Lipschitz Lifelong Reinforcement Learning

Erwan Lecarpentier[1, 2], David Abel[3], Kavosh Asadi[3, 4], Yuu Jinnai[3], Emmanuel Rachelson[1], Michael L. Littman[3]

[1]ISAE-SUPAERO, Université de Toulouse, France
[2]ONERA, The French Aerospace Lab, Toulouse, France
[3]Brown University, Providence, Rhode Island, USA
[4]Amazon Web Service, Palo Alto, California, USA

Problem statement: Lifelong RL

Problematic: in Lifelong RL, how to perform safe, distance-based, online knowledge transfer to accelerate learning of subsequent tasks?

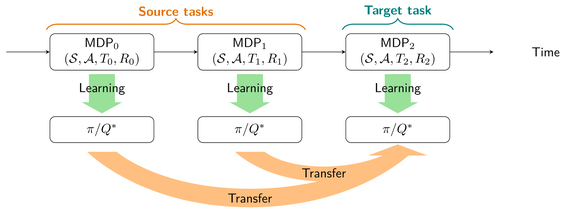


Figure: In Lifelong RL, an agent interacts sequentially with a series of MDPs.

Takeaway message

We consider the problem of knowledge transfer in Lifelong Reinforcement Learning (RL), *i.e.*, when an agent is facing a series of RL tasks, modeled by Markov Decision Processes (MDPs). Our contributions are as follows:

1. We introduce a novel metric pseudo-metric between MDPs;
2. We establish that the optimal value function Q^*_M is Lipschitz Continuous with respect to the MDP space endowed with this pseudo-metric;
3. From this theoretical result, we build a value-transfer method for Lifelong RL;
4. We adapt this method in an algorithm called Lipschitz RMax: the first **online**, **PAC-MDP**, **distance-based**, **non-negative transfer** method for Lifelong RL.

1 A pseudo-metric between MDP models

Definition (Pseudo-metric between models)

Given two MDPs $M = (\mathcal{S}, \mathcal{A}, R, T)$ and $\bar{M} = (\mathcal{S}, \mathcal{A}, \bar{R}, \bar{T})$, we define the *pseudo-metric between models* at $(s, a) \in \mathcal{S} \times \mathcal{A}$ as:

$$D_{sa}(M \| \bar{M}) \triangleq |R^a_s - \bar{R}^a_s| + \gamma \sum_{s' \in \mathcal{S}} V^*_M(s') |T^a_{ss'} - \bar{T}^a_{ss'}|.$$

2 Lipschitz continuity result

Proposition (Local pseudo-Lipschitz continuity)

For two MDPs $M, \bar{M}$, for all $(s, a) \in \mathcal{S} \times \mathcal{A}$,

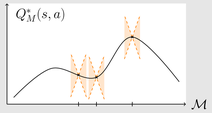
$$\left| Q^*_M(s, a) - Q^*_{\bar{M}}(s, a) \right| \leq \Delta_{sa}(M, \bar{M}),$$

with the local MDP pseudometric defined as

$$\Delta_{sa}(M, \bar{M}) \triangleq \min \left\{ d_{sa}(M \| \bar{M}), d_{sa}(\bar{M} \| M) \right\}, \quad (1)$$

and the local MDP dissimilarity $d_{sa}(M \| \bar{M})$ defined as the unique solution to the following fixed-point equation for d_{sa}:

$$d_{sa} = D_{sa}(M \| \bar{M}) + \gamma \sum_{s' \in \mathcal{S}} T^a_{ss'} \max_{a' \in \mathcal{A}} d_{s'a'}, \forall s, a.$$



3 Transfer method

Idea 1:
Close MDPs in the sense of Equation 1 have close Q^*.

Precisely, if we can measure the local pseudo-distance between two MDPs $M, \bar{M} \in \mathcal{M}$, we can deduce some information about their Q-values in the form of an upper-bound:

$$Q^*_M(s, a) \leq Q^*_{\bar{M}}(s, a) + \Delta_{sa}(M, \bar{M}).$$

We call this upper-bound the *Lipschitz bound on Q^*_M induced by $Q^*_{\bar{M}}$* and write it

$$U_{\bar{M}}(s, a) \triangleq Q^*_{\bar{M}}(s, a) + \Delta_{sa}(M, \bar{M}).$$

Idea 2:
Knowing a tight upper-bound on Q^* allows for fast learning.

From ideas 1 and 2, we build a transfer scheme for Lifelong RL:

1. Sample a new MDP $M \in \mathcal{M}$
2. Measure the distance between M and each source MDP and select $\bar{M}$, the closest MDP
3. Use $U_{\bar{M}}(s, a) = Q^*_{\bar{M}}(s, a) + \Delta_{sa}(M, \bar{M})$ as an upper-bound on Q^*_M to accelerate learning

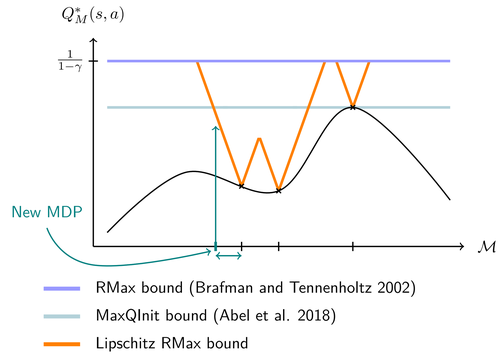


Figure: Upper-bounds on Q^*_M of the RMax, MaxQInit and Lipschitz RMax algorithms. Tighter upper-bounds potentially improve the sample efficiency of the algorithms.

Distance-based transfer scheme

This transfer method is **distance-based**, which we believe to be an important feature of an efficient transfer scheme. Intuitively, the amount of transferable knowledge should be proportional to a notion of similarity between tasks:

"Close tasks should allow for a large amount of transferable knowledge, and vice versa"

Questions:

1. How to compute the local pseudo-distance between two MDPs $\Delta_{sa}(M, \bar{M})$ online?
2. What happens if both the source M and the target MDP $\bar{M}$ are partially known?

Answer: We propose to make an approximation to be able to compute the induced Lipschitz bound $U_{\bar{M}}(s, a)$ online. This results in the Lipschitz RMax algorithm.

4 Lipschitz RMax algorithm

Lipschitz RMax practically implements the transfer method of Section 3 in the online Lifelong RL setting. The algorithm relies on two things:

1. An approximation of the induced Lipschitz upper-bound $U_{\bar{M}}(s, a)$;
2. The ability to use the maximum possible distance between models $D_{\max}$ in the form of prior knowledge.

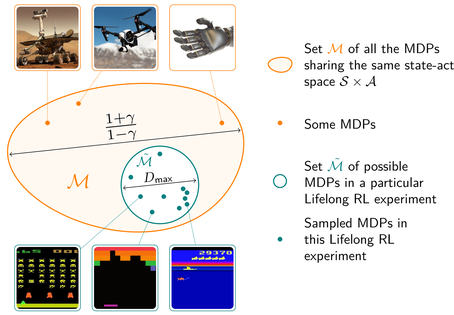


Figure: Illustration of the prior knowledge on the maximum pseudo-distance between models for a particular s, a pair. The maximum pseudo-distance between any MDPs reward and transition functions is $\frac{1+\gamma}{1-\gamma}$. In contrast, this distance, denoted by $D_{\max}$, is generally smaller in a particular Lifelong RL experiment.

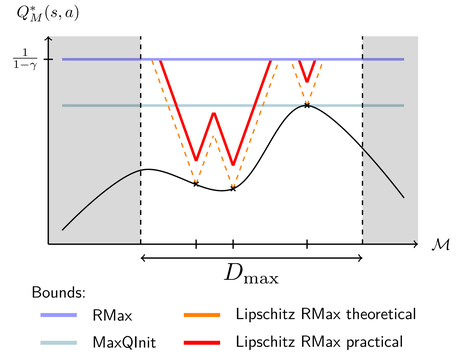


Figure: Illustration of the bounds, including the one practically used by Lipschitz RMax, represented in red. The knowledge of $D_{\max}$ in the form of prior knowledge allows Lipschitz RMax to reduce the space of possible MDPs in its approximation error.

Features of Lipschitz RMax:

- **Online:** the method can be applied online, without full knowledge of the target and source MDPs.
- **PAC-MDP** (Strehl, Li, and Littman 2009)**:** with probability higher than $1 - \delta$, Lipschitz RMax converges to an ϵ-optimal policy, with a polynomial *sample*, *computational* and *space* complexity in $(S, A, 1/\epsilon, 1/\delta, 1/(1-\gamma))$.
- **Distance-based:** the closer the MDPs, the higher the amount of transferred knowledge
- **Non-negative transfer:** with probability higher than $1 - \delta$, the computed induced Lipschitz bound is an upper-bound on Q^*_M, which prevents the reduction of performance by under-exploration.

5 Lifelong RL experiments

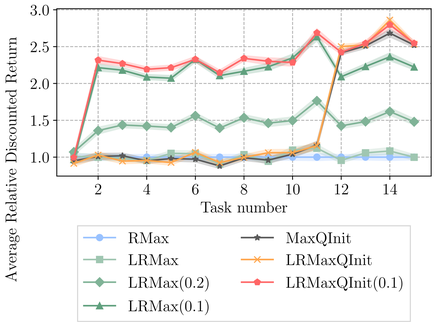


Figure: Performance of RMax, Lipschitz RMax, MaxQInit and a combination of Lipschitz RMax and MaxQInit in a Lifelong RL experiment featuring grid-world MDPs. The score is represented as a function of the task number.

Perspectives

1. Same approach with function approximation?
2. Other metrics than Equation 1: less conservative? Problem-dependent?
3. Reduce the linearly growing number of source tasks? Clustering?

References

Abel, D.; Jinnai, Y.; Guo, S. Y.; Konidaris, G.; and Littman, M. L. 2018. Policy and Value Transfer in Lifelong Reinforcement Learning. In Proceedings of the 35th International Conference on Machine Learning (ICML 2018).

Brafman, R. I.; and Tennenholtz, M. 2002. RMax – a General Polynomial Time Algorithm for Near-Optimal Reinforcement Learning. Journal of Machine Learning Research.

Strehl, A. L.; Li, L.; and Littman, M. L. 2009. Reinforcement Learning in Finite MDPs: PAC Analysis. Journal of Machine Learning Research 10(Nov): 24132444.