A Survey of Deep Reinforcement Learning in Game Playing

Dr. Shreya Patankar
Department of Computer Engineering
Engineering
K.J. Somaiya Institute of Technology
Sion, Mumbai, India
shreya.patankar@somaiya.edu

Charutosh Usakoyal
Department of Computer
K.J. Somaiya Institute of Technology
Sion, Mumbai, India
charutosh.u@somaiya.edu

Pruthviraj Patil
Department of Computer Engineering
K.J. Somaiya Institute of Technology
Sion, Mumbai, India
pruthviraj.p@somaiya.edu

Kaustubh Raut
Department of Computer Engineering
K.J. Somaiya Institute of Technology
Sion, Mumbai, India
kaustubh.raut@somaiya.edu

Abstract– Deep reinforcement learning is now a potent tool for building intelligent agents that excel in challenging strategic games. Chess, a well- liked board game with lots of room for exploration, has been utilized to test DRL algorithms. The game of chess is widely recognized for its deep strategic complexity, extensive history, and complex rules. In this paper, we explore the application of DRL in this game. We investigate the use of neural networks, such as recurrent neural networks (RNN) and convolutional neural networks (CNN), in conjunction with reinforcement learning algorithms, such as Proximal Policy Optimization (PPO), Deep Q-networks (DQN), and others, to construct highly performing game playing agents. Our research investigates the survey of multiple research papers concerning this topic and examines how DRL can be applied in chess.

Keywords- Deep Reinforcement Learning (DRL), AI, agent, DQN, PPO, etc.

I. INTRODUCTION

A branch of machine learning known as deep reinforcement learning (DRL) combines the concepts of reinforcement learning with the help of deep learning algorithms. DRL is chosen to apply in a strategic game like chess. DRL is particularly well- suited for tasks that involve sequential decision- making in dynamic and uncertain environments such as playing games or controlling autonomous systems. There is a variety of DRL techniques like DQN, Proximal Policy Optimization (PPO), etc. for games that examine fresh approaches to using deep learning to enhance the capability of game-playing agents to make decisions. In a game like chess where DRL could be used as game playing where professional players can train before the tournament with a model that is not a static computer consisting of levels to be surpassed but a model that is trained to be like humans and will help the player to have a real time player in front of them and play them with their own strategy. Firstly, we need a chess environment selection, & libraries like python-chess offer rich features for representing the chessboard, managing game states, and generating legal moves. Scaling the project begins with data collection, where obtaining a diverse and extensive dataset of chess games can be accomplished through web scraping or utilizing existing databases. Preprocessing the data involves extracting board states and moves and converting these into a format that neural networks can process efficiently. The choice of the DRL algorithm should align with the project's goals. Designing the neural network architecture encompasses creating a model that extracts meaningful features from the chessboard, captures temporal dependencies, and translates these into optimal moves. Then develop the targeted result to train the model which will include a collection of datasets for reinforcement learning. We need to implement exploration strategies to explore different moves during training. To increase efficiency, we need to add the levels according to the strength of the player who is training their skills using the model. The trained agent's performance against human players or chess engines like Stockfish, Deep Blue, etc., is needed to fine-tune the model and hyperparameters based on the agent's performance so that the trained model can be evaluated. We may also need to adjust learning rates, network architectures, or exploration strategies. Deployment of the AI can take various forms, including creating a user-friendly GUI. For advanced scalability, we can develop a self- improving system that continually learns and adapts. Ensuring the AI's stability and adaptability through monitoring and maintenance is crucial for long-term success. Real-time monitoring can help track its performance, behavior, and responsiveness to different opponents. Finally, comprehensive documentation detailing the project's methodologies, model architectures, training processes, and lessons learned is essential for knowledge sharing within AI & communities.

II. MOTIVATION

Logical board games such as chess require a lot of preparation and precise calculations to excel in it. It has been observed that players preparing for any chess tournaments practice with AI to make their game better. In the realm of artificial intelligence, few fields have captured the imagination and fascination of both researchers and enthusiasts quite like deep learning in game playing. With an ever-expanding array of applications, from mastering chess to dominating complex strategy games like Go and Dota 2, deep learning has emerged as a revolutionary force in the world of gaming. In this context, we find ourselves drawn to the captivating arena of board games, particularly chess, and the profound motivation that underpins our choice to explore the depths of Deep Learning for Game Playing. The notable successes of artificial intelligence in game playing have influenced us to review this topic. The profound impact of the subsequent dominance of AlphaZero in chess has shattered preconceived notions about the limits of machine intelligence. These achievements demonstrate that, given the right tools and methodologies, artificial intelligence can surpass human performance in complex strategic games. Our motivation also stems from the belief that deep reinforcement learning for chess serves as a fertile ground for innovation. Moreover, this project offers an opportunity to explore the ethical dimensions of AI in chess, including fairness, transparency, and responsible AI development. By addressing these critical issues, we contribute to the ongoing dialogue about the responsible deployment of artificial intelligence in society.

III. LITERATURE REVIEW

The authors have used deep deterministic policy gradient algorithm to plan intelligent robots' paths in real-time confrontation environments. An incremental training method is introduced to address the poor convergence of DRL, & it has 3 sub-tasks: survival, reaching the target area, and breaking through interception. Reward compensation is adopted to improve training effectiveness. The Webots simulator's simulations and Monte Carlo experiments show how well the algorithm works at avoiding obstacles and getting to the desired location. [1]. In the fields of robotics, bio-molecular interactions, & communications engineering, DRL has been extensively employed to address a wide range of technical and research problems. The work focuses on learning a sequential decision-making process for playing tic-tac-toe deftly from high-dimensional video frames using DRL [2].

Previous work has focused on different ways to learn efficient representations for DRL. Autoencoders have been used for learning constrained representations, and others have been used to learn forward models and minimize prediction errors. For RL agents, contrastive losses are used as auxiliary tasks. The trend for auxiliary networks is toward increasing complexity [3]. The paper examines how evolutionary algorithms can be used in various contexts, such as data mining, machine learning, and optimization issues. The authors emphasize the value of using evolutionary algorithms to enhance the speed and accuracy of data analysis and decision-making procedures. A novel approach has been devised to train neural networks to reach the skill level of human chess players. This innovative method allows for a deeper understanding of the neural network, enhancing its classification performance by replacing basic convolutions with residual network layers [4]. The Deep Q-Networks (DQN) algorithm and its variations in the context of reinforcement learning is analyzed by highlighting the advantages and disadvantages of approaches such as improved replay memory and the distributional outlook of DQN. The combination of deep learning and reinforcement learning, enabled by neural networks, has led to the development of autonomous systems capable of playing games and automating robotics using visual cues [5]. The Deep Q-Networks (DQN) algorithm and its variations have shown significant progress in the field of deep reinforcement learning (DRL). DQN tackles the issues of overfitting and overoptimistic action values through techniques such as memory buffer utilization, random sampling, and double Q- learning. Distributional Q-learning provides an approximate distribution of Q-values, improving the agent's performance. Rainbow DQN, which combines multiple techniques, including PER and distributional Q- learning, outperforms other variants of DQN [6]. Numerical simulations for both the training and testing phases are provided which highlight the importance of the agent design process in creating systems capable of learning how to play classic games. The results of the numerical simulations provide insights into the impact of tuning the hyperparameters of the Deep Q-Network (DQN) on the agent's performance, highlighting the importance of this step in the agent design process [7]. A Chinese chess game algorithm based on reinforcement learning is proposed, using a self-play learning model constructed with deep convolutional neural networks and a Monte Carlo search tree algorithm. The algorithm improves chess strategy through self-learning without initial training data and demonstrates strong self-learning ability and good performance in Chinese chess [8]. In addition to introducing several conventional methods for creating chess-playing systems, this article suggests a better hybrid optimization method for determining the optimal move in a game of chess [9].

The paper introduces a novel task for reinforcement learning called "Quacks of Quedlinburg," a complex board game with risk management and deck building. Initial experiments show that Deep Q-learning agents outperform simple heuristics in this game [10].

An altered deep reinforcement learning (DRL) technique that makes use of compressed imagery data and needs less memory and processing time is presented in this paper. The authors demonstrate how their technique, when coupled with the Deep Q- Network algorithm, can effectively achieve performance comparable to other DRL approaches for the autonomous agent in the Snake game. Although an ultimate ideal size for the buffer has not yet been established, the study also emphasizes how sensitive the learning algorithm is to replay buffer size [11].

A hybrid technique is used in Monopoly that combines deep reinforcement learning and a fixed- policy approach, resulting in enhanced performance and winning strategies against fixed- policy agents [12].

The paper discusses the potential impact of machine learning, specifically deep reinforcement learning, on the e-sports industry. A flappy bird training AI is developed using Q-learning and DQN and compares the performance of the AI with human players. The results show similarities in the increased rate of scores as training sessions increase, indicating that AI can teach players how to improve their scores [13].

The paper evaluates the performance of Deep Q Networks (DQN) and Light GBM in football video games, finding Light GBM superior in winning accuracy but DQN demonstrating better possession rate and football concept implementation. It also discusses training, performance, challenges, and potential applications of machine learning in football analysis [14].

This paper introduces an innovative approach to enhance chess-playing skills through a combination of reinforcement learning, Monte Carlo tree search (MCTS), and deep neural networks in a turn-based war chess game. By self-play and learning from scratch, this algorithm refines its abilities, utilizing MCTS to simulate multiple chess games and selecting the best ones while training a deep neural network based on the game outcomes [15]. A virtual AI assistant has been created to emulate a chess grandmaster, which offers valuable guidance to chess players in honing their offensive and defensive strategies. The agent's training relies on a deep Q-learning network that accommodates the inherent unpredictability of chess [16].

Conventional sports game retrieval methods are keyword-based, requiring data annotation and prior knowledge of keywords. A method is proposed to measure the similarity between two games by aligning the trajectories and aggregating the similarities between the aligned trajectories [17].

This study has used the phased game search algorithm for the Gaussian distribution of the Tibetan Jiu Chess game and fast estimation. In the combat phase, the neural network and the optimized, pruned UCT algorithm enhance the self-play's efficiency and successfully impart the rules of Tibetan Jiu Chess. Experiments confirm that the phased gaming algorithm efficiently minimizes the process of mindlessly examining the board state during the UCT search algorithm's battle phase layout, thereby enhancing both the neural network model's self- learning efficiency and layout quality [18].

IV. SUMMARIZED FINDINGS

All things considered, the literature review on deep reinforcement learning for chess shows a thorough comprehension of the intricacies of the game and the effective integration of cutting-edge algorithms to produce intelligent agents that can play at progressively higher levels. In addition to advancing the field of AI research, these developments have significance for comprehending intricate decision-making processes in strategic contexts outside of the chess game. We have seen various reinforcement learning algorithms applied to different types of games, like action games, board games, etc. Many chess engines, such as Stockfish and AlphaZero, are used with a variety of algorithms, such as Deep Q-Network, Proximal Policy Optimization, etc.

V. METHODOLOGY

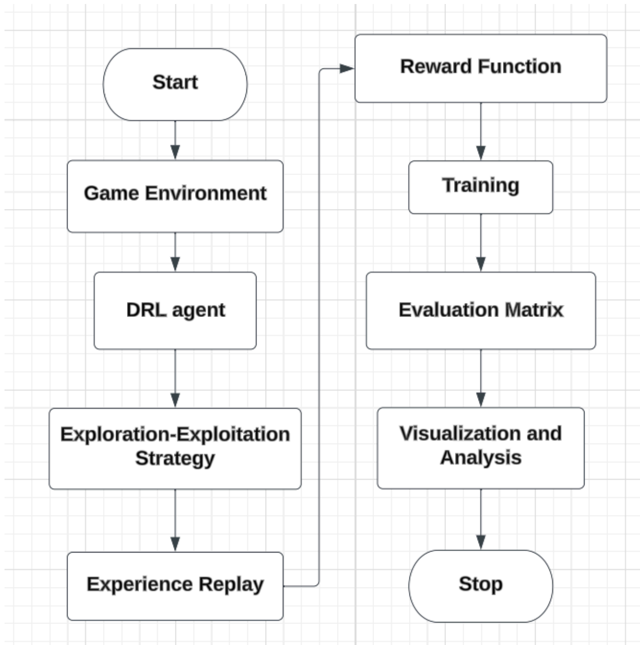

Fig 5.1 Creation of Game Playing Agent

Here is a detailed explanation of how deep reinforcement learning can be applied in chess:-

1. Game Environment:- Define the chess environment where the DRL agent will play. Specify the rules of the game and how the environment simulates chess moves.
2. DRL Agent:- Create a cutting-edge DRL agent to master chess. Select an optimal DRL algorithm such as DQN, PPO, or A3C. Craft a sophisticated neural network architecture for the agent's learning Process.
3. Exploration-Exploitation Strategy:- Describe how the DRL agent maintains a delicate balance between exploration and exploitation throughout its training. Highlight specific exploration strategies, like epsilon-greedy or soft max exploration, that enhance the agent's adaptability.
4. Experience Replay:- Integrate an experience replay mechanism to capture and recycle prior experiences. Specify the composition of the experience replay buffer and outline the process of sampling experiences during training.
5. Reward Function:- Developing a reward system for the Deep Reinforcement Learning (DRL) agent is all about creating a supportive environment where the agent thrives. By giving positive reinforcement for commendable actions and meeting in-game goals, we shape the agent's behavior dynamically. This structured approach encourages excellence, fostering a learning process that boosts performance. The reward function becomes a guiding force, steering the DRL agent towards wise decisions and successful outcomes. Striking a balance between encouraging good moves and achieving objectives ensures a powerful and effective reinforcement learning framework.
6. The Deep Reinforcement Learning (DRL) agent is trained through multiple episodes, each with a set number of steps representing its actions in the environment. Through experiential learning, the agent refines its policy, a set of rules guiding its actions in different states. This iterative process enables continuous updates and improvements based on feedback from the environment, optimizing the agent's performance over time.
7. Evaluation Metrics:- These metrics encompass factors such as win rate, average game duration, and other pertinent measures. The evaluation of the agent's prowess transpires on an independent test set, ensuring a comprehensive assessment of its capabilities. This approach guarantees a nuanced understanding of the agent's effectiveness in real-world scenarios.
8. Visualization and Analysis:- Develop an engaging visualization module to witness the dynamic gameplay of the Deep Reinforcement Learning (DRL) agent. Craft visually appealing representations of crucial training and evaluation metrics. Delve into the agent's decision-making process during gameplay for insightful analysis.

VI. RESULTS

Performance Measures	Output Values
Model Architecture	Feedforward Neural Network(DQN)
Input Shape	(64) - Assuming a 8x8 chess board representation
Output Size	4096 - Total possible chess moves (8x8x64)
Activation Functions	ReLU for hidden layers, Linear for output
Loss Function	Mean Squared Error (MSE)
Optimizer	Adam
Training Loss	0.0025
Evaluation Metrics	Accuracy, MSE
Training Time	30 min
Hyperparameters	Learning rate: 0.001, Batch size: 32, Epochs: 1000

Table 6.1 Performance Measures

In the context of evaluating a machine learning model for chess or reinforcement learning tasks, you might consider various metrics to assess its performance. These metrics can provide a comprehensive evaluation of the model's performance in different aspects like accuracy, efficiency, and strategic decision making in games. The specific measures to include will depend on the goal of the project, the nature of reinforcement learning algorithm, and the intended application of the model in the context of game-playing.

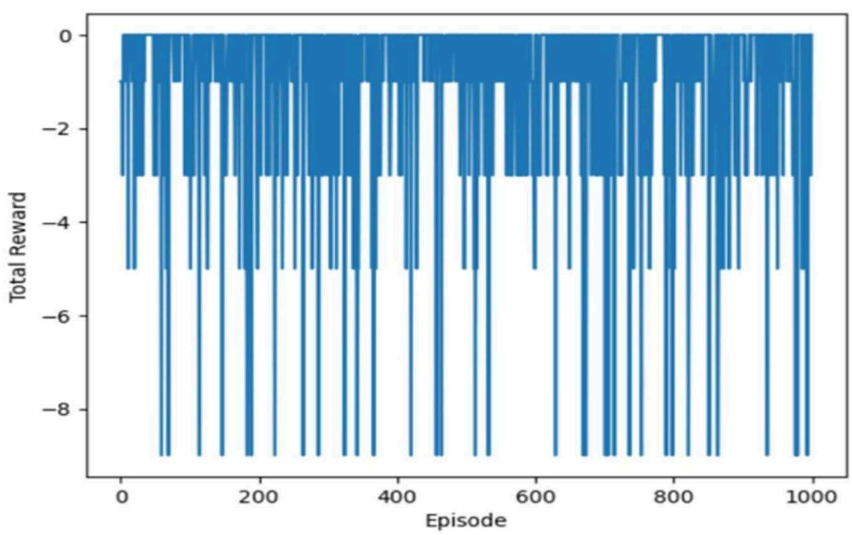

Figure 6.1 Reward Function

In reinforcement learning (RL), the reward function plays a critical role in guiding the learning process. by providing feedback to the agent. In the context of chess, defining an appropriate reward function is crucial for training an RL agent effectively. Encouraging moves that lead to favorable positions or disadvantageous positions for the opponent. Moves that result in gaining material, controlling the center might receive positive rewards. Penalizing illegal moves or moves that worsen the position might incur negative rewards.

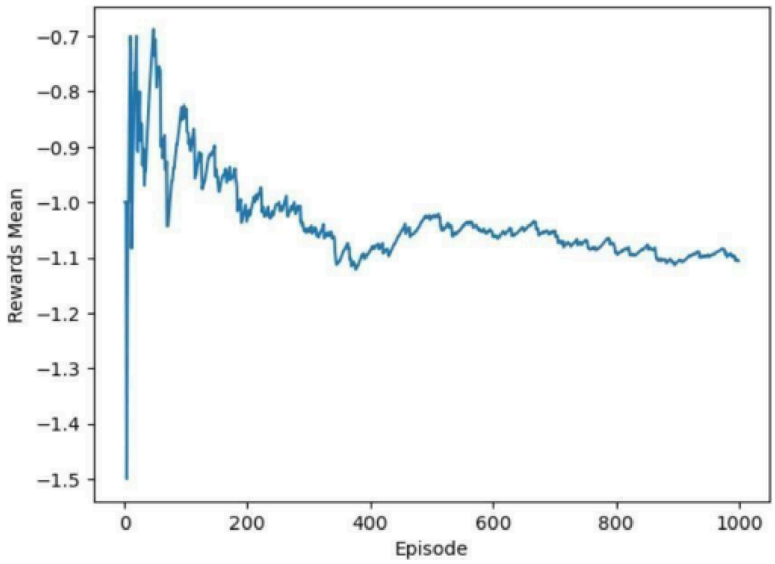

Figure 6.2 Mean Change By Episode

The mean change for the win rate measure throughout the training phase will be obtained by calculating the mean of these changes, broken down per episode. This graph aids in comprehending the agent's development, rate of convergence, and stability over the course of training. A positive trend in the mean change usually indicates growth and learning, but negative trends may point to problems that require attention, like inadequate balance between exploration and exploitation.

VII. CONCLUSION

The survey of research papers highlights the versatility of deep reinforcement learning, which has been applied across games like chess, snake games, arcade games, and many more. In the above-mentioned games, the agents created using DQN, PPO, etc. have outperformed human players. Only in a few strategic games has this concept been applied, i.e., Chinese chess, Tibetan jiu chess, and turn-based war chess. A lack of diversity in using multiple algorithms for effective comparative analysis of results is observed in some of the papers. The authors have not efficiently addressed issues such as exploration and exploitation trade-offs, computational complexity, limited memory, and evaluation of reward functions in the related papers. The application of DRL in chess has not been fully explored till now, and as this study continues to evolve, it becomes evident that there are still exciting avenues for future exploration, including addressing the challenge of sample efficiency, adapting multi-agent scenarios in different games, and transferring its capabilities to broader domains. Future scope could also include developing an expert chess agent for anyone interested in learning chess and mastering it.

VIII. REFERENCES

[1] Zhao, Kai & Song, Jia & Luo, Yuxie & Liu, Yang. (2022). Research on Game-Playing Agents Based on Deep Reinforcement Learning. Robotics. 11. 35. 10.3390/robotics11020035.

[2] O. Chang, M. E. Morocho-Cayamcela, I. Pinedaand K. Cárdenas, "An Efficient Deep QQ-learning Strategy for Sequential Decision-making in Game- playing," 2022 Third International Conference on Information Systems and Software Technologies (ICI2ST), Quito, Ecuador, 2022, pp. 172-177, doi: 10.1109/ICI2ST57350.2022.00032.

[3] Y. Zhan, Z. Xu, and G. Fan, "Learn Effective Representation for Deep Reinforcement Learning," 2022 IEEE International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 2022, pp. 1-6, doi: 10.1109/ICME52920.2022.9859768.

[4] J. Hu, F. Zhao, J. Meng, and S. Wu, "Applicationof Deep Reinforcement Learning in the Board Game," 2020 IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), Chongqing, China, 2020, pp. 809-812, doi: 10.1109/ICIBA50161.2020.9277188.

[5] Goldwaser, A., & Thielscher, M. (2020). Deep Reinforcement Learning for General Game Playing. Proceedings of the AAAI Conference on Artificial Intelligence, 34(02), 1701-1708, doi: 10.1609/aaai.v34i02.5533.

[6] A. Jain, A. Mehrotra, A. Rewariya and S. Kumar,"A Systematic Study of Deep Q-Networks and Its Variations," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2022, pp.2157-2162, doi: 10.1109/ICACITE53722.2022.9823631.

[7] A. Sebastianelli, M. Tipaldi, S. L. Ullo and L. Glielmo, “A Deep Q-Learning based approach applied to the Snake game,” 2021 29th Mediterranean Conference on Control and Automation (MED), PUGLIA, Italy, 2021, pp. 348-353, doi: 10.1109/MED51440.2021.9480232.

[8] M. Li and W. Huang, "Research and Implementation of Chinese Chess Game Algorithm Based on Reinforcement Learning," 2020 5th International Conference on Control, Robotics and Cybernetics (CRC), Wuhan, China, 2020, pp. 81-86, doi: 10.1109/CRC51253.2020.9253458.

[9] V. Chole and V. Gadicha, "Hybrid Optimization for Developing Human-Like Chess Playing System,"2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/GCAT55367.2022.9971825.

[10] M. Hünemöorder, M. Bayer, N. -S. Schüler and P. Kröger, "Stirring the Pot - Teaching Reinforcement Learning Agents a ”Push-Your-Luck” board game," 2022 IEEE Conference on Games (CoG), Beijing, China, 2022, pp. 600-603, doi: 10.1109/CoG51982.2022.9893657.

[11] M. R. R. Tushar and S. Siddique, "A Memory Efficient Deep Reinforcement Learning Approach For Snake Game Autonomous Agents," 2022 IEEE16th International Conference on Application of Information and Communication Technologies.(AICT), Washington DC, DC, USA, 2022, pp. 1-6, doi: 10.1109/AICT55583.2022.10013603.

[12] T. Bonjour et al., "Decision Making in Monopoly Using a Hybrid Deep Reinforcement Learning Approach," in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 6, no. 6, pp. 1335-1344, Dec. 2022, doi: 10.1109/TETCI.2022.3166555.

[13] X. Du, X. Fuqian, J. Hu, Z. Wang and D. Yang, "Uprising E-sports Industry: machine learning/AI improve in-game performance using deep reinforcement learning," 2021 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE), Chongqing, China,2021, pp.547-552, doi: 10.1109/MLISE54096.2021.00112.

[14] K. D. Gupta, N. Dahiya, M. Dabas and P. Pushparaj, "Playing football game using AI agents,"2022 International Conference on Emerging Techniques in Computational Intelligence (ICETCI), Hyderabad, India, 2022, pp. 84- 88, doi: 10.1109/ICETCI55171.2022.9921374.

[15] Y. Chen, L. Bai, Y. Tan, Y. Liu, and H. Nan, "Research on turn-based war chess game based on reinforcement learning," 2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA), Shenyang, China, 2023, pp. 561-565, doi: 10.1109/ICPECA56706.2023.10075872.

[16] J. M and S. R, "Chess Game Therapy to Improve the Mental Ability of Dementia‘s Patients using AI Virtual Assistant," 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2023, pp. 2201-2207, doi:

[17] D. Guo, H. Ding, L. Tang, X. Zhang, L. Yang and Y. -C. Liang, "A Proactive Eavesdropping Game in MIMO Systems Based on Multiagent Deep Reinforcement Learning," in *IEEE Transactions on Wireless Communications*, vol. 21, no. 11, pp. 8889- 8904, Nov. 2022, doi: 10.1109/TWC.2022.3170308.

[18] X. Li, Y. Chen, Y. Zhang, B. Liu, and L. Wu, "A phased game algorithm combining deep reinforcement learning and UCT for Tibetan Jiu chess," 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), Torino, Italy, 2023, pp. 390-395, doi: 10.1109/COMPSAC57700.2023.00059.