Project Proposal: Zap Q-Learning and its applications in Arcade Learning Environment (ALE)

Xuyin Zhu; Muchen Li; Xuanzhao Dong

### MOTIVATION & INTRODUCTION

Learning value or Q function plays an integral role in Reinforcement learning. However, all learning control methods face a dilemma: they seek to learn action values conditional on subsequent optimal behavior based on the fact that they first need to behave sub-optimally in order to explore all actions and hence find the optimal ones. As a result, different ways to learn optimal policy while exploring lead to different learning methods. As opposed to the “on-policy” learning approach, which chooses to learn action value function based on near-optimal policy, “off-policy learning’’ consists of two separate policies, one (target policy) that is evaluated and improved towards the optimal policy and another one (behavior policy) that explores and generates behavior. In general, “off-policy learning’’ methods are of greater variance and slower to converge [1]. In an attempt to solve this problem, many improvements have been proposed such as Watkins’ algorithm with a “polynomial learning rate” [2] and Speedy Q-Learning algorithm [3].

As a special case based on the stochastic Newton-Raphson algorithm, Zap Q-learning chooses to use matrix gain techniques to speed up convergence of traditional Watkins’ Q-learning [4] [5]. Furthermore, under two time-scale implementations, the result of Zap Q-learning algorithm will resemble and implementation of Newton-Raphson [5].

The goal of this project is to verify that Zap Q-learning converges faster than Deep Q-Networks (DQN) by comparing their performance on several Atari 2600 computer games. In previous work DQN has demonstrated strong ability to master difficult control policies for those games [6]. We plan to train agents using Zap Q-learning and DQN that can play Atari games based on in the Arcade Learning Environment(ALE), a simple reinforcement learning testbed that allows researchers and hobbyists to develop AI agents for Atari 2600 games [7]. Now ALE has been included in OpenAI Gym and researchers can easily access the environment by importing a Python library [8].

### Method

We will use OpenAI Gym ATARI as the environment to train two agents, one by Zap Q-learning and the other by DQN. Both agents are trained to play 6 games in Atari: Breakout, Enduro, Pong, Q\*bert, Seaquest and Space Invaders. Normalizing scores will be used to evaluate the performance of our learned policy across games [7]. Training time and/or episodes to converge will be used to evaluate the speed of convergence.

# References

|  |  |
| --- | --- |
| [1] | Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, Bradford Books, 2018. |
| [2] | E. Even-Dar and Y. Mansour, "Learning Rates for Q-learning," *Journal of Machine Learning Research,* no. 5, pp. 1-25, 2003. |
| [3] | Azar, M.G., Munos, R., Ghavamzadeh, M., & Kappen, H.J., "Speedy Q-Learning," *NIPS,* 2011. |
| [4] | Watkins, C.J.C.H., Dayan, P., "Q-learning," *Mach Learn,* no. 8, pp. 279-292, 1992. |
| [5] | Adithya M Devraj, Sean Meyn, "Zap Q-Learning," *NIPS,* 2017. |
| [6] | V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra and M. Riedmiller, "Playing Atari with Deep Reinforcement Learning," *NIPS,* 2013. |
| [7] | M. G. Bellemare, Y. Naddaf and J. Veness, "The Arcade Learning Environment: An Evaluation Platform for General Agents," *Journal of Artificial Intelligence Research,* no. 47, pp. 253-279, 2013. |
| [8] | "Atari," [Online]. Available: https://www.gymlibrary.dev/environments/atari/#references. [Accessed 9 11 2022]. |