Reinforcement Learning with Heterogeneous Policy Representations

Petar Kormushev and Darwin G. Caldwell
Department of Advanced Robotics, Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genova, Italy
Email: {petar.kormushev, darwin.caldwell}@iit.it

EXTENDED ABSTRACT

In Reinforcement Learning (RL) the goal is to find a policy π that maximizes the expected future return, calculated based on a scalar reward function $R(\cdot) \in \mathbb{R}$. The policy π determines what actions will be performed by the RL agent. Traditionally, the RL problem is formulated in terms of a Markov Decision Process (MDP) or a Partially Observable MDP (POMDP). In this formulation, the policy π is viewed as a mapping function ($\pi : s \longmapsto a$) from state $s \in S$ to action $a \in A$. This approach, however, suffers severely from the *curse of dimensionality*.

Alternatively, instead of trying to learn the explicit mapping from states to actions, it is possible to perform *direct policy search*, as shown in [1]. In this case, the policy π is considered to depend on some parameters $\theta \in \mathbb{R}^N$, and is written as a parameterized function $\pi(\theta)$. The episodic reward function becomes $R(\tau(\pi(\theta)))$, where τ is a trial performed by following the policy. The reward can be abbreviated as $R(\tau(\theta))$ or even as $R(\theta)$, which reflects the idea that the behaviour of the RL agent can be influenced by only changing the values of the policy parameters θ. Therefore, the outcome of the behaviour, which is represented by the reward $R(\theta)$, can be optimized by only optimizing the values θ. This way, the RL problem is transformed into a black-box optimization problem with cost function $R(\theta)$, as shown in [2] under the name *parameter-based exploration*.

A very important open question is: what is the best way to represent the policy $\pi(\theta)$? During the last decade, numerous policy representations have been proposed, and yet, there is not any deep understanding about which representation is most suitable for a given class of tasks. Considering the huge variety of tasks and numerous (sometimes conflicting) requirements towards the representation, we believe that it might never be possible to construct a single 'ubiquitous' representation that is suitable for any arbitrary task. Therefore, it is necessary to design methods that can automatically choose the most suitable policy representation for a given task.

In this paper, we propose a novel reinforcement learning approach for direct policy search that can simultaneously: (i) determine the most suitable policy representation for a given task; and (ii) optimize the policy parameters of this representation in order to maximize the reward and thus achieve the task. The approach assumes that there is a heterogeneous[1] set of policy representations available to choose from.

A naïve approach to solving this problem would be to take the available policy representations one by one, run a separate RL optimization process (i.e. conduct trials and evaluate the return) for each once, and at the very end pick the representation that achieved the highest reward. Such an approach, while theoretically possible, would not be efficient enough in practice.

Instead, our proposed approach is to conduct one single RL optimization process while interleaving simultaneously all available policy representations. This can be achieved by leveraging our previous work in the area of RL based on Particle Filtering (RLPF) [3], [4]. Particle filters, also known as Sequential Monte Carlo methods [5], originally come from statistics and are similar to importance sampling methods. Particle filters are able to approximate any probability density function, and can be

[1] By 'heterogeneous' we mean different from each other.

viewed as a 'sequential analogue' of Markov chain Monte Carlo (MCMC) batch methods. The main idea of RLPF is to use particle filtering as a method for choosing the sampling points, i.e. for calculating a parameter vector θ for each trial.

The key to linking particle filters and RL is to make the following observation. The landscape, defined by the reward function $R(\theta) \in \mathbb{R}$ over the whole continuous domain of the parameter space $\theta \in \Theta$, can be viewed as defining an improper probability density function (IPDF).

Once we make the assumption that $R(\theta)$ is just an IPDF, then the RL problem can be reformulated from a new point of view. Each trial $\tau(\pi(\theta))$ can be viewed as an independent sample from this unknown IPDF. The RL algorithm can be viewed as a method for choosing a finite number of sampling points for which to obtain the value of the IPDF. Finally, the RL problem can be viewed as the problem of finding the mode (or all modes, in the multi-modal case) of the unknown IPDF, given only a finite number of sampling points with their corresponding values of the IPDF, obtained by the RL algorithm. This view of RL immediately opens the path for applying particle filters, because they are a method for approximate estimation of an unknown PDF based on a finite number of samples.

We define a *policy particle* p_i to be the tuple $p_i = \langle \theta_i, \tau_i, R_i, w_i \rangle$, where the particle p_i represents the outcome of a single trial τ_i performed by executing an RL policy $\pi(\theta_i)$, where θ_i is a vector of policy parameter values modulating the behaviour of the RL policy π. The policy particle also stores the value of the reward function evaluated for this trial $R_i = R(\tau_i(\pi(\theta_i)))$. The variable w_i is the importance weight of this policy particle, and the way of its calculation is explained below.

Firstly, each policy particle p_i is assigned a scalar importance weight w_i derived from its corresponding reward R_i using a transformation function g, such that: $w_i \propto g(R_i)$. In the simplest case, $g(\cdot)$ could be the identity, but in the general case, it could be an arbitrary non-negative function. We apply the function g in such a way, that the importance weights are normalized, in the sense that: $\forall w_i \quad 0 < w_i < 1$, and also: $\sum w_i = 1$.

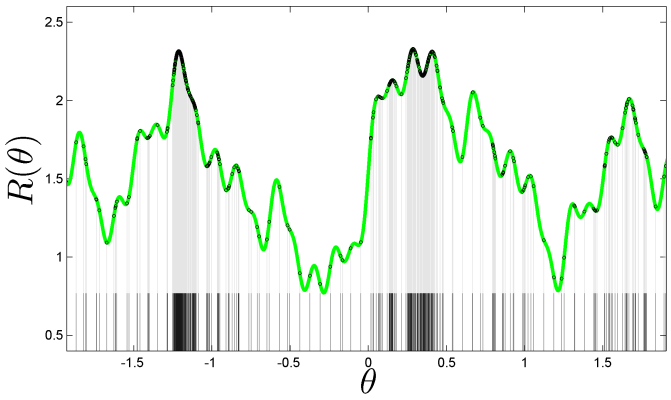

Fig. 1. An illustration of a typical run of RLPF (Reinforcement Learning based on Particle Filtering) on a 1-dimensional problem. The generated policy particles by RLPF are shown with vertical grey stripes. The corresponding reward values are shown with black circles on top of the reward function line, shown in green.

Secondly, we construct an auxiliary function $h(u) = \int_{-\infty}^{u} w_u du$, which in our discrete case takes the form $h(k) = \sum_{j=1}^{k} w_j$. This function can be thought of as the (approximate) cumulative density function (CDF) of the unknown PDF. Indeed, due to the way we create the importance weights, it follows directly that $\int_{-\infty}^{+\infty} w_u du = 1$, and thus $h(u)$ is a proper CDF. This is important because, given that $w_i > 0$, it guarantees that $h(u)$ is strictly monotonically increasing and therefore the inverse function h^{-1} exists.

We believe that this work opens up a novel research direction in RL. We can foresee many ways in which it can be extended in the future.

REFERENCES

[1] M. Rosenstein and A. Barto, "Robot weightlifting by direct policy search," in *International Joint Conference on Artificial Intelligence*, vol. 17, no. 1. Citeseer, 2001, pp. 839–846.

[2] T. Rückstieß, F. Sehnke, T. Schaul, D. Wierstra, Y. Sun, and J. Schmidhuber, "Exploring parameter space in reinforcement learning," *Paladyn. Journal of Behavioral Robotics*, vol. 1, no. 1, pp. 14–24, 2010.

[3] P. Kormushev and D. G. Caldwell, "Direct policy search reinforcement learning based on particle filtering," in *Proceedings of the 10th European Workshop on Reinforcement Learning*, 2012.

[4] ——, "Simultaneous discovery of multiple alternative optimal policies by reinforcement learning," in *Intelligent Systems (IS), 2012 6th IEEE International Conference*. IEEE, 2012, pp. 202–207.

[5] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo methods in practice*. Springer Verlag, 2001.