https://doi.org/ 10.48047/AFJBS.6.5.2024. 8986-8999

African Journal of Biological Sciences

Unveiling the Foundations and Frontiers of Reinforcement Learning

Ayesha Agrawal*, Computer Science & Engineering, Mody University of Science & TechnologyLakshmangarh, India.

VinodMaan , Computer Science & Engineering, Mody University of Science & Technology, Lakshmangarh, India.

Correspomding author: aishaagrawal33@gmail.com

Article History
Volume 6, Issue 5, 2024
Received: 15 May 2024
Accepted: 02 Jun 2024
doi: *10.48047/AFJBS.6.5.2024. 8986-8999*

Abstract:The field of reinforcement learning (RL) has been receiving significant attention due to the emergence of ambitious endeavours that include automated arm deception, 1v1 Dota, and Atari games. This expansion is consistent with the continued success of supervised deep learning, as demonstrated most significantly by the outcome of the 2012 ImageNet classification event. Neural networks with deep structures have recently been widely used in academics to address challenging issues; including comprehending sophisticated behaviours in changing contexts. As a branch of artificial intelligence, reinforcement learning (RL) presents a viable path towards achieving highly intelligent robotic behaviour. In contrast to supervised learning, which trains networks using labelled datasets, reinforcement learning (RL) is more appropriate for situations in which explicit input is not available since it incorporates experimental encounters with the environment. The fundamental ideas of reinforcement learning and how it is utilised in a variety of fields, such as games for computers, robots, and stock market assessment, are reviewed in this article, along with the different techniques for learning used in multi-agent scenarios. It also goes over how to formulate and solve difficulties related to reinforcement learning, offering perspective on the prospects and difficulties present in this quickly developing discipline.

Keywords—Reinforcement learning; Machine learning; Artificial learning; programming; Multi-armed bandits; the Markov decision process

1.INTRODUCTION

Reinforcement learning [1] has had a recent explosion with the emergence of numerous ambitious projects. Atari games from the 1980s, 1v1 Dota, and robotic arm manipulation are some notable significant applications of reinforcement learning. Furthermore, the rewards for supervised deep learning [2] have kept coming in, as seen by the 2012 Image Net classification competition. Furthermore, a wide spectrum of academics have been working with deep neural networks to solve a number of significant new problems, like understanding intelligent behaviors and attitudes in a complex dynamic environment. As a branch of machine learning, reinforcement learning is regarded as most advantageous and practical approaches to accomplish a high level of intelligence in robotic behavior [3]. It is regarded as a subfield of machine learning [3]. In machine learning projects and applications ([5], [6]), almost all researchers use supervised learning [4], in which a neural network model is given an input while precisely knowing what output it should produce; gradients are then calculated by using the back propagation method to train the network to output the results. The constructed dataset should have aspects of supervised learning. Given the quantity of pieces that must be collected for the dataset, this part of the procedure is not always simple to complete. Since an agent can never outperform a human at a game, the neural network will also be trained to mimic human player records and behaviors in a clear manner. After then, reinforcement learning ([7], [8], and [9]) is still the best option for dealing with the problem of an agent outperforming a human player and learning to play the game independently without assistance from a human. The general structure of supervised learning and reinforcement learning work similarly in that an input frame is continuously passed through certain neural network models and the network then produces an output action. It is not possible to identify the goal label in reinforcement learning, unlike supervised learning, because of the lack of a dataset.

The application of machine learning that extracts capabilities from data is called supervised learning. The information dataset used in this process is divided into two categories: the testing model with the dataset and the preparation information. The yield elements that need to be predicted or described are included in the prepared information, and the testing model makes use of the hidden tried information to determine the model's accuracy. Stated differently, an agent is employed to approximate the target values for every piece of data, after which it is stored in the memory for future usage.

The agent does not get impressions in unsupervised learning. By receiving and providing the precise contribution as a set of instructions, the agent must learn on its own. Since groups distorted the measurement setup, clusters are used here to effectively depict the

data. For this reason, this learning technique emphasizes the reduction phase and is mostly employed for bunching. Principal Component Analysis and K-Means Clustering are the two core techniques for dimensionality reduction and clustering.

In reinforcement learning, the decisions made are determined by the actions taken as a consequence. Therefore, the basis of Reinforcement Learning is the connection between an agent performing an action and the environment's response, which can be either positive or negative. Through experimentation and condition-based collaboration, the goal is achieved. Reinforcement learning combines the domains of supervised learning and dynamic programming to become a potent machine learning system. Strengthening numerous fields, including game theory, statistics, genetic algorithms, robotics, information theory, game theory, simulation-based optimization, and control theory, have successfully used learning.

The Recommender system learns from its users, particularly online users who can alter their websites to suit their preferences and needs. Users have two options for mining their information: collaborative suggestion and content-based recommendation. It lets the users to reconstruct their data with skillful, perceptive, and original recommendations.

2.LITERATURE REVIEW

As per Samuel, machine learning refers to the field of research where computers can acquire knowledge to learn without explicit programming. Anderson (1986) asserts that machine learning is associated with frameworks that enhance their performance as a result. According to Marsland (2015), machines can learn to solve specific problems on their own through machine learning. According to author (2018), machine learning stands as one of the prominent methods for handling approaches that carry out fault management and network data analysis. According to Lewis et al. (2008), reinforcement learning in the context of artificial intelligence can ideally resolve problems by collaborating with its environment and additionally by altering its control structures. According to Busoniu et al. (2009), reinforcement learning is used to discover the most effective configuration that maximizes the overall reward. It originated primarily from the tool of consistent learning and was also influenced by environmental rewards and penalties. According to Flore (2015), an agent must learn how to solve reinforcement learning problems through trial-and-error interfaces in a dynamic environment. As stated by Sutton in 1992 Experimentation and delayed results are always necessary for reinforcement learning. According to Tiwana et al. (2014), Fourth Generation (4G) Networks can be improved by using a framework for Quality of Services (QoS) based on reinforcement learning. Hou et al. (2017) presented a method that effectively addresses choice problems that are progressively improved. The Markov decision process is

used in programming, which is associated with reinforcement learning. According to Olafati (2006), algorithms for reinforcement learning are employed to reflect social and inevitable procedures. It makes use of state activity learning parameters, which exponentially increase the factor components. Reinforcement learning, according to Vidhate et al. (2016), is a methodology used to improve multi-agent learning and also a framework with novel tactics that not only validate simulated outcomes but also yield additional results. In 2017, Carluchoet al. suggested an adaptable PID control for portable robots based on a steady Q learning approach. Although prior knowledge is not necessary for this operation, it can comprehend procedures that deviate from traditional methods. It was also suggested by Hung et al. (2017) that small flocking fixed-wing UAVs be given Q learning methods to help them learn to navigate through a dynamic and unpredictable environment.

3. REINFORCEMENT LEARNING

Reinforcement learning is a supplementary form of learning method in which an agent examines the space of potential actions and provides feedback on the choices made. These options are discovered by exploratory communications in a dynamic environment. It is also distinguished by drawing comparisons between the issue and several machine learning research controls. Reinforcement learning problems can be solved using two main methods. Examining the space between activities to find one that works well in the surroundings is the first methodology. According to Sutton and Barto's 1998 study, this concept has been used to genetic computations, programming, and other more creative search techniques. The second methodology calculates the usefulness of engaging in activities under real-world settings by applying factual strategies and dynamic programming techniques.

3.1Architecture of ReinforcementLearning

Use either SI (MKS) or CGS as primary units. (SI units The fundamental framework of Reinforcement Learning is depicted in Fig. 1, where an agent uses sensor data to interact with a scenario, altering the surroundings and earning a reward for its efforts. The states serve as the environment's highlights or parameters. The value function evaluates the possible movements when in a specific state S. Therefore, one should anticipate some kind of recompense.

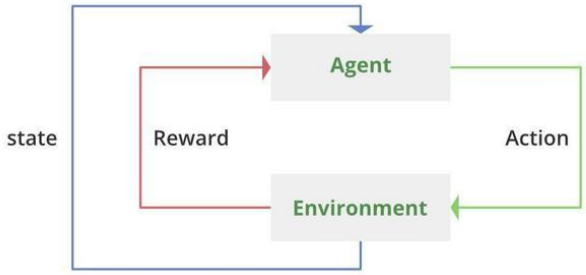

Figure. 1. Structure of Reinforcement Learning

So, one should anticipate receiving some kind of recompense. A reinforcement learning agent recognizes its surroundings and experiments with different conditions to determine the best course of action. It is a simple conduct and calculation learning approach. The agent will determine the best way to organize the new states. The degree to which the explanation corresponds to the ideal behavior depends on the work. In each emphasis, the agent indicates the state it is in right now ($s \in S$) and receives a reward signal ($r \in R$) after choosing an action ($a \in A$). In order to perform optimally in this process, the agent must have valuable experiences with states, activities, state developments, and rewards. The framework's evaluation occurs concurrently with the learning process. Finding ways to direct behavior while enhancing a reward signal is the main objective of reinforcement learning. A self-governing agent uses reinforcement learning to explore and learn the optimal course of action to take in each and every state to accomplish its goal.

The most common way to illustrate reinforcement learning is with a Markov Decision Process (MDP). For single agent Reinforcement Learning, it is the scientific established model. The model's problems stem from sequential decision-making, wherein actions must be selected at each state by referring to the relevant framework. These problems are ubiquitous in stochastic control theory, and their underlying principles are traceable.

3.2Steps of Reinforcement learning process

The Reinforcement Learning problem was approached using these stages prior to its emergence.

- Understanding Generalized problem: Not every problem really requires reinforcement learning. Before implementing the reinforcement learning method, every issue should be checked, and the following factors are taken into account: a) an experimental method of trial and error b) delayed rewards c) is able to show as MDP d) to determine whether or not the issue is under control.

- Stimulated Environment: Calculating the number of iterations is required prior to using the Reinforcement Learning methods. A simulation program is needed in order to adequately represent real-world items.
- Markov Process: These procedures must be followed in order to formulate the problem for every single problem. While designing the state space, action space, reward function, and other elements, the issue under consideration has to be represented as the MDP. All jobs for which an agent is paid will be completed by them.
- Algorithms: Diverse Reinforcement Learning algorithms are available and utilized to find the optimal policy or to learn the value function.

3.3Different uses of Reinforcement Learning

i) Traffic Forecasting Service

The number of vehicles on the road is increasing at a rapid rate, making traffic management seem like a major problem. Machines can be prepared and employed to address this problem in order to overcome it. Devices that superimpose a gauge on an enhanced traffic stream map to predict future traffic conditions. These frameworks can also aid in determining the traffic conditions within an area, both now and in the future, and provide steering recommendations to customers based on that information.

ii) Robotics

Under human supervision, robots are capable of carrying out incredible jobs, such as household chores and medical treatments. It is impossible to predict whether or not there will be a fully observable condition in this subject. It is not possible for this learning framework to predict information about several states that could appear to be identical.

iii) Computer Games

In the years that have passed, the gaming industry has grown significantly. To provide intelligent game information for the players, artificial intelligence agents are repurposed. These agents can play a variety of roles, such as rivals, allies, or other non-player characters. A game must meet a wide range of requirements, such as sound and unique visuals, in addition to interacting with human players.

iv) Machinery Applications

A type of machine learning computation known as reinforcement learning enables programming agents and machines to explicitly determine the best behavior in a given

situation and to increase its effectiveness. These programs are not modifiable. Manufacturing, Power Systems, Delivery Management, Inventory Management, and Finance Sector are all included.

v)Stock Market Analysis

For long-term success in the stock market, legitimate understanding of the constantly shifting trends of the stock exchange market is crucial. The forecasting of financial markets has been a key use of machine learning. Appropriate algorithms, such as reinforcement learning and support vector machines, have proven effective in tracking the stock market and enhancing the advantages of low-risk investment opportunities. In order to predict the daily stock pattern, it also combines market analysis that takes into account the choices of regular financial investors who make worldwide stock market investments.

vi) Learning Environments using Semantic Annotations

Functional learning is becoming increasingly important in all facets of life in the modern world. It provides a deeper understanding of the subject and aids in obtaining practical knowledge. Semantic computations are employed as a key component of skills that are founded on a very beneficial learning environment. Real-world scenario simulations aid in the development of practical skills such as problem solving, communication, teamwork, and decision making.

4. ALGORITHMS

One important viewpoint in reinforcement learning is that of an agent. It also goes by the names learner and decision maker. Anything deemed outside the agent's realm is seen as belonging to it. This Reinforcement Learning frameworks domain emerges as an effective agent for translating all conditions to activities through trial-and-error interactions. The single-agent and multi-agent frameworks, which differ in terms of their features, are used for these tasks. Other adjusting agents are also utilized in multi-agent a framework, which defies the Markov property that the traditional single agent depends upon and causes instability in the environment. The many reinforcement algorithms that are applied to multi-agent systems are as follows:

- Minimax-Q Learning Algorithm

The player has the opposite enthusiasm for the game. First, having the capacity to support learning computation is valuable. In this method, the player tries to increase its typical

incentive even in the event that the adversary makes the most appalling selection regarding what to do.

- Friend-or-Foe Q-Learning Algorithm

Each agent within the framework is referred as either a "friend" or a "foe" in the FFQ approach. The equilibrium in this case can be classified as either adversarial or coordinated. When comparing this algorithm to the Nash Q learning method, the former can guarantee convergence more strongly.

- Nash-Q Learning Algorithm

Wellman et al. (2003) developed a Nash Q learning computing method for multi-agent reinforcement learning techniques and suggested a zero-sum game structure of the Minimax Q learning technique to generic aggregate games. It is necessary to take into account multiple cooperative activities of participating agents rather than just individual actions in order to extend Q learning to the various multi-agent learning domains. The Q values for the learner and other players must be continued in this algorithm because to the notable differences between single agent and multi-agent reinforcement learning agents. The primary goal of identifying Nash equilibria at every state is to employ Nash equilibrium methods for Q- value updates. The Nash Q value must first be defined before the Nash Q learning technique can be used. The predicted sum of limited rewards at which point all agents must continue to adhere to the specified Nash equilibrium policies is what is meant to be defined by this value. Hu and Wellman (2003) also emphasized that, in some scenarios, this learning process in a multi-player setting interacts with Nash equilibrium tactics and adds additional expectations to the payout structures.

- rQ-Learning Algorithm

Large search space challenges are addressed by the rQ learning algorithm. R state and action set in this algorithm must always explicitly defined at the outset. When an action begins, it is triggered by a group of conditions before and after the action in a general way, while a state is triggered by basic relationships like a goal in front of it or a robot from the team to the left. An r action must meet a requirement in order to be defined appropriately: if it is appropriate for a specific case of a r state, it must also be appropriate for all occurrences of that state. This method works well for extensive searches.

- Fictitious Play Algorithm

When there is difficulty determining the outcomes of Nash equilibria in Nash equilibrium-based learning, the fictional play algorithm is employed to provide an additional method of

managing multi-agent frameworks. According to Cao, 1997 and Suematru, 2002, the participants must maintain Q values that belong to them, which are connected to the joint actions and are weighed by their conviction allocation, in addition to the other methods, which are represented by experimental dissemination. The algorithm adapts the specific Q-learning method for the stationary strategies of different players. It has also been applied to the non-stationary approaches of various players in modest games where players can showcase their rival adversaries.

- Policy Hill Climbing Algorithm

This algorithm adjusts the Q values similarly to fake play algorithm, but by carrying out the hill climbing in the space of these policies; it preserves the mixed policy, which is also referred to as the stochastic policy. Bowling and Velso proposed a PHC approach known as WoLF (Win or Learn Fast) by embracing the Win or Learn Fast concept and utilizing the variable learning rate. If the agent is not using this algorithm carefully and correctly, they will quickly learn from their mistakes. Because the learning rates will no longer be overfit to the many agents' changing tactics, the convergence will benefit from this adjustment.

- Multi-Agent SARSA Learning Algorithm

Nash Q and Minimax Q learning algorithms are categorized as off strategy Reinforcement Learning algorithms because they modify the max operator of a Q learning algorithm through their dominant response, called the Nash equilibrium policy. Regardless of the strategy chosen, an off approach learning technique in reinforcement learning constantly looks for ways to combine the best Q values of the optimal strategy. Sutton (1998) states that the SARSA algorithm is a policy Reinforcement Learning method that aims to converge to the optimal Q values of the currently implemented strategy. In his study, Suematru (2002) mentioned that he developed an SARSA-based multi-agent algorithm called EXORL (Extended Optimal Response Learning) to address this issue.

5. CONCLUSION

Machines were used in the past to lessen physical labor, but as artificial intelligence has advanced, humans have sought to create machines that possess both strength and intelligence. As a result, the idea of machine learning has emerged and is quickly becoming a popular field of study. This paper discusses the several categories of machine learning, including supervised and unsupervised learning, reinforcement learning and recommender system. It also illustrates the range of applications that fall under this umbrella. This research examines

several algorithms for reinforcement learning that can reduce the number of states in a test, increase learning productivity at the start of the test, and speed up convergence.

There is also a discussion of the Multi Agent Q learning algorithm, which is used to build the likely artificial intelligence field and adjust the Q values based on the available data. Here, applications of reinforcement learning are also covered, shedding light on how this technology has developed into something amazing and capable of producing amazing results across a wide range of difficult issues. Therefore, it is essential to use the most recent advancements in reinforcement learning together with innovative approaches to solve challenges pertaining to learning techniques, breaking down problems, making estimates, and incorporating real-life situations with partial information.

ACKNOWLEDGEMENT

The authors did not receive financing for the development of this research. The authors declare that there is no conflict of interest.

REFERENCES

[1] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learningbased control to safe reinforcement learning," Annual Review ofControl, Robotics, and Autonomous Systems, vol. 5, pp. 411–444, 2022.

[2] D. Yarats, R. Fergus, A. Lazaric, and L. Pinto, "Reinforcement learning with prototypical representations," in International Conferenceon Machine Learning. PMLR, 2021, pp. 11 920–11 931.

[3] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," Electronic Markets, vol. 31, no. 3, pp. 685–695, 2021.

[4] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in 2018 Fourth international conference on computing communication control and automation (ICCUBEA). IEEE, 2018, pp. 1–6.

[5] R. Kirk, A. Zhang, E. Grefenstette, and T. Rockt¨aschel, "A survey of generalisation in deep reinforcement learning," arXiv preprint arXiv:2111.09794, 2021.

[6] A. Oliver, A. Odena, C. A. Raffel, E. D. Cubuk, and I. Goodfellow,"Realistic evaluation of deep semi-supervised learning algorithms," Advances in neural information processing systems, vol. 31, 2018.

[7] M. Naeem, S. T. H. Rizvi, and A. Coronato, "A gentle introduction to reinforcement learning and its application in different fields," IEEE Access, vol. 8, pp. 209 320–209 344, 2020.

[8] J. Shin, T. A. Badgwell, K.-H. Liu, and J. H. Lee, "Reinforcement learning–overview of recent progress and implications for process control," Computers & Chemical Engineering, vol. 127, pp. 282– 294, 2019.

[9] Y. Qian, J. Wu, R. Wang, F. Zhu, and W. Zhang, "Survey on reinforcement learning applications in communication networks,"Journal of Communications and Information Networks, vol. 4, no. 2, pp. 30–39, 2019.

[10] M. Lanctot, E. Lockhart, J.-B. Lespiau, V. Zambaldi, S. Upadhyay, J. P´erolat, S. Srinivasan, F. Timbers, K. Tuyls, S. Omidshafiei et al.,"Openspiel: A framework for reinforcement learning in games," arXiv preprint arXiv:1908.09453, 2019.

[11] J. Oh, M. Hessel, W. M. Czarnecki, Z. Xu, H. P. van Hasselt, S. Singh, and D. Silver, "Discovering reinforcement learning algorithms," Advances in Neural Information Processing Systems, vol. 33, pp. 1060–1070, 2020.

[12] M. M. Afsar, T. Crump, and B. Far, "Reinforcement learning based recommender systems: A survey," ACM Computing Surveys, vol. 55, no. 7, pp. 1–38, 2022.

[13] T. Salimans and R. Chen, "Learning montezuma's revenge from a single demonstration," arXiv preprint arXiv:1812.03381, 2018.

[14] L. Canese, G. C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. Re, and S. Span`o, "Multi-agent reinforcement learning: A review of challenges and applications," Applied Sciences, vol. 11, no. 11, p. 4948, 2021.

[15] D. Mehta, "State-of-the-art reinforcement learning algorithms," International Journal of Engineering Research and Technology, vol. 8, pp. 717–722, 2020.

[16] H. Fei, X. Li, D. Li, and P. Li, "End-to-end deep reinforcement learning based co reference resolution," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics,2019, pp. 660–665.

[17] E. R. Watters, "Factors in employee motivation: Expectancy and equity theories," Journal of Colorado Policing, vol. 970, p. 4, 2021.

[18] D. A. Vidhate and P. Kulkarni, Enhanced Cooperative Multi-agent Learning Algorithms using Reinforcement Learning, *published in the proceedings of IEEE International Conference on Computing, Analytics and Security Trends*, pp 556-561, 2016.

[19] E Yang and DongbingGu, A Survey on Multi-agent Reinforcement Learning towards Multi Robot Systems, *published in the proceedings of IEEE 2005 Symposium on Computational Intelligence and Games, CIG '05 IEEE*, pp 292-299, 2005.

[20] E. F. Morales, Scaling up Reinforcement Learning with a Relational Representation, *published in the proceedings of the Workshop on Adaptability in Multi-Agent Systems., Sydney*, 2003.

[21] EJ Horvitz, J Apacible, R Sarin, L Liao, Prediction, Expectation, and Surprise: Methods, Designs, and Study of a Deployed Traffic Forecasting Service *published in the Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence (UAI2005)*, July, 2012.

[22] F. L. Lewis, G. Lendaris and D. Liv, Special Issue on Adaptive Dynamic Programming and Reinforcement Learning for Feedback Control, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 38, No. 4, pp 896-897, August, 2008.

[23] F. Musumeci, C. Rottondi, A. Nag, T. Malcaluso, D. Zibar, M. Ruffini and M. Tornatore, An Overview on Application of Machine Learning Techniques in Optical Networks, *published in IEEE Communications Surveys and Tutorials*, vol. 21, issue 2, pp. 1383-1408, November, 2018, DOI: 10.1109/COMST.2018.2880039.

[24] . Fujii and S. Managi, Trends and Priority Shifts in Artificial Intelligence Technology Invention: A Global Patent Analysis, *published in Econ. Anal. Policy*, vol. 58, pp. 60–69, 2018.

[25] Carlucho, M. D. Paula, S. A. Villar, G. G. Acosta, Incremental Q-Learning Strategy for Adaptive PID Control PF Mobile Robots, *Expert Systems with Applications*, Vol. 80, pp 183-199, September, 2017.

[26] J. Hou, Huali, J. Hu, C. Zhao, Y Guo, S. Li, and Q. Pan, A Review of the Applications and Hotspots of Reinforcement Learning, *published in the proceedings of IEEE International Conference on Unmanned Systems (ICUS)*, October, 2017, DOI: 10.1109/ICUS.2017.8278398.

[27] J. Hu and M. P. Wellman, Nash Q-Learning for General-Sum Stochastic Games, *Journal of Machine Learning, V*ol. 4, pp. 1039–1069, 2003.

[28] J. Mark Weal, T. DaniusMichaelides, K. Page, C. David De Roure, Fellow, IEEE, E. Monger, and M. Gobbi, Semantic Annotation of Ubiquitous Learning, *Journal Environments IEEE Transactions on Learning Technologies*, Vol. 5, No. 2, April-June 2012.

[29] J. R. Anderson, Machine Learning: An Artificial Intelligence Approach, Eds. Ryszard S. Michalski, et al., *published by Morgan Kaufmann*, vol. 2, 1986.

[30] J. Schmidhuber, A General Method for Multi-Agent Learning And Incremental Self-Improvement in Unrestricted Environments. In Yao, X (Ed),

Evolutionary Computation: Theory an Applications, *Journal Scientific Publications Co*., Singapore, 1998

[31] K. A. Wyrobek, E. H. Berger, HF M. Van der Loos, and K. Salisbury, Towards a Personal Robotics Development Platform: Rationale and Design of an Intrinsically Safe Personal Robot, *published in the Proceedings of International Conference on Robotics and Automation (ICRA),* 2008.

[32] L. Busoniu, R. Babuska, B. Deschutter and D. Ernst, Reinforcement Learning and Dynamic Programming using FuctionApproximators, *BocaRaton, FL: CRC Press*, 2009.

[33] Lanfranco, A. Castellanos, J. Desai, and W. Meyers. Robotic Surgery: A Current Perspective, *Journal of NCBI Annals of surgery*, 239(1):14, 2004.

[34] M. H. Bowling and M. M. Veloso, Multi-agent Learning Using a Variable Learning Rate, *Journal of Artificial Intelligence,* vol. 136, no. 2, pp. 215–250, 2002.

[35] M. H. Bowling, Multi-agent Learning in the Presence of Agents with Limitations, *published in the Ph.D. dissertation, School of Computer Science, Carnegie Mellon University,* Pittsburgh, May 2003.

[36] M. L. Littman, Markov Games as a Framework for Multi-agent Learning, *published in the Proceedings of 11th International Conference on Machine Learning, San Francisco*, pp. 157–163, 1994.

[37] M. Tiwana, S. Nawaz, A. Ikram and M. Tiwana, Self-Organizing Networks: A Packet Scheduling Approach for Coverage/Capacity Optimization in 4G Networks Using Reinforcement Learning, *ElektronikaIrElektrotechnika Journal*, vol. 20, pp. 59-64, 2014.

[38] N. Suematsu and A. Hayashi, A Multi-agent Reinforcement Learning Algorithm Using Extended Optimal Response, *published in the Proceedings of 1st International. Joint Conference on Autonomous. Agents & Multi agent Systems, Bologna, Italy*, pp. 370–377, July 15-19 2002.

[39] R. Ollfati-Saber, Flocking for Multi-agent Dynamics Systems: Algorithms and Theory, *IEEE Transactions on Automation Control*, Vol. 51, No. 3, pp 401-420, 2006.

[40] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed. Cambridge, MA: *published by MIT Press*, 2017.

[41] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed. Cambridge, MA: *published by MIT Press*, 2017.

[42] R. S. Sutton, Introduction: The Challenge of Reinforcement Learning, Machine Learning, *published by Kluwer Academic Publishers, Boston*, vol. 8, pp. 225-227, 1992.

[43] R. Sutton and A. Barto, Reinforcement Learning, *published by MIT Press*, ISBN 0-585-02445-6, 1998.

[44] S. Das, A. Dey, A. Pal and N Roy, Applications of Artificial Intelligence in Machine Learning: Review and prospect, *International Journal of Computer Applications (0975-8887),* Vol. 115, No 9, April 2015.

[45] S. M. Hung and S. N. Givigi, A Q-learning Approach to Flocking with UAV's in a Stochastic Environment, *IEEE Transactions on Cybernetics*, Vol. 47, No. 1, pp 186-197, 2017.

[46] S. Marsland, Machine learning: an algorithmic perspective, *published by CRC press*, 2015.

[47] Shen, Shunrong, H. Jiang, and T. Zhang, Stock Market Forecasting using Machine Learning Algorithms, 2012.

[48] T. Graepel, Playing Machines: Machine Learning Applications in Computer Games, ICML 2008 *published in the Tutorial Program Helsinki, Finland*, July 2008.

[49] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, Cooperative Mobile Robotics: Antecedents and Directions, *Journal Autonomous. Robots*, Vol. 4, pp. 1–23, 1997.