Universitat Pompeu Fabra
Barcelona

u*pf.*

Globally Optimal Hierarchical Reinforcement Learning for Linearly-Solvable Markov Decision Processes

Guillermo Infante, Anders Jonsson, Vicenç Gómez

AAAI 2022

Overview

Introduction

Introduction

- Hierachical Reinforcement Learning aims to make learning more efficient by exploiting of large problems (Wen et al., 2020).

- Novel approach to HRL using Linearly-solvable Markov Decision Processes (LMDPs).

- LMDPs are a ¡¿ (Todorov, 2006).

- Globally optimal value function (vs. *hierarchically optimal* or *recursively optimal*) (Dieterich, 2000).

Background

An LMDP (Kappen et al., 2012; Todorov, 2006) is as a tuple
$\mathcal{L} = \langle \mathcal{S}, \mathcal{T}, \mathcal{P}, \mathcal{R}, \mathcal{J} \rangle$:

- We define $\mathcal{S}^+ = \mathcal{S} \cup \mathcal{T}$ to denote the full set of states.
- $\mathcal{P} : \mathcal{S} \rightarrow \Delta(\mathcal{S}^+)$.
- $\mathcal{R} : \mathcal{S} \rightarrow \mathbb{R}$ and $\mathcal{J} : \mathcal{T} \rightarrow \mathbb{R}$.
- Learning agent follows a policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{S}^+)$.

- At time-step t, the agent observes s_t and receives a reward

$$\mathcal{R}(s_t, \pi) = \mathcal{R}(s_t) - \lambda \cdot \mathrm{KL}(\pi(\cdot|s_t) \| \mathcal{P}(\cdot|s_t)).$$

- The aim of the agent is to maximize

$$v^\pi(s) = \mathbb{E}\left[\sum_{t=1}^{T-1} \mathcal{R}(S_t, \pi) + \mathcal{J}(S_T) \,\middle|\, S_1 = s\right].$$

- We obtain the following Bellman optimality equation

$$\frac{1}{\lambda}v(s) = \frac{1}{\lambda}\mathcal{R}(s) + \max_\pi \mathbb{E}_{s' \sim \pi(\cdot|s)}\left[\frac{1}{\lambda}v(s') - \log\frac{\pi(s'|s)}{\mathcal{P}(s'|s)}\right] \quad (\forall s).$$

- Taking $z(s) = e^{v(s)/\lambda}$ for each $s \in \mathcal{S}^+$ leads to

$$z(s) = e^{\mathcal{R}(s)/\lambda} \sum_{s'} \mathcal{P}(s'|s)z(s').$$

- If $\mathcal{P}$ and $\mathcal{R}$ are known, then

$$\mathbf{z} = RP\mathbf{z}^+ \text{ where } R = \text{diag}(e^{\mathcal{R}(\cdot)/\lambda}).$$

- Alternatively, there is an online (corrected) update rule

$$\hat{z}(s_t) \leftarrow (1 - \alpha_t)\hat{z}(s_t) + \alpha_t e^{r_t/\lambda}\hat{z}(s_{t+1})\frac{\mathcal{P}(s_{t+1}|s_t)}{\hat{\pi}(s_{t+1}|s_t)}.$$

- Given a z, policies are derived using

$$\pi(s'|s) = \frac{\mathcal{P}(s'|s)z(s')}{\sum_{s''}\mathcal{P}(s''|s)z(s'')}.$$

Background - LMDPs (iv) - Compositionality

- Let $\{\mathcal{L}_1, \ldots, \mathcal{L}_n\}$ be a collection of LMDPs.
- Each LMDP $\mathcal{L}_i$ only differs in $\mathcal{J}_i(\mathsf{t})$.
- For a new LMDP $\mathcal{L}$ for which the next holds (Todorov, 2009)

$$e^{\mathcal{J}(\mathsf{t})/\lambda} = z(\mathsf{t}) = \sum_{k=1}^{n} w_k e^{\mathcal{J}_k(\mathsf{t})/\lambda}.$$

- Due to linearity, the following is also satisfied

$$z(s) = \sum_{k=1}^{n} w_k z_k(s) \ \forall s \in \mathcal{S}.$$

Hierarchical LMDPs

Hierarchical LMDPs (i)

- Given an LMDP $\mathcal{L}$, its state space $\mathcal{S}$ is partitioned into L subsets $\{\mathcal{S}_i\}_{i=1}^{L}$.
- Each such subset $\mathcal{S}_i$ induces a subtask, represented by an LMDP $\mathcal{L}_i = \langle \mathcal{S}_i, \mathcal{T}_i, \mathcal{P}_i, \mathcal{R}_i, \mathcal{J}_i \rangle$:
  - ▶ $\mathcal{T}_i$ includes states in $\mathcal{S}^+ \setminus \mathcal{S}_i$ that are one step away from any $s \in \mathcal{S}_i$.
  - ▶ For $\tau \in \mathcal{T}_i$,
    $$\mathcal{J}_i(\tau) = \begin{cases} \mathcal{J}(\tau) \text{ if } \tau \in \mathcal{T}, \\ \hat{v}(\tau) \text{ otherwise.} \end{cases}$$

Definition
Subtask equivalence (for some $\mathcal{L}_i$ and $\mathcal{L}_j$) imples a bijective relationship $f : \mathcal{S}_i \rightarrow \mathcal{S}_j$.

- Set of *exit states* $\mathcal{E} = \cup_{i=1}^{L} \mathcal{T}_i$
- $\mathcal{E}_i = \mathcal{E} \cap \mathcal{S}_i$ denotes the set of *exit states inside* subtask $\mathcal{L}_i$.
- Set of equivalence classes $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_C\}$, $C \leq L$.
- A single subtask $\mathcal{L}_j = \langle \mathcal{S}_j, \mathcal{T}_j, \mathcal{P}_j, \mathcal{R}_j, \mathcal{J}_j \rangle$ per equivalence class $\mathcal{C}_j \in \mathcal{C}$.
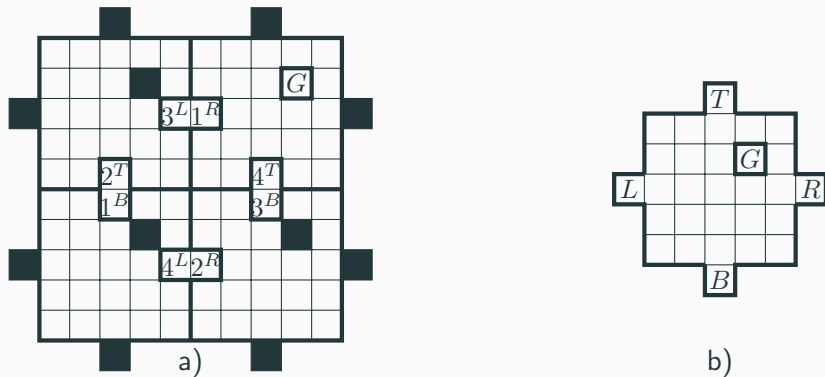
Figure 1: a) A 4-room LMDP, with all exit states highlighted; b) a single subtask with 5 terminal states G, L, R, T, B that is equivalent to all 4 room subtasks.

- Consider subtask $\mathcal{L}_j$ and its terminal set $\mathcal{T}_j = \{\tau_1, \ldots, \tau_n\}$.
- We define n base LMDPs $\mathcal{L}_j^1, \ldots, \mathcal{L}_j^n$, which only differ in $\mathcal{J}_j^k$.
- Concretely,

$$z_j^k(\tau) = \begin{cases} 1 \text{ if } \tau = \tau_k \rightarrow \mathcal{J}_j^k(\tau) = 0, \\ 0 \text{ otherwise } \rightarrow \mathcal{J}_j^k(\tau) = -\infty \end{cases}$$

- Thus, we can solve these base LMDPs to obtain $z_j^1, \ldots, z_j^n$.
- Having $\hat{z}(s)$ for each $t \in \mathcal{T}_j$, then thanks to compositionality,

$$\hat{z}(s) = \hat{z}(\tau_1) z_j^1(s) + \cdots + \hat{z}(\tau_n) z_j^n(s) \ \ \forall s \in \mathcal{S}_i, \forall \mathcal{L}_i \in \mathcal{C}_j.$$

Hierarchical LMDPs (v) - Subtask compositionality (ii)

- For all subtaks, terminal states $\tau_1 \ldots \tau_n$ are by definition in $\mathcal{E}$.
- Having access $\hat{z}_{\mathcal{E}} : \mathcal{E} \to \mathbb{R}$ and $z_j^1, \ldots, z_j^n$ for the base LMDPs is enough to represent $\hat{z}(s) \; \forall s \in \mathcal{S}$.
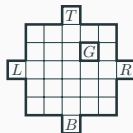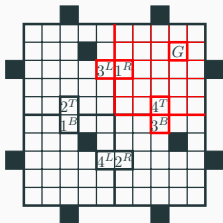- Solutions for base LMDPs can be reused.

Hierarchical LMDPs (vi) - Subtask compositionality (iii)

Remark

If $\hat{z}(s)$ is optimal for $s \in \mathcal{E}$, then $\hat{z}(s)$ for $s \in \mathcal{S}_i$ will also be optimal. With compositionality,

$$\hat{z}(s) = \hat{z}(3^L) * z_L(s) + \hat{z}(3^B) * z_B(s) + \hat{z}(G) * z_G(s)$$

For any s in $\mathcal{S}_i$. Thus, if $\hat{z} = z^*$ for $s \in \mathcal{E}$, then it will be optimal for the interior states as well.

Algorithms

Eigenvector algorithm (i)

- We can restrict

$$\hat{z}(s) = \hat{z}(\tau_1)z_j^1(s) + \cdots + \hat{z}(\tau_n)z_j^n(s) \ \ \forall s \in \mathcal{S}_i, \forall \mathcal{L}_i \in \mathcal{C}_j.$$

  to states $s \in \mathcal{E}$.

- Thus, we define

$$\mathbf{z}_\mathcal{E} = G\mathbf{z}_\mathcal{E}.$$

- This corresponds to an eigenvector problem.

- Value estimates for $s \in \mathcal{S} \setminus \mathcal{E}$ are obtained afterwards.

Eigenvector algorithm (ii) - Convergence proof

Lemma (1)
If the reward of each terminal state $t \in \mathcal{T}_i$ equals its optimal value in $\mathcal{L}$, i.e. $z_i(t) = z(t)$, the optimal value of each non-terminal state $s \in \mathcal{S}_i$ equals its optimal value in $\mathcal{L}$, i.e. $z_i(s) = z(s)$.

Lemma (2)
The solution to $\mathbf{z}_\mathcal{E} = G\mathbf{z}_\mathcal{E}$ is unique.

Lemma (3)
For each subtask $\mathcal{L}_i$ and state $s \in \mathcal{S}_i^+$, it holds that

$$z_i^1(s) + \cdots + z_i^n(s) \leq 1.$$

Online algorithm (i)

- We keep $\hat{z}_j^1, \ldots, \hat{z}_j^n$ for each $\mathcal{L}_j$.
- We can update all base LMDPs within a subtask $\mathcal{L}_j$ using intra-task learning (Kaelbling, 1993; Jonsson and Gómez, 2016).
- The online update rule (again, restricted to $s \in \mathcal{E}$)

$$\hat{z}_\mathcal{E}(s) \leftarrow (1 - \alpha_\ell)\hat{z}_\mathcal{E}(s) + \alpha_\ell[\hat{z}_\mathcal{E}(t_1)\hat{z}_j^1(s) + \cdots + \hat{z}_\mathcal{E}(t_n)\hat{z}_j^n(s)].$$

- Estimates at any level are learned in an episodic manner.
- When shall we update states in $\mathcal{E}$?

Online algorithm (ii)

We propose the following alternatives:

V_1: Update $s \in \mathcal{E}_i$ each time the agent transitions from s.

V_2: When the agent reaches $\tau \in \mathcal{T}_i$ of the subtask $\mathcal{L}_i$, update the values of every $s \in \mathcal{E}_i$.

V_3: When the agent reaches $\tau \in \mathcal{T}_i$ of the subtask $\mathcal{L}_i$, update the values of every $s \in \mathcal{E}_i$ and all exit states of subtasks in the equivalence class $\mathcal{C}_j$ of $\mathcal{L}_i$.

Experiments and results

Experiments - Rooms domain

- We varied the size of the rooms as well as the number of rooms.
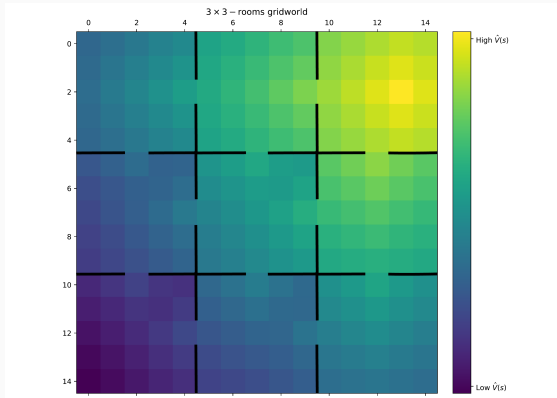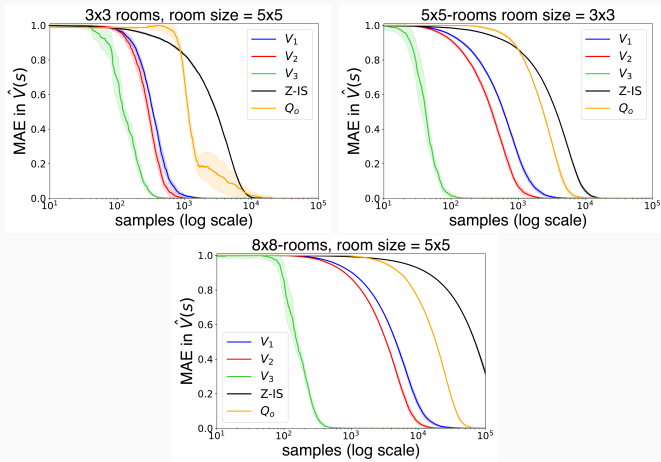
Results - Rooms domain

Figure 2: MAE over time for 3×3 (top-left), 5×5 (top-right) and 8×8 (bottom) room instances.

Experiments - Taxi domain

- A passenger is located at one of the four corners and he must be carried to a certain corner (excluding the pickup location).
- Base LMDPs here are going to each of the corners.
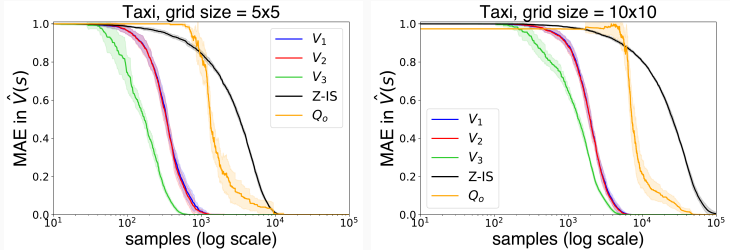- Sometimes there exist natural equivalence classes in facotres MDPs

Passenger at (0,0) and destination at (4,4)
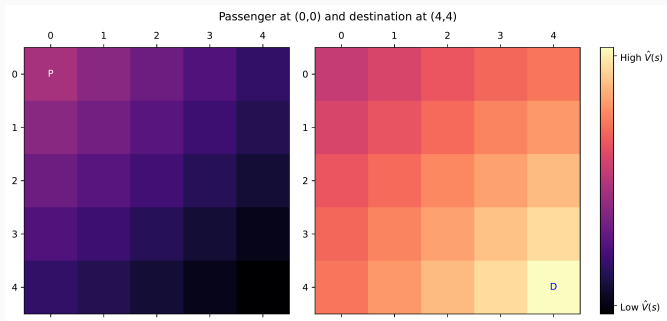
Results - Taxi domain

Figure 3: MAE over time for 5×5 (left) and 10×10 (right) grids of Taxi domain.

Contributions and Conclusion

- We define a novel scheme based on compositionality for solving subtasks.
- The subtasks decomposition is at the level of the value function, thus our approach does not suffer from non-stationarity in the online setting.
- Under mild assumptions, our method converges to the optimal value function.
- Unlike the options setting, we can retrieve the optimal value function for the full state space using the value estimates for the exit states.

Conclusion

- Unlike typical HRL approaches, we no longer need a high-level policy that selects among subgoals (e.g. non-stationarity in SMDPs).

- Instead, we are able to retrieve the optimal value function for each state thanks to the decomposition of the value function in terms of the values of base LMDPs.

References

Dietterich, T. G. (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. *J. Artif. Intell. Res.*, 13:227–303.

Jonsson, A. and Gómez, V. (2016). Hierarchical Linearly-Solvable Markov Decision Problems. In *Proceedings of the 26th International Conference on Automated Planning and Scheduling (ICAPS)*.

Kaelbling, L. P. (1993). Learning to Achieve Goals. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1094–1099.

Kappen, H. J., Gómez, V., and Opper, M. (2012). Optimal control as a graphical model inference problem. *Mach. Learn.*, 87(2):159–182.

Todorov, E. (2006). Linearly-solvable Markov decision problems. *Advances in Neural Information Processing Systems (NIPS)*, pages 1369–1376.

Todorov, E. (2009). Compositionality of optimal control laws. *Advances in Neural Information Processing Systems (NIPS)*, pages 1856–1864.

Wen, Z., Precup, D., Ibrahimi, M., Barreto, A., Van Roy, B., and Singh, S. (2020). On Efficiency in Hierarchical Reinforcement Learning. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*.