M} + 2B)\,|\mathcal{C}|\,. \quad (4)$$

Remark 14 *A natural question is how to ensure η is small enough since $\varepsilon_{\mathbf{M}}(0)$ is a priori unknown. One possible way to accommodate this is to consider near-optimality instead, with given precision $\tilde{\varepsilon}$, and simply choose $\eta = \tilde{\varepsilon}/2S$. In practice, we may choose η adaptively (i.e., $\eta = \eta_t$); we discuss in Appendix H some simple adaptive choices of η, and demonstrate that they lead to promising empirical performance, though not directly covered by Theorem 13.*

The regret bound in Equation (4) grows logarithmically with T, where the leading constant is determined by a notion of gap with respect to η-restricted gains. In this respect, the bound bears some similarity with logarithmic regret bounds for MABs. This is consistent with the design principle behind the rarely-switching algorithms wherein the MDP was viewed as a multi-policy MAB. In contrast, the bound in Equation (4) is inversely proportional to the square of the gap terms, which stems from the technical difficulties arising in the regret analysis in the average-reward setting. Jaksch et al. (2010) report a logarithmic regret bound for UCRL2 that depends on a similar notion of gap term. However, their bound involves an additive term, which depends on mixing time quantities and has an implicit dependence on $\log(T)$. It thus could grow very large. In empirical evaluation of UCRL2, it is often witnessed that the logarithmic regime in the regret actually kicks in after very long burn-in phase. While Equation (4) offers a bound with an optimal dependence on T, it is not clear whether the gap in terms of policy gains — appearing in both Equation (4) and (Jaksch et al., 2010) — is the best one could get. Indeed, we recall that regret *lower* bounds for (non-ergodic) average-reward MDPs are open and deriving them even for the case of known dynamics is a very interesting, yet challenging, topic of future research.

6. Choice of policies

In this section, we discuss the construction of the set Π_τ. Hereafter, we consider that a set of policies to be small if its size does not exceed 10^6, somewhat arbitrarily.

First, there are cases in which Π is small. This situation may typically happen in real-world applications when a learner must choose between a limited set of policies prescribed by experts. A typical example is that of agriculture in which policies are intervention plans carefully built by agronomists, with a few parameters, despite considering a complex system.

Then, even when Π is large, there are cases when $\Pi^\star$ is known to belong to a small set of policies. For instance in (Puterman, 1994)[Theorem 8.11.3], the author detail the case of an inventory problem when an optimal policy can be searched in a restricted set of $\binom{A+S-1}{S}$ many *non-decreasing* policies instead of all possible A^S ones. For an MDP with $S = 150$ states and $A = 4$ actions, there are over 10^{90} deterministic policies but only $585276 \simeq 10^6$ non-decreasing policies. Likewise, in goal-state MDPs, one can restrict to policies aiming at reaching (and staying) in a single state as fast as possible (they can be computed knowing the transitions of the MDP), yielding only S many policies to consider.

Finally, generic structural properties of the MDP can be used, such as restricting to stationary and unichain policies since an optimal policy satisfies both conditions. Also, when the MDP is known to be unichain, it then satisfies Assumption 3 with $k = 1$, which suggests to simply choose $\Pi_\tau = \mathcal{V}_{\widehat{\pi}^\star_\tau}(1)$. More generally, one can set $\Pi_\tau = \mathcal{V}_{\widehat{\pi}^\star_\tau}(k)$ provided that $|\mathcal{V}_\pi(k)| = \binom{S}{k}A^k$ is small. When k is unknown, one may choose $\Pi_\tau = \mathcal{V}_{\widehat{\pi}^\star_\tau}(\tilde{k}) \cup \Gamma$ where $\tilde{k}$ satisfies $\binom{S}{\tilde{k}}A^{\tilde{k}} \leqslant 10^6$ and Γ is a small set of policies uniformly randomly chosen in $\Pi \setminus \mathcal{V}_{\widehat{\pi}^\star_\tau}(\tilde{k})$. This indeed ensures that Π_τ contains an improving policy over $\widehat{\pi}^\star_\tau$ with positive probability, which may be interesting for the practitioner. In Appendix G, we detail an alternative way of exploiting policies having more than one recurrent class.

7. Numerical experiments

In this section, we discuss the practical implementation of the presented `IMED-KD` algorithm, and present some numerical experiments[3]. We consider three environments: *RiverSwim* (Fig. 1), which is difficult to navigate; *nasty* (Fig. 3), where two high reward cycles are separated by a bottleneck action; and 4-*rooms* (Fig. 4), which is a sparse reward environment with close-to-deterministic transitions.

Practical comparison In those environments, we illustrate the performance of `IMED-KD` against the strategies `UCRL3` (Bourel et al., 2020), `PSRL` (Osband et al., 2013) and Q-learning (run with discount $\gamma = 0.99$ and optimistic initialization). `PSRL` and `UCRL3` use a confidence parameter to control the quality of the MDP approximation, which is set to 0.05 in the experiments. Further, we adapt both strategies to receive exact knowledge of the transition. The η parameter of `IMED-KD` plays a similar role, and we therefore use $\eta = 0.05/|\mathcal{S}|$ to ensure a fair comparison. `IMED-KD` uses value iteration as a routine, which is faster than the extended value iteration used in `UCRL3`. Q-learning takes an exploration parameter, ε, or exploration scheme when ε is slowly decreased with time. We report regret curves averaged over 2048 independent runs along with quantiles 0.1 and 0.9.

RiverSwim In each of the L states, there are two actions: `RIGHT` and `LEFT`. In Fig. 1, the `LEFT` action is represented with a dashed line and the `RIGHT` with solid line. Rewards are located at the extremities of the MDP, with a small reward in left initial state s_1 and large reward in the rightmost state s_L. Starting from state s_1, this MDP has proven challenging because of the large amount of non-rewarding exploration necessary to find the optimal policy. We consider the 6-state and 25-state instances, which allows us to compare how algorithms behave depending on the amount of necessary exploration; see Fig. 2. Q-learning is struggling despite its optimistic initialization, while `IMED-KD` is on par with `PSRL` on both experiments. The regret of `UCRL3` scales differently with L than the one of `IMED-KD` and `PSRL`, although it remains controlled.

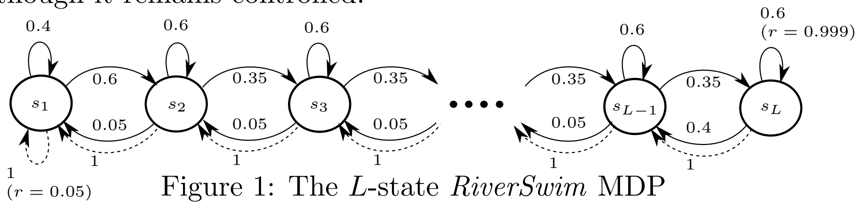

Figure 1: The L-state *RiverSwim* MDP

Nasty In this setting, there are two promising cycles separated by a small chain of one bottleneck state with no associated reward, which may induce an "oscillation" of a learner between the two cycles, paying the cost of the travel along the chain each time it changes cycle (policy). Q-learning exhibits a bad performance, suffering from a large, linear-shaped regret. `UCRL3` attains an even worse regret than Q-learning. In contrast, `IMED-KD` and `PSRL` are highly competitive and perform similarly.

n-rooms 4-rooms is a grid-like environment with 20 states and 4 cardinal actions where transitions are close to deterministic with a 0.8 chance of going in the intended direction. A

[3] Source code is available via https://github.com/fabienpesquerel/Logarithmic-regret-in-communicating-MDPs-Leveraging-known-dynamics-with-bandits.git.

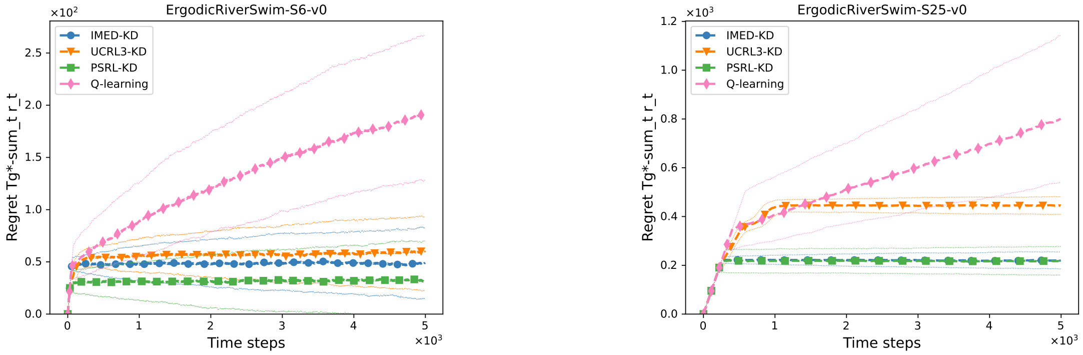

Figure 2: Regret on RiverSwim MDPs: 6-state (left) and 25-state (right)

reward of 0.99 is located in the goal state (highlighted in yellow), while it is zero elsewhere. Upon reaching the goal, the learner is positioned again in the initial red-state. As shown in Fig. 4, `IMED-KD` significantly outperforms the others in this environment —also in 2-rooms as shown in Appendix H. Even for horizons as large as 10^5, we cannot observe a bend in the Q-learning regret curve while it occurs around time step 6×10^4 for `UCRL3` (see Appendix H).

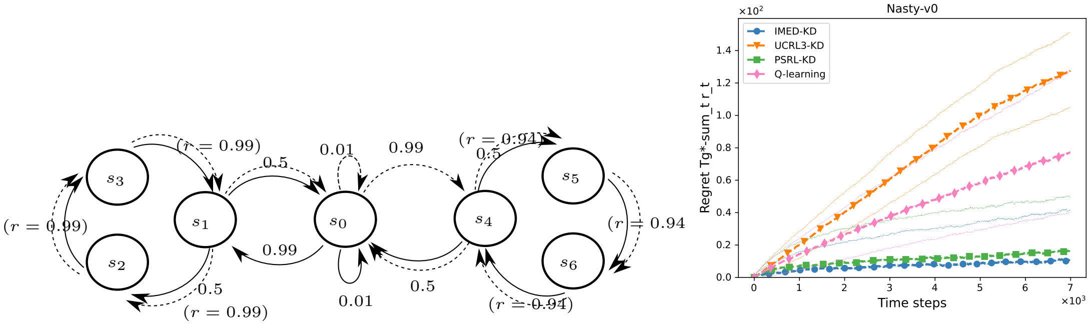

Figure 3: The *Nasty* environment (left) and regret curves (right)

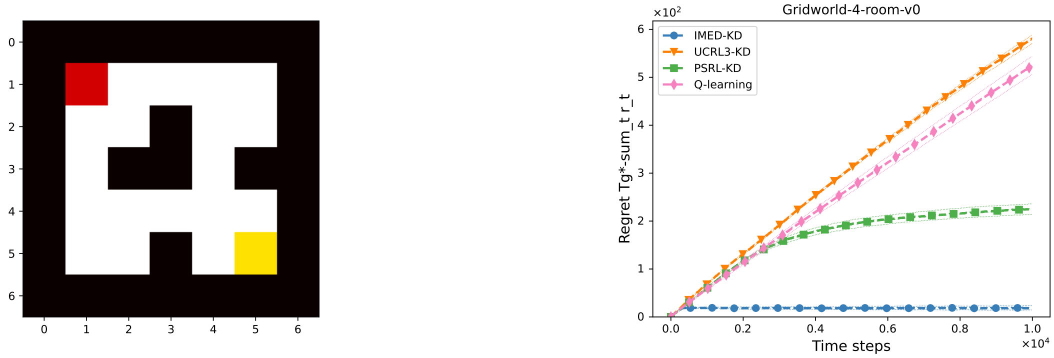

Figure 4: The *4-rooms* environment (left) and regret curves (right)

8. Conclusion

We studied regret minimization in communicating MDPs with known dynamics but unknown reward functions, and introduced a class of rarely-switching algorithms, whose design allows for leveraging the connectivity structure induced by the (known) transition function via considering the recurrent classes of the stationary policies. We presented `IMED-KD`, a rarely-switching algorithm that relies on an IMED-style index function. It admits an efficient implementation and significantly outperforms existing algorithms empirically. Under mild assumptions, we derived a finite-time, problem-dependent, and logarithmic regret bound for `IMED-KD`. Regret lower bounds for this setting (and communicating MDPs in general) are open, to our best knowledge, and deriving them is an interesting, yet challenging, direction for future work. Other interesting future directions include deriving adaptive rules to tune the parameter η (used to control the gains) and to relax the laziness assumption, even though some restrictive assumption seems required to ensure computational efficiency.

Acknowledgements The authors acknowledge the funding of the French National Research Agency, the French Ministry of Higher Education and Research, Inria, the MEL and the I-Site ULNE regarding project R-PILOTE 19-004-APPRENF. MST acknowledges the funding of the Independent Research Fund Denmark (DFF), under grant number 1026-00397B. The authors thank the anonymous reviewers for their careful reading of the paper and their suggestions for improvements.

References

S. Agrawal and R. Jia. Optimistic posterior sampling for reinforcement learning: Worst-case regret bounds. In *NIPS*, pages 1184–1194, 2017.

M. Azar, I. Osband, and R. Munos. Minimax regret bounds for reinforcement learning. In *ICML*, pages 263–272, 2017.

H. Bourel, O.-A. Maillard, and M. S. Talebi. Tightening exploration in upper confidence reinforcement learning. In *ICML*, pages 1056–1066, 2020.

A. N. Burnetas and M. N. Katehakis. Optimal adaptive policies for Markov decision processes. *Math. Oper. Res.*, 22(1):222–255, 1997.

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

W. Chen, Y. Wang, and Y. Yuan. Combinatorial multi-armed bandit: General framework and applications. In *ICML*, pages 151–159, 2013.

R. Combes, M. S. Talebi, A. Proutiere, and M. Lelareg. Combinatorial bandits revisited. In *NIPS*, pages 2116–2124, 2015.

R. Combes, S. Magureanu, and A. Proutiere. Minimal exploration in structured stochastic bandits. In *NIPS*, pages 1763–1771, 2017.

S. Filippi, O. Cappé, and A. Garivier. Optimism in reinforcement learning and Kullback-Leibler divergence. In *Allerton*, pages 115–122, 2010.

R. Fruit, M. Pirotta, and A. Lazaric. Near optimal exploration-exploitation in non-communicating Markov decision processes. In *NeurIPS*, pages 2998–3008, 2018.

A. Gopalan and S. Mannor. Thompson sampling for learning parameterized Markov decision processes. In *COLT*, pages 861–898, 2015.

W. K. Grassmann, M. I. Taksar, and D. P. Heyman. Regenerative analysis and steady state distributions for Markov chains. *Oper. Res.*, 33:1107–1116, 1985.

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

J. Honda and A. Takemura. Non-asymptotic analysis of a new bandit algorithm for semi-bounded rewards. *J. Mach. Learn. Res.*, 16:3721–3756, 2015.

T. Jaksch, R. Ortner, and P. Auer. Near-optimal regret bounds for reinforcement learning. *J. Mach. Learn. Res.*, 11:1563–1600, 2010.

O.-A. Maillard. Boundary crossing probabilities for general exponential families. *Math. Methods Stat.*, 27(1):1–31, 2018.

R. Ortner. Online regret bounds for Markov decision processes with deterministic transitions. In *ALT*, pages 123–137, 2009.

I. Osband, D. Russo, and B. Van Roy. (More) efficient reinforcement learning via posterior sampling. In *NIPS*, pages 3003–3011, 2013.

F. Pesquerel and O.-A. Maillard. IMED-RL: Regret optimal learning of ergodic Markov decision processes. In *NeurIPS*, pages 26363–26374, 2022.

M. L. Puterman. *Markov Decision Processes — Discrete Stochastic Dynamic Programming.* John Wiley & Sons, Inc., New York, NY, 1994.

H. Saber, P. Ménard, and O.-A. Maillard. Optimal strategies for graph-structured bandits. *arXiv preprint arXiv:2007.03224*, 2020.

M. Simchowitz and K. G. Jamieson. Non-asymptotic gap-dependent regret bounds for tabular MDPs. In *NeurIPS*, pages 1153–1162, 2019.

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

M. S. Talebi and O.-A. Maillard. Variance-aware regret bounds for undiscounted reinforcement learning in MDPs. In *ALT*, pages 770–805, 2018.

D. Tranos and A. Proutiere. Regret analysis in deterministic reinforcement learning. In *CDC*, pages 2246–2251, 2021.

C.-Y. Wei, M. Jafarnia Jahromi, H. Luo, H. Sharma, and R. Jain. Model-free reinforcement learning in infinite-horizon average-reward Markov decision processes. In *ICML*, pages 10170–10180, 2020.

Z. Zhang and X. Ji. Regret minimization for reinforcement learning by evaluating the optimal bias function. In *NeurIPS*, pages 2823–2832, 2019.