Bayesian-Adaptive Deep Reinforcement Learning via Ensemble Learning

Gilwoo Lee
gilwoo@cs.uw.edu

Jeongseok Lee
jslee02@cs.uw.edu

1 Introduction

While reinforcement learning is capable of controlling complex autonomous systems, RL algorithms typically require huge amounts of data and can overfit to a particular task or to be prone to disturbance. One of main challenges that needs to be addressed is train a policy robust to various model uncertainties and disturbances. In this project, we aim to address this challenge via an ensemble policy for Bayes-Adaptive Reinforcement Learning [1].

We assume that there exists a latent physics variable ϕ which determines the transition function of the underlying MDP, i.e. the transition function $P(s', \phi'|s, \phi, a)$ is now a function of state, action, and ϕ. We would like to learn a policy which maximizes the long term reward given ϕ. Formally, this is called Bayes-Adaptive MDP [1, 2], defined by a tuple $< \mathcal{S}', \mathcal{A}, P', P_0', R' >$ where

- $\mathcal{S}' = \mathcal{S} \times \Phi$ is the set of (states, physics variable),
- $\mathcal{A}$ is the set of actions,
- $P(\cdot|s, \phi, a)$ is the transition function between hyper-states, conditioned on action a being taken in hyper-state (s, ϕ),
- $P_0 \in \mathcal{P}(\mathcal{S} \times \Phi)$ combines the initial distribution over hyper-states,
- $R'(s, \phi, a)$ represents the reward obtained when action a is taken in hyper-state (s, ϕ).

We would like to find the optimal policy for the following Bellman equaton:

$$V^*(s, \phi) = \max_a \mathbb{E}\left[R(s, a, \phi) + \gamma \sum_{s', \phi'} P(s', \phi'|s, \phi, a) V^*(s', \phi')\right] \tag{1}$$

This formualtion is often refered as Bayes-Adaptive Reinforcement Learning (BARL) [1].

We make a simplification to BARL formulation. We assume that the dynamics of s' and ϕ' are independent given $P(s, \phi, a)$, i.e.

$$P(s', \phi'|s, \phi, a) = P(s'|s, \phi, a) \cdot P(\phi'|s, \phi, a).$$

This assumption allows us to simplify BARL with a gated ensemble policy learning method. At the high-level, we have a gating network that determines the *belief* of the physics parameters at time t,

$$b(\phi_t) = P(\phi_t|s_{t-1}, \phi_{t-1}, a_{t-1})$$

which is then used to compute the best policy from an ensemble of ϕ-dependent optimal policies, i.e., $\pi^*(\cdot; \phi)$ and $V^*(\cdot; \phi)$ are computed with typical RL algorithms for MDPs. Then the remaining task is to compute the one-step best action a:

$$a^* = \arg\max_a \mathbb{E}_{\phi \sim b(\phi)}\left[R(s, a, \phi) + \gamma \sum_{s', \phi'} P(s', \phi'|s, \phi, a) V^*(s', \phi')\right] \tag{2}$$

We model b_ϕ as a network capable of modeling evolving state change, e.g. Recurrent Neural Networks, or as a Bayes filter. At the low level, we plan to discretize Φ and have one actor-critic network per a discretized value of ϕ: each critic estimates $V^*(\cdot; \phi)$ and each actor has an optimal policy for a particular discretized value of $\pi^*(\cdot; \phi)$. Given b_ϕ and the set of actor-critic networks, it is straightforward to compute (2).

2 Background

Our work is closely related to QMDP [3, 4] which is an approximation for POMDP. QMDP approximates POMDP by assuming fully-observable MDP after 1-step, and approximating the Q-value at the current belief state $b(s)$ as $Q_a(b) = \sum_s b(s)Q_{MDP}(s, a)$. In our problem setup, we have a belief over the physics parameters ϕ of the MDP, $b(\phi)$, and we compute the policy $Q_a(s; b) = \sum_\phi b(\phi)Q_{MDP}(s, a; \phi)$.

The BAMDP formulation is also similar to POMDP formulation used in POMDP-lite [5] which assumes that the hidden state variables are are constant or only change deterministically. In our case, the hidden state variables correspond to the physics parameters ϕ. The authors of POMDP-lite have shown that such formulation is "equivalent to a set of fully observable Markov decision processes indexed by a hidden parameter" [5], which, in our case, could be viewed as a discretization of ϕ.

References

[1] M. Ghavamzadeh, S. Mannor, J. Pineau, A. Tamar, *et al.*, "Bayesian reinforcement learning: A survey," *Foundations and Trends® in Machine Learning*, vol. 8, no. 5-6, pp. 359–483, 2015.

[2] A. Guez, D. Silver, and P. Dayan, "Efficient bayes-adaptive reinforcement learning using sample-based search," in *Advances in Neural Information Processing Systems*, pp. 1025–1033, 2012.

[3] M. L. Littman, A. R. Cassandra, and L. P. Kaelbling, "Learning policies for partially observable environments: Scaling up," in *Machine Learning Proceedings 1995*, pp. 362–370, Elsevier, 1995.

[4] P. Karkus, D. Hsu, and W. S. Lee, "Qmdp-net: Deep learning for planning under partial observability," in *Advances in Neural Information Processing Systems*, pp. 4697–4707, 2017.

[5] M. Chen, E. Frazzoli, D. Hsu, and W. S. Lee, "Pomdp-lite for robust robot planning under uncertainty," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pp. 5427–5433, IEEE, 2016.