Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 9
AIDML 2025

Multi-agent Game Theory Applied in Artificial Intelligence

Chang Yuan *

College of Computer Science and Technology, Harbin Engineering University, Harbin, Heilongjiang, China

* Corresponding Author Email: yc18705277753@hrbeu.edu.cn

Abstract. With the further development of machine learning methods, algorithms are in the face of a more complicated environment than ever. The massive amount of data has brought severe pressure to present computing and decision-making strategies. In order to augment the performance of artificial intelligence in multi-agent interacting scenarios, researchers have adopted game theories to machine learning strategies. The article introduces the application of game theory from three aspects: the theoretical basis, the application fields, and the future prospects. The main purpose of the article is to summarize the current research of applying game theory to reinforcement learning algorithms and how the theory improves the performance of the witch. It is hoped that the article can provide a guide for interested researchers, and help push forward the achievements of the field. The research on multi-game theories not only improves the ability of logistic game related algorithms, but also positively pushes the development of very advanced artificial intelligence further.

Keywords: Game theory; multi-agent reinforcement learning; machine learning; artificial intelligence.

1. Introduction

With the rapid development of artificial intelligence and machine learning, relevant fields have become a major focus of current research. Multi-Agent Game Theory, which is widely applied in areas such as transportation, strategy games and robotics. Multi-agent system (MAS) is a complex system consisting of several game-theoretic agents [1]. Each of the agents possesses individual goals and strategies.

Multi-Agent Game Theory is developed based on game theory. It all began in 1950 when Nash introduced Nash Equilibrium [2]. Since which game theory has come into application in decision making. The theory was originally applied in chess playing programs. In 1951, Christopher Strachey developed the first computer program capable of playing checkers. In 1959, Arthur Samuel developed a checkers-playing program capable of learning from experience, which is one of the earliest examples of machine learning. In 1997, Deep Blue established by IBM defeated the world chess champion Garry Kasparov, showing the ability of artificial intelligence combined with game theory. In recent years, Multi-Agent Game Theory has been integrated with machine learning methods and gave birth to various applications such as Multi-Agent Deep Reinforcement Learning [3]. Such methods enable machines to solve problems with gaming strategy. Compared with single-agent systems, MAS is capable of tackling highly complex, dynamic decision-making problems. The applications of multi-agent game models are significantly outstanding in the following domains: transportation, robotics and game industry. Multi-agent game models have demonstrated great potential in real-world applications. It is highly promising to investigate further into the field and combine the strategy with deep learning models. There are currently multiple challenges remaining to be solved, such as the enormous complexity. It is essential to push forward the breakthrough of the research.

This paper reviews Multi-Agent Game Theory from three main perspectives: The fundamental theory, realistic applications and future investment. The main purpose of the paper is to present a thorough review of MAS, and inspire researchers to devote greater interest to relevant study.

2. Theoretical Foundation

2.1. Nash Equilibrium

Nash Equilibrium is an essential fundamental theory raised by Nash, J. F. in 1950. The theory describes a state where each agent selects the best strategy, and none of the participants can improve their payoff by changing the strategy supporting other participants to stay fixed.

That is to say, mathematically, given a state of game, for player i with strategy s_i, strategies of other players s_{-i} and a strategy s_i^*:

$$u_i(s_i^*, s_{-i}) \geq u_i(s_i, s_{-i}), \quad \forall s_i \in S_i \tag{1}$$

The formula indicates that each earn of a player is a loss for others.

Nash Equilibrium exists extensively for finite games. It reveals the criteria for how agents select the superior strategy. The theory is widely adopted in machine learning tasks.

2.2. Zero-Sum Game

A Zero-Sum Game is a game in which the payoff compensates for the loss of other players. In such a situation, the total gain of the agents remains a fixed figure. Given the payoff of each player as $u_i(s_1, s_2, ..., s_n)$, the total payoff satisfies:[4]

$$u_i(s_1, s_2, \ldots, s_n) + u_j(s_1, s_2, \ldots, s_n) = 0, \quad \forall i \neq j \tag{2}$$

That is to say any gain of a player will be a loss of another.

The zero-sum can be explained as the sum of a Nash Equilibrium added up to zero. Zero-sum game is commonly adopted in game trees. It plays an important role in the development of game theory.

2.3. Markov Game

Markov game is A game theory describing the Markov decision process (MDP). It is commonly used to tackle multi-agent decision-making, where the environment is rather probabilistic, such as reinforcement learning scenarios [5]. Compared to traditional games, Markov game shows state-dependence and temporal dynamics, which enable it to be applied in complex situations. There are several key factors that make a difference. Action Space is the set of actions that are available to the agent in each state. State Transition Function shows the probability of transitioning from one state to another given a joint action a. The function illustrates the transition from state to state. Reward Function describes the immediate reward when an action is taken. Discount Factor determines how potential rewards weigh compared to immediate ones. Finally, Policy is a mapping from states to a probability distribution over actions for an agent in Markov strategy. Policy derives only from the current state. Markov game combines game theory and reinforcement learning methods. The states in the Markov game always evolve over time. It can be seen as an expansion of traditional static gaming. The main equilibrium purpose of the Markov game is still to reach the Nash Equilibrium. Markov Nash Equilibrium demands all the participants adopt the Markov strategy, which only depends on the current status, and each be the optimal counterpart [6].

3. Application

Multi-agent games have been proven to be of great use in various fields. Focusing on complicated situations in terms of multiple units interfering with each other. For instance, transportation problems are in the face of massive complexity, consuming each vehicle as a player in a game. It is of great use to introduce the theory into traffic systems or city management. Similarly, multi-agent can be adopted into the robot industry and game industry.

3.1. Internet of Things

The Internet of Things (IoT) is a typical domain for game theory. Cloud computing is considered promising to improve the processing ability and storage capacity of IoT. However, cloud data centers are often located in remote areas, leading to delays while transmitting data. On the contrary, IoT is in command of real-time feedback [7].

In the last few years, researchers have begun applying game theory to IoT. It is considered of great use to analyze and optimize the system of IoT. In 2024, Zaineb Naaz and his team introduced a load-balancing mechanism in an IoT-edge environment (LMGT) established on intelligent game theory [8]. It augmented the Quality of Service (QoS) to the extreme and better manages the resources of the edge nodes. The method established an authentic decision-making framework to balance the load of the edge nodes. Researchers developed a systematic load-balancing algorithm that assessed the utilizable resources accurately and generated viable solutions accordingly to advocate the load of edge nodes. Edge computing improves efficiency by reallocating the computer process at the edge nodes [9]. The method includes three types of nodes at the beginning: n IoT nodes, m edge nodes, and cloud centers within a specific region. Each of them holds ample resources to process data from the IoT nodes. IoT nodes compare the distance between each edge node to create a priority list. Every time an IoT node establishes a routine targeted at the first node in the list and transfers data accordingly. The available resources of the edge nodes are calculated for every time ‘t’. The decision-making model equipped with load balancing game strategy will be activated to label edge nodes as ‘adequate’ or ‘depleted’ once the resource value has been quantified. If the result is ‘adequate’, the current path will remain substantial. Otherwise, it will be removed from the network and from the lists of the existing IoT nodes for a period of time called ‘renew time’. If all of the nodes are ‘depleted’, the system will establish a direct path between IoT nodes and the cloud center, ignoring all intermediary edge nodes along the route.

3.2. Smart City

As urbanization becomes the mainstream of global construction, urban managing strategies have been placed in an important status. Various industries are merged via information technology, which further provides better livelihood services for citizens [10].

However, the increasingly complicated internet system has brought greater difficulty in management simultaneously. To deal with the vast quantity of information, artificial intelligence has been deployed comprehensively in smart city construction. Reinforcement learning plays an important role in artificial intelligence. The key to reinforcement learning is multiple agents adopting different actions to interact with the environment. Therefore, they are able to actively learn the prior strategy adjusting to various environments. Plenty of scenarios in smart cities concern the interaction between multiple intelligent agents, such as the scheduling of vehicles in the Internet of Vehicles, and monitoring resources. To augment the performance of Reinforcement learning, researchers have applied multi-agent game theory to the strategy. Multi-agent game theory is capable of improving the algorithm of edge computing and thus enhancing the performance of IoT, which is an important method for applications such as data management.

The most commonly used Supervised learning and Semi-supervised learning methods rely on large amounts of data to conduct training. On the contrary, Reinforcement Learning based on Markov Decision Process can effectively avoid such problems. The intelligent agent in the reinforcement algorithm learns the optimal strategy by interacting with different environments. Agents make their decisions by observing the statement of the environment, and then adjust behaviors according to the environmental feedback. Since reinforcement learning does not depend on trainsets, agents demand exploring different actions and adjusting their strategies to find out the solution that maximizes the benefit. Common reinforcement learning models can deal with multi-agent problems, but they require a central controlling system to control all the targets. Such progress demands strictly on the stability, coverage, and instantaneity. Therefore, common reinforcement learning is not suitable for scenarios where communication is limited. In order to solve the listed conundrums, researchers invented Multi-Agent Reinforcement Learning. Each intelligent agent collects the relative status with sensors. The data is then introduced into the decision-making system to generate the responding action. After all agents finish the actions, they each collect the result and the changed environment status. The agents further upgrade the decision-making systems according to the new status.

Multi-agent reinforcement learning method collects various dimensions of urban data with sensors. Researchers are able to discover all kinds of problems in time and make more efficient decisions. Wang and his group have applied multi-agent reinforcement learning to schedule drones to assist agencies operating air-ground joint missions.

3.3. Entertainment Computing

Game theory originates from logistic games. Obviously, it can have great potential in entertainment computing. Reinforcement learning models combined with multi-agent game theory are appropriate for strategy games and robot designing.

Intelligent game agents adopt artificial intelligence to stimulate player behaviors and autonomously make decisions [11]. Intelligent game agents originate in chess games and have broadened to a wide range such as anticipating player behaviors, content generation, and result prediction [12]. The best-known milestone of intelligent game agents was in 2016 when AlphaGo from Google defeated Lee Sedol, one of the greatest chess players alive. Tracing back to the 1950s, these agents have iterated to outperform human beings in diverse types of games from real-time strategy games like StarCraft to Chinese card games such as Landlords and Mahjong.

The development of intelligent game agents has introduced massive classic methods and algorithms, generally divided into two groups: search and learning. The search method systematically finds solutions in the state space to assist intelligent agents in making decisions. The learning method empowers agents to improve their performance by learning from the experience and data.

Monte Carlo Tree Search (MCTS), introduced by Coulom, is widely adopted in searching methods, replacing the former position of the Minmax strategy. MCTS mainly runs through four stages: selection, expansion, simulation, and back-propagation. In-game agent tasks, MCTS nodes represent the various states of the game, while the root node symbolizes the present state and subsequent nodes stand for the potential states derived from the root node. The algorithm expands based on the subsequent nodes selected by the preset strategy until it reaches the defined search depth or encounters a node ready for expansion. The model simulates and evaluates the value of the nodes and feedback the results to the parent node. MTCS optimizes the strategies and valuation metrics through sustained simulation and updates.

4. Future Prospect

The algorithms combined with game theory strategy have so far performed outstandingly. Applications such as edge computing and Alphazero have redefined the capacity and efficiency of artificial intelligence. It can be concluded that these models have reached an extreme in learning and making decisions. What remains unsolved is that the current hardware conditions are not sufficient to support the full computational potential of these advanced algorithms. For example, alphazero has pushed the capacity of MTCS to a limit. MTCS is a commonly utilized method in strategic games like cards and chess. Within the process of MTCS, trillions of data and states have to be selected and back-propagated, which brings a mass burden for even the most advanced processor. Considering the compatibility of hardware cannot leap in a period, the best solution is to perfect the structure of the algorithm. The current relevant researches most focus on paralleling the Monte Caro Tree to relieve the pressure of the server. Yashar Naderzadeh and his team raised a method called PPB-MCTS in

April 2025[13]. It is designed to uncover sub-optimal actions at each decision-making point and accordingly improve the Methods for Pruning and back-propagation. Thus, the strategy can impressively improve the efficiency of training and processing of the MTCS algorithm.

5. Conclusion

This paper explored the theoretical foundations and practical applications of multi-agent game theory, focusing on its essential role in modifying strategic interactions among intelligent agents. By concluding the applications of the strategy, the paper reveals how game theory can function in edge computing and multi-agent reinforcement learning progress. Through case studies in fields like smart cities and entertainment computing, the paper reviews the real-world relevance of multi-agent game theory. Future work will focus on detailed method implementation and specific cases with typical models adopting game theory. This paper aims to provide a comprehensive overview of multi-agent game theory by bridging foundational concepts with practical applications across diverse domains. By highlighting its relevance in both theoretical development and real-world scenarios, the study contributes to a broader understanding of how strategic decision-making unfolds in complex, interactive environments. It is hoped that this work will offer valuable insights for researchers and practitioners alike, and serve as a stepping stone for continued exploration in this evolving field.

References

[1] Li Yichun, Liu Zejiao, Hong Yitian, et al. A Survey on Game Theory Based on Multi-Agent Reinforcement Learning. Acta Automatica Sinica, 2025, 51 (3): 1 – 19.

[2] Nash Jr J F. Equilibrium Points in N-Person Games. Proceedings of the National Academy of Sciences, 1950, 36 (1): 48 – 49.

[3] Gronauer S, Diepold K. Multi-Agent Deep Reinforcement Learning: A Survey. Artificial Intelligence Review, 2022, 55 (2): 895 – 943.

[4] Friedman S D, Christensen P, DeGroot J. Work and Life: The End of the Zero-Sum Game. Harvard Business Review, 1998, 76: 119 – 130.

[5] Littman M L. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In: Machine Learning Proceedings 1994. Morgan Kaufmann, 1994: 157 – 163.

[6] Hu J, Wellman M P. Nash Q-Learning for General-Sum Stochastic Games. Journal of Machine Learning Research, 2003, 4 (Nov): 1039 – 1069.

[7] Naaz Z, Joshi G, Sharma V. Load-Balancing Model Using Game Theory in Edge-Based IoT Network. Pervasive and Mobile Computing, 2025, 109: 102041.

[8] Naaz Z, Sharma V. Performance Analysis of Edge Node in IoT Network. In: 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT). IEEE, 2024, 5: 546 – 551.

[9] Garcia Lopez P, Montresor A, Epema D, et al. Edge-Centric Computing: Vision and Challenges. ACM SIGCOMM Computer Communication Review, 2015, 45 (5): 37 – 42.

[10] Du Wei, Ding Shifei. A Survey on Multi-Agent Reinforcement Learning. Computer Science, 2019, 46 (8): 1 – 8.

[11] Yildirim S, Stene S B. A Survey on the Need and Use of AI in Game Agents. InTech, 2010.

[12] Li H, Pang X, Sun B, et al. A Concise Review of Intelligent Game Agent. Entertainment Computing, 2024: 100894.

[13] Naderzadeh Y, Grosu D, Chinnam R B. PPB-MCTS: A Novel Distributed-Memory Parallel Partial-Backpropagation Monte Carlo Tree Search Algorithm. Journal of Parallel and Distributed Computing, 2024, 193: 104944.