Proceedings of CONF-APMM 2026 Symposium: Applied Mathematics and Quantitative Modeling
DOI: 10.54254/2753-8818/2026.PU35230

# *An Investigation on Reinforcement Learning Approaches to Maze Solving*

**Weijia Li**

*School of Automation (School of Artificial Intelligence), Hangzhou Dianzi University, Hangzhou, China*
*liweijia@hdu.edu.cn*

***Abstract.*** The problem of moving from a starting point to a destination through a maze of walls and passages, known as maze solving, has been an active field of research in mathematics and computer science for decades with applications to robotics, logistics, network routing and game AI. Classical search algorithms such as Depth First Search (DFS), Breadth First Search (BFS) and A* provide deterministic guarantees but require complete prior knowledge of the environment and fail in dynamic or partially observable settings. Reinforcement learning (RL) provides a completely different paradigm. Agents learn the optimal navigation strategy through trial and error interaction, without relying on predefined maps. This paper summarizes the maze solving methods based on RL. The research methods mainly include seven categories: Deep Q-Networks, Quantum Reinforcement Learning(QRL), Hierarchical RL, Curriculum Learning, Reward Shaping, Transfer Learning, and Multi-Agent RL. Each category has representative works, potential mechanisms and experimental results. Then the sample efficiency, scalability, observability requirements and applicability are compared to determine three urgent challenges: sample efficiency in large-scale maze, partial observability and processing of dynamic environment, and the gap between simulation and reality. The purpose of this survey is to provide researchers with a systematic understanding of the current landscape and determine the direction of future work.

***Keywords:*** Reinforcement Learning, Maze Solving, Transfer Learning, Multi-Agent Systems

## 1.Introduction

The idea of solving a maze is, essentially, the idea of finding a path from a starting point to a destination, through a complex network of passages and walls. From a graph-theoretic perspective, one can model a maze as, where vertices correspond to positions and edges to valid moves between neighboring cells [1]. This is an old problem, dating back decades in mathematics and computer science literature, yet it never seems to lose relevance. Researchers keep coming back to it because the underlying challenge shows up everywhere: robots trying to find their way through cluttered indoor spaces, logistics systems routing packages through warehouse floors, network packets finding the least congested path, and game characters chasing the player through dungeon corridors

© 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: 10.54254/2753-8818/2026.PU35230

[2, 3]. The sheer range of these applications explains why better maze-solving algorithms remain an active topic of investigation [1].

For a long time, the standard toolkit for this problem consisted of classical search algorithms. DFS and BFS were the obvious starting points, as both are simple, well-understood, and require no domain knowledge beyond the maze structure itself [2]. DFS plunges deep into one branch before retreating, which keeps memory usage low but frequently yields a decidedly suboptimal path. BFS, by contrast, fans out level by level and guarantees the shortest path, though the queue can grow unwieldy in larger mazes [4, 5]. The A* algorithm improved on both by bringing in a heuristic estimate that adds to the cost-so-far, pruning large portions of the search space and often finding good paths much faster [4, 6]. Dijkstra's algorithm and the left-hand rule have their own niches too, and practitioners have studied them extensively [2, 5]. The trouble is that all these methods assume the full maze layout is already known. They stumble when walls shift, when the environment is only partially observable, or when the task requires a solution that generalizes across many different maze shapes without restarting from scratch every time [3, 6].

Reinforcement Learning (RL) provided a radically new view. An RL agent immediately interacts with the maze instead of scanning the state space exhaustively, collecting rewards and progressively learning what works [7, 8]. First tabular approaches were used, like Q-learning and State-Action-Reward-State-Action (SARSA). They learn value functions from experience without a pre-built map, and so have an instant advantage over classical search [7, 8]. The real shift came with deep learning. Deep Q-Networks replaced lookup tables with neural networks, letting agents process raw visual inputs like camera frames instead of hand-crafted state representations. [3, 9]. Surveys have noted rapid progress in deep RL for visual navigation across several fronts [9]. For example, researchers have demonstrated that the end-to-end actor critic approach can follow natural language instructions in a 3D maze like environment. Hierarchical RL decomposes navigation into sub target sequences and relies on time abstraction to process tasks with a long time range [10]. Course learning provides another way to advance, gradually increasing the difficulty of the task, so that the agent can gradually establish skills [11]. The structure of the maze itself can be generated and analyzed by various algorithms [1], which directly determines the difficulty of the learning problem. Reward shaping technology has also become more sophisticated, and adaptive methods can now distinguish between beneficial shaping and harmful shaping [12]. At the same time, the transfer learning of subsequent features allows agents to reuse knowledge in mazes that share similar layouts.

There has been a lot of progress recently in reinforcement learning for maze solving, from tabular methods to deep networks, to hierarchical designs and even quantum-enhanced approaches, and the variety of outcomes obtained so far suggests tremendous potential for the future [3, 9, 13]. This velocity makes a methodical review of the current scene timely and important. Section 2 describes the main algorithms. Section 3 describes various approaches with their relative strengths and weaknesses, the current challenges in this field, and possible avenues for future research. Section 4 finishes the paper.

## 2. The utilization of reinforcement learning in maze resolution

Reinforcement learning has been brought to bear on maze solving in a variety of ways, and the resulting methods differ widely in how they represent the environment, what they assume about prior knowledge, and how they balance exploration against exploitation. Some approaches stick closely to the classical tabular Q-learning formulation, while others employ deep neural networks, hierarchical policy structures, or even quantum circuits. The following subsections walk through the

DOI: 10.54254/2753-8818/2026.PU35230

main strands of this literature, covering Deep Q-Networks, quantum RL, hierarchical and curriculum learning, reward shaping, transfer learning, and multi-agent settings.

## 2.1. Deep Q-Network

Deep Q-Network (DQN) is a widely adopted reinforcement learning method for maze solving, combining Q-learning with deep neural networks to approximate value functions.Nandy et al. [3] introduced a DQN-based maze solution for mobile robot navigation planning. DQN is a widely used reinforcement learning approach for maze solving. It extends Q-learning by using a deep neural network to approximate the action-value function. They use a fully connected network with two hidden layers. Each layer has 128 ReLU units. The Adam optimizer is used for training. Navigation is modeled as a Markov Decision Process (MDP). Rewards depend on how the distance to the goal changes. The learning rate decreases linearly. Epsilon-greedy exploration also decays gradually, so the agent keeps exploring throughout training. Two simulated mazes of different complexity, built with OpenCV and NumPy, were used for evaluation. The agent converged within 800 seconds on the smaller maze, but failed on the more complex one, suggesting that vanilla DQN struggles with highly branched maze structures.

Devo et al. [13] proposed a deep reinforcement learning approach for visual navigation with natural language instruction following in 3D maze-like environments. The model is trained with the Asynchronous Advantage Actor-Critic (A3C) algorithm and is composed of a language module based on Long Short-Term Memory (LSTM)-based language module. The language module encodes natural language instructions and combines the visual features extracted from the RGB framework to predict navigation actions. The Habitat simulator on Unreal Engine 4 generates photorealistic 3D mazes of any size. Unlike previous work relying on reference objects for instruction formulation, this approach targets environments without such reference points, closely resembling pure maze navigation. Results showed that the RL agent can effectively navigate in mazes of various sizes according to instructions.

## 2.2. Quantum Reinforcement Learning

Quantum Reinforcement Learning uses quantum mechanical phenomena, including superposition and coherence, to potentially accelerate the learning process. Two techniques are studied, namely comprehensive quantum frameworks and hybrid quantum-classical architectures. Dalla Pozza et al. [14] introduced a QRL framework for the maze problem. In this model, the maze is regarded as a quantum network defined by the adjacency matrix. The quantum Rover navigates it based on the random quantum roaming framework. The agent modifies the maze topology by adding or deleting connections, and the incentive depends on the escape probability of the rover. A hybrid protocol amalgamates QRL with traditional deep neural networks to optimize methods. The results show that the agent can develop optimal strategies in both classical and quantum fields, and quantum coherence improves the utilization of effective behavior in a short time. The architecture exhibits durability in noisy environments, suggesting potential applications for imminent NISQ devices.

Chen et al. [15] introduced a Hybrid Quantum Neural Network (HQNN) for deep Q-learning in maze challenges. The model combines a trainable Variational Quantum Circuit (VQC) with classical convolutional layers, using IBM Qiskit's Pauli-Z feature map and Real-Amplitudes ansatz within the PyTorch framework. The hybrid technique, assessed on 4×4 and 5×5 mazes. Compared with the single classical CNN, the hybrid technology achieves 85.19% and 93.13% win rates with fewer

DOI: 10.54254/2753-8818/2026.PU35230

parameters. This shows that the quantum classical hybrid method is feasible in the actual maze navigation task.

### 2.3. Hierarchical reinforcement learning

Conventional reinforcement learning algorithms often encounter difficulties in the task of expanding the field of view in the expansion maze with infrequent rewards. Hierarchical reinforcement learning tackles this by using temporal abstraction, breaking down navigation into a series of sub goals. Nachum et al. [10] introduced Data-Efficient Hierarchical Reinforcement Learning (HIRO), in which a higher-level policy suggests goal states for the lower-level policy to achieve. The main innovation is a non-policy correction technology, which can alleviate the non-stationarity caused by low-level policy changes and promote two-level stable nonpolicy training. Experiments on simulated robot tasks (such as maze navigation) show that Hiro can only obtain complex motion and object interaction behavior from millions of samples, which is much more than the previous strategy HRL method.

### 2.4 Curriculum reinforcement learning

Curriculum learning complements hierarchical reinforcement learning by gradually increasing task difficulty. Instead of breaking down the problem structure, it lets agents build on skills they already have. Ma et al. [11] studied curricular reinforcement learning for robotic navigation, focusing on collision avoidance and Navigation Amid Moveable Obstacles (NAMO). They devised a two-phase curriculum that begins with training in wide situations where wall-following is adequate, then transitions to scenarios with moveable impediments. A significant contribution is the analysis of the relationship between model convergence, navigation tactics, and difficulty measurement. The system includes distance and pace metrics adjusted for environmental challenges, confirmed through both simulated and actual robotic trials. The results indicated that curricular learning markedly expedites convergence and facilitates the acquisition of intricate NAMO policies that are unattainable via a foundational approach.

### 2.5. Reward structuring

Reward shaping is a method for integrating domain knowledge into reinforcement learning by providing supplementary reward signals to guide the learning process. Nevertheless, inadequately constructed shaping rewards might detrimentally affect performance, and adaptive strategies need to be implemented. Hu et al. [12] introduced BiPaRS, a bi-level optimization system designed for adaptive reward shaping. The model does not need to assume that all shaping rewards are beneficial, but obtains a parametric weight function to evaluate the contribution of each reward. The upper layer optimizes this weight to maximize the real reward, while the lower layer uses the weighted reward learning strategy. Experiments in sparse reward cartpole and MuJoCo shown that this method effectively utilizes the dominant shape, while ignoring or modifying the harmful shape, which provides a systematic method for the integration of domain knowledge.

### 2.6. Transfer learning

The transfer learning enables the reinforcement learning agent to use the knowledge acquired from one maze environment to deal with other maze environments that are structurally similar and so greatly shortens the training time for subsequent tasks. Zhang et al. [16] proposed a deep

DOI: 10.54254/2753-8818/2026.PU35230

reinforcement learning method based on subsequent features for navigation in a maze like environment. Their method splits the Q-function into successor features, which capture expected future state occupancies, and reward features that reflect task-specific preferences. Because the successor features are independent of the task, the system can reuse knowledge from previously learned tasks when facing a new task, requiring only limited additional training. The authors tested this in simulation and on a physical Robotino robot equipped with a Kinect sensor. After completing the first task, the learning time is greatly reduced, with the traditional plan and standard DQN and transfer learning. For robots with limited computing resources, the compact Q-function representation also maintains the feasibility of this method.

Zhu et al. [17] conducted a comprehensive survey of transfer learning in deep reinforcement learning and classified approaches based on objectives, approach, and suitable reinforcement learning frameworks. This classification places the successor-feature approach [16] into a broader taxonomy of policy transfer, representation transfer, and instance transfer, which provides a systematic framework for selecting techniques for maze-solving applications.

### 2.7. Multi-agent reinforcement learning

Multi-agent reinforcement learning refers to multiple agents need to navigate in a shared maze environment and need to cooperate and avoid collision. This extension extends the applicability of RL-based maze solving to socially interacting contexts. Martinez-Gil et al. [18] proposed MARL-Ped, a multi-agent reinforcement learning framework for modeling pedestrian groups in a maze environments. Each agent learns skills in path planning, route selection, and collision avoidance using model-free reinforcement learning. The framework was evaluated on three benchmark scenarios, namely shortest-path versus quickest-path selection, bidirectional flow in constricted corridors, and counter-directional navigation in a maze. The results show that autonomous reinforcement learning subjects naturally exhibit collective behaviors similar to human beings, such as lane creation. Pedestrian dynamics similar to those obtained with the social forces model by Helbing were observed, indicating that MARL-based approaches can produce realistic group behaviors without using manually written rules.

Sakai et al. [19] adopted a new method to solve the maze problem in a real plasma system through experiments. Using a hybrid DC/AC discharge in a square lattice of small channels, they showed how plasma from a given entrance converges towards the exit when increasing the voltage. This process is quantified by Shannon entropy and reproduced in a reinforcement learning model that converts potentials into symbolic rewards. This provides a fascinating physical analogy to reinforcement learning-based maze solving, indicating that route-finding concepts might be relevant in classical, quantum and physical systems.

## 3. Comparison and discussion of models

### 3.1. The merits and demerits of various models

Table 1 highlights a clear pattern across the spectrum of RL-based maze-solving methods. Value-based approaches such as DQN [3, 13] are the easiest starting point. They need little domain-specific engineering and have already worked well on 2D grids, 3D visual mazes, and instruction-following tasks. Their weaknesses are equally obvious. Convergence demands a massive number of environment interactions, so these methods become impractical for large-scale or continuous spaces.

DOI: 10.54254/2753-8818/2026.PU35230

Performance also falls off significantly when rewards are sparse and delayed. This gap motivates the structural methods that follow.

Table 1. Comparison of RL-based maze-solving methods

| Method | Strengths | Weaknesses |
|---|---|---|
| Deep Q-Network [3, 13] | End-to-end learning from original visual inputs | Sample inefficient |
| Quantum RL [14, 15] | Potential speedup; fewer parameters in hybrid form | NISQ device constraints; limited scalability |
| Hierarchical RL [10] | Temporal abstraction; better sample efficiency | Complex architecture; inter-level non-stationarity |
| Curriculum Learning [11] | Faster convergence; enables complex behavior learning | Requires careful curriculum design |
| Reward Shaping [12] | Improves learning in sparse-reward settings | Risk of harmful shaping without adaptive control |
| Transfer Learning [16, 17] | Rapid adaptation to related environments; compact representation | Requires task similarity |
| Multi-Agent RL [18, 19] | Emergent collective behaviors; cross-domain physical analogues | Concurrent-learning non-stationarity |

Hierarchical RL [10] and curriculum learning [11] attack the scalability problem from different angles. HIRO introduces temporal abstraction through goal-conditioned sub-policies, effectively reducing the search depth at the cost of architectural complexity and non-stationarity between levels. In contrast, course learning retains a single policy, but gradually increases the environmental difficulty, allowing agents to gradually accumulate skills. The two strategies are not mutually exclusive. A curriculum of increasingly complex hierarchical tasks is a natural way to combine them, yet this direction remains largely unexplored.

Reward shaping [12] works differently. It leaves the policy structure and training protocol untouched, and instead adjusts the reward signal to make learning easier. Transfer learning [16, 17] extends the horizon even further, allowing knowledge reuse across similar environments, which is particularly useful in the context of robot deployment where each new maze topology would otherwise require retraining from scratch. Multi-agent RL [18] and physical analogues [19] extend the boundaries in very different directions. MARL-Ped demonstrates that decentralized RL agents can spontaneously recapitulate collective behaviors observed in human crowds, while the plasma-based maze solver [19] provides a more fundamental insight. The principles of reward-driven exploration and convergence to optimal paths may be universal enough to manifest in physical systems governed by discharge dynamics rather than neural computation. Whether such analogues can inform the design of better RL algorithms remains an open and fascinating question.

### 3.2. Challenges and future outlook

Despite the progress so far, many problems still plague the use of reinforcement learning in maze solving.Efficiency in sampling and scalability. The main issue is the large number of environmental interactions required by modern deep reinforcement learning approaches. Even with advances in hierarchical and curriculum-based methods, training agents on large mazes with hundreds or thousands of cells is still computationally expensive. Future work could explore model-based reinforcement learning techniques , establish the internal representation of maze dynamics, and

DOI: 10.54254/2753-8818/2026.PU35230

allow much less interaction between planning and the real world. A promising way forward is to marry reinforcement learning with classical search. Classical algorithms can provide initial guidance or break the problem into sub-goals, while the RL component can adaptively fine tune the strategy.

Most modern reinforcement learning methods to maze solving assume that the agent has access to a complete observation of the maze state. However, the agent has limited sensing abilities and must navigate using incomplete information. Partial Observable Markov Decision Process (POMDP) formulations and recurrent neural network architectures provide partial solutions, but it is still difficult to extend these methods to large mazes. Moreover, limited studies have considered scenarios where the maze itself is dynamic, such as walls that can move, doors that can open or close, and new obstacles that can appear. Robust and efficient adaptation of reinforcement learning agents to changing environments opens up immediate applications in search-and-rescue robotics, and autonomous navigation in dynamic indoor environments.

## 4. Conclusion

This paper has reviewed the use of reinforcement learning for maze solving. The discussed methods are tabular Q-learning, Deep Q-Networks, hierarchical architectures, quantum enhanced methods and multi-agent frameworks. Each method has its own advantages: DQN-based methods can learn end-to-end from raw inputs; hierarchical and curriculum strategies can increase scalability by using abstraction and progressive difficulty; transfer learning can speed up adaptation in related environments; quantum and multi-agent extensions open up new theoretical and practical possibilities. The comparative analysis indicates that there is no single best approach for all scenarios and the choice is dependent on the maze size, observability and whether the environment is static or dynamic. In future, addressing sample efficiency, partial observability and the simulation-to-reality gap will be key towards the transfer of RL based maze solvers from lab demonstrations to real-world deployment. With rapid progress in this area, continued collaboration between the RL, robotics and quantum computing communities will likely yield increasingly powerful maze solving systems.

## References

[1] Gabrovšek, P. (2017) Analysis of maze generating algorithms. Proc. Int. Conf. Information and Communication Technology Convergence (ICTC), 1054-1058.
[2] Elkari, B., et al. (2024) Exploring maze navigation: A comparative study of DFS, BFS, and A* search algorithms. Stat., Optim. Inf. Comput., 12, 761-781.
[3] Nandy, A., Subash, S. and Sarkar, A. (2023) Maze solving using deep Q-network. Proc. 6th Int. Conf. Advances in Robotics (AIR), 1-5.
[4] Pathak, M.J., Patel, R.L. and Rami, S.P. (2018) Comparative analysis of search algorithms. Int. J. Computer Applications, 179, 50, 1-6.
[5] Permana, S.D.H., Bintoro, K.B.Y., Arifitama, B. and Syahputra, A. (2018) Comparative analysis of pathfinding algorithms A*, Dijkstra, and BFS on maze runner game. Int. J. Information System and Technology, 1, 2, 1-8.
[6] Barnouti, N.H., Al-Dabbagh, S.S.M. and Naser, M.A.S. (2016) Pathfinding in strategy games and maze solving using A* search algorithm. J. Computer and Communications, 4, 11, 15-25.
[7] Reddy, M.V.V., G, M.R., Rahul, B., Bondala, C.R., Parimi, N. and S, R.I. (2025) Smart maze solver using reinforcement learning. Proc. 3rd Int. Conf. Networks, Multimedia and Information Technology (NMITCON), 1-6.
[8] Harwin, L. and Supriya, P. (2019) Comparison of SARSA algorithm and temporal difference learning algorithm for robotic path planning for static obstacles. Proc. Int. Conf. Inventive Systems and Control (ICISC), 1-6.
[9] Li, F., Wang, C. and Ge, S.S. (2020) A survey on visual navigation for artificial agents with deep reinforcement learning. IEEE Access, 8, 135426-135452.

DOI: 10.54254/2753-8818/2026.PU35230

[10] Nachum, O., Gu, S., Lee, H. and Levine, S. (2018) Data-efficient hierarchical reinforcement learning. Advances in Neural Information Processing Systems (NeurIPS), 3303-3313.
[11] Ma, H.-C., et al. (2023) Curriculum reinforcement learning: From avoiding collisions to navigating among movable obstacles in diverse environments. IEEE Robot. Autom. Lett., 8, 5, 2740-2747.
[12] Hu, Y., Liang, L., Liu, M., Luo, J. and Zhang, J. (2020) Learning to utilize shaping rewards: A new approach of reward shaping. Advances in Neural Information Processing Systems (NeurIPS), 15931-15941.
[13] Devo, A., Costante, G. and Valigi, P. (2020) Deep reinforcement learning for instruction-following visual navigation in 3D maze-like environments. IEEE Robot. Autom. Lett., 5, 2, 1175-1182.
[14] Dalla Pozza, N., Buffoni, L., Martina, S. and Caruso, F. (2022) Quantum reinforcement learning: The maze problem. Quantum Machine Intelligence, 4, 11, 1-10.
[15] Chen, H.-Y., Chang, Y.-J., Liao, S.-W. and Chang, C.-R. (2024) Deep Q-learning with hybrid quantum neural network on solving maze problems. Quantum Machine Intelligence, 6, 2, 1-11.
[16] Zhang, J., Springenberg, J.T., Boedecker, J. and Burgard, W. (2017) Deep reinforcement learning with successor features for navigation across similar environments. Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS), 2371-2378.
[17] Zhu, Z., Lin, K., Jain, A.K. and Zhou, J. (2023) Transfer learning in deep reinforcement learning: A survey. IEEE Trans. Pattern Anal. Mach. Intell., 45, 11, 13344-13362.
[18] Martinez-Gil, F., Lozano, M. and Fernández, F. (2014) MARL-Ped: A multi-agent reinforcement learning based framework to simulate pedestrian groups. Simulation Modelling Practice and Theory, 47, 259-275.
[19] Sakai, O., Karasaki, T., Ito, T., Murakami, T., Tanaka, M. and Kambara, M. (2024) Maze-solving in a plasma system based on functional analogies to reinforcement-learning model. PLOS ONE, 19, 5, e0300842.