OPTIMIZING EXPLORATION PARAMETER IN DUELING DEEP Q-NETWORKS FOR COMPLEX GAMING ENVIRONMENT

MUHAMMAD SHEHRYAR KHAN

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Master of Computer Science

School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia

MAY 2019

DEDICATION

This thesis is dedicated to my Father who aspired me to continue my studies,
my Mother who always kept me in her prayers.

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many researchers and academicians. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my thesis supervisor, Associate Professor Dr. Siti Zaiton Mohd Hashim, for encouragement, advice and guidance. Without her continued support and interest, this thesis would not have been the same as presented here.

I am also thankful to Dr. Maqsood and Dr. Rashid who provided me with a workstation to meet the requirements of research and Dr.Waqar for having fruitful discussions on my thesis.

I express my gratitude to my parents and brothers who encouraged me in every situation. My sincere appreciation also extends to all my friends, colleagues and family members who helped and supported me.

ABSTRACT

Reinforcement Learning is being used to solve various tasks. A Complex Environment is a recent problem at hand for Reinforcement Learning, which employs an Agent who interacts with the surroundings and learns to solve whatever task has to be done. To solve a Complex Environment efficiently using a Reinforcement Learning Agent, a lot of parameters are to be kept in perspective. Every action that the Agent takes has a consequence in the form of a Reward Function. Based on the value of this Reward Function, our Agent develops a Policy to solve the Environment. The Policy is generally developed to maximize the Reward Functions. The Optimal Policy employs an Exploration Strategy which is used by the Agent. Reinforcement Learning Architectures are relying on the Policy and Exploration Strategy of the Agent to solve the Environment efficiently. This research is based upon two parts. Firstly, the optimization of a Deep Reinforcement Learning Architecture “Dueling Deep Q-Network” is conducted by improving its Exploration strategy. It combines a recent and novel Exploration technique, Curiosity Driven Intrinsic Motivation, with the Dueling DQN. The performance of this Curious Dueling DQN is checked by comparing it with the existing Dueling DQN. Secondly, the performance of the Curious Dueling DQN is validated against Noisy Dueling DQN, a combination of Dueling DQN with another recent exploration strategy called Noisy Nets, hence, finding an optimal exploration strategy. The performance of both solutions is evaluated in the environment of Super Mario Bros based on Mean Score and Estimation Loss. The proposed model improves the Mean Score by 3 folds, while the loss is increased by 28%.

ABSTRAK

Reinforcement Learning digunakan untuk menyelesaikan pelbagai masalah. Perselitaran yg kompleks adalah satu masalah baru dalang Reinforcement Learning, yang menggunakkan ejen bagi berinteraksi dengen persekitaran dan belajar menyelesaikan apa jua tugasen yang perlu difuat. Bagi menyelesaikan persekitaran kompleks dengen lebih cekap, banyak parameter perlu difelit. Setiap tindakan ejen aken mempengerahu Fungsi Ganjaran. Berdasarkan nilai ganjaran, Ejen yang dibangukan menghasilkan satu polisi untuk menyelesaikan penselataren terselat. Polisi itu secara umumay memaksimumkan fungsi ganjaran. Strategi penjelajeran digune ken untulemen dapatuan polisi optimal. Rangka kerja Reinforcement Learning bergantung kepada polisi dan strategi penjelajeran ejen. Penyelidikan ini terbenag kepada dua. Pertama, pensoptinuman Deep Reinforcement Learning ”Dueling Deep Q-Network” difuat dengan memperbaiki strategi penjelajeran. Ia menggabungkuan teknik penjelajeran Curiosity Driven Intrinsic Motivation dengen Dueling DQN. Prestasi Curious Dueling DQN dibandingkan dengan Dueling DQN yang ada. Kedua, prestasi Curious Dueling DQN dibandingkan dengan Noisy Dueling DQN, gabungan Dueling DQN den strategi penjelajeran yang terluhi Noisy Nets, yang menghasilkan strategi penjelajeran yg optimal. Prestasi kedua-dua cadagan penyelesaian disalikan delan persekitaran Super Mario berdasarkan Skor Min dan Estimation Loss. Model yang dicadangkan meningkatkan Skor Min sebanyak 3 lipatan, manakala loss meningkat sebanyak 28%.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DQN	-	Deep Q-Networks
RL	-	Reinforcement Learning
AI	-	Artificial Intelligence
ALE	-	Arcade Learning Environment
MDP	-	Markov Decision Process
SARSA	-	State-Action-Reward-State-Action
DDQN	-	Dueling Deep Q-Networks
A3C	-	Asynchronous Actor-Critic
CPU	-	Central Processing Unit
GPU	-	Graphics Processing Unit
ICM	-	Intrinsic Curiosity Module
TRPO	-	Trust Region Policy Optimization

CHAPTER 1

INTRODUCTION

1.1 Motivation

In this technology dependent era of digital gadgets, Artificial Intelligence (AI) has gained immense importance. It is being used in every business enterprise, applications, operating systems and websites. The use of AI is valid as it makes our life easier. It provides us those services which were unheard of in the past. With the rapid growth in AI, the appropriate decision making mechanism is a dire need of the hour. Reinforcement Learning is one of those domains in AI which tackles this aspect.

Reinforcement Learning empowers the concept of cognition (Huhns and Singh, 1998) as it learns things and makes decisions in the most humanized way possible. It uses trial and error to learn from its experiences which is the same as a normal human being. Humans interact with an environment based upon what they like and dislike about it. They tend to go for the experiences which provided them with some amusement and they never think about doing something which they rather disliked. Reinforcement learning works on similar principle.

Reinforcement Learning has number of applications and advantages based upon its improvements over the past few years. It is being applied to a number of fields for the ease of work without requiring a human worker. It is being applied in Robotics (Kober et. al, 2013), Web System configuration (Bu et. al, 2009), Optimizing reactions in Chemistry (Zhou et. al, 2017), Cluster Management (Mao et. al, 2016), Traffic Light Control networks (Arel et. al, 2010), Bidding and Advertising (Jin et. al, 2018) and most prominently in solving Gaming Environments (Silver et. al, 2016).

Reinforcement learning when combined with Deep Neural Networks is further modified. The decision making capability of Reinforcement learning combined with learning capability of Deep Neural Networks can solve Complex Real World problems. Deep Reinforcement Learning is capable of performing on par with humans and even exceeding it to have Super-Human Performance (Mnih et. al, 2013).

Google's DeepMind has been very active in Deep Reinforcement Learning research. In fact, DeepMind's AlphaGo Zero (Silver et. al, 2017), the Deep Q Learning based algorithm, was able to defeat the world champion in the game of GO and achieved Super-Human Performance (Mnih et. al, 2013). Similarly, a lot of other Deep RL Frameworks have been developed since then, which are performing even better than the DQN approach.

The algorithms are applied to specific characters which solve the game or environment, called "Reinforcement Learning Agents". A lot of environments have been developed for the research purposes, to check the performance of our RL Agent. Many of these Benchmarks are related to Complex Environments like games or situations in which our Agent has to solve them efficiently. Arcade Learning Environment (ALE) was developed which is based on the Atari 2600 (Bellamare et. al, 2013). Super Mario Bros is another environment in which a lot of research has been done (Pathak et. al, 2017). The RL Agents can be trained and tested on these Benchmarks to check their performance.

Such a Deep RL Agent was proposed called 'Dueling Deep Q-Network'. This Model of RL is an extension of DQN in which they break the state value function and action advantage function into two separate streams. This algorithm was trained and tested on ALE and it outperformed the state of the art agent (Wang et. al, 2016). But the Policy used in this approach was the traditional one which learns by maximizing the reward functions. Changing the Policy which applies Optimal Exploration techniques can further improve the performance of the Dueling Networks (Pathak et. al, 2017).

There are many state of the art Exploration techniques which have come up recently. Curiosity Driven Intrinsic Motivation (Pathak et. al, 2017) is one of them which don’t focus on the extrinsic rewards of the environment but rather its own internal curiosity to explore the environment, leading to much efficient results. Noisy Networks is a Noise Driven exploration strategy which adds learned Perturbations to the weights, thus increasing the exploration (Fortunato et. al, 2017). Thus, enhancing the exploration parameter can give a performance boost.

1.2 Problem Background

Recently, a lot of work has been done on Deep Reinforcement learning algorithms and frameworks. A lot of new Models have come forward with each tackling a specific parameter and improving it further. Some of these Models have also been combined together to form Hybrids which tackle multiple parameters together, hence enhancing the performance of these Models (Mnih et. al, 2016).

Many Deep RL Models have been developed recently. Deep Q Learning (Mnih et. al, 2015), Double Q Learning (Hasselt et. al, 2016), Asynchronous Actor-Critic (Mnih et. al, 2016) and Deep Dueling Network Architectures (Wang et. al, 2016) are some of them. All of these models have Super-Human Performance (Mnih et. al, 2013) and tackle different perspectives. But these algorithms focus much on upgradation of the Network Architecture instead of local parameters. However, improving these parameters can also result in better performance.

Each parameter in Reinforcement Learning has different impact on the learning problem. RL Architectures tackle the state-action functions to improve the response of the agent (Tsay et. al, 2011), optimize the policy by reward shaping (Harutyunyan et. al, 2015), optimize the value functions to maximize the rewards (Mohan and Laird, 2010), optimize search space (Brys et. al, 2014) and improve exploration strategies (Kormelink et. al ,2018). Some of these parameters are related together and changing or improving one of them impacts the other parameter. While each parameter tackles different perspectives, all of them contribute to the overall performance of the agent.

The performance of the agent depends upon the timing and sample efficiency. To solve some complex problem, the RL Agent first needs to be trained on that environment using the basic Reinforcement Learning approach. The agent is given a set of actions through which it can interact with the environment. Using these actions, the agent goes through multiple iterations ranging from hundreds to thousands, to learn the working of the environment. The amount of time and the iterations it takes accounts for the timing efficiency. Based upon these samples, the agent forms a policy to solve the environment. This accounts for the sample efficiency.

The policy of the agent depends on the exploration strategy used by it. Exploration means how the agent likes to go about interacting with the environment. It can go for the rewards like coins, it can go straight for the finish line, can explore new hidden areas, can kill enemies etc. The type of exploration a policy employs in turn affects the overall performance.

Some exploration techniques have been proposed in the recent past (Kormelink et. al, 2018). These exploration techniques are usually based on the extrinsic rewards present in the environment. But these external rewards are sparse and in turn affect the exploration of the agent. These techniques are usually used in some of the Architectures. Recently, a novel technique was proposed which went for the curiosity factor of the agent which comes from the intrinsic motivation to explore different areas in the environment (Pathak et. al, 2017). It accounted for the sparse rewards and time efficiency and involved more exploration. This approach was combined with policy based A3C Architecture (Mnih et. al, 2016), and it was shown that it clearly lifted the performance of the existing A3C. Another approach, Noisy Networks (Fortunato et. al, 2017) was proposed which adds measured noise in the weights of the agent, and uses noise to drive it to explore further.

The effect of Intrinsic Motivation was not tested further on other Deep RL Architectures. It has not been combined with a value based algorithm yet. Noisy Nets was tested on some, and it improved the performance, but it still remained to seek which of these techniques is the better one. It is hence clear that no attention was given to test how well intrinsic motivation works on other models. Moreover, the optimal strategy for exploration was not found out of these latest approaches. Finding the optimal technique will result in further improvement of any Architecture which comes forward in the future.

1.3 Problem Statement

The performance of the agent is optimized by upgrading the frameworks, which enhances the timing and sample efficiency significantly, but the parameters are kept out of perspective, which can further improve the performance of the agent.

This research investigates the combination of a Deep Reinforcement Learning Architecture, Dueling DQN, with a recent Exploration strategy called Curiosity Driven Intrinsic Motivation, to optimize the performance of our Agent to solve the environment efficiently. The results of the Curious Dueling DQN are compared with the existing architecture to check the performance optimization. Moreover, another exploration technique, Noisy networks combined with Dueling DQN, which is state of the art, is compared with Curious Dueling DQN, thus to evaluate the performance and to find the optimum strategy out of the two.

1.4 Research Aim

The aim of this research is to produce an effective Deep Reinforcement Learning Agent which can solve a Complex Environment efficiently using the enhanced exploration approach of Curiosity Driven Intrinsic Motivation.

1.5 Research Objectives

The objectives of the research are:

1. To propose Curious Dueling DQN, an enhancement of Dueling DQN by optimizing its exploration parameter using Curiosity Driven Intrinsic Motivation.
2. To evaluate the performance of Curious Dueling DQN against state of the art Noisy Nets Dueling DQN.

1.6 Research Scope

The scope of this research is:

1. Based upon the Deep Reinforcement Learning Architecture, this research focuses to improve the Exploration perspective of the learning.
2. The Complex Environment of 2d Super Mario game is used for Experiment setup and testing of the Agent.
3. Performance Evaluation is done by comparing it with the Existing Dueling Network Architecture and state of the art Noisy Dueling DQN.
4. The Optimum Exploration strategy is found by comparing the two approaches.

1.7 Research Contribution

This Research contributes to the Performance Optimization of the Reinforcement learning agent as follows:

1. It applies Dueling DQN to Super Mario Environment, which has not been used for this Architecture before.

2. It optimizes the performance of Dueling DQN by applying recent exploration strategies.

3. It finds the Optimum Exploration Strategy by comparison of Curious Dueling DQN and Noisy Dueling DQN which has not been done before.

1.8 Report Organization

The research proposal is comprised of six chapters, each chapter giving specific details about the proposed research.

Chapter 1 is the Introduction. It gives the basic idea of what the research is about. It includes the introduction, problem statement, background, objects, scope and significance of the research.

Chapter 2 is Literature review. It focuses on the previously done work related to this research. It discusses Markov Decision Process, Models, Policies, Reinforcement Learning Models, Deep Reinforcement Learning and its Models, different Exploration Strategies and Complex Environments.

Chapter 3 is Methodology. It gives the Operational Framework to solve the problem. It also describes the Environment is which our Agent is trained and tested, and the software used to implement the basic functionality.

Chapter 4 gives the Research Design and Implementation. It explains in detail the flow of each objective and how it was implemented. The implementation details contain flowcharts and pseudocodes. The specifications and working of the Super Mario environment are also given.

Chapter 5 is the Analysis and Discussion of the results. It shows all the performance graphs of training and testing of the agent. These graphs are used for validation and comparison of the techniques.

REFERENCES

Aronson, J. W. (1995). Analysis of a randomized greedy matching algorithm.

Arel, I., Liu, C., Urbanik, T., & Kohls, A. G. (2010). Reinforcement learning-based multi-agent system for network traffic signal control. *IET Intelligent Transport Systems*, *4*(2), 128-135.

Bellemare, M. G., Naddaf, Y., Veness, J., & Bowling, M. (2013). The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, *47*, 253-279.

Bellemare, M. G., Dabney, W., & Munos, R. (2017). A distributional perspective on reinforcement learning. *arXiv preprint arXiv:1707.06887*.

Bu, X., Rao, J., & Xu, C. Z. (2009, June). A reinforcement learning approach to online web systems auto-configuration. In *Distributed Computing Systems, 2009. ICDCS'09. 29th IEEE International Conference on* (pp. 2-11). IEEE.

Brys, T., Harutyunyan, A., Vrancx, P., Taylor, M. E., Kudenko, D., & Nowé, A. (2014, July). Multi-objectivization of reinforcement learning problems by reward shaping. In *Neural Networks (IJCNN), 2014 International Joint Conference on*(pp. 2315-2322). IEEE.

Carmel, D., & Markovitch, S. (1997, August). Exploration and adaptation in multiagent systems: A model-based approach. In IJCAI (1) (pp. 606-611).

Cohn, D. A. (1994). Neural network exploration using optimal experiment design. In Advances in neural information processing systems (pp. 679-686).

Fortunato, M., Azar, M. G., Piot, B., Menick, J., Osband, I., Graves, A., ... & Blundell, C. (2017). Noisy networks for exploration. *arXiv preprint arXiv:1706.10295*.

Handa, H. (2011, June). Dimensionality reduction of scene and enemy information in Mario. In *Evolutionary Computation (CEC), 2011 IEEE Congress on* (pp. 1515-1520). IEEE.

Harutyunyan, A., Brys, T., Vrancx, P., & Nowé, A. (2015, May). Shaping mario with human advice. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems* (pp. 1913-1914). International Foundation for Autonomous Agents and Multiagent Systems.

Hasselt, H. V. (2010). Double Q-learning. In *Advances in Neural Information Processing Systems* (pp. 2613-2621).

Huhns, M. N., & Singh, M. P. (1998). Cognitive agents. *IEEE Internet computing*, *2*(6), 87-89.

Jin, J., Song, C., Li, H., Gai, K., Wang, J., & Zhang, W. (2018). Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising. *arXiv preprint arXiv:1802.09756*.

Kauten, C. (2018). Super Mario Bros for OpenAI Gym. GitHub Repository, https://github.com/Kautenja/gym-super-mario-bros (as of May 2019)

Kauten, C. (2018). Playing Mario with Deep Reinforcement Learning, GitHub Repository, https://github.com/Kautenja/playing-mario-with-deep-reinforcement-learning (as of May 2019)

Kober, J., Bagnell, J. A., & Peters, J. (2013). Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, *32*(11), 1238-1274.

Konda, V. R., & Tsitsiklis, J. N. (2000). Actor-critic algorithms. In *Advances in neural information processing systems* (pp. 1008-1014).

Kormelink, J. G., Drugan, M. M., & Wiering, M. A. (2018). Exploration Methods for Connectionist Q-learning in Bomberman. In *ICAART (2)* (pp. 355-362).

Krening, S., Harrison, B., Feigh, K. M., Isbell, C., & Thomaz, A. (2016, May). Object-focused advice in reinforcement learning. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems* (pp. 1447-1448). International Foundation for Autonomous Agents and Multiagent Systems.

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Machine Learning Proceedings 1994* (pp. 157-163).

Mao, H., Alizadeh, M., Menache, I., & Kandula, S. (2016, November). Resource management with deep reinforcement learning. In *Proceedings of the 15th ACM Workshop on Hot Topics in Networks* (pp. 50-56). ACM.

Mohan, S., & Laird, J. (2010). Relational reinforcement learning in infinite mario. *Ann Arbor*, *1001*, 48109-2121.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... & Kavukcuoglu, K. (2016, June). Asynchronous methods for deep reinforcement learning. In *International conference on machine learning* (pp. 1928-1937).

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Ortega, J., Shaker, N., Togelius, J., & Yannakakis, G. N. (2013). Imitating human playing styles in super mario bros. *Entertainment Computing*, *4*(2), 93-104.

Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017, May). Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning (ICML)* (Vol. 2017).

Sarjant, S., Pfahringer, B., Driessens, K., & Smith, T. (2011, August). Using the online cross-entropy method to learn relational policies for playing different games. In *Computational Intelligence and Games (CIG), 2011 IEEE Conference on* (pp. 182-189). IEEE.

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv preprint arXiv:1511.05952*.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... & Dieleman, S. (2016). Mastering the game of Go with deep neural networks and tree search. *nature*, *529*(7587), 484.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Chen, Y. (2017). Mastering the game of Go without human knowledge. *Nature*, *550*(7676), 354.

Sutton, R. S. (1992). Introduction: The challenge of reinforcement learning. In *Reinforcement Learning* (pp. 1-3). Springer, Boston, MA.

Thrun, S. B. (1992). Efficient exploration in reinforcement learning.

Tsay, J. J., Chen, C. C., & Hsu, J. J. (2011, November). Evolving intelligent mario controller by reinforcement learning. In *Technologies and Applications of Artificial Intelligence (TAAI), 2011 International Conference on* (pp. 266-272). IEEE.

Wang, Z., Schaul, T., Hessel, M., Van Hasselt, H., Lanctot, M., & De Freitas, N. (2016). Dueling network architectures for deep reinforcement learning. *arXiv preprint arXiv:1511.06581*.

Watkins, C. J., & Dayan, P. (1992). Q-learning. Machine learning, 8(3-4), 279-292.

Van Hasselt, H., Guez, A., & Silver, D. (2016, February). Deep Reinforcement Learning with Double Q-Learning. In *AAAI* (Vol. 2, p. 5).

Zhou, Z., Li, X., & Zare, R. N. (2017). Optimizing chemical reactions with deep reinforcement learning. *ACS central science*, *3*(12), 1337-1344.