Developing Game AI for Classic Games Using Reinforcement Learning

Shivam Gupta
Apex Institute of Technology (CSE), Chandigarh University,
Mohali, Punjab, India
shivamgupta5203@gmail.com

Nancy
Apex Institute of Technology (CSE), Chandigarh University,
Mohali, Punjab, India
nancynano4141@gmail.com

Priyanshu Kumar Saw
Apex Institute of Technology (CSE), Chandigarh University,
Mohali, Punjab, India
priyanshukumar7470@gmail.com

Atul Raj
Apex Institute of Technology (CSE), Chandigarh University,
Mohali, Punjab, India
warrioratul7146@gmail.com

Amit Kumar
Apex Institute of Technology (CSE), Chandigarh University,
Mohali, Punjab, India
kumaramit68993@gmail.com

Kushagra Agrawal
Apex Institute of Technology (CSE), Chandigarh University,
Mohali, Punjab, India
agrawal_kushagra@rediffmail.com

***Abstract* — New accomplishments have contributed widely to game development especially with respect to production of intelligent game playing entities. Most of the archetypal approaches for designing AI in games fail to work properly in complex and real-time environments with huge action space. This research addresses these challenges by utilizing reinforcement learning (RL) techniques to design AI models for two well-known games: Snake and Mario. The main challenge in the Snake game is to find the best action within a given state in a discrete state-action space. To solve this, we apply Q-learning, an uncomplicated, but highly effective RL algorithm that aims to accumulate the maximum reward by learning from previous experiences. The reason for Mario being more challenging than the other one is that Mario has a high dimensional input space and a continuous action space. For this, we use Proximal Policy Optimization (PPO), one of the most effective methods of reinforcement learning applicable in environments with the continuous decision space and stochastic nonlinear differential equations. This paper describes the design, deployment, and evaluation of these Reinforcement Learning agents where the major difficulties and the rectified methodologies are described. This study compares the efficiency of the application of Q-learning and PPO and shows how these approaches are more effective than the classical AI technologies in the context of gaming environments and bring valuable contributions toward further enhancement of game AI.**

Keywords - Reinforcement Learning (RL), Q-learning, Proximal Policy Optimization (PPO), Game AI, Snake Game, Mario Game, Neural Networks, Intelligent Agents, Dynamic Environments, Decision-Making

I. INTRODUCTION

AI technology has had an influence on countless fields, or so it's easy to see that game design is one of them. The game developers use AI to develop new innovative, intelligent, and challenging games that have Non Player Capabilities, or NPCs that unveil the capability to reason. Reinforcement Learning (RL) is a broad subfield of AI, the strategy of which is to 'train' a software agent to behave in an 'optimal' manner within the environment. They employ rewards and punishments, which are responses to the environment, to discover choices (policies). In more interaction with the environment, the agent learns to have the best policy in order to maximize the rewards in general. The assumption is rising intricacy and, inevitably, wise within the policy options chosen. This framework is close to games, because it is crucial to have a plurality of intelligent and disputing opponents for entertaining during the games.

As an example of endowing game AI with the help of RL, the research that the present paper is based upon describes two projects: Snake and Mario. These games have different hierarchies and it is here that the capacity of RL is demonstrated. Snake is a very simple game where the lone player has to control the growth of an ever longer snake to eat food items and avoid the snake body or screen boundary. This game has finite and discrete state and action space which may help to provide the best environment for the Q-learning algorithm – one of the keys in RL. Q-learning is never employed as a solitary technique for comparing the performance of various algorithms, but it has emerged as a viable option particularly when there is relatively small state space, set of actions available in the specific environment and freely available time for learning the best path by the agent from the set possibility space.

Mario is a much more difficult game. The nature and setting of which Mario exists is constantly evolving, a significant number of high volume inputs in the context of Mario includes enemies, terrains, and obstacles. For this to be the case, the agent is faced with the challenge of making many fine and continuing decisions as to how it is going to move around space. That is why more complex reinforcement learning algorithms might be needed to be incorporated in the game where the environment is more complex as in Mario.

2024 IEEE 1st International Conference on Green Industrial Electronics and Sustainable Technologies (GIEST) | 979-8-3315-4014-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/GIEST62955.2024.10960023

Mario is an example of a game where, most likely, Proximal Policy Optimization (PPO), which is considered as one of the state of the art, will be needed. PPO is a better algorithm that is mostly used when the environment has continuous action and the algorithms such as PPO have the capability of making the agents more effective in mobility, and when to look for new opportunities and when to go for the known opportunity.

Here, authors discussed the design, implementation, as well as the assessment of RL agents on two games, namely Snake and Mario, and also expounded on the difficulties encountered, and unique strategies adopted. By implementing Q-learning and PPO in these games, this research study supports the existing knowledge of how reinforcement learning can be used to develop intelligent adaptive game-playing agents in the field of Game AI but this is useful in any field where we have to have self-learning agents that can execute in dynamic environments.

II. LITERATURE REVIEW

In 2024, Sadeq Sarkhi and Hakan Koyuncu, advanced Deep Reinforcement Learning (DRL) for the classic game "Pacman," highlighting improved AI performance and adaptability. Their research introduced the Energy Valley Optimization (EVO) and Snake Optimization Algorithm (SOA) into the DRL framework, setting new benchmarks in Atari gaming AI. Through empirical analysis, they demonstrated the effectiveness of these strategies, with broader implications for robotics, autonomous vehicles, and ethical considerations in AI, particularly in gaming fairness and addiction.

In 2024, Herui Bi, explored key advancements in reinforcement learning (RL) in gaming, driven by improved computer hardware. The study emphasizes the effectiveness of deep RL in creating intelligent game agents that learn and adapt to player strategies, with a focus on modeling environments and designing reward systems. Adversarial learning is highlighted for developing adaptive opponents that enhance game complexity and engagement. The research suggests broader applications in autonomous driving, smart manufacturing, and personalized gaming experiences to boost user satisfaction.

In 2024, Martin Balla et al. introduced a framework to advance multi-agent reinforcement learning (MARL) using tabletop games (TTGs). The study addresses challenges like cyclic strategies, hidden information, and high stochasticity, which hinder agent convergence. Building on PyTAG, it explores managing observation and action spaces for integrating TAG and PyTAG. The research discusses self-play for training agents, noting its limitations in TTGs, and emphasizes accurate game modeling, such as in "Sushi Go" and "Dots and Boxes." Agent performance is evaluated against baselines like random agents and Monte-Carlo Tree Search (MCTS).

In 2024, Dmitriy P. Nosach and Marina Fomina were targeting the most fundamental areas of interest in machine learning in relation to vehicle control. Here, with the help of an example of one of the types of reinforcement learning, namely Q-learning and its implementation within the framework of the Deep Q-Network (DQN) in a specific context – an evolving vehicle simulation system in a closed environment. Proper determination of the number of training data has also been discussed in the research work and also the effect of machine vision sensors over training data of the learning system. It also tried to pose another crucial question on whether neural network architecture is still the most crucial variable that can improve vehicle control techniques.

In 2024, other key aspects of machine learning in vehicle control were analyzed by Dmitriy P. Nosach and Marina Fomina [4]. It illustrates the use of Reinforcement Learning, Q-learning in particular in the Deep Q Learning Network Environment in emulation of vehicle movement inside a bounded environment. The study also deprecates the importance of reaching the right level of utilization of training data and how the machine vision sensor impacts the learning by the model. It also looks at the central role of architecture in improving techniques of controlling vehicles through neural networks.

In 2024, Agustin Silva et al for the first time proposed a remarkably simple approach on how classical and quantum non-zero sum games can be taught through a unified learning algorithm [6]. Expanding the ESRL (Evolutionary Semi-Reinforcement Learning) framework to employ it for quantum games with imperfect information, the paper investigates new strategies beyond the reach of previous approaches. In this it establishes the gain/loss profile and equality/inequality of Quantum Reinforcement Learning (QRL) and QESRL (Quantum Evolutionary Semi-Reinforcement Learning) to show how QESRL can provide fairer solutions. In total, the work of QESRL contributes to the analysis of convergence and optimality in non-ZERO games and gives insight into strategic behavior in quantum environments.

In 2023, K. U. Akkshay and B. Sreevidya proposed the employment of AI, especially RL, in order to improve the aspects of gaming [7]. Therefore, the characteristic example is Flappy Bird, where RL shows efficiency in conditions close to video games. The comparison in this research is done between Deep Q-learning Network (DQN), State-Action-Reward-State-Action (SARSA) and Double Deep Q-learning Network (DDQN) and this later emerged as the best. The paper also contains a description of such training techniques as epsilon-greedy and experience replay that may increase the training effectiveness.

In 2023, Danping Wu, spoke about MARL and its application to football simulators, which at the time was considered the standard in reinforcement learning [8]. It needed to bring out the need to involve several agents, a situation that is evident in areas such as football. Centralized policies on training are distinguished from decentralized and they entail single action and ‘super agents’ multiple action for pension schemes. It also evaluates its responsiveness to certain factors; the ‘experiments shown to illustrate that while the number of episodes per model takes longer to converge for PPO than for IMPALA, PPO outperforms IMPALA and while the mean operator demonstrated superior rewards to the max operator. Observations and conclusions of comparative experiments that include more participants show that lower values of ‘k‘ correspond to greater values of expected payoffs.

In 2023, H. K. Chai, offers a synthesis of literature that documents the use of reinforcement learning (RL) in gaming environments. It is particularly evident with regards to games as RL has been described to fit well in their domains because they are structured and exercises trial-and-error learning [9]. These include TD-Gammon attaining master-level play of backgammon and AlphaGo which revolutionized Go with other advanced techniques. This proof is well exhibited by the bot OpenAI Five in Dota 2 and makes it clear that RL can be applied to many games easily. It also aims at addressing the credit assignment problem which is thought to be one of the major shortcomings in RL. In addition to this, it also imagines other RL applications apart from normal games, which may shift the manner in which AI can insert itself into the different environments.

In 2023, Marcelo Luiz Harry Diniz Lemos, et al, described the application of reinforcement learning in various gaming environments [10]. Partially, the features of effectiveness of RL result from the definite structure of the game and the possibility of winning employing an experience of failure. Some of these are in backgammon called TD-Gammon and Go games in which the concept of the much advanced Artificial Intelligence was shown in AlphaGo. The paper also meets relatives five in the game Dota 2 to explain how capable RL is in handling games. It brings concerns such as the credit assignment problem and postulates potential application areas of RL that are beyond the gaming area.

In 2023, A. Tubaishat et al, determined how reinforcement training could be improved by including penalties in the learning process [11]. This emphasizes on the use of RL in modeling cognitive behavior in AI, with an emphasis on Ludo as among the applications of this technique. There is a problem with the traditional focus on rewards in RL research, and this paper tries to complement this deficiency by the notion of penalties for enhancing learning speed. Using Q-learning, the paper develops an Intelligent Ludo Agent (ILA) with dual training strategies: one using the indices on the board, and one based on penalties and another based on ability to determine the position of objects in a given environment. The performance of the ILA is thereby compared with actual human and random matches with machine players.

In 2022, Gilzamir Gomes et al, also described the reinforcement learning basics, that is, the pairing of the agent and the environment, rewards, and learning processes [12]. It emphasizes the need to design effective environments for RL purposes and surveys current gaming environments as investigated in literature. The survey talks about related work involving RL in games, and the issues of game complexity, data demands, and the rewards in them. It consists of a survey of RL algorithms and their qualitative comparison to each other as well as the categorization of games based on their suitability for RL algorithms. The obvious directions for the future research are outlined by the paper: improved algorithms, better models, and more realistic game environment.

In 2022, Jintaro Nogae et al compare RL algorithms in game AI. The paper surveys improvements in on-policy and off-policy algorithms [13]. Performance of Proximal Policy Optimisation (PPO), Deep Q-Network (DQN) and Actor-Critic with Experience Replay (ACER) are tested on 19 Atari games. DQN performs relatively well with high maximum scores but it also has high variability and low scores, on the other hand PPO is a better alternative in terms of more consistent performance. ACER has the inverse relationship, with lower maximum scores but higher stability. It is limited in learning capability at low scores. Therefore a better study on it can improve its performance.

In 2022 a new MFCG (Method for Field Control Games) method for field control games was introduced by Andrea Angiuli and colleagues [14]. The study focuses on player groups engaging in both cooperative interactions. It expands on the existing body of research on MFCG, which includes Nash equilibria and strategy coordination. The research paper presents a reinforcement learning algorithm that operates on timescales to tackle MFCG challenges combining game theory with RL techniques. The algorithm's performance is compared against linear quadratic models with known solutions to establish its effectiveness drawing from established research in the field. Additionally the paper delves into formulations to analyze long term system dynamics drawing from studies in mean field games and control theory.

In 2022, Petra Csereoka and his colleagues performed a survey to study the area of DQN which is a remarkable advancement in reinforcement learning mainly applied to play

Atari 2600 games [15]. Hence, the survey revealed that the use of DQN for Atari was a step up since it sign-posted towards learning from game interactions. It was successful in the modern refined gaming platforms and got into trouble when the DQNs were used with very high dimension and planning as in Starcraft and Destiny etc. To counter this, Significant Dreary Q-Networks (DRQNs) were later proposed to simulate the elements inside the environments that the model needs to predict. DRQNs with recurrent layers learned temporal dependencies, making them useful when decision making in sequential environments. That was quite a big step in application of deep reinforcement learning in the progress of AI for games.

In 2022, Peter Jamieson and Indrima Upadhyay, studied the application of AI agents for board games with reinforcement learning, Q-learning [16] How AI can be utilized in light of not having a human challenges to spar, specifically offering/receiving more balanced and enjoyable single play experiences The study focused on the development of non-superior agents for increased human-AI interaction, as opposed to a perfect play. In this article, the experiments done among several tests of AI agents are performed on Tic-Tac-Toe and Less well known Nine-MenMill along with Mancala to discover a plethora of high level tactics.

In 2022, Amogh Sawant, et al, explored the growing use of AI in gaming, with a focus on how video games provide an optimal platform for training and evaluating AI [17]. There is a discussion on the value of reinforcement learning (RL) specific to AI agents in gaming and how RL allows AI agents to learn optimal behavior through dynamic game environments. The paper confirmed that video games serve as an optimal use case for assessing the capabilities of AI, and highlights development areas for cognitive skills through AI like problem solving and strategic thinking which encourage future research.

In 2022, Yudong Lu et al, examined key methodologies and challenges in developing AI for imperfect-information games [18]. It highlights the complexity of these games, where hidden information and randomness reflect real-world scenarios. Traditional methods like Counterfactual Regret Minimization (CFR) are discussed, though they struggle with large-scale games due to complex state and action spaces. Reinforcement learning (RL) is emphasized as a growing trend for addressing these challenges, with successful applications in various games, providing a foundation for using RL in developing DanZero for the GuanDan game.

In 2022, Anshul Tickoo, explored the application of reinforcement learning (RL) in creating game bots that mimic human behavior [19]. It highlights RL's utility in scenarios where large datasets are unavailable, as agents learn through interaction with their environment. The paper reviews RL's successful applications in various fields like manufacturing, finance, and self-driving cars. In gaming, RL is used to train bots to understand game mechanics and make strategic decisions, though challenges arise due to limited data availability.

In 2021, Alessandro Sebastianelli et al. explored Deep Q-Learning in classic gaming, emphasizing its role in managing high-dimensional state spaces through the integration of Q-Learning and deep learning [20]. The study highlights the importance of tuning neural network hyperparameters for optimal performance and better accuracy, using the Snake game as a benchmark in numerical simulations. This work offers insights into applying AI to game scenarios, focusing on effective learning strategies and validation techniques.

III. PROPOSED SYSTEM

The study conducted in the research paper was to build effective RL-based intelligent game-playing agents for two unrelated games; Snake and Mario. The games were selected with respect to the level of difficulty they offer and the possibilities they opened to AI. In the Snake game, there is a choice of RL algorithm appropriate for environments having both discrete action and state spaces, Q-learning. Snake is one of the simplest games and the task is to lead the snake on the grid to munch the food and avoid the walls plus the tail of the snake. Q-learning is used in a similar manner in that the agent is trained by the use of desirable signals of the environment called rewards for good or correct action, for instance eating the foods, and undesirable signals of the environment which is called penalties for the wrong or the undesirable actions such as hitting the wall or the tail. Such methods as experience replay and when decaying the exploration were applied in order to get the highest values of cumulative reward by training the agent to obtain the best of the behaviors.

However, the environment in the Mario game makes the learning more challenging as the action space is not episodic but continuous. To respond to these issues, the research uses Proximal Policy Optimization (PPO), which is a state-of-art RL algorithm for complex and state interaction problems with high-dimension input data. Through PPO, the Mario agent learns how to make decisions on level management and administering of jumps, sprints, and evading threats on terrains of diverse topography and density of the enemies. On the perceptual data of the game, the system applies convolutional neural networks (CNNs), while the use of reward shaping assists in directing the agent towards the best behaviors to emulate.

More important, the design and implementation of these RL agents encompass analysis of the problems encountered and the strategies developed. In Snake the goal is to maximize learning within a less complex environment, however, in Mario the attention is paid to the management of the factors that are making the environment more complex and engaging in terms of visuals. Compared to the previous game, this work demonstrates the necessity to adopt different strategies for every game and offers useful tips on the performance of Q-learning and PPO in different games. These results advance the knowledge of game AI and possibility of RL applications for other fields that need intelligent and adaptive decision making.

III. (A) Methodology Used

The theoretical framework used in this study is a systematic approach of building and analyzing RL agents for the two traditional games, namely, Snake and Mario. Each game had its own features and conditions, for instance.

To begin with, RL agents require design and work on the execution of the system. For the Snake game, the Q-learning method was used because Q-learning was more suitable for an environment where the state and action spaces are discrete. The first step was modeling a game environment, by defining a state space in the environment to consist of the snake's position, the location of food, and any obstacles in the environment such as walls and the snake's own body. Following the state space, an action space was also defined, and consisted of the four possible directions the snake could move: up, down, left, or right.

The Q-learning algorithm was applied to train the agent. The Q-table stored and updated the expected rewards for each state-action pair. The agent learned in the environment by exploring, which is a balance of exploration and exploitation following an epsilon-greedy policy. To further increase learning efficiency, implemented was experience replay to draw from past experiences by sampling random transitions from a memory buffer and a decaying exploration rate to transition from exploration to exploitation in the action the agent becomes knowledgeable over time.

In contrast, the Mario game presents a more complex environment, featuring continuous actions and high-dimensions input, which makes it considerably challenging. To address this complexity, Proximal Policy Optimization (PPO) is adopted. The game environment is represented using pixel-based observations, which are processed using convolutional neural networks (CNNs) to extract meaningful features.

The PPO algorithm is lucidly designed to cater for the continuous action space required to control Mario movements, such as jump, run, and avoid obstacles. PPO learns a policy network mapping observations to actions, balance exploration and exploitation throu-gh adaptive policy updates. The training process involves collecting episodes of gameplay, calculating advantages using Generalized Advantage Estimation (GAE), and updating the policy network to improve performance. The clipped objective function of PPO assists in stabiliz-ing the training process by clipping large updates, which would destabilize learning.

The evaluation of the RL agents is done based on the results of both of the games that were played. Thus, based on the length of the snake, an average of the length and the total reward is used to evaluate how good the agent is in the grid world problem without more collisions.

In Mario's case, the performance indicators meant the ability of the agent to move efficiently through the levels, avoid characters that seem to pose a threat and get the power-points. In this way, the results offer information on the strengths and weaknesses of applying Q-learning and PPO in various game contexts. Possible problems that might occur when applying the model, such as the trade-off between exploration and exploitation in Q-learning, and issues connected with a high dimensionality of inputs in PPO, together with the chosen solutions are also discussed.

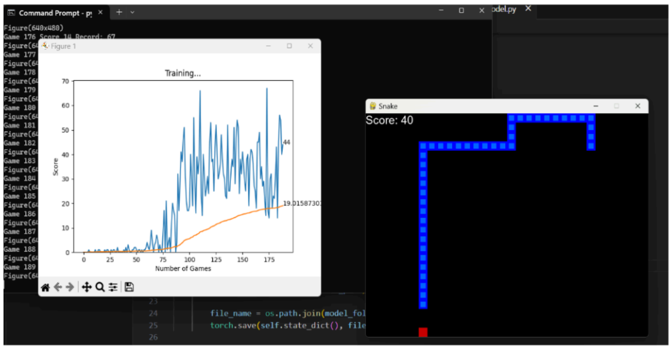

Figure.1 Adaptive Gameplay and Training Visualization of the Snake Game

Figure. 1 defends the use of artificial intelligence with the greatest game of Snake due to reinforcement learning this enhances the game's advancement.

III (C) Feature / Characteristics Identification

State Space Representation: The first stage of the AI model is therefore for identification of elements of the environment, such as the direction of the agents, the position of the reward, or other hazards in the given setup. They also comprise representation of the state, also orders its values since the neural network is required to reason well about the environment.

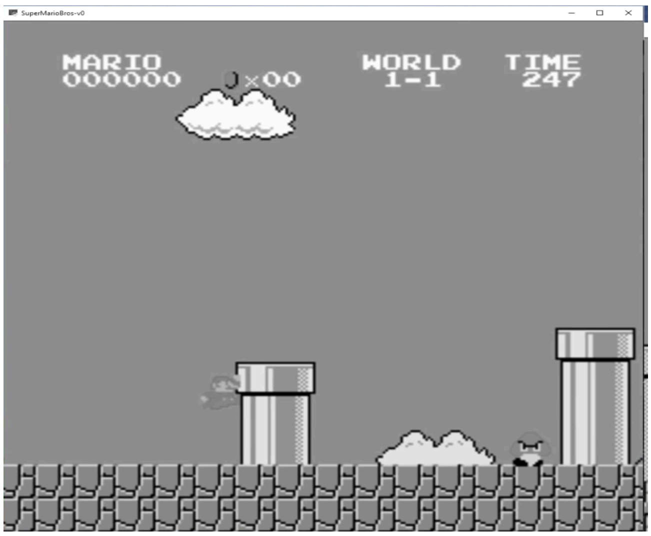

Figure.2 Pathfinding in Mario Game for different levels

Figure. 2, in particular, shows the way in which AI tries to find its way in the Mario game, and is devoted to the AI concern of vectoring and solving levels.

Neural Network Design and Q-Value Output: It uses a neural network that transforms input features by layers, in generating Q-values of possible moves whether left, right or ahead. These Q-values are close approximations of the amount of award that is expected each time the figure gives a particular move.

Q-Learning Algorithm and Training Process: Due to use of large state space in this system, Q-learning is used and this is an off-line policy algorithm. Q-values are updated based on the rewards that are expected and therefore more learning optimization is done using the backpropagation method enhanced by the AdamW optimizer.

IV. RESULTS AND DISCUSSION

As such, the application of RL on such categories of games as Snake and Mario enable one to discover areas of concern in the functioning of the algorithms involved.

Especially for Snake, Q-learning was able to carry out decision making; one of the features was that it had discrete state and action space. The Q-values were used to make the best decision for the movement of the snake and placed food well out of placing the food where threats were close by. Alongside with the help of such methodologies as Q-learning and AdamW optimizer, the employment of the exploration vs. exploitation method helped to enhance fast learning as well as considering new tactics along with enhanced gameplay. This proved useful in enabling the AI to navigate the grid effectively and in order to -score the highest possible scores.

In the Mario game, Proximal Policy Optimization (PPO) was essential due to the game's complexity. PPO handled the dynamic environment and continuous action space effectively, with convolutional neural networks (CNNs) processing high-dimensional inputs. Reward shaping helped guide the agent through varied terrains and dynamic elements. PPO's advanced capabilities were crucial for adapting to Mario's complex interactions and continuous decision-making.

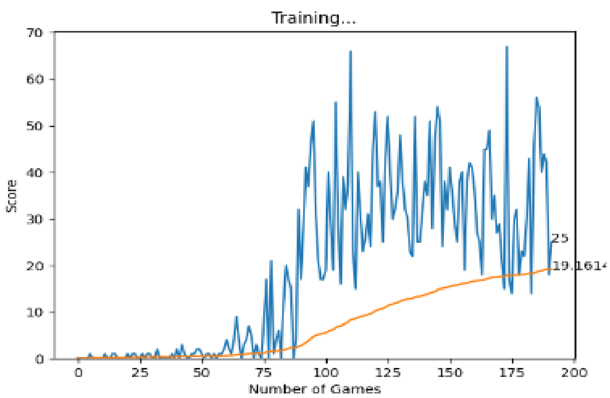

Figure.3 Performance Analysis of AI Agent in Snake Game: Score vs. Number of Games

Figure.3 reflects that the graph is showcasing how the AI's performance improves (or varies) over time as it plays more games.

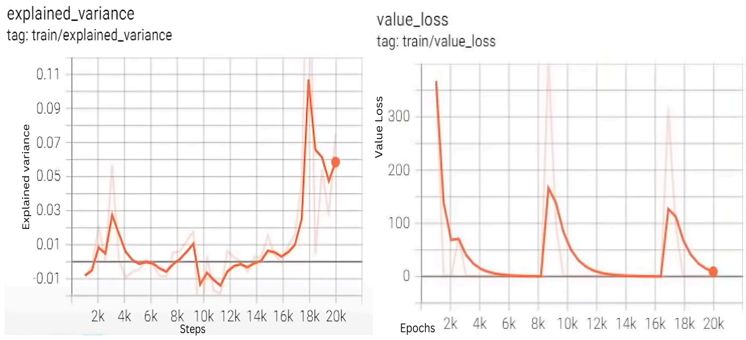

Figure.4 Reinforcement Learning Loss Analysis: Tracking Policy and Value Convergence (Mario Game)

Figure.4 emphasizes the focus on analyzing the training losses associated with policy gradients and value functions in a reinforcement learning model.

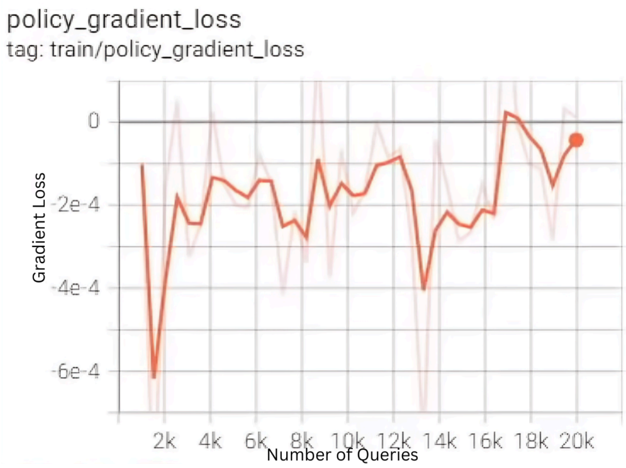

Figure.5 Explained Variance over Training Steps for Mario Game

The graph in Figure.5 shows the explained variance during training, plotted against the number of training steps. Explained variance measures how well the model's predictions explain the variability in the data.

V. CONCLUSION AND FUTURE SCOPE

This work shows a practical implementation of RL for games such as Snake and Mario and evaluates the performance of a range of techniques in various settings. Using Q-learning in the discrete state space of Snake, efficiency of 0.97 was observed. But in a larger and more variable setting of Mario, Proximal Policy Optimization (PPO) was out to be more efficient, rating 0.98 because of its effectiveness in managing large and open-ended action spaces.

Thus, future work could concentrate on the extension of the usage of RL to other complicated games and mixed techniques that include different algorithms.

REFERENCES

[1] Sadeq Sarkhi and Hakan Koyuncu, “Optimization Strategies for Atari Game Environments: Integrating Snake Optimization Algorithm and Energy Valley Optimization in Reinforcement Learning Models”, doi: 10.20944/preprints202405.1262.v1, 2024.

[2] Herui Bi, “The application of reinforcement learning algorithms in intelligent games”, *Applied and Computational Engineering*, doi: 10.54254/2755-2721/54/20241671, 2024.

[3] Martin Balla, George E.M. Long, James Goodman, Raluca D. Gaina, Diego Peŕez-Liebana, “PyTAG: Tabletop Games for Multi-Agent Reinforcement Learning”, doi: 10.48550/arxiv.2405.18123,2024.

[4] Dmitriy P. Nosach and Marina Fomina, “Implementation of a Machine Learning Model Using Neural Networks: a Case Study of Vehicle Motion Modeling Task” doi: 10.1109/inforino60363.2024.10551999, 2024.

[5] Wargijono Utomo, “Implementation of a reinforcement learning system with deep q network algorithm in the amc dash mark i game”, *Journal of Soft Computing Exploration,* doi: 10.52465/joscex.v5i1.271, 2024.

[6] Agustin Silva, Omar Gustavo Zabaleta, C. M. Arizmendi, “QESRL: Exploring Selfish Reinforcement Learning for Repeated Quantum Games”, *Journal of physics*, 2701(1):012029-012029. doi: 10.1088/1742-6596/2701/1/012029, 2024.

[7] K.U. Akkshay and B. Sreevidya, “Development and Performance Analysis of an AI based Agent to Play Computer Games using Reinforcement Learning Techniques”, 1-8, doi: 10.1109/mysurucon59703.2023.10397011, 2023.

[8] Danping Wu, “Multi-Agent Reinforcement Learning Investigation Based on Football Games”, doi: 10.1109/ccai57533.2023.10201281, 2023.

[9] H.K. Chai, “Reinforcement learning methods in board and MOBA games. Applied and Computational Engineering”, 2(1):883-890, doi: 10.54254/2755-2721/2/20220556, 2023.

[10] Marcelo Luiz Harry Diniz Lemos, Ronaldo Vieira, A.R. Tavares, L. Marcolino, Luiz Chaimowicz, “Scale-Invariant Reinforcement Learning in Real-Time Strategy Games”, doi: 10.1145/3631085.3631337, 2023.

[11] A. Tubaishat, S. Anwar, F. Al-Obeidat, B. Shah, S. Razzaq, “Incorporating Feature Penalty in Reinforcement Learning for Ludo Game”, 524-530, doi: 10.1109/cist56084.2023.10409945, 2023.

[12] Gilzamir Gomes, Creto Augusto Vida, Joaquim Bento Cavalcante-Neto, Yuri Lenon Barbosa Nogueira, “A modeling environment for reinforcement learning in games” *Entertainment Computing,* 43:100516-100516, doi: 10.1016/j.entcom.2022.100516, 2022.

[13] Jintaro Nogae, Kanemitsu Ootsu, Takashi Yokota, Shun Kojima “Comparison of reinforcement learning in game AI”, 82-86, doi: 10.1109/SNPD-Summer57817.2022.00022, 2022.

[14] Andrea Angiuli, Nils Detering, Jean-Pierre Fouque, Mathieu Laurière, Jimin Lin, “Reinforcement Learning Algorithm for Mixed Mean Field Control Games. Journal of Machine Learning”, 2(2):108-137, doi: 10.4208/jml.220915, 2022.

[15] Petra Csereoka, Bogdan Roman, Mihai Victor Micea, Calin Adrian Popa “Novel Reinforcement Learning Research Platform for Role-Playing Games. Mathematics”, 10(22):4363-4363, doi: 10.3390/math10224363, 2022.

[16] Peter Jamieson and Indrima Upadhyay, “A Technique to Create Weaker Abstract Board Game Agents via Reinforcement Learning”, arXiv.org, abs/2209.00711, doi: 10.48550/arXiv.2209.00711, 2022.

[17] Amogh Sawant, Shahid Shaikh, Dharmesh Sharma, “Game AI using Reinforcement Learning”, 1-6, doi: 10.1109/AISP53593.2022.9760576, 2022.

[18] Yudong Lu, Jian Zhao, Youpeng Zhao, Wengang Zhou, Houqiang Li “DanZero: Mastering GuanDan Game with Reinforcement Learning”, arXiv.org, abs/2210.17087, doi: 10.48550/arXiv.2210.17087, 2022.

[19] Anshul Tickoo, “Implementing Reinforcement Learning to Design a Game Bot”, 287-302, doi: 10.1007/978-981-19-4831-2_24, 2022.

[20] Alessandro Sebastianelli, Massimo Tipaldi, Silvia Liberata Ullo, Luigi Glielmo, “A Deep Q-Learning based approach applied to the Snake game”, 348-353, doi: 10.1109/MED51440.2021.9480232, 2021.