Proceedings of the 4th International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/2026.34601

Virtual Pet AI Algorithm: A Comparative Study of Growth-Adaptive Systems Based on A Path Planning and Reinforcement Learning*

Liyuan Liang

Yucai Middle School, Chongqing, China
Shirleyliyuan@outlook.com

Abstract. This paper explores the differences in the application of A * path planning and reinforcement learning in the adaptive AI system for virtual pet growth, and implements virtual pet path navigation, autonomous behavior decision-making, and dynamic growth regulation based on the two algorithms. At present, virtual pets have become the mainstream digital products for emotional companionship and leisure relaxation in people's daily lives. The market and user demand are continuously increasing. But there is a problem that the intelligence level of most virtual pet products on the market is low, and there are generally fixed and rigid behavior patterns that cannot grow and evolve independently based on user interaction and usage scenarios. The anthropomorphic interaction experience and intelligent autonomy are seriously insufficient. The A * path planning algorithm, with its advantages of efficient pathfinding and precise obstacle avoidance, has become the mainstream solution for character movement navigation in virtual scenes; And 'reinforcement learning' improves strategies through autonomous trial and error training and environmental interaction, enabling intelligent agents to make behavioral decisions, provide emotional feedback, and dynamically grow and iterate. Under the premise of industry development and technological application, conducting a comparative study on the virtual pet growth adaptive system based on A * path planning and reinforcement learning can not only clarify the adaptability and advantages and disadvantages of the two algorithms in virtual pet construction, but also provide algorithm selection for highly anthropomorphic and autonomous virtual companion AI products, which has strong practical application value and academic research significance.

Keywords: Virtual pets, A* path planning, Reinforcement learning, Adaptive growth, AI algorithms

1. Introduction

Nowadays, with the rapid development of the digital companion industry, virtual pets have gradually become popular digital interactive products among the public due to their leisure, entertainment, and emotional comfort attributes. However, at present, the intelligence level of most virtual pets is relatively low. They not only have rigid walking methods and insufficient path planning efficiency,

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.34601

but also find it difficult to autonomously learn user behavior habits and form dynamic adaptive interaction modes, resulting in a poor overall anthropomorphic experience. In response to the existing problems mentioned above, this study focuses on the intelligent movement and adaptive behavior of virtual pets. It combines A * path planning algorithm with Q-learning and PPO reinforcement learning algorithms to explore and optimize the movement and obstacle avoidance ability of pet scenes. At the same time, the application differences between the two reinforcement learning algorithms are compared to achieve autonomous learning and behavior evolution of pets based on user habits. This study can effectively improve the functional deficiencies of traditional virtual pets and provide feasible technical ideas and practical references for the development of highly intelligent anthropomorphic virtual companion products.

In recent years, path planning technology has become a hot topic in many technical fields due to its broad application prospects and scientific research value [1]. In terms of research status and path planning both domestically and internationally, the A * algorithm remains the benchmark method in the field of game AI [2]; In the field of reinforcement learning, Researchers pioneered a new paradigm of deep RL with the DQN algorithm [3]. Cascaded system based on modular design [4]; End to end neural symbolic system. The former has good real-time performance but poor scalability, while the latter has high flexibility but high training costs. The core research focuses on in-depth exploration of key technologies and application scenarios of virtual pet AI algorithms, with a focus on five major research questions. Firstly, it clarifies the core role and value boundaries of path planning in virtual pet interaction; The second is to systematically analyze the basic principles, execution steps, and landing adaptation solutions of the A * algorithm in virtual pet scenes; Thirdly, by comparing the performance of A * algorithm with other mainstream path planning algorithms, the advantages of its application in virtual pet motion control are extracted; The fourth is to explore how to use Q-learning and PPO reinforcement learning algorithms to achieve autonomous learning and dynamic behavior evolution of virtual pets on player behavior habits; The fifth is to look forward to the future development direction of virtual pet path planning and adaptive AI technology, and propose targeted optimization paths to provide theoretical support and practical guidance for the research and development of highly anthropomorphic virtual pet systems.

2. Core technical solution

Firstly, there is the A * path planning algorithm: the core engine of intelligent movement [5]. The A * algorithm is the core technology for implementing intelligent movement of virtual pets in this project [6]. The A * algorithm is a heuristic search algorithm that comprehensively evaluates the proxy value of each extended search node by setting appropriate heuristic functions. By comparing the proxy values of each extended node, the most promising point is selected for expansion until the target node is found. The advantages are fewer expansion nodes, good robustness, and fast response to environmental information; The disadvantage is that in practical applications, the node limitation caused by the volume of the moving body itself is ignored, but improvement can be achieved by adding nodes. Its core logic is based on the optimal path search of the evaluation function. The algorithm evaluates the function f (n)=g (n)+h (n). Calculate the node cost, where g (n) represents the actual distance from the starting point to the current node, h (n) represents the estimated distance from the current node to the endpoint, and f (n) is the total cost of the node. The algorithm achieves efficient optimal path search by prioritizing the exploration of nodes with the lowest total cost.

The specific execution steps are as follows. The first step is to initialize the open list (set of nodes to be explored) and the closed list (set of explored nodes); Step 2: Calculate the f-value of the starting node and add it to the open list; Step three, select the node with the smallest f value from the

DOI: 10.54254/2755-2721/2026.34601

open list in a loop and move it to the closed list; Step four, if the current node is the endpoint, backtrack the node to generate a path; If the endpoint has not been reached, explore the adjacent nodes of the current node, update the g, h, and f values of the adjacent nodes, and add them to the open list; Step five, if the open list is empty, it is determined that there is no feasible path and the algorithm terminates. This algorithm balances search efficiency and path optimality through heuristic search, perfectly adapting to the real-time requirements of game scenes.

Reinforcement Learning Framework: The Evolutionary Basis for Behavior Adaptation. To achieve the behavioral growth and adaptive ability of virtual pets, the project adopts a reinforcement learning framework combining Q-learning and PPO (Proximal Policy Optimization) [7]. Q-learning constructs a "state action reward" Q-table to enable virtual pets to learn optimal behavior through interaction with players and the environment. It is suitable for players' habitual learning in simple scenarios, such as memorizing areas for home activities and preferred interaction distances; The PPO algorithm improves the evolutionary efficiency of complex behaviors by optimizing the policy update process, and is suitable for complex scenarios such as dynamically following policy adjustments and obstacle avoidance style optimization.

3. Key application implementation

The three core application scenarios of the A * algorithm are "autonomous pathfinding". When a virtual pet receives a player's location specified command through the command interaction module, such as clicking on scene coordinates or mapping the position with voice commands, the A * algorithm will start the scene space modeling process. Firstly, the scene obstacle preprocessing module scans the static obstacles in the game scene, including furniture, walls, terrain boundaries, and other preset collision bodies. Based on the grid method or polygon decomposition method, the passable area of the scene map is divided, and a pathfinding mesh containing node weights is constructed; Subsequently, starting from the current coordinates of the pet and ending at the location specified by the player, the optimal path nodes are selected using the estimation function f (n)=g (n)+h (n) (g (n) is the actual cost of movement from the starting point to the current node, and h (n) is the estimated cost of Manhattan distance/Euclidean distance from the current node to the endpoint). Heuristic search is implemented through dynamic updates of the OPEN and CLOSED tables; The final output path will be smoothed to avoid the mechanical feeling of line movement, while ensuring the shortest path length and collision free, balancing movement efficiency and behavioral rationality, and adapting to various types of virtual scenes such as indoor and outdoor [8]. Next is "intelligent following". When the player triggers the following command, the system will obtain real-time coordinate information of the player's character and dynamically update it as the "dynamic endpoint" of the A * algorithm, constructing a continuous iterative path planning mechanism. To avoid problems such as lagging, taking long detours, or getting too close/too far during the following process, the algorithm sets a dynamic distance threshold. When the actual distance between the pet and the player exceeds the threshold range, the path will be recalculated; Simultaneously introducing path caching and incremental update strategies, only re evaluating nodes in areas that have changed after player movement, rather than searching the entire map, significantly reducing computational costs; In addition, by optimizing the node expansion strategy, the pet's tracking trajectory is ensured to fit the player's movement path, achieving a smooth following experience of "not falling behind, not competing for space, and not taking detours", matching different movement speeds of players such as walking and running. Finally, "Dynamic Obstacle Avoidance": In response to sudden dynamic obstacles in the scene (such as props temporarily placed by players, other virtual characters, moving objects generated by scene interaction, etc.), the system

DOI: 10.54254/2755-2721/2026.34601

quickly captures the position and movement status of obstacles through collision detection modules (such as real-time collision monitoring based on bounding boxes), triggering the emergency path replanning process of the A * algorithm. To meet the real-time requirements of the game, the algorithm adopts preprocessing and parallel computing optimization - pre caching dynamic obstacle types that frequently appear in the scene, establishing a priority collision response mechanism, and strictly controlling the response time of path recalculation within 3ms through GPU parallel acceleration of node search process; When generating a path, the algorithm will combine the movement trend of obstacles to plan a detour path, ensuring a smooth and uninterrupted detour process, avoiding collisions with dynamic obstacles while not deviating from the original following/moving target, achieving a millisecond level closed-loop response of "perception decision execution".

The adaptive ability driven by reinforcement learning and player habit learning is based on the Q-learning algorithm to construct a player behavior preference learning model. Through multi-dimensional data collection modules, real-time capture of player interaction behavior characteristics is achieved, including high-frequency activity areas in virtual scenes, such as the living room and residential areas at home, preferred distances when interacting with pets, such as close call or long-distance call, as well as high-frequency triggered interactive triggers, such as feeding and caressing, and regular interaction periods in the morning or evening. The algorithm maps these behavioral features into a state space (S), using the pet's response behavior as the action space (A), and sets a reward function (R) - giving positive rewards when the pet's behavior matches the player's preferences, and negative punishments when it does not. The behavior decision strategy is optimized by iteratively updating the Q-table. Through continuous learning, pets can autonomously generate personalized behavior patterns: they automatically go to high-frequency activity areas during players' free time to wait and interact, dynamically adjust the following distance according to players' historical preferences to maintain a distance of 0.8-1.2 meters, and for players who prefer private space, maintain a distance of 1.5-2.0 meters, actively triggering high-frequency interactive actions and achieving an adaptive companionship experience of "understanding players".

Dynamic behavior evolution, relying on PPO (Proximal Policy Optimization) algorithm to construct a behavior evolution framework, solves the problems of slow convergence speed and insufficient behavior diversity of Q-learning algorithm in complex scenarios [9]. The algorithm encapsulates the core behavior of pets (such as following, obstacle avoidance, and interactive response) as the optimization goal of the policy network. Through continuous interactive data training and optimization strategy, in the evolution of following behavior, early pets follow based on basic path planning, which is prone to frequent detours, lost tracks, and other problems. The PPO algorithm collects players' movement track data, trains the prediction model to predict the next move direction of players, adjusts the starting point of path planning in advance, reduces the following detour rate by more than 60%, and achieves "predictive tracking"; In the evolution of obstacle avoidance behavior, a rigid zigzag path is initially adopted. The algorithm learns scene environment features (such as obstacle shape, scene channel width) and player acceptance feedback to optimize the smoothness and rationality of the obstacle avoidance path, generate curved paths that fit the environment, and avoid mechanical feeling; In addition, the algorithm supports personalized evolution of behavioral styles. For example, some players prefer "lively" pets, whose obstacle avoidance actions are more agile and their following distance fluctuates more. Some players prefer "stable heavy" pets, whose behavior is smoother and their paths are more conservative. Through continuous interactive training and strategy iteration, pet behavior will gradually evolve from standardized patterns in the early stages to personalized features that fit players' usage habits. As the

DOI: 10.54254/2755-2721/2026.34601

duration of companionship increases, the refinement and adaptability of behavior evolution continue to improve, allowing pets to present unique "personality traits".

4. Algorithm comparison and advantage analysis

By comparing with mainstream path planning algorithms such as Dijkstra's algorithm, BFS (breadth first search), and genetic algorithm, the A * algorithm has significant advantages in adapting to virtual pet scenes. As shown in Table 1, different algorithm types have varying degrees of adaptation to virtual pets. Table 1 shows that different algorithm types have varying degrees of adaptation to virtual pets.

Table 1. Comparison of algorithm types' adaptability to virtual pets

Algorithm Type	core strengths	disadvantage	Virtual pet compatibility
A* algorithm	Efficiently finding the optimal path; Adjustable h value balance speed and accuracy; Low resource consumption.	Complex terrain requires optimization of h-value design.	perfect match
Dijkstra's algorithm	Ensure the optimal path	No inspiring information, low search efficiency.	Compare matches
BFS (breadth first)	simple implementation	Only look for the shortest number of steps, without considering distance.	Compare matches
Genetic Algorithm	Adapt to complex dynamic environments	Large computational load and poor real-time performance.	General matching

To sum up, the core advantage of the A * algorithm lies in balancing "optimal path" and "real-time performance", controllable resource consumption, and the ability to meet the core requirements of "multiple obstacles, dynamic targets, and low latency" in virtual pet scenes. At the same time, it is also easy to integrate with reinforcement learning frameworks, providing a stable path foundation for behavior adaptation.

5. Research limitations and future prospects

5.1. Research limitations

Firstly, during the real-time bottleneck period of the A * algorithm in a super large open world scenario, the path planning of the A algorithm faces a significant increase in computational time in super large open world scenarios with a map area of over 1000 square meters [10]. The core reason is that the spatial complexity of such scenes is extremely high, and the number of grid nodes or polygon decomposition units that need to be processed increases exponentially - compared to small closed scenes, super large scenes contain more terrain details, such as undulating landforms, multi area partitions, massive static obstacles, such as building clusters, natural vegetation, prop displays, etc., which leads to a significant expansion of the node expansion scale of OPEN and CLOSED tables when the algorithm performs heuristic search, and a sharp increase in the calculation frequency of the valuation function f (n)=g (n)+h (n). At the same time, in ultra large scenarios, the distance between the starting and ending points is usually far, and the depth and breadth of path search are significantly improved. Traditional single threaded A algorithm is difficult to complete optimal path filtering and smoothing processing in a short period of time, often resulting in a

DOI: 10.54254/2755-2721/2026.34601

calculation delay of more than 10ms, exceeding the real-time requirements of virtual pet interaction (usually controlled within 5ms), which leads to pet movement lag and instruction response delay, seriously affecting the user experience. In addition, if there is a dynamic loading area in the scene, the A * algorithm needs to frequently rebuild the scene model and pathfinding grid, further increasing computational overhead and reducing the continuity and efficiency of path planning.

Secondly, there is the issue of cold start and data dependency in reinforcement learning models. Currently, virtual pet adaptive systems based on Q-learning and PPO algorithms have significant "cold start" bottlenecks and data dependency characteristics. The core logic of reinforcement learning is to optimize strategies through a large number of interactive trial and error iterations. However, pets lack accumulated player behavior data in the early stages, resulting in insufficient Q-table initialization and ineffective training of PPO strategy network parameters. At this time, pets' behavior decisions rely more on random exploration or basic preset rules, and their intelligence is significantly insufficient. For example, during the player's habitual learning stage, pets may not be able to accurately determine player needs due to insufficient data collected on high-frequency activity areas, preferred interaction distances, and other factors. This may result in frequent trips to non target areas and unreasonable following distances; At the level of dynamic behavior evolution, due to the lack of sufficient interactive feedback samples, the strategy updates of PPO algorithm are difficult to converge, and the behaviors of pets such as following and avoiding obstacles still exhibit standardized and mechanical characteristics, making it difficult to quickly form personalized patterns that fit player habits. In addition, there are significant differences in the interaction habits of different players, such as some players having low interaction frequency and some players preferring diversified instructions. If the interaction data of a certain type of player is insufficient, the model may suffer from overfitting or poor generalization ability, resulting in unstable adaptive effects and inability to adapt to diverse user needs. The current obstacle avoidance logic is only applicable to static and simple dynamic obstacles, and lacks adaptability to obstacles with complex motion trajectories.

5.2. Future development direction

Multi modal fusion path planning, combining visual recognition, virtual physical collision and other multimodal information, enhances pets' obstacle avoidance ability in complex environments; Secondly, personalized path style shaping: Based on the "personality settings" of pets, such as lively, gentle, and timid, adjust the path planning strategy to present differentiated movement characteristics for pets with different personalities; Then, edge computing optimization, migrating part of path planning tasks to terminal devices, reducing server computing pressure, adapting to emerging scenarios such as cloud games; Finally, reinforcement learning pre training optimization reduces the reliance on real player interaction data by simulating data to pre train the basic model and improve initial behavioral intelligence.

6. Conclusion

This study focuses on the intelligent movement and behavior adaptation of virtual pets, and constructs a dual module AI algorithm system based on A * path planning and reinforcement learning (Q-learning, PPO). Targeted research is conducted around the efficiency of path planning and the effectiveness of behavior adaptation. At the level of path planning, through performance comparison with other mainstream path planning algorithms such as Dijkstra and RRT *, it has been confirmed that the A * algorithm exhibits optimal adaptability in virtual pet scenes due to its

DOI: 10.54254/2755-2721/2026.34601

heuristic search characteristics. By accurately balancing search efficiency and path optimality through an evaluation function, it can efficiently meet the three core requirements of precision in autonomous pathfinding, fluency in intelligent following, and real-time dynamic obstacle avoidance (response time ≤ 3ms), significantly outperforming other algorithms in terms of computational cost and path rationality performance in complex scenes. At the level of behavior adaptation, the application differences between Q-learning and PPO reinforcement learning methods were compared. The Q-learning algorithm discretizes player habits (such as activity areas and interaction preferences) by constructing a Q-table, which has the advantages of lightweight model and fast training speed, and can quickly adapt to players' basic behavior preferences. However, it has limitations such as slow convergence speed and insufficient behavior diversity in complex scenarios; The PPO algorithm, relying on policy gradient optimization and trust domain constraints, can handle high-dimensional behavior spaces and perform better in pet dynamic behavior evolution (such as predictive following and personalized obstacle avoidance styles). It can achieve continuous refinement iteration of behavior patterns, but it relies more on interactive data volume and has slightly higher initial training costs. The comprehensive research results indicate that the A * path planning algorithm is the optimal solution for intelligent movement of virtual pets, while the collaborative application of Q-learning and PPO reinforcement learning (Q-learning quickly completes basic habit adaptation, PPO achieves high-order behavior evolution) can maximize the adaptive ability of virtual pets, ultimately achieving the core goal of "efficient movement+personalized companionship", providing key technical support for the development of highly humanized virtual pet systems.

At present, it still faces core challenges, such as high computation time of A * algorithm in ultra large open scenarios, insufficient cold start intelligence in the early stage of reinforcement learning, and weak obstacle avoidance ability in complex trajectory dynamic obstacles. These are still the key directions that need to be addressed in subsequent technical optimization and functional improvement.

For a best viewing experience the used font must be Times New Roman, on a Macintosh use the font named times, except on special occasions, such as program code.

References

[1] Han, H., Zhang, H., Chen, Y., Liu, J. (2025). The evolution and optimization of game AI: From rule-driven to deep reinforcement learning. Neurocomputing 652, 131036.
[2] Zhang, G., Hu, X., Chai, J., Zhao, L. and Yu, T. (2011) Overview of Path Planning Algorithms and Their Applications. 1002-6886(2011)05-0085-06.
[3] Pang, J. and Feng, Z. (2023) Research Progress on the Integration of Heuristic Search and Deep Reinforcement Learning in Game AI Planning. Computer Application Research.
[4] Hua, W. and Zhang, Y. (2022) Neural-Symbolic Chain of Logic Reasoning. EMNLP Findings.
[5] Li, J. and Sun, X. (2008) Research on UAV Trajectory Planning Algorithm Based on Improved A-Star Algorithm. Journal of Ordnance Engineering, (7), 787-792.
[6] Fox, D. (2008) A Probabilistic Approach to Collaborative Robot Localization. Autonomous Robots, 8, 325-344.
[7] Watkins, C.J.C.H. and Dayan, P. (1992) Q-learning. Machine Learning, 8(3), 279-292.
[8] Purnomo, M.R. et al. (2021) Dynamic Pathfinding for Non-Player Character Follower on Game. in Proc. International Conference on Game Development and Multimedia.
[9] Savarese, S., Gupta, A. and Li, F.F. (2025) Emergence of Complex Behaviors in Virtual Pets via Proximal Policy Optimization. arXiv preprint arXiv: 2506.00138.
[10] Liu, Z., Wu, H., Li, X. et al. (2025) Intelligent Automated Guided Vehicle (AGV) Path Planning and Virtual Reality Simulation for Space Launch Sites. China Manned Space Engineering Network.