Describing the role of Artificial Neural Networks in Reinforcement Learning

Thilo Stegemann
University of Applied Sciences
Applied Computer Science
12459 Berlin, Wilhelminenhofstraße 75A
Email: t.stegemann@gmx.de

***Abstract*—TODO : Write an awesome abstract at the end!**

I. Introduction to Deep Reinforcement Learning

Artificial Intelligence is a huge, rewarding, rising and complex research field. More and more people are interested in AI every day. Students, researchers, economics, engineers, CEO's and investors are highly encouraged to use, understand and/or improve AI technologies. At some point in time an AI newcomer will get to the problems of Reinforcement Learning (RL) and therefore to Artificial Neural Networks (ANN's). Andrew Ng. describes AI as the new upcomming electricity: AI will change many different industries and it will have a huge general impact in everyday life. RL problems consider an agent-environment interaction framework. The agent (RL logic and learning algorithms) will interact with the environment (a Markov Decision Process). The agent will get rewards and states from the environment and the environment will get actions from the agent. The agent tries to learn optimal behaviour through trial and error attempts. The agent wants to know which actions in which states get the most long-term reward and fit this knowledge into a policy representation. A few main problems of this RL framework are:

- The agent only gets a numerical reward from the environment at the end of a decision-sequence.
  ∼ *Delayed Reward*
- How should the reward be assigned to the different steps of a decision-sequence?
  ∼ *Credit Assignment Problem*
- How to handle vast action- and state-spaces?
  ∼ *Generalization Problem*

A major goal of RL is to find a global optimal policy. A policy is a function which maps states to actions. This policy will additionally get a vector of parameters. The parameter-vector changes the policy output. This parametrisation of the policy function is called "function approximation" and Artificial Neural Networks are a really great approach for approximating a policy function. With this approximation the problem of vast action- and state-spaces can be solved. To optimise the parameter-vector methods like "Monte-Carlo Policy Gradient" or "Actor-Critic Policy Gradient" are used. Applications like TD-Gammon by Gerald Tesauro proved that learning complex strategy games with Artificial Neural Networks is possible and promising.

II. Related Work

A. Asynchronous Methods for Deep Reinforcement Learning

The scientists from Google DeepMind and Montreal Institute for Learning Algorithms introduced asynchronous deep learning algorithms. These asynchronous algorithms are based on four standard reinforcement learning algorithms: One-step Q-learning, one-step Sarsa, n-step Q-learning and advantage actor-critic. The paper explains the background of reinforcement learning and how the asynchronous reinforcement learning methods works. The study was approved by an experiment in an Atari 2600 evaluation environment . All four asynchronous algorithms where tested within the test environment . The Atari 2600 environment tests where used to compare the performance of the four algorithms. The main finding of this study is that all four asynchronous deep reinforcement learning algorithms are able to train neural network controllers on a variety of domains in a stable manner. In addition their results show that stable train ing of neural networks through reinforcement learning is possible with both value-based and policy-based methods, off-policy as well as on-policy methods, and in discrete as well as continuous domains.

B. Deep Reinforcement Learning with Double Q-Learning

Aim of this paper is to determine if the recent DQN (Deep Q Network) algorithm, which combines Q-learning with a deep neural network, suffers from substantial overestimations in some games in the Atari 2600 domain. Furthermore the Google DeepMind contributors point out how the Double Q-learning algorithm can be generalized to work with large-scale function approximation to successfully reduce the DQN overoptimism, resulting in more stable and reliable learning. Finally they propose a specific adaptation to the DQN algorithm and show that the resulting algorithm (Double DQN) not only reduces the observed overestimation, as they hypothesized, but that this also leads to much better performance on several Atari 2600 games.

C. Human-level controll through deep reinforcement learning

The paper is about how to reach human-level control through a deep Q-network, that can learn successful policies directly from high-dimensional sensory inputs using end-to-end reinforcement learning. The deep Q-network agent (in reinforcement learning an agent is the executing learning algorithm) is tested on the challenging classic Atari 2600 game environment. The result of this test demonstrated that the deep Q-network agent, receiving only the pixels and the game score as inputs, was able to surpass the performance of all previous algorithms and achieve a level comparable to that of a professional human games tester across a set of 49 games, using the same algorithm, network architecture and hyperparameters.

D. Mastering the game of Go with deep neural networks and tree search

E. Playing Atari with Deep Reinforcement Learning

- Cite [1]
- Cite [2]

F. Reinforcement Learning

G. Artificial Neural Networks

REFERENCES

[1] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," *CoRR*, vol. abs/1602.01783, 2016. [Online]. Available: http://arxiv.org/abs/1602.01783

[2] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," *CoRR*, vol. abs/1509.06461, 2015. [Online]. Available: http://arxiv.org/abs/1509.06461