Machine Learning
Benchmarking neural-fitted Actor Critic with state of the art reinforcement learning algorithms*

H. Maathuis [a] M. Groefsema [b] S. Warmelink [b] T. Oosterhuis [a]

[a] *University of Groningen, 9747AG Groningen*

Abstract

1 Introduction

With reinforcement learning an agent is placed in an environment in a state where it can execute a number of actions. Each action the agent takes in a state entails a reward or a punishment for the agent as well as bringing it to a new state. By maximizing tt's reward the agent gradually learns the value of each state (either by keeping track of this value in a lookup table, or when the environment is to complex for this to be feasible, by approximating the value function for the environment with a function approximator such as a multi-layer perceptron - MLP), allowing it to autonomously learn the optimal policy for its environment. Reinforcement learning agents can learn complex behavior this way, such as solving a maze, or obeying traffic rules in a driving simulation.
When they were first developed reinforcement learning algorithms were designed to deal with discretized state and action spaces. In real life however state spaces aren't discritized and although action spaces can be, this is not ideal as different actions may require different levels of discritization to be accurate to get the desired reward. Furthermore the required level of discritization can not be known in all cases, and having to discover it is time intensive. This potential problem can be solved by keeping the state and action space continuous in the environment representation of the agent.
Several new reinforcement learning algorithms and adaptations of existing reinforcement algorithms were developed to model continuous state and action spaces. In this paper we will benchamark one such algorithm, named Neural Fitted Actor Critic (NFAC), which was develeped in August 2016 against two established (at the time of the writing of this paper) continuous reinforcement learning algorithms, named Continuous Actor Critic Learning Automaton (CACLA) and SARSA (State Action Reward State Action) with gradient descent (referred to in this paper as GD-SARSA).
First a more extensive background will be given on reinforcement learning in general including on model free reinforcement learning amd exploration, on function approximation with MLP's and on continuous state and action spaces, then the three algorithms that are compared in this paper are explained in detail, the implementation of these algorithms and the setup of the benchmarking experiment will be discussed, after that the benchmarking results will be presented and the contribution of NFAC as a reinforecemnt learning algorithm will be evaluated.

*In case of an extended abstract refer to the original paper in a footnote such as "The full paper has been published in *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 13–20, 2013." Also, please keep the title and authors exactly the same as the original.

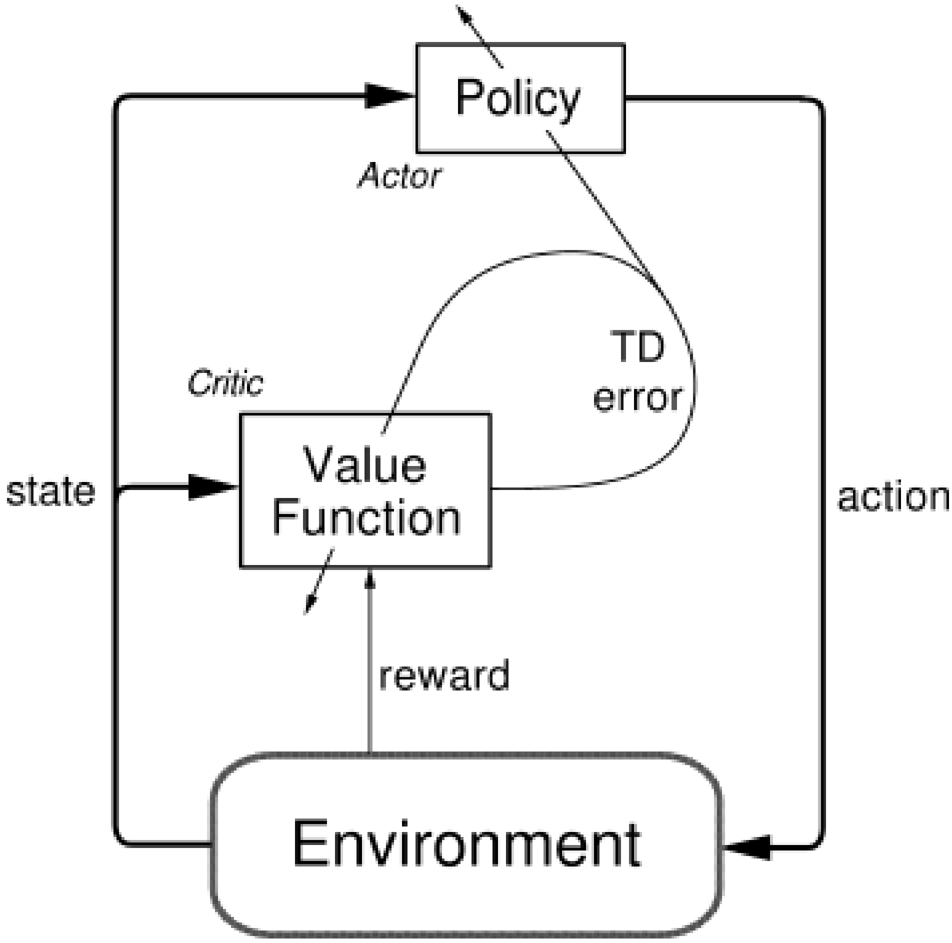

Figure 1: Actor Critic system. Reprinted from [2]

2 Background

3 CACLA

A well established algorithm in Reinforcement Learning is Continuous Actor Critic Learning Automaton (CACLA) [1]. The algorithm is capable of dealing with continuous state and action spaces. It implements an Actor Critic system in which both the Actor and Critic are operationalized using a Multi-Layer Perceptron. In the algorithm, the Actor is responsible for selecting the current action given the policy. The Critic is used in the calculation of the TD-error which drives the learning of the Actor. This method allows for a seperation between the representation of the policy and the value function. A visualization of the Actor-Critic system is shown in Figure 1

4 Experimental setup

5 Conclusions and further work

6 Contributions

References

[1] H. Van Hasselt and M.A. Wiering. Reinforcement learning in continuous action spaces. In *Proceedings of the IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, 2007*, pages 272–279. IEEE, 2007.

[2] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.