Off-Policy Neural Fitted Actor-Critic

Toward a data efficient neural actor-critic

{ Matthieu.Zimmer, Yann.Boniface and Alain.Dutech }@loria.fr

Université de Lorraine — Inria — Loria, Laboratoire lorrain de recherche en informatique et ses applications

Problem

Reinforcement learning (RL) is a framework for solving sequential decision problems where an agent interacts with its environment and adapts its policy based on a reward signal.
We present an RL algorithm respecting two main requirements while being most **data efficient** possible :

1. dealing with **continuous action and state** spaces,
2. knowledge added by the designer to the agent should be minimal.

Contributions

Inspired by Fitted Q Iteration (FQI) [1] and Deep Deterministic Policy Gradients (DDPG) [2], we formulated a new **off-policy**, **non-linear**, **off-line**, **model-free**, **actor-critic** algorithm. Unlike FQI, it deals with continuous action and state spaces and performs better than DDPG on three experimental environments.

Algorithm

Data: $\mathcal{D}$ replay buffer of N samples, Q_0 value-function, π_b previous policies, K number of fitted iteration, G number of gradient descent for actor updates, $inverting\_gradient$ strategy, $reset\_critic$ strategy

for $k \leftarrow 1$ **to** K **do**
  for $(s_t, a_t, u_t, r_{t+1}, s_{t+1}) \in \mathcal{D}$ **do**
    $q_{k,t} \leftarrow r_{t+1}$
    if $s_{t+1} \notin S^*$ **then**
      $q_{k,t} \leftarrow q_{k,t} + \gamma Q_{k-1}(s_{t+1}, \pi_{k-1}(s_{t+1}))$
    end
  end
  $Q_k \leftarrow Q_{k-1}$
  if $reset\_critic$ **then**
    $Q_k \leftarrow$ randomly initialize critic network
  end
  Update critic by minimizing the loss:

$$\frac{1}{N}\sum_{t=1}^{N} min\big(1, \frac{\pi_{k-1}(a_t|s_t)}{\pi_b(a_t|s_t)}\big)\Big(q_{k,t} - Q_k(s_t, a_t)\Big)^2$$

  Randomly initialize actor network π_k
  for $g \leftarrow 1$ **to** G **do**
    Update the actor policy using the batch gradient over $\mathcal{D}$:
    if $inverting\_gradient$ **then**

$$\nabla_a = \nabla_a . \begin{cases} \frac{a_{max} - a}{a_{max} - a_{min}} & \text{if } \nabla_a < 0 \\ \frac{a - a_{min}}{a_{max} - a_{min}}, & \text{otherwise} \end{cases}$$

    end

$$\nabla_{\theta^{\pi_k}} \pi_k = \frac{1}{N}\sum_{t=1}^{N} \nabla_a Q(s_t, a)|_{a=\pi_k(s_t)} \nabla_{\theta^{\pi_k}} \pi_k(s_t)$$

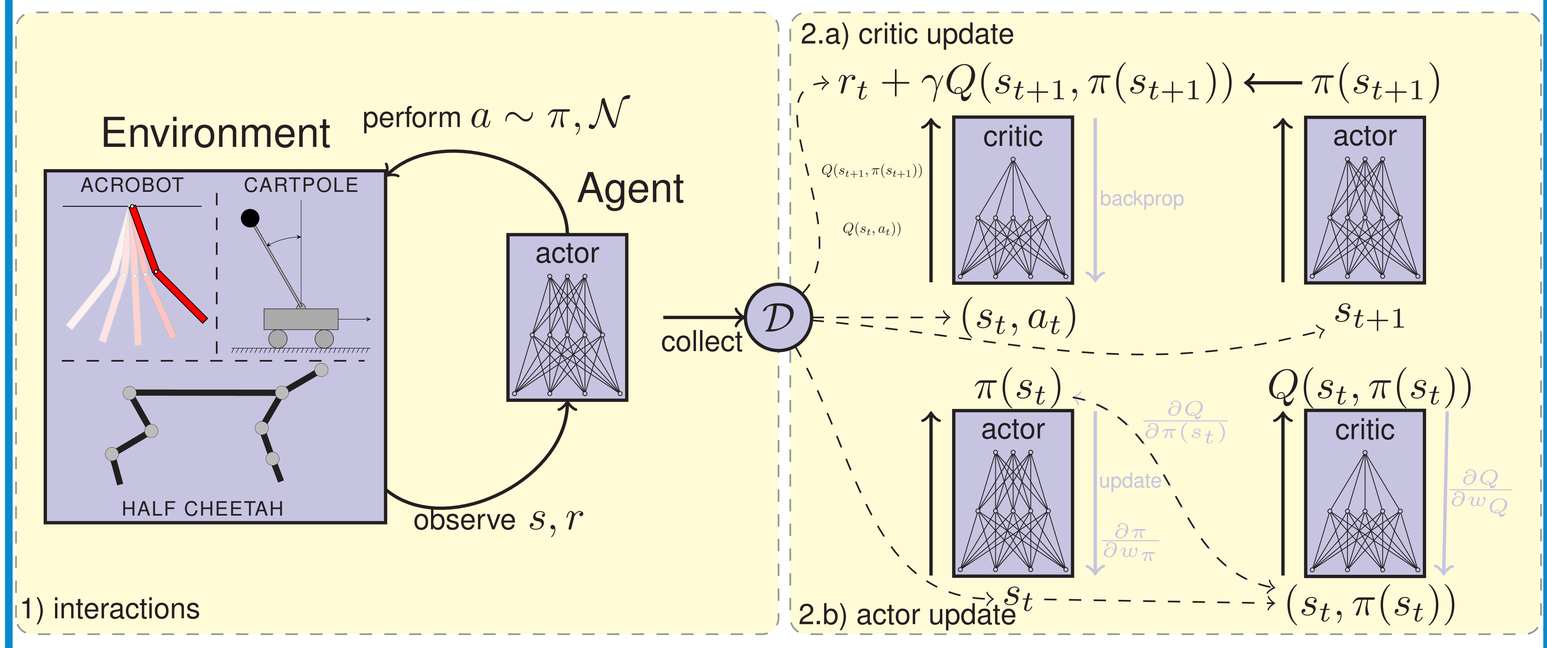
  end
end

Method

Actor-critic is a solution to handle **continuous actions spaces** in Markov Decision Process (S, A, R, T, γ). The critic learns the value-function Q_π that describes the expected return following π after the state s. The goal of the actor is to decide which action to take in which state, by finding the best policy according to Q :

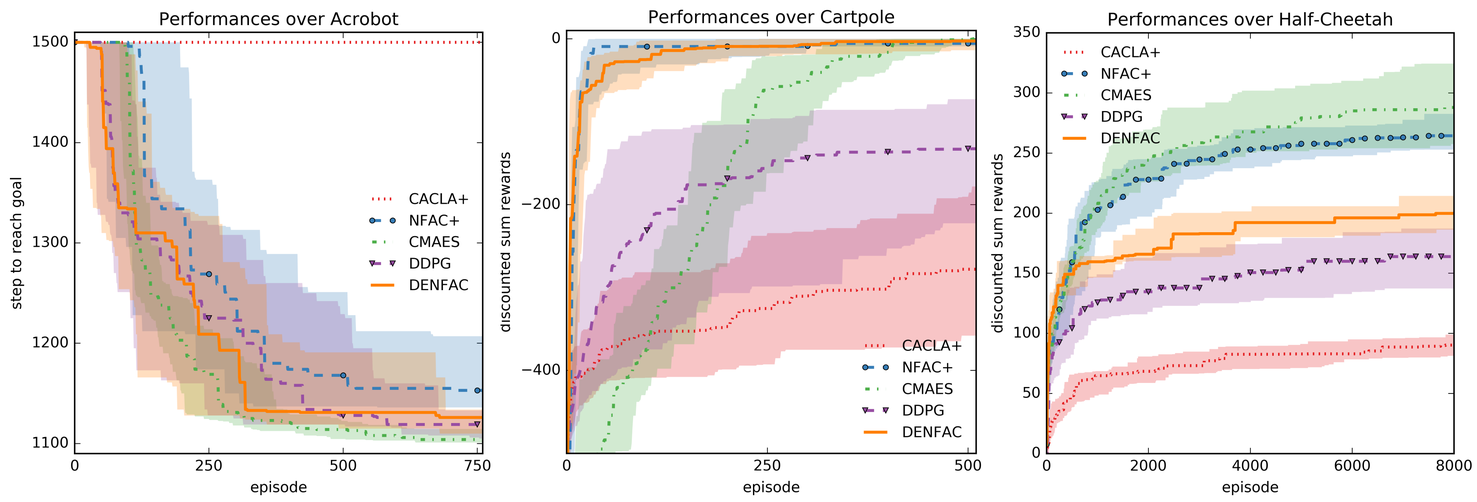
$$Q_{k+1} = \underset{Q \in \mathcal{F}_c}{\text{argmin}} \sum_{s_t, a_t, r_{t+1}, s_{t+1} \in \mathcal{D}} min\big(1, \frac{\pi_{k-1}(a_t|s_t)}{\pi_b(a_t|s_t)}\big)\Big[Q(s_t, a_t) - \Big(r_{t+1} + \gamma Q_k(s_{t+1}, \pi_k(s_{t+1}))\Big)\Big]^2,$$

$$\pi_{k+1} = \underset{\pi \in \mathcal{F}_a}{\text{argmax}} \sum_{s_t \in \mathcal{D}} Q_{k+1}\big(s_t, \pi_k(s_t)\big).$$



	FAC [3]	DDPG [2]	NFAC+ [4]	CACLA+	DENFAC
Offline & Batch	×		×		×
Off-policy	×	×			×
Fitted Critic	×		×		×
Actor updated through ∇Q	×	×			×
Learn Q	×	×		×	×
Reset Networks			×		×
Retrace					×
Target Networks		×			
Batch Normalization		×	× (+)	× (+)	×

Results



Median and quartile of the best registered performance in Acrobot (lower better) and Cartpole (higher better) environment during RL learning.

References

[1] Martin Riedmiller. Neural fitted Q iteration - First experiences with a data efficient neural Reinforcement Learning method. In *Lecture Notes in Computer Science*, volume 3720 LNAI, pages 317–328, 2005.

[2] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[3] András Antos, Rémi Munos, and Csaba Szepesvari. Fitted Q-iteration in continuous action-space MDPs. 2008.

[4] Matthieu Zimmer, Yann Boniface, and Alain Dutech. Neural Fitted Actor-Critic. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2016.

[5] John Schulman, Sergey Levine, Michael Jordan, and Pieter Abbeel. Trust Region Policy Optimization. *International Conference on Machine Learning*, page 16, 2015.

Futures Directions

In order to increase **data efficiency**, it should be analyzed if a First-In First-Out (FIFO) queue is the best choice for $\mathcal{D}$. Slowing down the change in the policy might increase his stability [5]. Finally, a better exploration strategy, that also take into account the previous collected data, is another possibility to deepen.

Source Code

The source code and data are available at :

`drl.gforge.inria.fr`