State Abstractions for Lifelong Reinforcement Learning (Appendix)

David Abel [1] **Dilip Arumugam** [1] **Lucas Lehnert** [1] **Michael L. Littman** [1]

We here include proofs omitted from the paper.

....

Theorem 3.1 (Efficient Abstractions)**.** *Consider any transitive predicate on state pairs, p, that takes computational complexity c_p to evaluate for a given state pair. The state abstraction type ϕ_p that induces the smallest abstract state space can be computed*[1] *in $\mathcal{O}(|\mathcal{S}|^2 \cdot c_p)$.*

Proof. Let c_p denote the computational complexity associated with computing the predicate p for a given state pair. Consider the algorithm consisting of the following four rules for constructing abstract clusters (which define the abstract states) using queries to each of the $|\mathcal{S}|^2$ state pairs. Let (s_i, s_j) denote the current state pair:

1. If $p(s_i, s_j)$ is true, and neither state is in an abstract cluster yet, make a new cluster consisting of these two states.
2. If $p(s_i, s_j)$ is true and only one of the states is already in a cluster, add the other state to the existing cluster.
3. If $p(s_i, s_j)$ is true and both s_i and s_j are in different cluster, merge the clusters.
4. If $p(s_i, s_j)$ is false, add each state not yet in a cluster to its own cluster.

Running this algorithm makes one query per state pair, of which there are $|\mathcal{S}|^2$. Thus, the complexity is $O\left(|\mathcal{S}|^2 \cdot c_p\right)$.

From steps 1-3, after iterating through the possible state pairs, there cannot exist a state pair (s_x, s_y) such that $p(s_x, s_y)$ is true but s_x and s_y are in different clusters. Further, by transitivity, when we apply the cluster merge in step 3, we are guaranteed that every state pair in the resulting cluster necessarily satisfies the predicate. Thus, we compute the smallest clustering definable by p. □

....

Theorem 3.2. *The $\phi_{Q_d^*}$ abstraction type is a subclass of $\phi_{Q_\varepsilon^*}$, studied by Abel et al. (2016) and Hutter (2016), with $d = \varepsilon$, and therefore, for a single MDP:*

$$V^*(s_0) - V^{\pi_{\phi_{Q_d^*}}}(s_0) \leq \frac{2d\text{R}\textsc{Max}}{(1-\gamma)^2}. \tag{1}$$

Proof. For any two state-action pairs that satisfy the predicate ϕ_d^*, we know by definition of the predicate that for each action a, there exists a Q_{lower} such that:

$$Q_{lower} \leq Q(s_1, a) \leq Q_{lower} + d,$$
$$Q_{lower} \leq Q(s_2, a) \leq Q_{lower} + d.$$

Therefore, for each action a:

$$|Q(s_1, a) - Q(s_2, a)| \leq d. \tag{2}$$

Therefore, $\phi_{Q,d}^*$ is a subclass of $\phi_{Q,\varepsilon}^*$. □

....

Theorem 3.2 (Abstract State Space Size)**.** *For a given d, the function belonging to the transitive abstraction type $\phi_{Q_d^*}$ that induces the smallest possible abstract state space size is at most $2^{|\mathcal{A}|}$ times larger than that of the maximally compressing instance of type $\phi_{Q,\varepsilon}$, for $d = \varepsilon$. Thus, letting $\mathcal{S}_d$ denote the abstract state space associated with the maximally compressing $\phi_{Q_d^*}$, and letting $\mathcal{S}_\varepsilon$ denote the abstract state space associated with the maximally compressing ϕ_{Q_ε},:*

$$|\mathcal{S}_\varepsilon| \cdot 2^{|\mathcal{A}|} \geq |\mathcal{S}_d|. \tag{3}$$

Proof. Let M be an arbitrary MDP. Consider a set of states $\tilde{S} \subset \mathcal{S}$ clustered together under $\phi_{Q_\varepsilon^*}$ and, in particular, consider the Q-values of all states in $\tilde{S}$ for a particular action, $a \in \mathcal{A}$. Note that, by construction of $\phi_{Q_\varepsilon^*}$, for any

$$\forall_{s,s' \in \tilde{S}} : |Q(s,a) - Q(s',a)| \leq \varepsilon,$$

Recall that, intuitively, $\phi_{Q_d^*}$ is a discretization of the interval $[0, \text{V}\textsc{Max}]$ where d controls the placement of boundaries, forming buckets of Q-values. The Q-values for all states in $\tilde{S}$ and for action a reside in a single sub-interval of length ε.

Letting $d = \varepsilon$, the placement of boundaries that form $\phi_{Q_d^*}$ could break the ε-interval of Q-values for the non-transitive

[1] Notably, the complexity of c_p dictates the overall complexity of computing ϕ_p.

cluster $\tilde{S}$ no more than once, resulting in the creation of at most two new state clusters in $\phi_{Q_d^*}$. Repeating the process $\forall a \in \mathcal{A}$, these separations within the original cluster compound, resulting in at most $2^{|\mathcal{A}|}$ such subdivisions and, accordingly, $2^{|\mathcal{A}|}$ clusters in $\phi_{Q_d^*}$ for each cluster in $\phi_{Q_\varepsilon^*}$. □

....

Corollary 3.3.1 (PAC Value Loss)**.** *Consider any state-abstraction type ϕ_p with value loss τ_p, that is, in the traditional single task setting:*

$$\forall_{s\in\mathcal{S}} : V^*(s) - V^{\pi_{\phi_p^*}}(s) \le \tau_p. \tag{4}$$

Then, the PAC abstraction ϕ_p^δ, in the lifelong setting, has value loss:

$$\mathop{\mathbb{E}}_{M\sim D}\left[V_M^*(s) - \forall_{s\in\mathcal{S}} : V_M^{\pi_{\phi_p^*}}(s)\right] \le \\ \varepsilon(1-3\delta)\tau_p + 3\delta\text{VM}\textsc{ax}. \tag{5}$$

Proof. By definition of PAC abstractions, with probability $1-\delta$, the abstraction function ϕ_p^δ aggregates iff $\rho_{\delta+\varepsilon}^p$, for some small $\varepsilon \in (-\delta, \delta)$.

Then, with probability $1-\delta$, there is at least a $1-\delta-\varepsilon$ chance that the predicate holds for a particular state, by definition of ρ_δ^p. Thus, by definition of ρ_δ^p, with probability $(1-\delta)(1-\delta-\varepsilon)$, the state abstraction correctly aggregates, and consequently the inherited value loss τ_p bound holds. If the abstraction incorrectly aggregates, the value loss can be up to VMAX.

Letting $\varepsilon = \delta$, we see that the PAC loss is at worst upper bounded by a convex mixture of τ_p with probability $(1-3\delta)$, and with probability 3δ, is VMAX. Thus, the value loss of ϕ_p^δ is:

$$\forall_{s\in\mathcal{S}} : \mathop{\mathbb{E}}_{M\sim D}\left[V_M^*(s) - V_M^{\pi_{\phi_p^*}}(s)\right] \le \\ \varepsilon(1-3\delta)\tau_p + 3\delta\text{VM}\textsc{ax}. \quad \square \tag{6}$$

....

Theorem 3.4 (PAC Abstraction Sample Bound)**.** *Let $\mathscr{A}_p$ be an algorithm that given an MDP $M = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \gamma\rangle$ as input can determine if $p(s_1, s_2)$ is true for any pair of states, for any state abstraction type.*

Then, for a given $\delta \in (0,1]$ and $\varepsilon \in (-\delta, \delta)$, we can compute $\hat{\phi}_p^{\delta+\varepsilon}$ after $m \ge \frac{\ln\left(\frac{2}{\delta}\right)}{\varepsilon^2}$ sampled MDPs from D.

Proof. We are given as input a $\delta \in (0,1]$, a distribution over MDPs D, and the algorithm $\mathscr{A}_p$ which, given an MDP M and a state pair outputs $p_M(s, s')$.

Consider an arbitrary pair of states s and s'. For m sampled MDPs, the algorithm $\mathscr{A}_p$ can produce a sequence of m predicate evaluations:

$$p_1(s, s'), \cdots, p_m(s, s'). \tag{7}$$

Let $\hat{p}$ be the empirical mean over the predicate sequence:

$$\hat{p} = \frac{1}{m}\sum_{i=1}^{m} p_i(s, s'). \tag{8}$$

The clustering algorithm is quite simple: for our input $\delta \in (0,1]$, cluster all state pairs (s, s') such that $\hat{p}(s, s') \ge 1-\delta$ after m samples.

We now prove that, for a particular setting of m, the resulting cluster assignments constitute a state abstraction that clusters a pair of states only if the predicate is true with high probability.

First, let $\overline{p}$ denote the probability that p is true over the distribution:

$$\overline{p}(s, s') = \Pr_{M\sim D}\{p(s, s') = 1\}. \tag{9}$$

Using Hoeffding's bound, we upper bound the probability that $\hat{p}$ deviates from $\overline{p}$ by more than some small $\varepsilon \in (0, \delta)$:

$$\Pr\left\{|\hat{p}(s, s') - \mathbb{E}\left[\hat{p}(s, s')\right]| \ge \varepsilon\right\} \tag{10}$$

$$= \Pr\left\{|\hat{p}(s, s') - \overline{p}(s, s')| \ge \varepsilon\right\} \le 2e^{-2m\varepsilon^2}. \tag{11}$$

Thus, for $\delta = 2e^{-2m\varepsilon^2}$:

$$\Pr\left\{|\hat{p}(s, s') - \overline{p}(s, s')| < \varepsilon\right\} > 1-\delta. \tag{12}$$

Rewriting:

$$\Pr\left\{|\hat{p}(s, s') - \overline{p}(s, s')| < \varepsilon\right\} > 1-\delta \tag{13}$$

$$\iff \Pr\left\{-\varepsilon < \hat{p}(s, s') - \overline{p}(s, s') < \varepsilon\right\} > 1-\delta, \tag{14}$$

By algebra, note that, when $m \ge \frac{\ln\frac{2}{\delta}}{\varepsilon^2}$, the condition of Equation 12 holds.

Let ρ_δ^p denote the predicate that is true if and only if p is true over the distribution with high probability for a given δ:

$$\rho_\delta^p(s_1, s_2) = \begin{cases} 1 & \overline{p} \ge 1-\delta \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

Now, we form our state abstraction under the following rule:

$$\hat{\phi}_p^\delta(s_1) = \hat{\phi}_p^\delta(s_2) \equiv \hat{p}(s, s') > 1-\delta. \tag{16}$$

If, after m samples, $\hat{p}$ were identical to $\overline{p}$, then we would have:

$$\forall_{s,s'} : \Pr_{M\sim D}\{\rho_\delta^p(s, s') \equiv \hat{\phi}_p^\delta(s_1) = \hat{\phi}_p^\delta(s_2)\} \ge 1-\delta. \tag{17}$$

Hence, $\hat{p}$ deviates from $\overline{p}$ by at most ϵ with probability $1-\delta$. Thus, for some $\varepsilon \in (-\delta, \delta)$, $\hat{p} + \varepsilon = \overline{p}$. Therefore, the clustering rule defined by Equation 16 ensures there exists an ε such that, with high probability, we cluster according to:

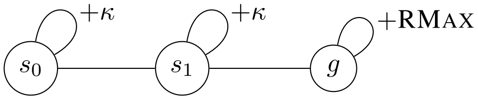
$$\forall_{s_1,s_2} : \rho^p_{\delta+\varepsilon}(s_1, s_2) \equiv \phi^\delta_p(s_1) = \phi^\delta_p(s_2). \tag{18}$$

We conclude that, for $m \geq \frac{\ln \frac{2}{\delta}}{\varepsilon^2}$ sampled and solved MDPs, we compute a lifelong PAC state abstraction $\hat{\phi}^\delta_p$. □

....

Theorem 3.5. *Consider an MDP M and an instance of R-Max (Brafman & Tennenholtz, 2002) that breaks ties using round-robin selection over actions. This algorithm is PAC-MDP in the raw state space. Next, pair a domain with any state-abstraction function ϕ. If R-Max interacts with M by projecting any received state s through ϕ, then R-Max is no longer guaranteed to be PAC-MDP in M. In fact, the number of mistakes made by R-Max can be arbitrarily large.*

Proof. Consider the simple three state chain:



The agent has three actions, `left`, `right`, and `loop`, associated with their natural effects (`left` in s_0 is a self loop with reward 0, while `right` moves the agent to s_1, and so on).

In states s_0 and s_1, let the reward for `loop` be some small constant κ, and let the `loop` action in s_3 yield RMAX reward.

Let $\varepsilon = 0.1$, $\gamma = 0.95$, s_0 define the initial state, and $\kappa = 0.001$. Then

$$\forall_{s \in \{s_0, s_1, s_2\}} : \max_{a_1, a_2} Q^*(s, a_1) - Q^*(s, a_2) \leq \varepsilon.$$

Therefore, for $\varepsilon = 0.1$, a valid clustering assigns $\phi(s_0) = \phi(s_1)$. The R-Max knownness parameter for a state-action pair is given as m.

To break ties, we suppose R-Max chooses actions according to a *round-robin* policy, starting with action `left`. Thus, in the abstract, R-Max first chooses left, then right, then self loop, then left, right, self loop, and so on, until each state-action pair is known.

In the above problem, this sequence of actions will *never* lead the agent out of state s_0 or s_1. Therefore, after m executions of these three actions across states s_0 and s_1, R-Max with ϕ will compute a transition model that never includes the ability to transition to g. Further, the action `loop` will have the largest reward associated with it—κ, a reward chosen to be arbitrarily small—which is thus arbitrarily worse than the goal reward. So, R-Max will make an unbounded number of mistakes. □

....

Corollary 3.5.1. *For any RL algorithm $\mathscr{A}$ whose policy updates during learning and an arbitrary state abstraction ϕ.*

Let $\mathscr{A}_\phi$ denote the algorithm yielded by projecting all incoming states to $\phi(s)$ before presenting them to $\mathscr{A}$, and let $M_\phi = \langle \mathcal{S}_\phi, \mathcal{A}, \mathcal{T}_\phi, \mathcal{R}_\phi, \gamma \rangle$, denote the abstract MDP induced by ϕ on M, where:

$$\begin{aligned} \mathcal{S}_\phi &= \{\phi(s) : \forall_{s \in \mathcal{S}}\}, \\ \mathcal{R}_\phi(\phi(s), a) &= \sum_{g \in \phi^{-1}(\phi(s)))} w(g) \mathcal{R}(g, a), \\ \mathcal{T}_\phi(s, a, s') &= \sum_{g \in G(s)} \sum_{g' \in G(s')} \mathcal{T}_\phi(g, a, g') w(g), \end{aligned}$$

with $w(s)$ is a fixed weighting function and $G(s) = \phi^{-1}(\phi(s))$. That is, $G(s)$ gets all of the true environmental states in the same cluster as s.

The process yielded by $\mathscr{A}_\phi$ interacting with M is not identical to $\mathscr{A}$ interacting with M_ϕ. That is, the expected trajectory taken by the agent is not the same in the two situations. Formally:

$$\mathbb{E}_{\mathscr{A}} \left[s_t \mid s_0, \pi \right] \neq \mathbb{E}_{\mathscr{A}_\phi} \left[s_t \mid s_0, \pi \right], \tag{19}$$

where s_t is the state the agent arrives in after t time steps.

Proof. Note that when M_ϕ is computed directly, the functions $\mathcal{R}_\phi$ and $\mathcal{T}_\phi$ assume a fixed weighting function $w(s)$.

Again consider the three state chain from the previous proof.

During typical interaction between M and $\mathscr{A}_\phi$, however, no such fixed weighting function exists *for any algorithm $\mathscr{A}$ that updates its policy*. That is, the distribution of states the agent finds itself in will change as its policy changes, and therefore, $w(s)$ must change, too.

Thus, the process of $\mathscr{A}_\phi$ interacting with M induces a sequence of interactions with abstract MDPs whose transition and rewards change along with the policy the agent follows. Thus, for any non-identity ϕ, for any algorithm $\mathscr{A}$ whose policy changes over time, the resulting interaction is non-identical. □