Exploration of Game Artificial Intelligence: Key Technologies and Case Analysis

Yang Meng[a]
College of Tourism and Cultural Industry, Guizhou University, Guizhou, China

Keywords: Game AI, Reinforcement Learning, Evolutionary Strategies, Monte Carlo Tree Search.

Abstract: Artificial Intelligence (AI) has seen rapid advancements in recent years, with game AI emerging as a key area for testing and refining AI technologies. Games have become valuable platforms for evaluating AI performance, exemplified by notable successes like AlphaGo and OpenAI's Dota 2 bots. This paper provides a comprehensive review of game AI development, focusing on the background and significance of game-based AI research. The paper is structured to: 1) introduce the foundations of game AI; 2) highlight the key characteristics of games used for AI testing; 3) present core algorithms such as Evolutionary Strategies (ES), Reinforcement Learning (RL), and Monte Carlo Tree Search (MCTS), detailing their basic principles; 4) discuss the practical applications of these algorithms in various games; 5) analyze the strengths and limitations of these techniques. Furthermore, the paper outlines the historical progression of game AI, its broader significance, and identifies the challenges and potential future research directions in this field. The goal is to offer beginners a clear understanding of game AI, while motivating deeper exploration of its technical complexities. Future work will delve into detailed studies of specific algorithms, expanding on their implementation and practical relevance.

1 INTRODUCTION

Games are widely recognized as popular benchmarks for Artificial Intelligence (AI) with known tasks and defined rules (Schaeffer, 2001). Multiple cutting-edge techniques could be applied to combat tasks and finally reach the human-level performance. By human-computer gaming, a wide range of key AI technologies are tested and examined through decades, which have made contribution to the prosperity of AI applications in many industries.

Originating from 1950 when Alan Turing proposed the first method to verify the capability of machines (Turing, 2009), constantly evolving AI algorithms attempted to mimic humans to challenge many later games as different as Chess, Go, first person shooting games (FPS), Real-Time Strategy (RTS) games and Multiplayer Online Battle Arena (MOBA) games. These electronic games can to some extent reduce the cost of physical devices in task simulations and generally provide simulation environments with controllable complexity (Buro, 2004). Under the influence of mutiple factors, thesis have observed technology exhibit remarkable performance. AlphaGo Zero, for instance (Silver et.al, 2017), employing deep learning, self-play, and Monte Carlo Tree Search (MCTS), beated several professional go players and demonstrated effective tactics for large state perfect information games. Moreover, Texas Hold'em (Moravčík et.al, 2017), Starcraft (Vinyals et.al, 2019), Dota 2 (Berner et.al, 2019), HoK and many other games are considered representatives of AI creating milestones in various types of games (Ye et.al, 2020). Besides, attempts that prevent solutions becoming overly focused on a particular kind of game, such as Arcade Learning Environment (ALE), developed by Bellamare et. al. (Bellemare and et.al, 2013), weakens the rule specificity caused by the differences in rules between different games, thus providing AI with the challenge of more generalized ability requirements.

This study focuses on organizing and summarizing the relevant concepts and backgrounds of AI, in an attempt to propose a comprehensive overview of Game AI. The paper delves into an analysis of core technologies and their performances

[a] https://orcid.org/0009-0008-1332-393X

Meng and Y.
Exploration of Game Artificial Intelligence: Key Technologies and Case Analysis.
DOI: 10.5220/0013517500004619
In *Proceedings of the 2nd International Conference on Data Analysis and Machine Learning (DAML 2024)*, pages 349-353
ISBN: 978-989-758-754-2

in the realm of Game AI. Following this, the study evaluates the advantages and disadvantages of these technologies, offering insights into their current limitations and potential improvements. The paper concludes by summarizing the conclusions of the study and outlining potential directions for Game AI development in the future.

2 METHODOLOGY

2.1 Game and Its Features That Challenge the AI

Although different games have different features and test different AI capabilities, there are some features of game that are widely recognized as challenging for AI. This article will introduce these features through the example of the famous video game StarCraft. StarCraft is an RTS game. In this game that can be played against both computers and players, players gather resources on the battlefield to form troops, whose characteristics are determined by the race they control. The victory condition of the game is to destroy the opponent's core base. The popularity of this project has given birth to highly developed professional competitions in South Korea, with many players competing in TV league (OGN, 2019). StarCraft is a typical example of a game with imperfect information, long time horizon, and heterogeneous features.

Since in games with imperfect information, players can only infer the complete information of other players through limited information, the algorithm needs to seek Nash equilibrium, and cannot use Zermelo's theory for perfect information games to find the optimal solution (Schwalbe and Walker, 2001). For instances in StarCraft, different players do not share the same view of the map. The long-time horizon refers to a game that can last for several minutes or even more than an hour. This means that in video games such as StarCraft, an AI system may need to make thousands of decisions for thousands of frames of a game.

The heterogeneous feature means that players have different identities, and each identity has a unique game mechanism. Although the rules of the game are the same for all players, the different identities of the players make each player's strategy different. Take StarCraft as an example, players can choose different races, and different races have completely different hero features and special development strategies.

2.2 Game AI Techniques

This article will introduce three widely used techniques, namely Evolutionary Strategies (ES), Reinforcement Learning (RL), and MCTS. First, this article will focus on explaining the concepts of these techniques and briefly introduce their algorithmic mechanisms, then give their representative applications in game AI and point out their advantages and limitations. Figure 1 shows the basic structure of this section of the article.

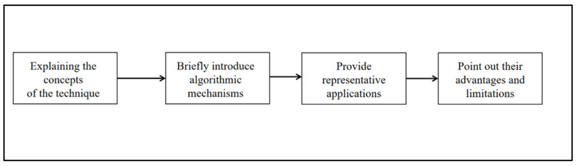

Figure 1: Basic structure of this section of the article (Picture credit: Original).

2.2.1 Evolutionary Strategies

The evolutionary methods are inspired by natural selection. It defines many populations. The populations have chromosomes, which are usually a string of codes that represent the characteristics of the solution and the ability to adapt to the environment. The code is diversified through a defined variation operator, and the gene pool is used to limit the scope of diversity, define the domain of the problem, and limit the space of possible solutions. Then, selective pressure is used to continuously optimize the fitness of the population to the problem (usually represented by a function). Figure 2 shows the structure of the mechanism above.

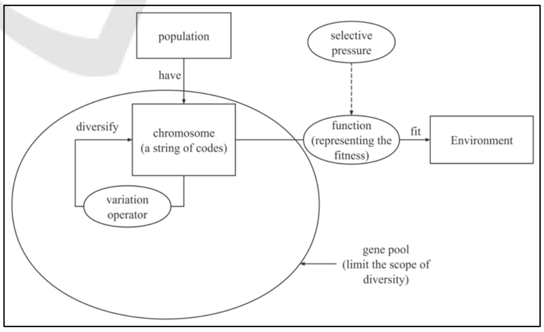

Figure 2: The mechanism of evolutionary methods (Picture credit: Original).

The ES is a highly favoured variant of evolutionary algorithms. In the ES, the problem is defined as finding a real n-dimensional vector x associated with the extreme value of a function, F(x): Rn→R. ES performs similarly to RL in some Atari games. The strategy of the Wargus game was produced by Ponsen et al. using evolutionary algorithms in 2005 (Ponsen et.al, 2005), but The Evolutionary Methods have two limitations: Firstly, the standard deviation of the constant in each dimension (average step size) slows down the convergence to the optimal value; Secondly, the instability of point-to-point search may cause it to stop at a local minimum.

2.2.2 RL

The principal components of RL are an agent, an environment, a state, an action, and a reward. The environment will change to a new state when the agent acts, and it will then signal the new state with a reward (either positive or negative). The Figure 3 briefly demonstrates the mechanism above.

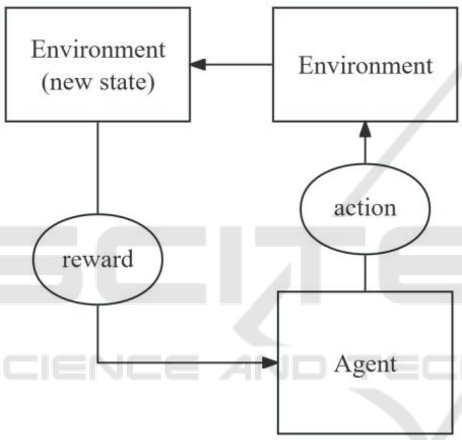

Figure 3: The brief mechanism of RL (Picture credit: Original).

Based on the new state and the reward provided by the environment, the agent then executes fresh actions in accordance with a specific strategy. The interaction between the agent and the environment through state, action, and reward is represented by the above process, which could be viewed as a Markov Decision Process (MDP). Within the field of machine learning, RL stands apart from the more popular supervised and unsupervised learning approaches. RL specifically seeks to identify the best course of action for continuous time series through interactive goal-oriented learning. In comparison, unsupervised learning is the process of identifying hidden patterns in unlabelled data. It typically refers to algorithms like clustering and dimensionality reduction, and supervised learning is the process of learning rules through labelled data, typically referring to regression and classification problems.

Using a RL algorithm, AlphaZero was able to master chess, go, and shogi. RL has also been shown to be effective in tactical decision-making in some RTS games. However, RL produces sample inefficient problems because it needs a massive amount of data for policy learning (Yu, 2018) and RL is rarely used in strategic decision-making due to the delayed reward problem.

2.2.3 MCTS

MCTS is often used in board games, since board games are likely to be perfect information games such as Go, Othello, chess, Texas Hold'em, etc. To put it simply, a perfect information game is one in which every player at any one time has perfect knowledge of every action taken before. But knowing every move does not imply that one can compute and deduce every conceivable result. For instance, there are more than 10^{170} legitimate positions that can exist in Go. In order to choose the most advantageous course of action based on the best simulation results, it simulates both its own and the opponent's conduct in the game beforehand and store the results in the tree. MCTS consists of four steps: selection, expansion, simulation, and back propagation. In each simulation, simulating the end of the round state through rollout strategy, which mainly utilises a concept of Upper Confidence Bound, requires first using tree strategy to select paths in the search tree, then expand a leaf node and using the final score to update the state operation values on the path to complete the simulation.

The field of computer Go made tremendous progress from 2005 to 2015, as demonstrated by AlphaGo, thanks in part to MCTS. MCTS can also be combined with many other AI techniques to achieve outstanding performance. A good example is JueWu's success in RTS games (Yin and et.al, 2023). However, MCTS can hardly obtain the best practices in some games with complex player behaviours, such as Mahjong and DouDiZhu.

3 RESULT AND DISCUSSION

The ongoing confrontation between AI and human players, or game scripts, is a significant driver for the continuous evolution of AI, providing a crucial foundation for research and applications beyond gaming. The advancements in the gaming industry, coupled with the evolution of game mechanics, have not only introduced new research tools for Game AI but also sparked the development of innovative algorithms. These developments have attracted greater public attention, as seen with the societal impact following AlphaGo's success, which catalysed a broader interest in AI research. As the gaming industry evolves, more complex games suitable for AI research continue to emerge, offering AI increasingly challenging environments to navigate. To keep up with the dynamic demands of human players, many games have grown in complexity, presenting new opportunities for AI to demonstrate its potential. Different games, and even distinct tasks within a single game, possess unique characteristics that demand varied AI capabilities. For instance, the strategic decision-making required in StarCraft is vastly different from the tactical responses necessary for other games, illustrating the diverse demands placed on AI. Due to the transparent nature of game rules, AI research benefits from a controlled, low-cost, and easily testable environment, where various algorithmic technologies can be rigorously evaluated. These experimental settings allow researchers to uncover both the strengths and limitations of AI, generating valuable insights that drive further technological advancements.

However, Game AI currently faces several limitations. One of the primary challenges is versatility; many AI algorithms are tailored to specific tasks, and their performance suffers when applied to different games or tasks. Although DRL has become a widely-used paradigm in Game AI, it does not guarantee success across all games. Additionally, AI designed for human-computer competition often struggles to align with the central objective of most games, which is to enhance the player's experience. The economic feasibility of implementing AI in the gaming industry remains a hurdle, as high-level AI development is costly and inaccessible to smaller research teams, despite the decreasing technical barriers. Looking ahead, several trends hold promise for the future of Game AI. These include fostering competitions to promote the development of more versatile AI, creating new performance evaluation metrics, and applying AI technologies to reduce game development costs. The creation of low-resource AI and the development of new, challenging games will also shape the future of this field. These trends aim to address current limitations, pushing the boundaries of what Game AI can achieve while ensuring it remains accessible and practical in real-world applications.

4 CONCLUSIONS

This article offers a comprehensive introduction to the role of AI in human-machine confrontation, particularly within the context of game AI. It begins by discussing the unique features of games that pose challenges for AI development and then introduces the fundamental concepts, core principles, and mechanisms behind representative AI technologies, such as RL and decision-making algorithms. The article examines these technologies in terms of their applications, advantages, and drawbacks, providing a well-rounded perspective on their current capabilities. Furthermore, the limitations of existing game AI systems are highlighted, including issues related to versatility, cost-effectiveness, and applicability in real-world scenarios. The article also outlines potential future trends for the development of game AI, such as the creation of more generalized AI systems, improvements in performance evaluation criteria, and the reduction of game development costs through AI integration. By focusing on providing beginners with a clear overview of the field, this article attempts to enable them to quickly grasp core concepts and motivate them to further learn. In future work, this article will explore more specialized literature, conduct targeted experiments, and provide an in-depth analysis of the implementation details of key technologies, aiming to contribute more comprehensively to the advancement of game AI research.

REFERENCES

Bellemare, M.G., Naddaf, Y., Veness, J., and Bowling, M., 2013. The arcade learning environment: an evaluation platform for general agents. Journal of Artificial Intelligence Research, 47(1):253–279.

Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., et al. 2019. Dota 2 with large scale deep reinforcement learning. arXiv print:1912.06680.

Buro, M., 2004. Call for AI Research in RTS Games. In Challenges in Game AI: Papers from the AAAI Workshop, 139–142.

Moravčík, M., Schmid, M., Burch, N., Lisỳ, V., Morrill, D., Bard, N., Davis, T., Waugh, K., Johanson, M., Bowling, M., 2017. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. Science, 356, 508–513.

OGN, 2019. On gamenet StarLeague. Retrieved from: https://www.chinaz.com/tags/hanguoxingjizhiyeliansai.shtml

Ponsen, M., Muñoz-Avila, H., Spronck, P., and Aha, D., 2005. Automatically Acquiring Domain Knowledge for Adaptive Game AI Using Evolutionary Learning. In

Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, 1535–1540.

Schaeffer, J., 2001. A Gamut of Games. AI Magazine, 22(3), 29–46.

Schwalbe, U., Walker, P., 2001. Zermelo and the early history of game theory. Games Econ, 34, 123–137.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, M., Chen, Y.T., Lillicrap, T., Hui, F., Sifre, L., Van Dendriessche, G., Graepel, T., Hassabis, D., 2017. Mastering the game of go without human knowledge. Nature, 550(7676), 354–359.

Turing, A.M., 2009. Computing machinery and intelligence. In Parsing the Turing Test; Springer: Berlin/Heidelberg, Germany, 23–65.

Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W.M., Dudzik, A., Huang, A., Georgiev, P., Powell, R., et al. 2019. Alphastar: Mastering the real-time strategy game starcraft ii. DeepMind Blog, 2.

Ye, D., Chen, G., Zhang, W., Chen, S., Yuan, B., Liu, B., Chen, J., Liu, Z., Qiu, F., Yu, H., et al. 2020. Towards playing full moba games with deep reinforcement learning. Adv. Neural Inf. Process. Syst, 33, 621–632.

Yin, Q.Y., Yang, J., Huang, K.Q., et al. 2023. AI in Human-computer Gaming: Techniques, Challenges and Opportunities. Mach. Intell. Res. 20, 299–317.

Yu, Y., 2018. Towards sample efficient reinforcement learning. In Proceedings of the International Joint Conference on Artificial Intelligence, 5739–5743.