PLAYING LUNAR LANDER WITH DEEP REINFORCEMENT LEARNING

Carlos Souza*
Georgia Institute of Technology
São Paulo, SP, Brazil
souza@gatech.edu

June 28th, 2019

ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

***K*eywords** Reinforcement learning · Deep Reinforcement Learning · Deep Q-Networks

1 Introduction

Reinforcement learning (RL) theory provides mathematical background and classic algorithms to enable agents to learn optimally control an environment. However, to successfuly apply those algorithms in problems with real-world complexity, several challenges must be tackled, must notably i) how to derive efficient representations of the environment from high-dimensional sensory inputs, and ii) how to use these to generalize past experience to new situation (Mnih et al. 2015).

Recent advances in deep neural networks originated a new type of agent, known as Deep Q-Network agent, capable of overcoming these challenges, which enabled RL algorithms to perform effectively. All the many recent successes in applying RL to complex sequential decision-making problems were kick-started by the Deep Q-Network algorithm (DQN; Mnih et al. 2015). Since then, many extensions have been proposed to improve its stability and/or speed. Double DQN (DDQN; Hasselt, Guez, and Silver 2016) addresses the problem of overestimation of action values, by decomposing the max operation in the target into action selection and action evaluation. The Dueling Network architecture (Dueling DDQN; Wang et al. 2016) better generalize learning accross actions by proposing an architecture that explicitly separates the representation of state values and (state-dependent) action advantages. Prioritized Experience Replay (PER; Schaul et al. 2016) improves data efficiency, by replaying more of- ten transitions from which there is more to learn.

References

[1] Volodymyr Mnih et al. "Human-level control through deep reinforcement learning". In: *Nature* 518.7540 (Feb. 2015), pp. 529–533. ISSN: 00280836. URL: http://dx.doi.org/10.1038/nature14236.

*Latest Git hash: INCLUDE LATEST GITHASH

[2] Hado van Hasselt, Arthur Guez, and David Silver. “Deep Reinforcement Learning with Double Q-Learning”. In: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI’16. Phoenix, Arizona: AAAI Press, 2016, pp. 2094–2100. URL: `http://dl.acm.org/citation.cfm?id=3016100.3016191`.

[3] Tom Schaul et al. “Prioritized Experience Replay”. In: *CoRR* abs/1511.05952 (2016).

[4] Ziyu Wang et al. “Dueling Network Architectures for Deep Reinforcement Learning”. In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1995–2003. URL: `http://proceedings.mlr.press/v48/wangf16.html`.