Proceedings of Machine Learning Research 101:204–219, 2019 ACML 2019

Model-Based Reinforcement Learning Exploiting State-Action Equivalence

Mahsa Asadi MAHSA.ASADI@INRIA.FR
Mohammad Sadegh Talebi SADEGH.TALEBI@INRIA.FR
Hippolyte Bourel HIPPOLYTE.BOUREL@ENS-RENNES.FR
Odalric-Ambrym Maillard ODALRIC.MAILLARD@INRIA.FR
Inria Lille – Nord Europe

Editors: Wee Sun Lee and Taiji Suzuki

Abstract

Leveraging an equivalence property in the state-space of a Markov Decision Process (MDP) has been investigated in several studies. This paper studies equivalence structure in the reinforcement learning (RL) setup, where transition distributions are no longer assumed to be known. We present a notion of similarity between transition probabilities of various state-action pairs of an MDP, which naturally defines an equivalence structure in the state-action space. We present equivalence-aware confidence sets for the case where the learner knows the underlying structure in advance. These sets are provably smaller than their corresponding equivalence-oblivious counterparts. In the more challenging case of an unknown equivalence structure, we present an algorithm called `ApproxEquivalence` that seeks to find an (approximate) equivalence structure, and define confidence sets using the approximate equivalence. To illustrate the efficacy of the presented confidence sets, we present `C-UCRL`, as a natural modification of `UCRL2` for RL in undiscounted MDPs. In the case of a known equivalence structure, we show that `C-UCRL` improves over `UCRL2` in terms of *regret* by a factor of $\sqrt{SA/C}$, in any communicating MDP with S states, A actions, and C classes, which corresponds to a massive improvement when $C \ll SA$. To the best of our knowledge, this is the first work providing regret bounds for RL when an equivalence structure in the MDP is efficiently exploited. In the case of an unknown equivalence structure, we show through numerical experiments that `C-UCRL` combined with `ApproxEquivalence` outperforms `UCRL2` in ergodic MDPs.

Keywords: Reinforcement Learning, Regret, Confidence Bound, Equivalence.

1. Introduction

This paper studies the Reinforcement Learning (RL) problem, where an agent interacts with an unknown environment in a single stream of observations, with the aim of maximizing the cumulative reward gathered over the course of experience. The environment is modeled as a Markov Decision Process (MDP), with finite state and action spaces, as considered in most literature; we refer to (Puterman, 2014; Sutton and Barto, 1998) for background materials on MDPs, and to Section 2. In order to act optimally or nearly so, the agent needs to learn the parameters of the MDP using the observations from the environment. The agent thus faces a fundamental trade-off between *exploitation vs. exploration*: Namely, whether to gather more experimental data about the consequences of the actions (exploration) or acting

consistently with past observations to maximize the rewards (exploitation); see (Sutton and Barto, 1998). Over the past two decades, a plethora of studies have addressed the above RL problem in the undiscounted setting, where the goal is to minimize the regret (e.g., Bartlett and Tewari (2009); Jaksch et al. (2010); Gheshlaghi Azar et al. (2017)), or in the discounted setting (as in, e.g., Strehl and Littman (2008)) with the goal of bounding the sample complexity of exploration as defined in (Kakade, 2003). In most practical situations, the state-space of the underlying MDP is too large, but often endowed with some *structure*. Directly applying the state-of-the-art RL algorithms, for instance from the above works, and ignoring the structure would lead to a prohibitive regret or sample complexity.

In this paper, we consider RL problems where the state-action space of the underlying MDP exhibits some *equivalence structure*. This is quite typical in many MDPs in various application domains. For instance, in a grid-world MDP when taking action 'up' from state s or 'right' from state s' when both are away from any wall may result in similar transitions (typically, move towards the target state with some probability, and stay still or transit to other neighbors with the remaining probability); see, e.g., Figure 1 in Section 2. We are interested in exploiting such a structure in order to speed up the learning process. Leveraging an equivalence structure is popular in the MDP literature; see (Ravindran and Barto, 2004; Li et al., 2006; Abel et al., 2016). However, most notions are unfortunately not well adapted to the RL setup, that is when the underlying MDP is *unknown*, as opposed to the known MDP setup. In particular, amongst those considering such structures, to our knowledge, none has provided performance guarantees in terms of regret or sample complexity. Our goal is to find a near-optimal policy, with controlled regret or sample complexity. To this end, we follow a model-based approach, which is popular in the RL literature, and aim at providing a generic model-based approach capable of exploiting this structure, to speed up learning. We do so by aggregating the information of state-action pairs in the same equivalence class when estimating the transition probabilities or reward function of the MDP.

Contributions. We make the following contributions. (i) We first introduce a notion of similarity between state-action pairs, which naturally yields a *partition* of the state-action space $\mathcal{S} \times \mathcal{A}$, and induces an equivalence structure in the MDP (see Definition 1–2). To our knowledge, while other notions of equivalence have been introduced, our proposed definition appears to be the first, in a discrete RL setup, explicitly using profile (ordering) of distributions. (ii) We present confidence sets that incorporate equivalence structure of transition probabilities and reward function into their definition, when the learner has access to such information. These confidence sets are smaller than those obtained by ignoring equivalence structures. (iii) In the case of an unknown equivalence structure, we present `ApproxEquivalence`, which uses confidence bounds of various state-action pairs as a proxy to estimate an empirical equivalence structure of the MDP. (iv) Finally, in order to demonstrate the application of the above equivalence-aware confidence sets, we present `C-UCRL`, which is a natural modification of the `UCRL2` algorithm (Jaksch et al., 2010) employing the presented confidence sets. As shown in Theorem 13, when the learner knows the equivalence structure, `C-UCRL` achieves a regret which is smaller than that of `UCRL2` by a factor of $\sqrt{SA/C}$, where C is the number of classes. This corresponds to a massive improvement when $C \ll SA$. We also verify, through numerical experiments, the superiority of `C-UCRL` over `UCRL2` in the case of an unknown equivalence structure.

Related Work. There is a rich literature on state-abstraction (or state-aggregation) in MDPs; we refer to (Li et al., 2006) on earlier methods, and to (Abel et al., 2016) for a good survey of recent approaches. (Ravindran and Barto, 2004) introduces aggregation based on homo-morphisms of the model, but with no algorithm nor regret analysis. (Dean et al., 1997; Givan et al., 2003) consider a partition of state-space of MDPs based on the notion of *stochastic bi-simulation*, which is a generalization of the notion of bi-simulation from the theory of concurrent processes to stochastic processes. This path is further followed in (Ferns et al., 2004, 2011), where *bi-simulation metrics* for capturing similarities are presented. Bi-simulation metrics can be thought of as quantitative analogues of the equivalence relations, and suggest to resort to optimal transport, which is intimately linked with our notions of similarity and equivalence (see Definition 1). However, these powerful metrics have only been studied in the context of a *known* MDP, and not the RL setup. The approach in (Anand et al., 2015) is similar to our work in that it considers state-action equivalence. Unlike the present paper, however, it does not consider orderings, transition estimation errors, or regret analysis. Another relevant work to our approach is (Ortner, 2013) on aggregation of states (but not of pairs, and with no ordering) based on concentration inequalities, a path that we follow. We also mention the works (Brunskill and Li, 2013; Mandel et al., 2016), where clustering of the state-space is studied. As other relevant works, we refer to (Leffler et al., 2007), where *relocatable action model* is introduced, and to (Diuk et al., 2009) that studies RL in the simpler setting of factored MDPs. We also mention interesting works revolving around complementary RL questions including the one on selection amongst different state representations in (Ortner et al., 2014) and on state-aliasing in (Hallak et al., 2013).

As part of this paper is devoted to presenting an equivalence structure aware variant of UCRL2, we provide here a brief review of the literature related to *undiscounted* RL. Undiscounted RL dates back at least to (Burnetas and Katehakis, 1997), and is thoroughly investigated later on in (Jaksch et al., 2010). The latter work presents UCRL2, which is inspired by multi-armed bandit algorithms. Several studies continued this line, including (Bartlett and Tewari, 2009; Maillard et al., 2014; Gheshlaghi Azar et al., 2017; Dann et al., 2017; Talebi and Maillard, 2018; Fruit et al., 2018), to name a few. Most of these works present UCRL2-style algorithms, and try to reduce the regret dependency on the number of states, as in, e.g., (Gheshlaghi Azar et al., 2017; Dann et al., 2017) (restricted to the episodic RL with a fixed and known horizon). Although the concept of equivalence is well-studied in MDPs, no work seems to have investigated the possibility of defining an aggregation that both is based on state-action pairs (instead of states only) for RL problems, and uses optimal transportation maps combined with statistical tests. Especially, the use of profile maps seems novel and we show it is also effective.

2. Model and Equivalence Classes

2.1. The RL Problem

In this section, we describe the RL problem, which we study in this paper. Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p, \nu)$ be an undiscounted MDP[1], where $\mathcal{S}$ denotes the discrete state-space with cardinality S, and $\mathcal{A}$ denotes the discrete action-space with cardinality A. Here, p represents the

1. Our results can be extended to the discounted case as well.

transition kernel such that $p(s'|s,a)$ denotes the probability of transiting to state s', starting from state s and executing action a. Finally, ν is a reward distribution function on $[0,1]$, whose mean is denoted by μ.

The game proceeds as follows. The learner starts in some state $s_1 \in \mathcal{S}$ at time $t = 1$. At each time step $t \in \mathbb{N}$, the learner chooses one action $a \in \mathcal{A}$ in its current state s_t based on its past decisions and observations. When executing action a_t in state s_t, the learner receives a random reward $r_t := r_t(s_t, a_t)$ drawn independently from distribution $\nu(s_t, a_t)$, and whose mean is $\mu(s_t, a_t)$. The state then transits to a next state $s_{t+1} \sim p(\cdot|s_t, a_t)$, and a new decision step begins. We refer to (Sutton and Barto, 1998; Puterman, 2014) for background material on MDPs and RL. The goal of the learner is to maximize the *cumulative reward* gathered in the course of interaction with the environment. As p and ν are unknown, the learner has to learn them by trying different actions and recording the realized rewards and state transitions. The performance of the learner can be assessed through the notion of *regret*[2] with respect to an optimal oracle, being aware of p and ν. More formally, as in (Jaksch et al., 2010), under a learning algorithm $\mathbb{A}$, we define the T-step regret as

$$\mathfrak{R}(\mathbb{A}, T) := Tg_\star - \sum_{t=1}^{T} r_t(s_t, a_t)\,,$$

where $g_\star$ denotes the *average reward* (or *gain*[3]) attained by an optimal policy, and where a_t is chosen by $\mathbb{A}$ as a function of $((s_{t'}, a_{t'})_{t'<t}, s_t)$. Alternatively, the objective of the learner is to minimize the regret, which calls for balancing between exploration and exploitation. In the present work, we are interested in exploiting *equivalence structure* in the state-action space in order to speed up exploration, which, in turn, reduces the regret.

2.2. Similarity and Equivalence Classes

We now present a precise definition of the equivalence structure considered in this paper. We first introduce a notion of similarity between state-action pairs of the MDP:

Definition 1 (Similar state-action pairs) *The pair (s', a') is said to be ε-similar to the pair (s, a), for $\varepsilon = (\varepsilon_p, \varepsilon_\mu) \in \mathbb{R}^2_+$, if*

$$\|p(\sigma_{s,a}(\cdot)|s,a) - p(\sigma_{s',a'}(\cdot)|s',a')\|_1 \leqslant \varepsilon_p \quad \text{and} \quad |\mu(s,a) - \mu(s',a')| \leqslant \varepsilon_\mu\,,$$

where $\sigma_{s,a} : \{1,\dots,S\} \to \mathcal{S}$ indexes a permutation of states such that $p(\sigma_{s,a}(1)|s,a) \geqslant p(\sigma_{s,a}(2)|s,a) \geqslant \dots \geqslant p(\sigma_{s,a}(S)|s,a)$. We refer to $\sigma_{s,a}$ as a **profile mapping** *(or for short,* **profile***) for (s,a), and denote by $\boldsymbol{\sigma} = (\sigma_{s,a})_{s,a}$ the set of profile mappings of all pairs.*

The notion of similarity introduced above naturally yields a *partition* of the state-action space $\mathcal{S} \times \mathcal{A}$, as detailed in the following definition:

Definition 2 (Equivalence classes) *$(0,0)$-similarity is an equivalence relation and induces a canonical partition of $\mathcal{S} \times \mathcal{A}$. We refer to such a canonical partition as* **equivalence classes** *or* **equivalence structure***, denote it by $\mathcal{C}$, and let $C := |\mathcal{C}|$.*

2. We note that in the discounted setting, the quality of a learning algorithm is usually assessed through the notion of *sample complexity* as defined in (Kakade, 2003).
3. See, e.g., (Puterman, 2014) for background material on MDPs.

In order to help understand Definitions 1 and 2, we present in Figure 1 an MDP with 13 states, where the state-action pairs (6,Up) and (8,Right) are equivalent up to a permutation: Let the permutation σ be such that $\sigma(2) = 9$, $\sigma(6) = 8$, and $\sigma(i) = i$ for all $i \neq 2, 6$. Now $p(\sigma(x)|6, \mathsf{Up}) = p(x|8, \mathsf{Right})$ for all $x \in \mathcal{S}$, and thus, the pairs (8,Right) and (6,Up) belong to the same class.

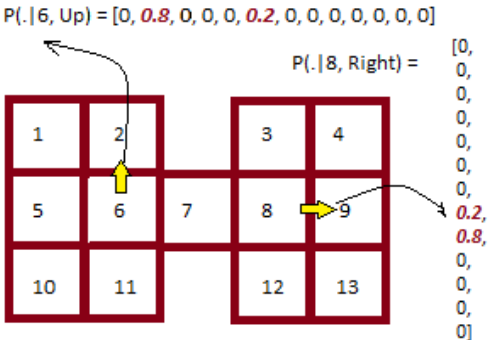


Figure 1: A grid-world MDP showing similar transitions from state-action pairs (6,Up) and (8,Right).

Remark 3 *Crucially, the equivalence relation is not only stated about states, but about* state-action pairs. *For instance, pairs (6,Up) and (8,Right) in this example are in the same class although corresponding to playing different actions in different states.*

Remark 4 *The profile mapping $\sigma_{s,a}$ in Definition 1 may not be unique in general, especially if distributions have sparse supprts. For ease of presentation, in the sequel we assume that the restriction of $\sigma_{s,a}$ to the support of $p(\cdot|s, a)$ is uniquely defined. We also remark that Definition 1 can be easily generalized by replacing the $\|\cdot\|_1$ norm with other contrasts, such as the KL divergence, squared distance, etc.*

In many environments considered in RL with large state and action spaces, the number C of equivalent classes of state-action pairs using Definitions 1–2 stays small even for large SA, thanks to the profile mappings. This is the case in typical grid-world MDPs as well as in *RiverSwim* shown in Figure 2. For example, in *Ergodic RiverSwim* with L states, we have $C = 6$. We also refer to Appendix F for additional illustrations of grid-world MDPs. This remarkable feature suggests that leveraging this structure may yield significant speed-up in terms of learning guarantees if well-exploited.

3. Equivalence-Aware Confidence Sets

We are now ready to present an approach that defines confidence sets for p and μ taking into account the equivalence structure in the MDP. The use of confidence bounds in a model-based approach is related to strategies implementing the *optimism in the face of uncertainty* principle, as in stochastic bandit problems (Lai and Robbins, 1985; Auer et al., 2002). Such an approach relies on maintaining a set of plausible MDPs (models) that are consistent with the observations gathered, and where the set contains the true MDP with high probability. Exploiting equivalence structure of the MDP, one could obtain a more precise estimation of mean reward μ and transition kernel p of the MDP by *aggregating* observations from various state-action pairs in the same class. This, in turn, yields *smaller* (hence, better) sets of models.

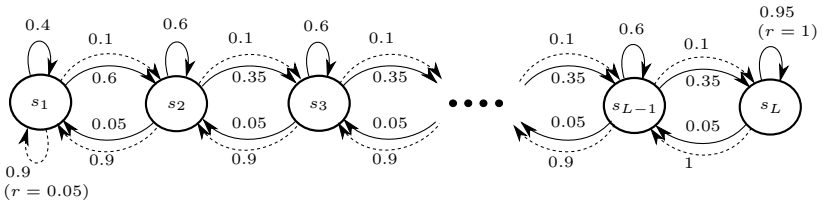


Figure 2: The L-state *Ergodic RiverSwim* MDP

Notations. We introduce some necessary notations. Under a given algorithm, for a pair (s,a), we denote by $N_t(s,a)$ the total number of observations of (s,a) up to time t. Let us define $\widehat{\mu}_t(s,a)$ as the empirical mean reward built using $N_t(s,a)$ i.i.d. samples from $\nu(s,a)$, and $\widehat{p}_t(\cdot|s,a)$ as the empirical distribution built using $N_t(s,a)$ i.i.d. observations from $p(\cdot|s,a)$. For a set $c \subseteq \mathcal{S}\times\mathcal{A}$, we denote by $n_t(c)$ the total number of observations of pairs in c up to time t, that is $n_t(c) := \sum_{(s,a)\in c} N_t(s,a)$. For $c \subseteq \mathcal{S}\times\mathcal{A}$, we further denote by $\widehat{\mu}_t(c)$ and $\widehat{p}_t(\cdot|c)$ the empirical mean reward and transition probability built using $n_t(c)$ samples, respectively; we provide precise definitions of $\widehat{\mu}_t(c)$ and $\widehat{p}_t(\cdot|c)$ later on in this section.

For a given confidence parameter δ and time t, we write $\mathcal{M}_{t,\delta}$ to denote the set of plausible MDPs at time t, which may be generically expressed as

$$\mathcal{M}_{t,\delta} = \{(\mathcal{S},\mathcal{A},p',\nu') : p' \in \mathsf{CB}_{t,\delta} \text{ and } \mu' \in \mathsf{CB}'_{t,\delta}\}\,, \tag{1}$$

where $\mathsf{CB}_{t,\delta}$ (resp. $\mathsf{CB}'_{t,\delta}$) denotes the confidence set for p (resp. μ) centered at $\widehat{p}$ (resp. $\widehat{\mu}$), and where μ' is the mean of ν'. Note that both $\mathsf{CB}_{t,\delta}$ and $\mathsf{CB}'_{t,\delta}$ depend on $N_t(s,a)$, $(s,a)\in\mathcal{S}\times\mathcal{A}$. For ease of presentation, in the sequel we consider the following confidence sets[4] used in several model-based RL algorithms, e.g., (Jaksch et al., 2010; Dann et al., 2017):

$$\begin{aligned}
\mathsf{CB}_{t,\delta} &:= \{p' : \|\widehat{p}_t(\cdot|s,a) - p'(\cdot|s,a)\|_1 \leqslant \beta_{N_t(s,a)}(\tfrac{\delta}{SA}),\ \forall s,a\}\,, \\
\mathsf{CB}'_{t,\delta} &:= \{\mu' : |\widehat{\mu}_t(s,a) - \mu'(\cdot|s,a)| \leqslant \beta'_{N_t(s,a)}(\tfrac{\delta}{SA}),\ \forall s,a\}\,, \qquad \text{where}
\end{aligned} \tag{2}$$

$$\beta_n(\delta) = \sqrt{\frac{2(1+\frac{1}{n})\log\left(\sqrt{n+1}\,\frac{2^S-2}{\delta}\right)}{n}} \quad \text{and} \quad \beta'_n(\delta) = \sqrt{\frac{(1+\frac{1}{n})\log(\sqrt{n+1}/\delta)}{2n}}\,,\ \forall n. \tag{3}$$

These confidence sets were derived by combining Hoeffding's (Hoeffding, 1963) and Weissman's (Weissman et al., 2003) concentration inequalities with the Laplace method (Peña et al., 2008; Abbasi-Yadkori et al., 2011), which enables to handle the random stopping times $N_t(s,a)$ in a sharp way; we refer to (Maillard, 2019) for further discussion. In particular, this ensures that the true transition function p and mean reward function μ are contained in the confidence sets with probability at least $1-2\delta$, uniformly over all time t.

Remark 5 *As the bounds for μ and p are similar, to simplify the presentation, from now on we assume the mean reward function μ is known*[5]*.*

We now provide modifications to $\mathsf{CB}_{t,\delta}$ in order to exploit the equivalence structure $\mathcal{C}$, when the learner knows $\mathcal{C}$ in advance. The case of an unknown $\mathcal{C}$ is addressed in Section 4.

4. The approach presented in this section can be extended to other concentration inequalities, as well.
5. This is a common assumption in the RL literature; see, e.g., (Bartlett and Tewari, 2009).

3.1. Case 1: Known Classes and Profiles

Assume that an oracle provides the learner with a perfect knowledge of the equivalence classes $\mathcal{C}$ as well as profiles $\boldsymbol{\sigma} = (\sigma_{s,a})_{s,a}$. In this ideal situation, the knowledge of $\mathcal{C}$ and $\boldsymbol{\sigma}$ allows to straightforwardly aggregate observations from all state-action pairs in the same class to build more accurate estimates of p and μ. Formally, for a class $c \subset \mathcal{C}$, we define

$$\widehat{p}_t^{\boldsymbol{\sigma}}(x|c) = \frac{1}{n_t(c)} \sum_{(s,a)\in c} N_t(s,a)\widehat{p}_t(\sigma_{s,a}(x)|s,a)\,, \qquad \forall x \in \mathcal{S}, \tag{4}$$

where we recall that $n_t(c) = \sum_{(s,a)\in c} N_t(s,a)$. The superscript $\boldsymbol{\sigma}$ in (4) signifies that the aggregate empirical distribution $\widehat{p}_t^{\boldsymbol{\sigma}}$ depends on $\boldsymbol{\sigma}$. Having defined $\widehat{p}_t^{\boldsymbol{\sigma}}$, we modify the confidence set (2) by modifying the L_1 bound there as follows:

$$\|\widehat{p}_t^{\boldsymbol{\sigma}}(\boldsymbol{\sigma}^{-1}(\cdot)|c) - p'(\cdot|c)\|_1 \leqslant \beta_{n_t(c)}(\tfrac{\delta}{C})\,, \quad \forall c \in \mathcal{C}, \tag{5}$$

and further define: $\mathsf{CB}_{t,\delta}(\mathcal{C},\boldsymbol{\sigma}) := \{p' \colon (5) \text{ holds}\}$, where $(\mathcal{C},\boldsymbol{\sigma})$ stresses that $\mathcal{C}$ and $\boldsymbol{\sigma}$ are provided as input. Then, for all time t and class $c \in \mathcal{C}$, by construction, the true transition p belongs to $\mathsf{CB}_{t,\delta}(\mathcal{C},\boldsymbol{\sigma})$, with probability greater than $1-\delta$.

Remark 6 *It is crucial to remark that the above confidence set does not use elements of $\mathcal{C}$ as "meta states" (i.e., replacing the states with classes), as considered for instance in the literature on state-aggregation. Rather, the classes are only used to* group *observations from different sources and build more refined estimates for each pair: The plausible MDPs are built using the same state-space $\mathcal{S}$ and action-space $\mathcal{A}$, unlike in, e.g., (Ortner, 2013).*

3.2. Case 2: Known Classes, Unknown Profiles

Now we consider a more realistic setting when the oracle provides $\mathcal{C}$ to the learner, but $\boldsymbol{\sigma}$ is unknown. In this more challenging situation, we need to *estimate* profiles as well. Given time t, we find an *empirical profile mapping* (or for short, empirical profile) $\sigma_{s,a,t}$ satisfying

$$\widehat{p}_t(\sigma_{s,a,t}(1)|s,a) \geqslant \widehat{p}_t(\sigma_{s,a,t}(2)|s,a) \geqslant \ldots \geqslant \widehat{p}_t(\sigma_{s,a,t}(S)|s,a)\,,$$

and define $\boldsymbol{\sigma}_t = (\sigma_{s,a,t})_{s,a}$. We then build the modified empirical estimate in a similar fashion to (4): For any $c \in \mathcal{C}$,

$$\widehat{p}_t^{\boldsymbol{\sigma}_t}(x|c) = \frac{1}{n_t(c)} \sum_{(s,a)\in c} N_t(s,a)\widehat{p}_t(\sigma_{s,a,t}(x)|s,a)\,, \quad \forall x \in \mathcal{S}.$$

Then, we may modify the L_1 inequality in (2) as follows:

$$\|\widehat{p}_t^{\boldsymbol{\sigma}_t}(\boldsymbol{\sigma}_t^{-1}(\cdot)|c) - p'(\cdot|c)\|_1 \leqslant \frac{1}{n_t(c)} \sum_{(s,a)\in c} N_t(s,a)\beta_{N_t(s,a)}(\tfrac{\delta}{C})\,, \quad \forall c \in \mathcal{C}, \tag{6}$$

which further yields the following modified confidence set that uses only $\mathcal{C}$ as input: $\mathsf{CB}_{t,\delta}(\mathcal{C}) := \{p' : (6) \text{ holds}\}$. The above construction is justified by the following non-expansive property of the ordering operator, as it ensures that Weissman's concentration inequality also applies to the ordered empirical distribution:

Lemma 7 (Non-expansive ordering) *Let p and q be two discrete distributions, defined on the same alphabet $\mathcal{S}$, with respective profile mappings σ_p and σ_q. Then,*

$$\|p(\sigma_p(\cdot)) - q(\sigma_q(\cdot))\|_1 \leqslant \|p - q\|_1 \,.$$

The proof of Lemma 7 is provided in Appendix B. An immediate corollary follows.

Corollary 8 *The confidence set $\mathsf{CB}_{t,\delta}(\mathcal{C})$ contains the true transition function p with probability at least $1-\delta$, uniformly over all time t.*

4. Unknown Classes: The ApproxEquivalence Algorithm

In this section, we turn to the most challenging situation when both $\mathcal{C}$ and $\boldsymbol{\sigma}$ are unknown to the learner. To this end, we first introduce an algorithm, which we call ApproxEquivalence, that finds an approximate equivalence structure in the MDP by grouping transition probabilities based on statistical tests. ApproxEquivalence is inspired by (Khaleghi et al., 2016) that provides a method for clustering time series. Interestingly enough, ApproxEquivalence does not require the knowledge of the number of classes in advance.

We first introduce some definitions. Given $u, v \subseteq \mathcal{S} \times \mathcal{A}$, we define the distance between u and v as $d(u,v) := \|p^{\boldsymbol{\sigma}_u}(\cdot|u) - p^{\boldsymbol{\sigma}_v}(\cdot|v)\|_1$. ApproxEquivalence relies on finding subsets of $\mathcal{S} \times \mathcal{A}$ that are *statistically close* in terms of the distance function $d(\cdot,\cdot)$. As $d(\cdot,\cdot)$ is unknown, ApproxEquivalence relies on a lower confidence bound on it: For $u, v \subseteq \mathcal{S} \times \mathcal{A}$, we define the *lower-confidence distance function* between u and v as

$$\widehat{d}(u,v) := \widehat{d}_{t,\delta}(u,v) := \left\|\widehat{p}_t^{\boldsymbol{\sigma}_{u,t}}(\cdot|u) - \widehat{p}_t^{\boldsymbol{\sigma}_{v,t}}(\cdot|v)\right\|_1 - \varepsilon_{u,t} - \varepsilon_{v,t} \,,$$

where for $u \in \mathcal{S} \times \mathcal{A}$ and $t \in \mathbb{N}$, we define $\varepsilon_{u,t} := \frac{1}{n_t(u)} \sum_{\ell \in u} N_t(\ell) \beta_{N_t(\ell)}\big(\frac{\delta}{SA}\big)$. We stress that, unlike $d(\cdot,\cdot)$, $\widehat{d}(\cdot,\cdot)$ is not a distance function.

Definition 9 (PAC Neighbor) *For a given equivalence structure $\mathcal{C}$, and given $c \in \mathcal{C}$, we say that $c' \in \mathcal{C}$ is a* PAC Neighbor *of c if it satisfies: (i) $\widehat{d}(c,c') \leqslant 0$; (ii) $\widehat{d}(\{j\},\{j'\}) \leqslant 0$, for all $j \in c$ and $j' \in c'$; and (iii) $\widehat{d}(\{j\}, c \cup c') \leqslant 0$, for all $j \in c \cup c'$. We further define $\mathcal{N}(c) := \{c' \in \mathcal{C} \setminus \{c\} : \text{(i)–(iii) hold}\}$ as the set of all PAC Neighbors of c.*

Definition 10 (PAC Nearest Neighbor) *For a given equivalence structure $\mathcal{C}$ and $c \in \mathcal{C}$, we define the* PAC Nearest Neighbor *of c (when it exists) as:*

$$\mathsf{Near}(c, \mathcal{C}) \in \operatorname*{argmin}_{u \in \mathcal{N}(c)} \widehat{d}(c,u) \,.$$

ApproxEquivalence proceeds as follows. At time t, it receives as input a parameter $\alpha > 1$ that controls the level of aggregation, as well as $N_t(s,a)$ for all pairs (s,a). Starting from the trivial partition of $\{1,\dots,SA\}$ into $\mathcal{C}^0 := \{\{1\},\dots,\{SA\}\}$, the algorithm builds a coarser partition by iteratively merging elements of $\mathcal{C}^0$ that are *statistically close*. More precisely, the algorithm sorts elements of $\mathcal{C}^0$ in a non-increasing order of $n_t(c), c \in \mathcal{C}^0$ so as to promote pairs with the tightest confidence intervals. Then, starting from c with the largest $n_t(c)$, it finds the PAC Nearest Neighbor c' of c, that is $c' = \mathsf{Near}(c, \mathcal{C}^0)$. If $\frac{1}{\alpha} \leqslant \frac{n_t(c)/L(c)}{n_t(c')/L(c')} \leqslant \alpha$, where $L(c) = |c|$, the algorithm merges c and c', thus leading to a novel

partition $\mathcal{C}^1$, which contains the new cluster $c \cup c'$, and removes c and c'. The algorithm continues this procedure with the next set in $\mathcal{C}^0$, until exhaustion, thus finishing the creation of the novel partition $\mathcal{C}^1$ of $\{1, \dots, SA\}$. `ApproxEquivalence` continues this process, by ordering the elements of $\mathcal{C}^1$ in a non-increasing order, and carrying out similar steps as before, yielding the new partition $\mathcal{C}^2$. `ApproxEquivalence` continues the same procedure until iteration k when $\mathcal{C}^{k+1} = \mathcal{C}^k$ (convergence). The pseudo-code of `ApproxEquivalence` is shown in Algorithm 1.

Algorithm 1 `ApproxEquivalence`

Input: N_t, α
  Initialization: $\mathcal{C}^0 \leftarrow \{\{1\}, \{2\}, \dots, \{SA\}\}$; $n \leftarrow N_t$; $L \leftarrow \mathbf{1}_{SA}$
  changed $\leftarrow$ True; $k \leftarrow 1$;
  while changed **do**
    $\mathcal{C}^{k+1} \leftarrow \mathcal{C}^k$;
    changed $\leftarrow$ False;
    Index $\leftarrow$ argsort(n);
    for all $i \in$ Index **do**
      if $n(i) = 0$ **then**
        Break;
      end if
      if Near$(i, \mathcal{C}^{k-1}) \neq \emptyset$ **then**
        $j \leftarrow$ Near$(i, \mathcal{C}^{k-1})$;
        if $\frac{1}{\alpha} \leqslant \frac{n(i)/L(i)}{n(j)/L(j)} \leqslant \alpha$ **then**
          $\widehat{p}_t^{\boldsymbol{\sigma}^{j,t}}(\cdot|j) \leftarrow \frac{1}{n(j)+n(i)} \Big(n(j) \widehat{p}_t^{\boldsymbol{\sigma}^{j,t}}(\cdot|j) + n(i) \widehat{p}_t^{\boldsymbol{\sigma}^{i,t}}(\cdot|i) \Big)$
          $L(i) \leftarrow L(j) + L(i)$; $n(i) \leftarrow n(j) + n(i)$;
          $n(j) \leftarrow 0$, $L(j) \leftarrow 0$;
          $\mathcal{C}^{k+1} \leftarrow \mathcal{C}^{k+1} \setminus (\{\{i\}, \{j\}) \cup \{i, j\}$;
          changed $\leftarrow$ True;
        end if
      end if
    end for
    $k \leftarrow k + 1$;
  end while
output $\mathcal{C}^k$

The purpose of condition $\frac{1}{\alpha} \leqslant \frac{n_t(c)/L(c)}{n_t(c')/L(c')} \leqslant \alpha$ is to ensure the stability of the algorithm. It prevents merging pairs whose numbers of samples differ a lot. We note that a very similar condition (with $\alpha = 2$) is considered in (Ortner, 2013) for state-aggregation. Nonetheless, we believe such a condition could be relaxed.

Remark 11 *Since at each iteration, either two or more subsets are merged,* `ApproxEquivalence` *converges after, at most,* $SA - 1$ *steps.*

We provide a theoretical guarantee for the correctness of `ApproxEquivalence` for the case when α tends to infinity. The result relies on the following separability assumption:

Assumption 1 (Separability) *There exists some* $\Delta > 0$ *such that*

$$\forall c \neq c' \in \mathcal{C}, \forall \ell \in c, \forall \ell' \in c', \quad d(\{\ell\}, \{\ell'\}) \geqslant \Delta \,.$$

Proposition 12 *Under Assumption 1, provided that* $\min_{s,a} N_t(s,a) > f^{-1}(\Delta)$, *where* $f : n \mapsto 4\beta_n(\frac{\delta}{SA})$, `ApproxEquivalence` *with the choice* $\alpha \to \infty$ *outputs the correct equivalence structure* $\mathcal{C}$ *of state-action pairs with probability at least* $1 - \delta$.

The proof of Proposition 12 is provided in Appendix C. We note that Assumption 1 bears some similarity to the separability assumption used in (Brunskill and Li, 2013). Note further although the proposition relies on Assumption 1, we believe one may be able to derive a similar result under a weaker assumption as well. We leave this for future work.

Now we turn to defining the aggregated confidence sets. Given t, let $\mathcal{C}_t$ denote the equivalence structure output by the algorithm. We may use the following confidence set:

$$\mathsf{CB}_{t,\delta}(\mathcal{C}_t) := \Big\{p' : \|\widehat{p}_t^{\boldsymbol{\sigma}_t}(\boldsymbol{\sigma}_t^{-1}(\cdot)|c) - p'(\cdot|c)\|_1 \leqslant \sum_{(s,a)\in c} \tfrac{N_t(s,a)}{n_t(c)} \beta_{N_t(s,a)}\big(\tfrac{\delta}{SA}\big)\,,\ \forall c \in \mathcal{C}_t\Big\}\,. \quad (7)$$

5. Application: The C-UCRL Algorithm

This section is devoted to presenting some applications of equivalence-aware confidence sets introduced in Section 3. We present C-UCRL, a natural modification of UCRL2 (Jaksch et al., 2010), which is capable of exploiting the equivalence structure of the MDP. We consider variants of C-UCRL depending on which information is available to the learner in advance.

First, we briefly recall UCRL2. At a high level, UCRL2 maintains the set $\mathcal{M}_{t,\delta}$ of MDPs at time t,[6] which is defined in (1). It then implements the optimistic principle by trying to compute $\overline{\pi}_t^+ = \operatorname{argmax}_{\pi:\mathcal{S}\to\mathcal{A}} \max_{M\in\mathcal{M}_{t,\delta}} g_\pi^M$, where g_π^M denotes the gain of policy π in MDP M. This is carried out approximately by the `Extended Value Iteration (EVI)` algorithm that builds a near-optimal policy π_t^+ and MDP $\widetilde{M}_t$ such that $g_{\pi_t^+}^{\widetilde{M}_t} \geqslant \max_{\pi,M\in\mathcal{M}_{t,\delta}} g_\pi^M - \frac{1}{\sqrt{t}}$. Finally, UCRL2 does not recompute π_t^+ at each time step. Instead, it proceeds in internal episodes (indexed by $k \in \mathbb{N}$), and computes π_t^+ only at the starting time t_k of each episode, defined as $t_1 = 1$ and for all $k > 1$, $t_k = \min\Big\{t > t_{k-1} : \exists s,a, \nu_{t_{k-1}:t}(s,a) \geqslant N_{t_{k-1}}(s,a)^+\Big\}$, where $\nu_{t_1:t_2}(s,a)$ denotes the number of observations of pair (s,a) between time $t_1{+}1$ and t_2, and where for $z \in \mathbb{N}$, $z^+ := \max\{z,1\}$. We provide the pseudo-code of UCRL2 in Appendix A.

5.1. C-UCRL: Known Equivalence Structure

Here we assume that the learner knows $\mathcal{C}$ and $\boldsymbol{\sigma}$ in advance, and provide a variant of UCRL2, referred to as C-UCRL$(\mathcal{C},\boldsymbol{\sigma})$, capable of exploiting the knowledge on $\mathcal{C}$ and $\boldsymbol{\sigma}$. Given δ, at time t, C-UCRL$(\mathcal{C},\boldsymbol{\sigma})$ uses the following set of models

$$\mathcal{M}_{t,\delta}(\mathcal{C},\boldsymbol{\sigma}) = \Big\{(\mathcal{S},\mathcal{A},p',\nu) : p' \in \mathsf{Pw}(\mathcal{C}) \ \text{ and } \ p'_{\mathcal{C}} \in \mathsf{CB}_{t,\delta}(\mathcal{C},\boldsymbol{\sigma})\Big\}\,,$$

where $\mathsf{Pw}(\mathcal{C})$ denotes the state-transition functions that are piece-wise constant on $\mathcal{C}$, and where $p'_{\mathcal{C}}$ denotes the function induced by $p' \in \mathsf{Pw}(\mathcal{C})$ over $\mathcal{C}$ (that is $p'(\cdot|s,a) = p'_{\mathcal{C}}(\cdot|c)$ for all $(s,a)\in c$). Moreover, C-UCRL$(\mathcal{C},\boldsymbol{\sigma})$ defines

$$t_{k+1} = \min\Big\{t > t_k : \exists c \in \mathcal{C} : \sum_{(s,a)\in c} \nu_{t_k:t}(s,a) \geqslant n_{t_k}(c)^+\Big\}\,.$$

We note that forcing the condition $p' \in \mathsf{Pw}(\mathcal{C})$ may be computationally difficult. To ensure efficient implementation, we use the same `EVI` algorithm of UCRL2, where for $(s,a) \in c$,

6. This set is described by the Weismann confidence bounds combined with the Laplace method. The original UCRL2 algorithm in (Jaksch et al., 2010) uses looser confidence bounds relying on union bounds instead of the Laplace method.

we replace $\widehat{p}_t(\cdot|s,a)$ and $\beta_{N_t(s,a)}(\frac{\delta}{SA})$ respectively with $\widehat{p}^{\boldsymbol{\sigma}}_t(\cdot|c)$ and $\beta_{n_t(c)}(\frac{\delta}{C})$. The precise modified steps of C-UCRL$(\mathcal{C},\boldsymbol{\sigma})$ are presented in Appendix A for completeness. An easy modification of the analysis of (Jaksch et al., 2010) yields:

Theorem 13 (Regret of C-UCRL$(\mathcal{C},\sigma)$) *With probability higher than $1-3\delta$, uniformly over all time horizon T,*

$$\mathfrak{R}(\textit{C-UCRL}(\mathcal{C},\sigma),T) \leqslant 18\sqrt{CT\big(S+\log(2C\sqrt{T+1}/\delta)\big)} + DC\log_2(\tfrac{8T}{C})\,.$$

The proof of Theorem 13 is provided in Appendix D. This theorem shows that efficiently exploiting the knowledge of $\mathcal{C}$ and $\boldsymbol{\sigma}$ yields an improvement over the regret bound of UCRL2 by a factor of $\sqrt{SA/C}$, which could be a huge improvement when $C \ll SA$. This is the case in, for instance, many grid-world environments thanks to Definitions 1-2; see Appendix F.

5.2. C-UCRL: Unknown Equivalence Structure

Now we consider the case where $\mathcal{C}$ is unknown to the learner. In order to accommodate this situation, we use ApproxEquivalence in order to estimate the equivalence structure.

We introduce C-UCRL, which proceeds similarly to C-UCRL$(\mathcal{C},\boldsymbol{\sigma})$. At each time t, ApproxEquivalence outputs $\mathcal{C}_t$ as an estimate of the true equivalence structure $\mathcal{C}$. Then, C-UCRL uses the following set of models taking $\mathcal{C}_t$ as input:

$$\mathcal{M}_{t,\delta}(\mathcal{C}_t) = \Big\{(\mathcal{S},\mathcal{A},p',\nu) : p' \in \mathsf{Pw}(\mathcal{C}_t) \;\text{ and }\; p'_{\mathcal{C}_t} \in \mathsf{CB}_{t,\delta}(\mathcal{C}_t)\Big\}\,.$$

Further, it sets the starting step of episode $k+1$ as:

$$t_{k+1} = \min\Big\{t > t_k : \exists c \in \mathcal{C}_{t_k},\, \textstyle\sum_{(s,a)\in c} \nu_{t_k:t}(s,a) \geqslant n_{t_k}(c)^+ \text{ or } \exists s,a,\, \nu_{t_k:t}(s,a) \geqslant N_{t_k}(s,a)^+\Big\}\,.$$

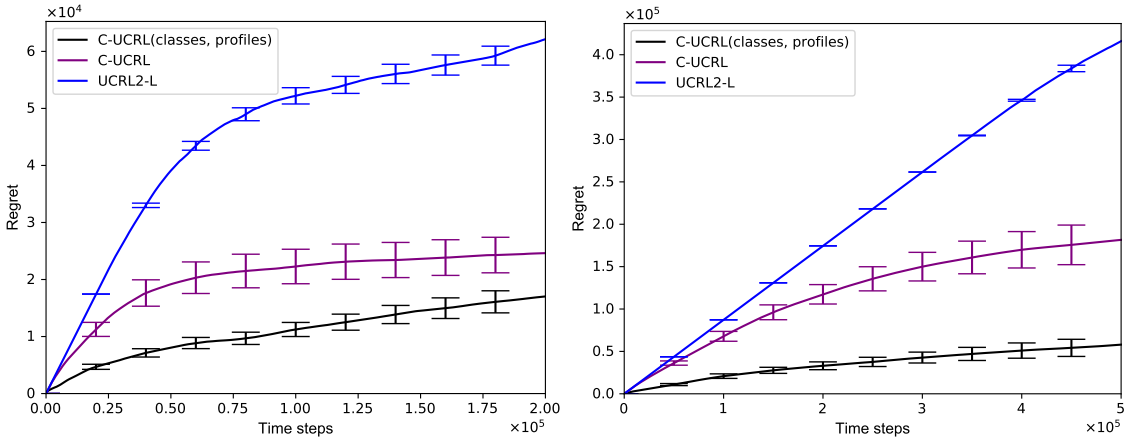
Similarly to C-UCRL$(\mathcal{C},\boldsymbol{\sigma})$, we use a modification of EVI to implement C-UCRL.

Remark 14 *Note that $\mathcal{M}_{t,\delta}(\mathcal{C}) \neq \mathcal{M}_{t,\delta}(\mathcal{C}_t)$ as we may have $\mathcal{C}_t \neq \mathcal{C}$. Nonetheless, the design of ApproxEquivalence, which relies on confidence bounds, ensures that $\mathcal{C}_t$ is informative enough, in the sense that $\mathcal{M}_{t,\delta}(\mathcal{C}_t)$ could be much smaller (hence, better) than a set of models that one would obtain by ignoring equivalence structure; this is also validated by the numerical experiments in ergodic environments in the next subsection.*

5.3. Numerical Experiments

We conduct numerical experiments to examine the performance of the proposed variants of C-UCRL, and compare it to that of UCRL2-L[7]. In our experiments, for running ApproxEquivalence, as a sub-routine of C-UCRL, we set $\varepsilon_{u,t} = \beta_{n_t(u)}(\frac{\delta}{3SA})$ for $u \in \mathcal{S}\times\mathcal{A}$ (in the definition of both $\widehat{d}(\cdot,\cdot)$ and $\mathsf{CB}_{t,\delta}(\mathcal{C}_t)$). Although this could lead to a biased estimation of p, such a bias is controlled thanks to using $\alpha > 1$ (in our experiments, we set $\alpha = 4$).

7. UCRL2-L is a variant of UCRL2, which uses confidence bounds derived by combining Hoeffding's and Weissman's inequalities with the Laplace method, as in (2). We stress that UCRL2-L attains a smaller regret than the original UCRL2 of (Jaksch et al., 2010).



(*a*) 25-state Ergodic RiverSwim (*b*) 50-state Ergodic RiverSwim

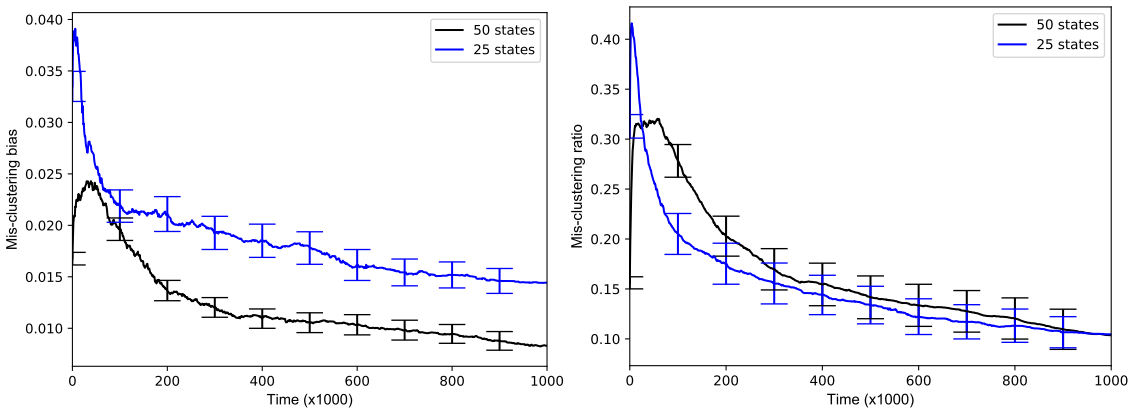
Figure 3: Regret of various algorithms in Ergodic RiverSwim environments

In the first set of experiments, we examine the regret of various algorithms in ergodic environments. Specifically, we consider the ergodic *RiverSwim* MDP, shown in Figure 2, with 25 and 50 states. In both cases, we have $C = 6$ classes. In Figure 3, we plot the regret against time steps under C-UCRL($\mathcal{C}, \boldsymbol{\sigma}$), C-UCRL, and UCRL2-L executed in the aforementioned environments. The results are averaged over 100 independent runs, and the 95% confidence intervals are shown. All algorithms use $\delta = 0.05$, and for C-UCRL, we use $\alpha = 4$. As the curves show, the proposed C-UCRL algorithms significantly outperform UCRL2-L, and C-UCRL($\mathcal{C}, \boldsymbol{\sigma}$) attains the smallest regret. In particular, in the 25-state environment and at the final time step, C-UCRL($\mathcal{C}, \boldsymbol{\sigma}$) attains a regret smaller than that of UCRL2-L by a factor of approximately $\sqrt{SA/C} = \sqrt{50/6} \approx 2.9$, thus verifying Theorem 13. Similarly, we may expect an improvement in regret by a factor of around $\sqrt{SA/C} = \sqrt{100/6} \approx 4.1$ in the other environment. We however get a better improvement (by a factor of around 8), which can be attributed to the increase in the regret of UCRL2-L due to a long burn-in phase (i.e., the phase before the algorithm starts learning).

We now turn our attention to the quality of approximate equivalence structure produced by ApproxEquivalence (Algorithm 1), which is run as a sub-routine of C-UCRL. To this aim, we introduce two performance measures to assess the quality of clustering: The first one is defined as the total number of pairs that are *mis-clustered*, normalized by the total number SA of pairs. We refer to this measure as the *mis-clustering ratio*. More precisely, let $\mathcal{C}_t$ denote the empirical equivalence structure output by ApproxEquivalence at time t. For a given $c \in \mathcal{C}_t$, we consider the restriction of $\mathcal{C}$ to c, denoted by $\mathcal{C}|c$. We find $\ell(c) \in \mathcal{C}|c$ that has the largest cardinality: $\ell(c) \in \operatorname{argmax}_{x \in \mathcal{C}|c} |x|$. Now, we define

$$\text{mis-clustering ratio at time } t := \frac{1}{SA} \sum_{c \in \mathcal{C}_t} |c \setminus \ell(c)| \,.$$

Note that the mis-clustering ratio falls in $[0, 1]$ as $\sum_{c \in \mathcal{C}_t} |c| = SA$ for all t. Our second performance measure accounts for the error in the aggregated empirical transition probability



(*a*) Mis-clustering bias (*b*) Mis-clustering ratio

Figure 4: Assessment of quality of approximate equivalence structures for Ergodic RiverSwim with 25 and 50 states

due to mis-clustered pairs. We refer to this measure as *mis-clustering bias*. Precisely speaking, for a given pair $e \in \mathcal{S} \times \mathcal{A}$, we denote by $c_e \in \mathcal{C}_t$ the set containing e in $\mathcal{C}_t$. Then, we define the mis-clustering bias at time t as

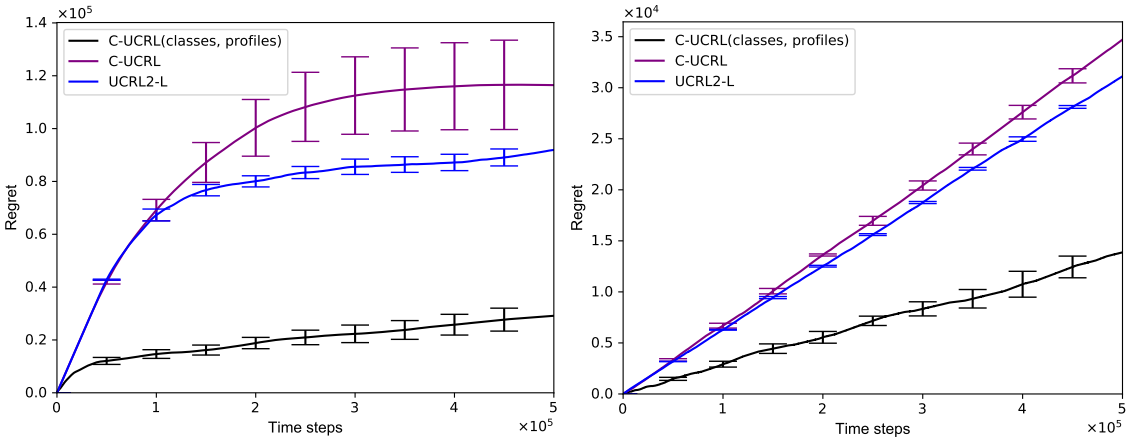
$$\text{mis-clustering bias at time } t := \sum_{c \in \mathcal{C}_t} \sum_{e \notin \ell(c)} \|\widehat{p}_t(\cdot|c_e) - \widehat{p}_t(\cdot|c_e \setminus \{e\})\|_1 \,.$$

In Figure 4, we plot the "mis-clustering ratio" and "mis-clustering bias" for the empirical equivalence structures produced in the previous experiments. We observes on the figures that the errors in terms of the aforementioned performance measures reduce. These errors do now vanish quickly, thus indicating that the generated empirical equivalence structures do not agree with the true one. Yet, they help reduce uncertainty in the transition probabilities, and, in turn, reduce the regret; we refer to Remark 14 for a related discussion.

In the second set of experiments, we consider two communicating environments: *4-room grid-world* (with 49 states) and *RiverSwim* (with 25 states). These environments are described in Appendix E. In Figure 5, we plot the regret against time steps under `C-UCRL`$(\mathcal{C}, \boldsymbol{\sigma})$, `C-UCRL`, and `UCRL2-L`, and similarly to the previous case, we set $\delta = 0.05$ and $\alpha = 4$. The results are averaged over 100 independent runs, and the 95% confidence intervals are shown. In both environments, `C-UCRL`$(\mathcal{C}, \boldsymbol{\sigma})$ significantly outperforms `UCRL2-L`. However, `C-UCRL` attains a regret, which is slightly worse than that of `UCRL2-L`. This can be attributed to the fact that `ApproxEquivalence` is unable to find an accurate enough equivalence structure in these non-ergodic environments.

6. Conclusion

We introduced a similarity measure of state-action pairs, which induces a notion of equivalence of profile distributions in the state-action space of a Markov Decision Process. In the



(*a*) 25-state RiverSwim (*b*) Grid-World

Figure 5: Regret of various algorithms in communicating environments

case of a known equivalence structure, we have presented confidence sets incorporating such knowledge that are provably tighter than their corresponding counterparts ignoring equivalence structure. In the case of an unknown equivalence structure, we presented an algorithm, based on confidence bounds, that seeks to estimate an empirical equivalence structure for the MDP. In order to illustrate the efficacy of our developments, we further preseted `C-UCRL`, which is a natural modification of `UCRL2` using the presented confidence sets. We show that when the equivalence structure is known to the learner, `C-UCRL` attains a regret smaller than that of `UCRL2` by a factor of $\sqrt{SA/C}$ in communicating MDPs, where C denotes the number of classes. In the case of an unknown equivalence structure, we show through numerical experiments that in ergodic environments, `C-UCRL` outperforms `UCRL2` significantly. The regret analysis in this case is much more complicated, and we leave it for future work. We believe that the presented confidence sets can be combined with model-based algorithms for the discounted setup, which we expect to yield improved performance in terms of sample complexity both in theory and practice.

Acknowledgements

This work has been supported by CPER Nord-Pas-de-Calais/FEDER DATA Advanced data science and technologies 2015-2020, the French Ministry of Higher Education and Research, Inria, and the French Agence Nationale de la Recherche (ANR), under grant ANR-16-CE40-0002 (project BADASS).

References

Y. Abbasi-Yadkori, D. Pál, and Cs. Szepesvári. Improved algorithms for linear stochastic bandits. In *Proc. of NIPS*, pages 2312–2320, 2011.

D. Abel, D. Hershkowitz, and M. L. Littman. Near optimal behavior via approximate state abstraction. In *Proc. of ICML*, pages 2915–2923, 2016.

A. Anand, A. Grover, Mausam, and P. Singla. ASAP-UCT: Abstraction of state-action pairs in UCT. In *Proc. of IJCAI*, pages 1509–1515, 2015.

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

P. L. Bartlett and A. Tewari. REGAL: A regularization based algorithm for reinforcement learning in weakly communicating MDPs. In *Proc. of UAI*, pages 35–42, 2009.

E. Brunskill and L. Li. Sample complexity of multi-task reinforcement learning. In *Proc. of UAI*, page 122, 2013.

A. N. Burnetas and M. N. Katehakis. Optimal adaptive policies for Markov decision processes. *Mathematics of Operations Research*, 22(1):222–255, 1997.

C. Dann, T. Lattimore, and E. Brunskill. Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning. In *Proc. of NIPS*, pages 5711–5721, 2017.

T. Dean, R. Givan, and S. Leach. Model reduction techniques for computing approximately optimal solutions for Markov decision processes. In *Proc. of UCAI*, pages 124–131, 1997.

C. Diuk, L. Li, and B. R. Leffler. The adaptive k-meteorologists problem and its application to structure learning and feature selection in reinforcement learning. In *Proc. of ICML*, pages 249–256, 2009.

N. Ferns, P. Panangaden, and D. Precup. Metrics for finite Markov decision processes. In *Proc. of UAI*, pages 162–169, 2004.

N. Ferns, P. Panangaden, and D. Precup. Bisimulation metrics for continuous Markov decision processes. *SIAM Journal on Computing*, 40(6):1662–1714, 2011.

R. Fruit, M. Pirotta, A. Lazaric, and R. Ortner. Efficient bias-span-constrained exploration-exploitation in reinforcement learning. In *Proc. of ICML*, pages 1573–1581, 2018.

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

R. Givan, T. Dean, and M. Greig. Equivalence notions and model minimization in Markov decision processes. *Artificial Intelligence*, 147(1-2):163–223, 2003.

A. Hallak, D. Di-Castro, and S. Mannor. Model selection in Markovian processes. In *Proc. of ACM SIGKDD*, pages 374–382, 2013.

W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

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

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

A. Khaleghi, D. Ryabko, J. Mary, and P. Preux. Consistent algorithms for clustering time series. *JMLR*, 17(1):94–125, 2016.

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

B. R. Leffler, M. L. Littman, and T. Edmunds. Efficient reinforcement learning with relocatable action models. In *Proc. of AAAI*, volume 7, pages 572–577, 2007.

L. Li, T. J. Walsh, and M. L. Littman. Towards a unified theory of state abstraction for MDPs. In *ISAIM*, 2006.

O.-A. Maillard. Mathematics of statistical sequential decision making. *Habilitation à Diriger des Recherches*, 2019.

O.-A. Maillard, T. A. Mann, and S. Mannor. How hard is my MDP? "the distribution-norm to the rescue". In *Proc. of NIPS*, pages 1835–1843, 2014.

T. Mandel, Y.-E. Liu, E. Brunskill, and Z. Popovic. Efficient Bayesian clustering for reinforcement learning. In *Proc. of IJCAI*, pages 1830–1838, 2016.

R. Ortner. Adaptive aggregation for reinforcement learning in average reward Markov decision processes. *Annals of Operations Research*, 208(1):321–336, 2013.

R. Ortner, O.-A. Maillard, and D. Ryabko. Selecting near-optimal approximate state representations in reinforcement learning. In *Proc. of ALT*, pages 140–154, 2014.

V. H. Peña, T. L. Lai, and Q.-M. Shao. *Self-normalized processes: Limit theory and Statistical Applications*. Springer Science & Business Media, 2008.

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

B. Ravindran and A. G. Barto. Approximate homomorphisms: A framework for non-exact minimization in Markov decision processes. In *Proc. of KBCS*, 2004.

A. L. Strehl and M. L. Littman. An analysis of model-based interval estimation for Markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT Press Cambridge, 1998.

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

T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu, and M. J. Weinberger. Inequalities for the L1 deviation of the empirical distribution. *Hewlett-Packard Labs, Technical Report*, 2003.