Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/37/20230486

A research of artificial intelligence game agent application

Yuqi Lan[1], Zhenghao Li[2, 3]

[1]HDU-ITMO Joint Institute, Hangzhou Dianzi University, China
[2]School of Public Policy and Administration, Xi'an Jiaotong University, China

[3]zhlz233@stu.xjtu.edu.cn

Abstract. Currently, large language models are on the rise with breakthrough progress in artificial intelligence. Existing reviews of AI game agents have not covered these latest developments, requiring a combing and analysis of the newest research advancements in game AI agents. This paper summarizes the application scenarios of game AI agents in four aspects: combat AI, Non-Player Character (NPC) interaction, automated testing, and Artificial General Intelligence (AGI) testing. In combat AI, there is a progressive developmental trend, with the introduction of Monte Carlo tree search and reinforcement learning enabling AI game agents to fully surpass humans in traditional board games. In NPC interaction, full AI is unnecessary. Game developers only need to incorporate AI for abilities related to player experience to increase appeal, with controllable generation results. In automated testing, game AI agents lack generalizability for testing so far. In AGI testing, academia has helpfully explored general game AI, but capabilities remain limited to certain games. Introducing large language models to game AI agents shows unprecedented capabilities. Finally, this paper provides an outlook on the hot topics and future directions of this research subject.

Keywords: AI game agent, NPC interaction, automated game testing, artificial general intelligence testing.

1. Introduction

Since 2023, there has been rapid development in the field of artificial intelligence, especially in large language models and multimodal models. The complex logical reasoning abilities of "GPT-4"have reached the top human level in some areas. Artificial intelligence has long been used as intelligent agents to play games. However, it hasn't been long since artificial intelligence started defeating top professional human players. "Deep Blue"[1] first defeated a professional chess player in 1997. Before the advent of "AlphaGo"[2], defeating human professional Go players with AI was considered impossible. This boosted researchers' confidence in realizing more powerful artificial intelligence. Now, there is emerging research combining large language models and game-playing agents, such as "Voyager" which leveraged GPT-4 to actively explore and improve skills in the Minecraft world [3].

Research on AI game-playing agents is thriving. Cutting-edge research has become quite different from earlier models based on search algorithms. Instead, it combines multiple AI methods and enables general game intelligence with the support of multimodal models. However, existing review studies are mostly limited to board games and video games characterized by competition and strategy [4]. They have not covered survival games, leaving great potential for supplementary research. Games can create simplified human worlds. Moreover, OpenAI Five has demonstrated the ability to beat the

DOI: 10.54254/2755-2721/37/20230486

professionals at complex video game Dota 2 [5]. The performance of game-playing agents in them can not only improve the efficiency of automated game testing, but also transfer to robotics, social science simulation, and more. There is great research potential.

In summary, current research has not kept up with the latest developments in this field. There have been transformative technologies, necessitating a review and analysis of the progress in game AI agents. This article summarizes the application scenarios of game-playing agents in terms of combat AI, NPC interaction, test automation, and AGI testing. It also analyses and summarizes these scenarios, proposing future outlooks.

2. Application scenarios

2.1. Combat AI

The development of combat AI has progressed from simple to complex. Pathfinding algorithms are the most basic, only requiring a search for reachable paths, laying the foundation for search algorithms like depth-first search, breadth-first search, and A* search [6]. After that, the academic world long sought breakthroughs in board games, as they have turn-based gameplay and a relatively small action space per turn, reducing the computational load of searching [7]. Using the Minimax algorithm to deduce victorious decisions, Deep Blue, which first defeated top professional human players, implemented this with Alpha-Beta Pruning and prior chess knowledge [1]. However, this approach does not apply to Go, which has a larger action space that makes searching too time-consuming. AlphaGo addressed this issue by using Monte Carlo tree search and neural networks, defeating the world champion Ke Jie at the time. AlphaGo Zero further introduced reinforcement learning, enhancing AlphaGo's abilities even more [2]. Since then, AI has been able to completely defeat humans in traditional board games.

Video games pose even more complexity, with both turn-based and real-time confrontations, drastically expanding the action space for agents. Take League of Legends as an example. The movement direction of heroes is continuous, with complex movement speeds influenced by many factors like hero type, active skills, passive skills, items, summoner skills, runes, buffs, ally skills, and enemy skills. Each hero also has 4 different active skills. There are many skill choices, and the direction of non-targeted skills is also continuous. Decision making in the game is very complex. Such games currently use deep reinforcement learning for training, like OpenAI Five which performed remarkably in Dota 2[5]. It is even more complex in FPS games, set in 3D environments and combining computer vision to extract information, training AI agents with deep reinforcement learning [8].

The methods used in the above cases exhibit a progressive developmental trend, with important turning points. The Minimax algorithm elevated the capabilities of game AIs to play adversarial games. The introduction of Monte Carlo tree search and neural networks allowed searching without traversing enormous trees, greatly reducing the computational scale. With scarce high-quality human game data, introducing deep reinforcement learning allowed agents to improve beyond human levels through self-play. At this stage, model training takes long, and agents have high computational overhead when running. The generalization capabilities of models are also poor.

2.2. Application of ai agents in non-player characters (NPCs)

In current commercial applications, the ways Non-Player Characters (NPCs) interact with players are relatively fixed. Most games have interactions designed by developers in advance, including dialogues between NPCs and players, as well as facial expressions and actions during conversations. Alternatively, they make enemies appropriately challenging without being completely unbeatable for players. Therefore, interactions with NPCs in most games have the characteristics of being mechanical, standardized, and easy for developers to control. Although fully AI-driven NPCs can bring freshness and interactive fun to players, it is unrealistic to fully automate NPCs in terms of commercial application and hardware.

DOI: 10.54254/2755-2721/37/20230486

Commercially, with NPCs fully controlled by AI, their behaviours would be unpredictable [9], posing great challenges for developers. In terms of hardware, neural network models require extensive computational resources. For example, OpenAI used large amounts of CPUs and GPUs for Dota 2. For players' hardware capabilities, fully AI NPCs are not realistic. Thus, NPCs are currently only partially AI-driven in games.

Emotion generation AI. Emotional NPCs can build stronger virtual relationships with the player, making the game's story compelling. FAtiMA, built on cognitive science theories, is an important AI model for NPC emotion generation and has been commercially applied. Corresponding open-source toolkits have also been developed [10]. In 2023, breakthroughs in large language models enabled game NPCs to leverage models like NVIDIA's NeMo, exhibiting natural and rich linguistic capabilities. Developers can also freely customize their language styles [11].

Simple behaviour AI. Freeing NPCs from fixed behaviour scripts and providing rich and varied feedback on player behaviours can greatly increase the player's freedom in the game and improve the game's playability. Research suggests that in games with combat scenarios, NPCs designed manually by developers lack playability, while AI-developed NPCs are more favoured by players [12]. NPC behaviours should be controllable for developers, so behaviour trees are often used to control AI NPCs. To further enhance the determinism of AI NPC behaviours for developers, the advent of trained behaviour trees optimized this [13].

In summary, fully AI-driven is not necessary for NPCs like it is for humanoid robots. Game developers only need to incorporate AI into abilities relevant for player experience to increase appeal, with controllability over AI outputs.

2.3. Application of AI game agents in development testing

As a software product, game usually contains very complex operation mechanism and content structure, and needs to adapt to the user's complex software environment and hardware conditions, and often needs to be updated and maintained by the publisher for a long time, so game testing is very necessary. Conventional human playtesting is often costly and time-consuming, and testers develop path-dependent gameplay over time, making it difficult to exhaust all possible states of the game. Therefore, it is necessary to use intelligent agents to perform automatic game testing. Unlike combat AI that aims for optimal performance, AI game agents in development testing are for tasks like exploring, validating, balancing, and finding bugs. This research is greatly significant, as it can transition test engineers from tedious testing to more productive development work, improving industrial productivity.

In 2D video games, researchers have compared human test engineers and AI game agents based on Monte Carlo tree search (MCTS) in bug finding capabilities, discovering AI agents have abilities approaching human test engineers [14]. For 3D games, research found reinforcement learning can achieve automated testing in test environments but has limitations in transferability to other 3D games [15]. Game testing involves more than the above. Academia has proposed a tool called RiverGame for comprehensive automated testing of sound, graphics, haptics, gameplay, and more [16].

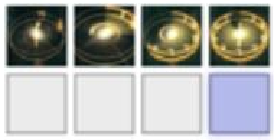
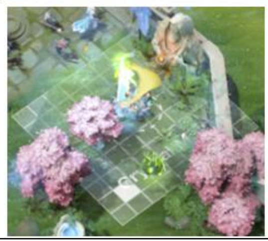
However, using AI for automated testing is not yet industry-ready. Research has found low willingness among game companies to use AI game agents for testing. Main reasons are the inability to integrate AI testing into development workflows, and lack of readily available general testing tools, requiring custom in-house development at high cost [17].

In summary, academia has explored using AI game agents for game testing to some extent, but challenges remain. Developing testing tools requires specialized model training for each game, lacking generalizability. Some models require the player's game data to train, which does not prevent the player from playing a bad game. In addition, many games introduce events that contain randomness, which makes the game performance not conform to the AI agent's expectations. And finally, testing capabilities of AI game agents also have limitations. Gameplay experience involves subjective human emotional judgment, which cannot be tested even with mature AI automated game testing, because AI

DOI: 10.54254/2755-2721/37/20230486

agents can't play games like human minds. But it also allows AI agents to find problems that would be hard for human testers to think of. Table 1 shows the action parameters of OpenAI [5] .

Table 1. OpenAI five action selection parameters [5].

Action Parameters	Usage	
Delay	In order to tell agent when to take action	
Unit Selection	In order to select target	
Offset	The offset is expressed in coordinates	

2.4. Artificial general intelligence in game

Artificial General Intelligence (AGI) is one of the ultimate goals of AI research, and games are a great platform to explore it. First, a game is a carefully designed cognitive environment in which players learn and have fun while playing, which means that the game is a sampling of the real world. Second, the game is a controlled environment, and researchers can design problems of varying difficulty for the agents in it. Finally, with the help of various game engines, many costly tests can be simulated in games. Compared to using robots to test AGI in real life, using agents to test AGI in games has obvious advantages.

Building universal game agents that excel in all games remains an elusive dream. Current general game AI can only perform remarkably in some games. A groundbreaking general agent is AlphaZero, which surpassed all previous algorithms in chess, shogi and Go [18]. This was further developed into MuZero, extending it to 2D video games [19]. The GVGAI (General Video Game AI), which contains hundreds of 2D arcade-style games described in specific languages, provides a unified benchmark for evaluating general game AI, but participating agents have relatively low win rates [20]. However, these studies focus on the ability of agents to solve problems, and the harder task is to actively recognize problems. To test stronger AI, the agent should not be confined to a specific framework, it needs to recognize the game environment through vision, hearing, and natural language, just like a human. In addition to using traditional pixel-based vision perception, Gaina et al.'s research built general game AI using only audio [21].

With the advent of large language models exhibiting unprecedented generalization capabilities, researchers are inspired to build general game AI with them. Voyager first leveraged leading large language model GPT-4, surpassing all prior Minecraft agents. Compared to previous algorithms, Voyager obtained the most items, progressed fastest in skills, and explored the widest [4]. Park et al. built a virtual town with inhabitants driven by a large language model, enabling natural language interaction with players to create unprecedented general game AI [22].

DOI: 10.54254/2755-2721/37/20230486

In summary, academia has conducted beneficial explorations into general game AI, but capabilities remain restricted to certain games and certain frameworks. While studies based on images and sound as input compensate for this shortcoming, these studies rarely involve the exploration of intelligence itself. With the emergence of large language models, incorporating them into game AI has demonstrated unprecedented capabilities. However, the heavy computational burden of large language models makes it impossible to deploy locally, limiting the performance and scenarios tested.

3. Open problems and outlook

Based on current progress in game AI research, applications are in algorithm evaluation, enhancing interactivity, automated testing, and general AI testing. There remain many challenges in this field, mainly around requiring human intervention for game state acquisition, lack of desired generalizability in game AI, and limited applicability.

A key aspect of game AI is acquiring game states to inform decisions. This complexity varies for different games. For Go, only the positions of black and white pieces are needed, e.g. AlphaGo uses 19x19 pixel images. For video games, state acquisition is extremely complex and differs by game, posing challenges for model generalization. Taking League of Legends as an example, state parameters include agent position, health, shields, etc. Full state information is also unavailable in video games, limited to observable areas. Thus, intricate game states are an obstacle for both academia and industry, resulting in large neural network inputs for video games. Game state acquisition still lacks intelligence, requiring human judgment and analysis. Current general game AI is still limited to certain game genres. While large language models have strong generalization capabilities, their slow response makes them unsuitable for real-time games.

With the increasing popularity and demand for AI in NPCs, the development of NPC agents is gradually moving towards general purpose and easy deployment. In 2023, NVIDIA introduced undisclosed commercial products enabling natural language NPC interactions and generating simulated facial movements from audio. Unity's ML-Agents greatly simplified NPC AI development without requiring machine learning expertise[23]. In general, in the process of interaction between NPCs and players, most studies focus on the authentic and credible feedback of NPCs to players, but the player's information is rarely perceived by NPCs. In reality, people can improve the efficiency of interactive information transfer through tone, expression, movement, etc., but the player's interaction with NPCs is mostly limited to character control and optional choices. Although there have been attempts to capture player expressions, movements, and biometric signals to assist NPCs in their interaction with players, the research is still lacking. Moreover, this is an area where academia is way ahead of commercial applications. While both add to the believability and immersion of the game, it's clear that people are more interested in beautiful graphics than more intelligent NPCs. With the advancement of multimodal research, NPCs will also develop towards multimodality. Then the unpredictability of AI-driven NPC agent behavior outcomes will remain an important research direction, and enhancing NPC AI agent interactivity will also increase uncontrollability. Therefore, in-depth research is still needed in this field.

Currently, game AI can only assist development testing, not fully automate it. Many factors restrict full automation. Firstly, general game AI is still lacking, preventing guaranteed generalizability of automated testing tools. Many test agents rely on a specific test framework or even a specific descriptive game language. On the other hand, in order to make a distinctive game, developers often develop their unique game engine, which makes it difficult to rely on the underlying interface of the game for a generality implementation. Secondly, game development involves multimodal data, while multimodal research is still nascent. As the mainstream games that players love move from simple 2D to complex 3D games, agents can no longer rely on logical calculations for testing, but are forced to make decisions by combining images, sounds, and words like humans do. Thirdly, fully automated testing requires software engineering integration, not just post-development evaluation. Testing during development is necessary to ensure that the original error is corrected in a timely manner and that the whole thing is not broken, but a new development process is needed to match it.

Although the large language model is considered the most promising model for implementing AGI, its input and output are mostly limited to text. Existing AGI studies using large language models usually require the input from other modes into conceptual text, and then the output text into other modes. This complex transformation not only leads to the loss of information, but also to low efficiency. Fundamentally, models that can directly understand images, sounds, and other modal data in the same way that large language models understand text have not yet been realized. Moreover, even if we were able to test AGI agents in games, it would still be an unpredictable challenge to transfer them to the real world. Some cognitive scientific properties of the game environment need to be defined so that they can be referenced to reality.

Future applications of game AI. There is currently no game AI to train professional esports players. While traditional board games use game AI to enhance professional player skills, modern video game esports has yet to utilize AI training for professionals, with promising prospects. In social sciences, experiments are sometimes ethically prohibited, but academia has simulated human interactions with large language models[22]. Using game AI for survival in virtual worlds can enable social science simulations, e.g., predicting policy effects by implementation in the virtual world, avoiding ethics concerns of real-world trials. This aids research institutions and government policymaking. Game AI also provides testing environments for risky robotics like self-driving cars, ensuring safety prior to real-world trials.

4. Conclusion

With the background of immense progress in AI, this paper reviews game AI agent literature, summarizing key research areas of combat AI, development assistance, and general game AI. The fastest progress has been in the past decade, achieving many breakthroughs with reinforcement learning. Current limitations around requiring human intervention for game state acquisition, inadequate general game AI capabilities, and narrow applicability are also highlighted.

Future game AI agent research will trend towards generalization and stronger 3D game capabilities. Achieving generalization will enable realistic automated testing. This not only represents a breakthrough in game AI, but the goal pursued across all AI fields. While large models provide a glimpse towards generalization, their computational expense makes them unsuited for real-time games. This can be addressed through hardware improvements or model optimization. Most current game AI focuses on 2D games, with some implementations in FPS and Minecraft. Future research should target 3D game breakthroughs. Reinforcement learning is well-suited for the low-data regime of game AI agent training and should continue being applied until disruptive new techniques emerge. Mature AI agent automated testing can be realized on top of progress in these areas.

Authors contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Campbell M, Hoane A J and Hsu F 2002 Deep Blue *Artif. Intell.* **134** 57–83
[2] Silver D et al 2016 Mastering the game of Go with deep neural networks and tree search *Nature* **529** 484–489
[3] Wang G, Xie Y, Jiang Y, Mandlekar A, Xiao C, Zhu Y, Fan L, and Anandkumar A 2023 Voyager: An Open-Ended Embodied Agent with Large Language Models Preprint arxiv/2305.16291
[4] Lu Y and Li W 2022 Techniques and Paradigms in Modern Game AI Systems *Algorithms* **15** 282
[5] OpenAI et al 2019 Dota 2 with Large Scale Deep Reinforcement Learning Preprint arxiv/1912.06680
[6] Webster R W 1991 Useful AI tools-a review of heuristic search methods *IEEE Potentials* **10** 51–54

DOI: 10.54254/2755-2721/37/20230486

[7] Yannakakis G N and Togelius J 2018 *Artificial Intelligence and Games* (Berlin: Springer)

[8] Huang S, Su H, Zhu J, Chen T 2019 Combo-Action: Training Agent For FPS Game with Auxiliary Tasks *Proc. AAAI Conf. Artif. Intell.* **33** 954–961

[9] Kopel M and Hajas T 2018 Implementing AI for Non-player Characters in 3D Video Games 2018 *Intell. Inf. Database Syst.* 610–619

[10] Mascarenhas S, Guimarães M, Prada R, Santos P A, Dias J and Paiva A 2022 FAtiMA Toolkit: Toward an Accessible Tool for the Development of Socio-emotional Agents *ACM Trans. Interact. Intell. Syst.* **12** 8:1-8:30

[11] Kuchaiev O et al 2019 NeMo: a toolkit for building AI applications using Neural Modules Preprint arxiv/1909.09577

[12] Soni B and Hingston P 2008 Bots trained to play like a human are more fun *2008 IEEE Int. Jt. Conf. Neural Netw. IEEE World Congr. Comput. Intell.* 363–369

[13] Edwards G, Subianto N, Englund D, Goh J W, Coughran N, Milton Z, Mirnateghi N and Ali Shah S A 2021 The Role of Machine Learning in Game Development Domain - A Review of Current Trends and Future Directions *2021 Digit. Image Comput. Tech. Appl.* 01–07

[14] Sagredo-Olivenza I, Gómez-Martín P P, Gómez-Martín M A and González-Calero P A 2019 Trained Behavior Trees: Programming by Demonstration to Support AI Game Designers *IEEE Trans. Games* **11** 5–14

[15] Ferdous R, Kifetew F, Prandi D and Susi A 2023 Towards Agent-Based Testing of 3D Games using Reinforcement Learning *Proc. 37th IEEEACM Int. Conf. Autom. Softw. Eng.* 1–8

[16] Paduraru C, Paduraru M and Stefanescu A 2022 RiverGame - a game testing tool using artificial intelligence *2022 IEEE Conf. Softw. Test. Verification Valid.* 422–432

[17] Politowski C, Guéhéneuc Y and Petrillo F 2022 Towards automated video game testing: still a long way to go *Proc. 6th Int. Workshop Games Softw. Eng. Eng. Fun Inspir. Motiv.* 37–43

[18] Silver D et al 2018 A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play *Science* **362** 1140–1144

[19] Schrittwieser J et al 2020 Mastering Atari, Go, chess and shogi by planning with a learned model *Nature* **588** 604–609

[20] Perez-Liebana D, Samothrakis S, Togelius J, Schaul T and Lucas S 2016 General Video Game AI: Competition, Challenges and Opportunities *Proc. AAAI Conf. Artif. Intell.* 30

[21] Gaina R D and Stephenson M 2019 "Did You Hear That?" Learning to Play Video Games from Audio Cues *2019 IEEE Conf. Games CoG.* 1–4

[22] Park J S, O'Brien J C, Cai C J, Morris M R, Liang P and Bernstein M S 2023 Generative Agents: Interactive Simulacra of Human Behavior Preprint *arxiv/2304.03442*

[23] Youssef A E, Missiry S E, Nabil El-gaafary I, ElMosalami J S, Awad K M and Yasser K 2019 Building your kingdom Imitation Learning for a Custom Gameplay Using Unity ML-agents *2019 IEEE 10th Annu. Inf. Technol. Electron. Mob. Commun. Conf.* 0509–0514