Journal of Electronic Research and Application, 2026, Volume 10, Issue 8
http://ojs.bbwpublisher.com/index.php/JERA
Online ISSN: 2208-3510
Print ISSN: 2208-3502

A Review of Path Planning Algorithms for Mobile Robots

Xinrui Li*

Zhangjiagang Campus, Jiangsu University of Science and Technology, Zhenjiang 212003, China

Corresponding author: Xinrui Li, 3783574526@qq.com

Abstract: Path planning for mobile robots is a key technology for achieving autonomous navigation. It aims to search for a collision-free optimal path from the starting point to the target point in complex environments. This paper provides a systematic review of existing path planning algorithms. First, according to the principles of path planning algorithms, the existing methods are categorized into classical algorithms, intelligent optimization algorithms, and artificial intelligence algorithms. Second, the basic principles, advantages, disadvantages, and improvement strategies proposed by other scholars to address the limitations of each type of algorithm are elaborated in detail. Finally, based on the trend of algorithm fusion, the development potential of algorithm fusion mechanisms is prospected. The review concludes that a single algorithm struggles to adapt to complex and dynamic environments, and the integration of the advantages of multiple algorithms has become the current research trend. In the future, improving the real-time performance and robustness of algorithms in dynamic and unstructured environments remains a major challenge.

Keywords: Mobile robot; Path planning algorithm; Classical algorithm; Intelligent optimization algorithm

Online publication: Sept 9, 2026

1. Introduction

With the advent of the "Industry 4.0" era and the intelligent service age, the application of mobile robots in industrial manufacturing, intelligent logistics, medical services, and home assistance has become increasingly widespread [1]. As a core technology for autonomous navigation, path planning's primary task is to find a collision-free path or trajectory from the initial state to the target state for a robot with specific kinematic or dynamic constraints in complex environments [2]. The key to achieving autonomous, efficient, and safe operation lies in advanced motion planning and control algorithms. At the same time, it imposes high requirements on the real-time performance, robustness, and optimality of path planning. How to efficiently plan collision-free paths in unknown or partially known environments has become a research hotspot [3]. The quality of path planning algorithms directly determines the intelligence level and application scope of mobile robots. Related research holds significant theoretical and engineering value [4].

This paper systematically reviews the mainstream algorithms for mobile robot path planning, covering the principles, advantages, disadvantages, and improvement strategies of three major categories: classical algorithms, intelligent optimization algorithms, and artificial intelligence algorithms (as shown in **Figure 1**). It further discusses algorithm fusion mechanisms and provides an outlook on future trends based on existing research. The aim is to provide reference and inspiration for research on mobile robot path planning and to promote the development of this field.

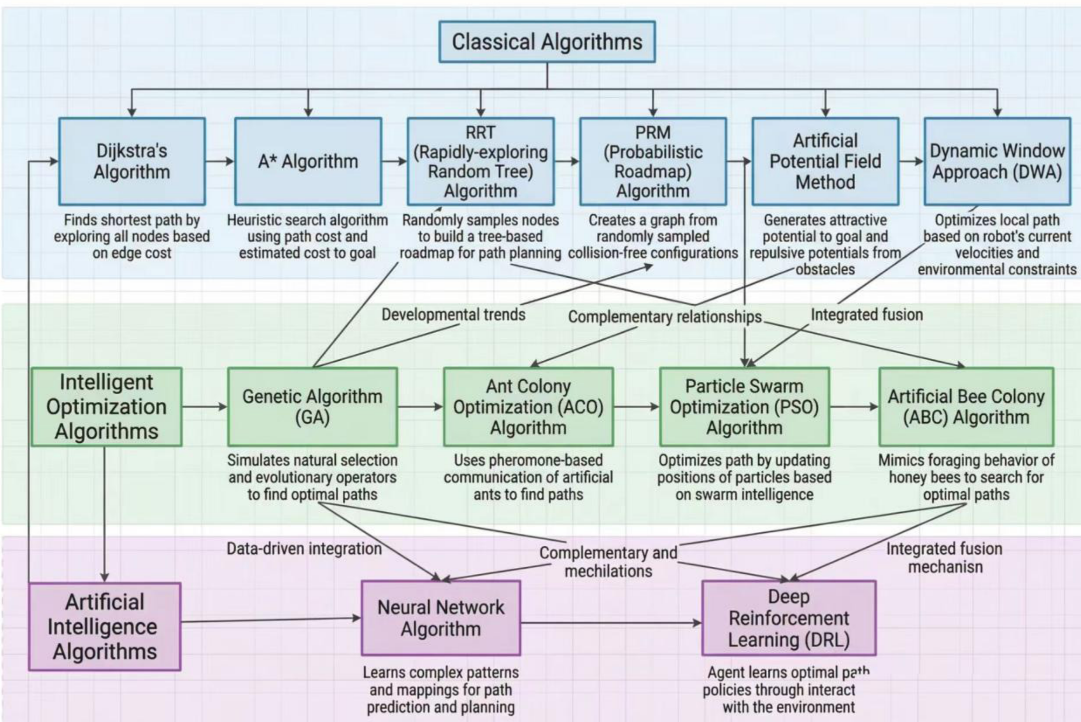


Figure 1. Classification of mobile robot path planning algorithms.

2. Path Planning algorithms for mobile robots

2.1. Classical algorithms

2.1.1. Graph search-based algorithms

With the rapid development of artificial intelligence, path planning has become one of the key technologies in mobile robot applications [5]. Graph search-based algorithms solve problems by systematically exploring graph structures, with core strategies including breadth-first search, depth-first search, heuristic search, and Dijkstra's algorithm. This paradigm laid the foundation for mobile robot path planning, among which the most representative algorithms are Dijkstra's algorithm and the A* algorithm [6].

Dijkstra's algorithm is a single-source shortest path graph search algorithm. As the theoretical cornerstone of shortest path searching, it employs a breadth-first strategy to expand nodes layer by layer, ensuring the acquisition of the optimal path in known environments [7]. However, when dealing with large-scale environments, its unique mechanism leads to high computational complexity and reduced efficiency.

To address the above limitations of Dijkstra's algorithm, Li proposed a relaxed Dijkstra algorithm based on grid environment modeling, which is used to solve real-time path planning problems for mobile robots in large-scale and densely obstructed working environments [8]. Luo et al. proposed an extended Dijkstra algorithm that simulates the terrain environment through Delaunay triangulation, constructs a two-dimensional developable and passable corridor on the surface, and solves for the optimal path of this corridor, thereby improving the accuracy of surface-optimized paths [7]. Gong et al. found that when mobile robots move along paths planned by the Dijkstra algorithm in complex environments, frequent turning problems often occur [9]. Therefore, they proposed a smooth path planning method based on the Dijkstra algorithm by utilizing geometric topology methods combined with actual scene information. This method uses the Dijkstra algorithm to search for the optimal path in a discrete graph as a guide path, thereby improving algorithm efficiency. However, traditional smoothing techniques often result in poor curvature handling and suboptimal trajectories. Ahmad et al. proposed a Dijkstra algorithm integrated with dynamic waypoint allocation using piecewise cubic Bézier curves [10]. Based on the path constructed by the Dijkstra algorithm, this method achieves greater adaptability across different scenarios through context-aware path allocation and iterative optimization based on curve smoothing.

The A* algorithm is a widely used heuristic algorithm. It introduces a heuristic function on the basis of Dijkstra's algorithm. By prioritizing the expansion of nodes with the smallest total cost through the cost function, it significantly improves search efficiency. The cost function of the A* algorithm is defined as:

$$f(n) = g(n) + h(n) \tag{1}$$

where g(n) is the actual cost from the start point to node n, and h(n) is the heuristic estimated cost from node n to the goal.

The A* algorithm performs well in static environments, but its performance highly depends on the design of the heuristic function. In complex environments, it often suffers from excessive node expansion, resulting in increased computational burden, and it struggles to effectively handle dynamic obstacles.

Although the A* path planning algorithm can generate a collision-free path, it still has several shortcomings, such as suboptimal paths, too many redundant points, and insufficient smoothness. To address the deficiencies of the A* algorithm, Zhang et al. adopted a bidirectional optimization method to remove useless nodes, shorten the path length, reduce the number of inflection points, and enhance path smoothness, proposing an improved version of the A* algorithm [5]. Zhao et al. eliminated redundant nodes and unnecessary inflection points in the path to obtain a path containing only the start point, necessary inflection points, and the goal point, and further optimized the path planned by the traditional A* algorithm using geometric methods [11]. Since the A* algorithm has difficulty adapting to complex scenarios and fails to meet real-time requirements, Ou et al. improved the A* algorithm for obstacle space path planning by equipping the mobile robot platform with LiDAR and an inertial measurement unit, solving the problems of excessive turning points and slow search speed in traditional A* algorithms [12]. Yin et al. innovatively adopted a grid environment modeling method to improve the traditional A* algorithm [13]. The improved A* algorithm demonstrates higher search efficiency and fewer traversed nodes compared to both the traditional A* and bidirectional A* algorithms, thereby enhancing the efficiency of mobile robots.

2.1.2. Sampling-based algorithms

Sampling-based algorithms efficiently approximate solutions to complex problems by randomly selecting data samples. They do not rely on systematic traversal but utilize probabilistic and statistical principles, making them suitable for high-dimensional, computationally intensive, or imprecise model scenarios. Their core idea is to trade acceptable error for time and resource efficiency. The results exhibit probabilistic convergence rather than absolute precision [14]. The Probabilistic Roadmap (PRM) is one of the classic path planning methods. Its advantages include simple principles, probabilistic completeness, fast planning speed, and the ability to form asymptotically optimal paths. However, it performs poorly in dynamic obstacle avoidance [15].

Francis et al. adopted a hierarchical robot navigation approach, using PRM as a planning tool and reinforcement learning (RL) for robot control [16]. They proposed a PRM-RL long-range navigation method equipped with AutoRL, which completed short-range obstacle avoidance tasks. To meet various navigation tasks in dynamic scenarios, Liu et al. required mobile robots to achieve high planning success rates, fast planning, dynamic obstacle avoidance, and shortest paths [15]. They adopted a hierarchical planning concept by introducing D* during the network construction and planning process of PRM, and proposed the PRM-D* method to improve dynamic obstacle avoidance performance. Facing the difficulty of offline path planning algorithms in obtaining available paths in real time, Ye et al. proposed a PRM algorithm based on an obstacle potential field sampling strategy, called the Obstacle Potential Field Probabilistic Roadmap method [17]. Chen et al. adopted Bézier curves to enhance the smoothness of paths generated by PRM [14]. To address the shortcomings of PRM in complex environments, they dynamically adjusted PRM sampling points to improve the success rate of path detection. Experimental results showed that the improved PRM algorithm outperforms the original PRM, Rapidly-exploring Random Tree Star (RRT*), and bidirectional random expansion methods in path generation.

Compared with the PRM method, RRT* has attracted widespread attention in trajectory planning due to its asymptotic optimality. Wang et al. addressed the randomness of path generation in the RRT-Connect algorithm and the slow convergence speed of the B-RRT* algorithm by introducing an intelligent sampling function to enhance the directionality of random tree expansion [18]. They proposed an efficient path planning algorithm based on an improved B-RRT*. This algorithm incorporates a fast expansion strategy into the EB-RRT* framework, allowing the use of RRT-Connect in free space and the improved RRT* in obstacle space, thereby improving expansion efficiency while avoiding local optima. Khuat et al. introduced a new algorithm called MultiRRT to solve collaboration requirements during path searching [19]. This algorithm extends RRT by incorporating two new mechanisms—node reduction and Bézier interpolation—to ensure the feasibility and optimality of the generated paths. Yu et al. improved the traditional PQ-RRT* algorithm by combining the posture adjustment angle of the mobile robot and introducing a new distance evaluation function, proposing a hybrid planning algorithm named New Potential Fast RRT* [20]. Bian et al. proposed a Quadtree Rapidly-exploring Random Tree Star algorithm, which shortens the trajectory length of RRT*, improves node quality, and reduces trajectory cost [21]. It performs excellently in terms of node count, convergence time, and path length, thereby enhancing the potential of the algorithm in path planning.

2.1.3. Local optimization-based algorithms

Local optimization algorithms aim to start from an initial solution and iteratively search and move toward better solutions within its "neighborhood," quickly converging to a local optimum [22]. These methods are highly efficient and easy to implement, serving as core solving modules for many complex optimization problems. However, their results heavily depend on the initial point and are prone to falling into local optima rather than global optima.

The Artificial Potential Field (APF) method is a local approach for real-time robot path planning. It assumes that the goal point generates an attractive force and obstacles generate repulsive forces, together forming a virtual force field. The robot is treated as a point in the field and moves toward the goal under the guidance of the resultant force. However, traditional APF algorithms are highly susceptible to local minima, which reduces motion reliability in complex environments. To address this issue, Ma et al. proposed a novel hybrid obstacle avoidance algorithm called Deflection Simulated Annealing-Adaptive Artificial Potential Field by combining an improved simulated annealing mechanism with an enhanced APF model and embedding a direction deflection mechanism during the annealing process [23]. Yang et al. proposed an improved APF algorithm based on paths generated by multi-objective models and quadratic programming, which improved path smoothness and reduced the time required for path planning and trajectory optimization [24]. Zhai et al. addressed the problem of avoiding dynamic and static obstacles in unstructured environments by adding virtual target points and establishing a water-drop-shaped repulsive potential field function [25]. They proposed a local obstacle avoidance trajectory planning method based on an improved APF algorithm. In large-scale environments, goal searching and obstacle avoidance based on RL dynamic optimization decision-making are still limited by slow convergence and low computational efficiency. To solve this problem, Jin et al. proposed a hybrid framework combining RL and APF, named the SAC-APF hybrid framework, which improved the goal search algorithm [26]. Experiments showed that this method significantly outperforms standalone RL algorithms in both goal search efficiency and obstacle avoidance performance.

In recent years, autonomous mobile robots have significantly increased in real-world adoption due to their ability to perform diversified tasks. A critical aspect of their large-scale application is effective local path planning. In this context, the Dynamic Window Approach (DWA) is widely regarded as a robust local path planning solution.

When mobile robots face changes in environmental conditions such as congestion levels, road width, and obstacle density, the evaluation module may select inefficient paths or even cause collisions. To overcome this challenge, Kobayashi et al. proposed dynamic weight coefficients based on Q-learning [27]. This algorithm can dynamically select optimal paths and weight coefficients using a pre-learned Q-table to better adapt to different environmental conditions. Chang et al. introduced a bidirectional search strategy and a variable-weight trajectory evaluation function to address the low efficiency of traditional obstacle avoidance algorithms in unknown complex environments [28]. They proposed an improved DWA algorithm and integrated it with the Optimal Reciprocal Collision Avoidance method, ultimately presenting an enhanced DWA fusion algorithm. To solve the problem that the original DWA algorithm cannot balance safety and speed due to fixed parameters in complex environments, Gong et al. proposed an improved DWA algorithm by modifying the original velocity cost function and adding a distance cost function for the target point, which is used for local obstacle avoidance of robots [29]. Regarding the difficulty of satisfying global optimality and real-time obstacle avoidance in dynamic environments, Duan Qiuhui proposed a fusion algorithm combining an improved D* algorithm and an improved DWA [30]. This algorithm first rasterizes the environment map for modeling; then optimizes the cost function in the D* algorithm by establishing a regional obstacle complexity quantification index vector; finally, it designs the evaluation function of the dynamic window method based on key node information from the globally optimal path, thereby quickly planning a globally optimal smooth path.

2.2. Intelligent optimization algorithms

2.2.1. Genetic algorithm

The Genetic Algorithm (GA) is a meta-heuristic algorithm that simulates the processes of selection, crossover, and mutation in biological evolution. It can be used as a global random search method for optimizing mobile robot paths [31]. Its flowchart is shown in **Figure 2**.

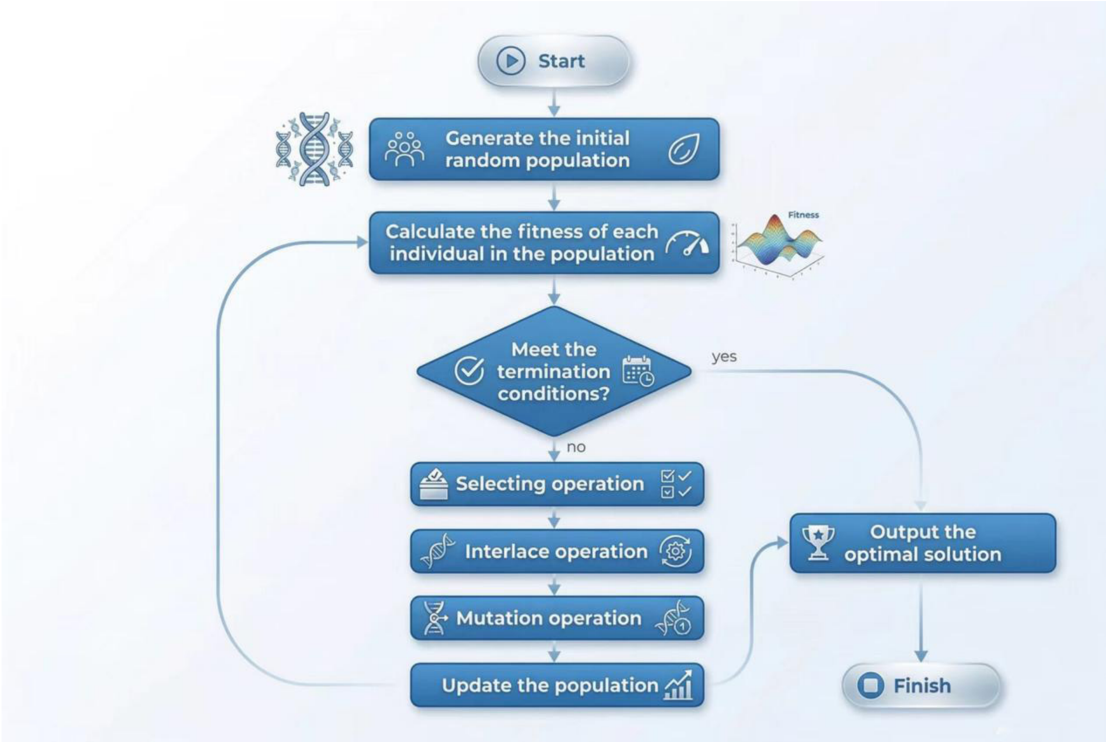


Figure 2. Flowchart of the genetic algorithm.

To address the shortcomings of the standard GA, many improved variants have been proposed. Li et al. incorporated domain-specific knowledge into robot path planning and introduced effective methods for evaluating path quality and accurately detecting collisions [32]. They proposed a novel knowledge-based GA that enables the algorithm to find optimal or near-optimal robot paths in both static and dynamic environments. Li et al. proposed an improved multi-objective GA [33]. The algorithm uses a heuristic median insertion method to generate the initial population, thereby improving the feasibility of initial paths. A multi-objective fitness function is constructed based on three criteria: path length, path safety, and path energy consumption, to ensure the quality of the planned path. Ab Wahab et al. proposed an enhanced GA that improves previous models by introducing a novel population initialization method and combining various genetic operators, resulting in higher-quality optimal paths [34]. In complex environments, when mobile robots perform tasks with different risk levels, various road factors need to be considered. Feng et al. proposed a novel hybrid adaptive genetic algorithm [35]. This algorithm integrates optimized dual optimization operators with an enhanced adaptive GA, achieving efficient path planning under diverse tasks and complex road conditions.

2.2.2. Ant colony optimization algorithm

The Ant Colony Optimization (ACO) algorithm is inspired by the foraging behavior of ants and is a probabilistic swarm intelligence optimization method. Ants deposit pheromones along their paths, and subsequent ants select paths based on pheromone concentration, forming a positive feedback mechanism that ultimately guides the colony toward the optimal solution [36]. Its principle is illustrated in **Figure 3**. In global path planning for mobile robots, the ACO algorithm is widely adopted due to its swarm intelligence characteristics. However, it still suffers from tendencies to fall into local optima and difficulty in obtaining globally optimal paths.

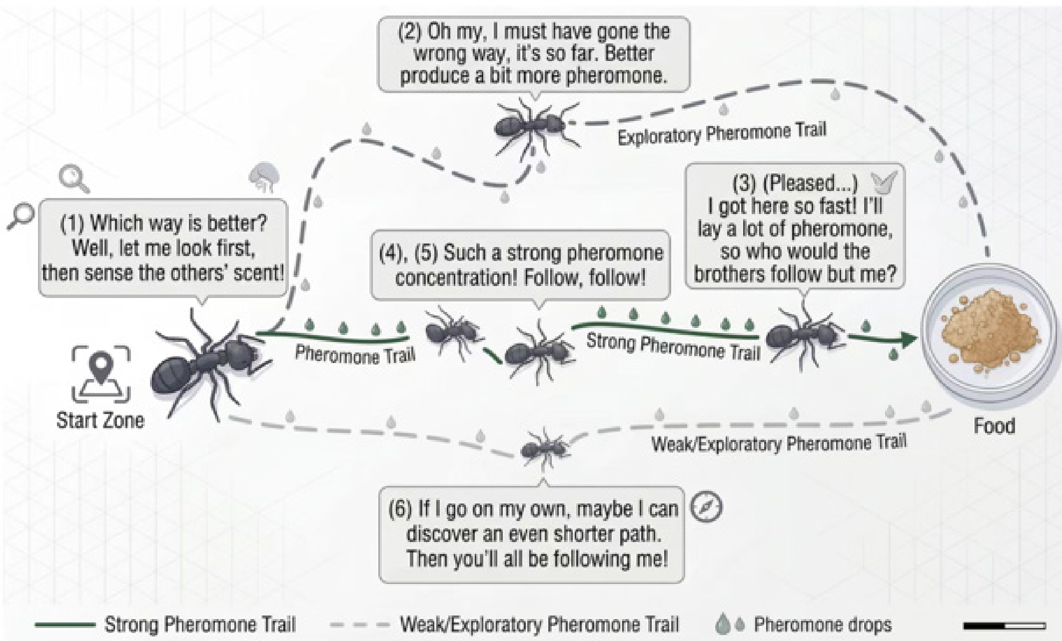


Figure 3. Schematic diagram of the principle of the ant colony optimization algorithm.

To overcome issues such as low search efficiency, slow convergence speed, and susceptibility to local optima, Zhang et al. proposed a novel multi-strategy enhanced Ant Colony Optimization algorithm for mobile robot path planning under non-holonomic constraints [37]. Zhao et al. introduced Q-Learning into the traditional ACO algorithm to enhance its learning capability in dynamic environments, proposing an improved ACO algorithm based on Q-Learning [38]. This algorithm effectively reduces the number of iterations, shortens path optimization time and path length, and improves other performance metrics, while enhancing global search capability and convergence speed. Li et al. proposed an intelligently enhanced ACO algorithm [39]. By implementing non-uniform initial pheromone distribution, adopting an ε-greedy strategy to adjust state transition probabilities, and dynamically balancing pheromone and heuristic functions, the algorithm outperforms traditional ACO in terms of path length, convergence speed, and solution stability, highlighting its practical value in solving global path planning problems for mobile robots. Shi et al. addressed problems of the basic ACO algorithm such as excessively long convergence time, poor global path quality, and unsuitability for dynamic and unknown environments [40]. They proposed an improved ACO algorithm by designing a new probability transfer function and a novel path exploration strategy for unknown environments, effectively solving the weak path exploration capability of ACO in unknown environments.

2.2.3. Particle swarm optimization algorithm

The core idea of the Particle Swarm Optimization (PSO) algorithm originates from the social behavior of bird flocks foraging for food. Each potential solution is regarded as a "particle" in the search space, endowed with velocity and position attributes. By tracking individual and global optimal solutions, particles continuously adjust their states during iterations and eventually converge to the optimal solution of the problem. The algorithm is simple to implement, converges quickly, and is commonly used for optimization problems [41]. Its flowchart is shown in **Figure 4**.

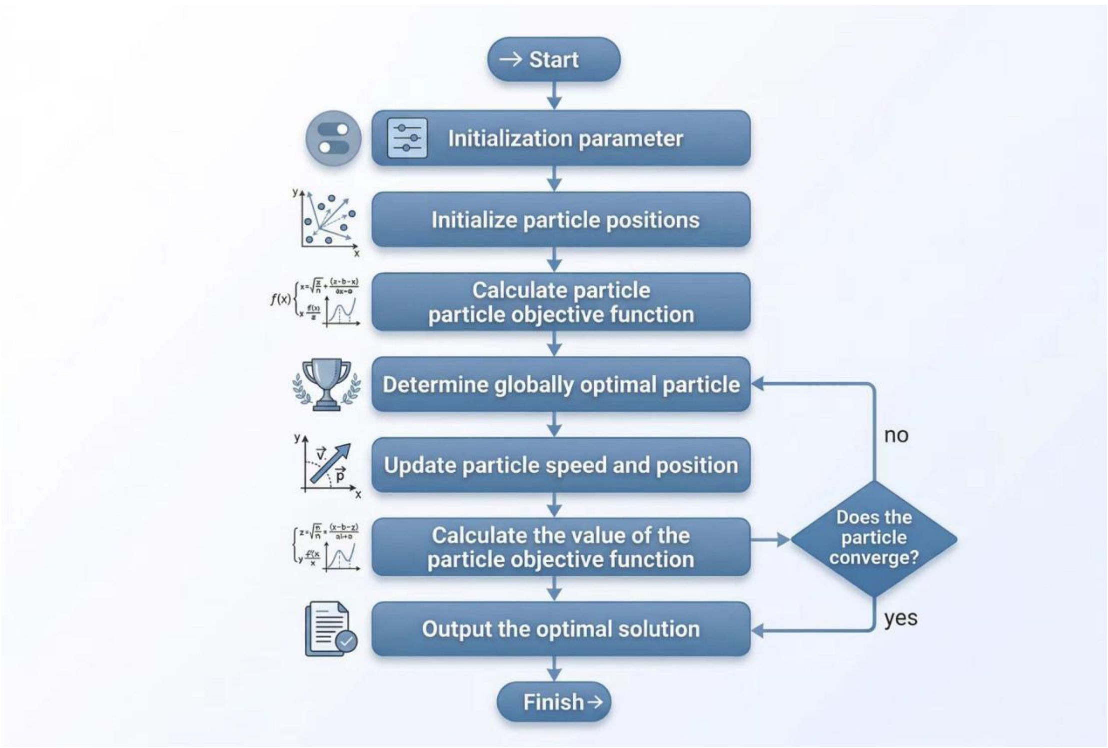


Figure 4. Flowchart of the particle swarm optimization algorithm.

For mobile robots operating under various constraints, finding a near-optimal path remains a challenging optimization problem. Traditional PSO algorithms have limitations when solving constrained problems, such as premature convergence. To address these challenges, Tao et al. proposed a dual-particle PSO algorithm with a random perturbation strategy [42]. This algorithm outperforms existing PSO algorithms and other mature path planning algorithms in terms of path quality and running time, demonstrating its feasibility in solving mobile robot path planning challenges. Lian et al. proposed a novel chaotic adaptive particle swarm optimization algorithm, which maintains the smoothness of robot path movement by optimizing control points in cubic spline interpolation [43]. Lu et al. proposed a multi-objective optimization algorithm for path planning based on improved particle swarm optimization. The algorithm constructs a multi-objective optimization mathematical model and uses PSO to solve multi-objective nonlinear optimization problems, thereby achieving global optimization of path planning [44]. Wang et al. proposed a novel particle swarm optimization method based on a hybrid learning model [45]. To prevent the algorithm from falling into local optima, this method employs a hybrid learning model and a multi-pool fusion strategy, which improves convergence speed while increasing population diversity. Zeng et al. proposed a switching PSO algorithm based on dynamic neighborhoods [46]. The algorithm utilizes a new velocity update mechanism and a novel switching learning strategy to fully exploit the population evolution information of the entire swarm and conduct a comprehensive search of the problem space. In addition, it is hybridized with the differential evolution algorithm to alleviate premature convergence. Experimental results show that the algorithm outperforms several existing PSO algorithms in terms of solution accuracy and convergence performance, and is particularly suitable for complex multimodal optimization problems.

2.2.4. Artificial bee colony algorithm

The Artificial Bee Colony (ABC) algorithm is a swarm intelligence optimization algorithm that simulates the foraging behavior of honeybees. It achieves optimization through the division of labor and cooperation among employed bees, onlooker bees, and scout bees: employed bees exploit food sources and share information, onlooker bees select high-quality food sources based on the shared information, and scout bees randomly explore new areas [47]. The algorithm flowchart is shown in **Figure 5**. This positive feedback and exploration mechanism gives the algorithm strong global optimization capability, making it commonly used for solving numerical and combinatorial optimization problems [48]. However, the ABC algorithm is prone to falling into local optima, suffers from premature convergence, and exhibits poor adaptability when dealing with high-dimensional or discrete problems.

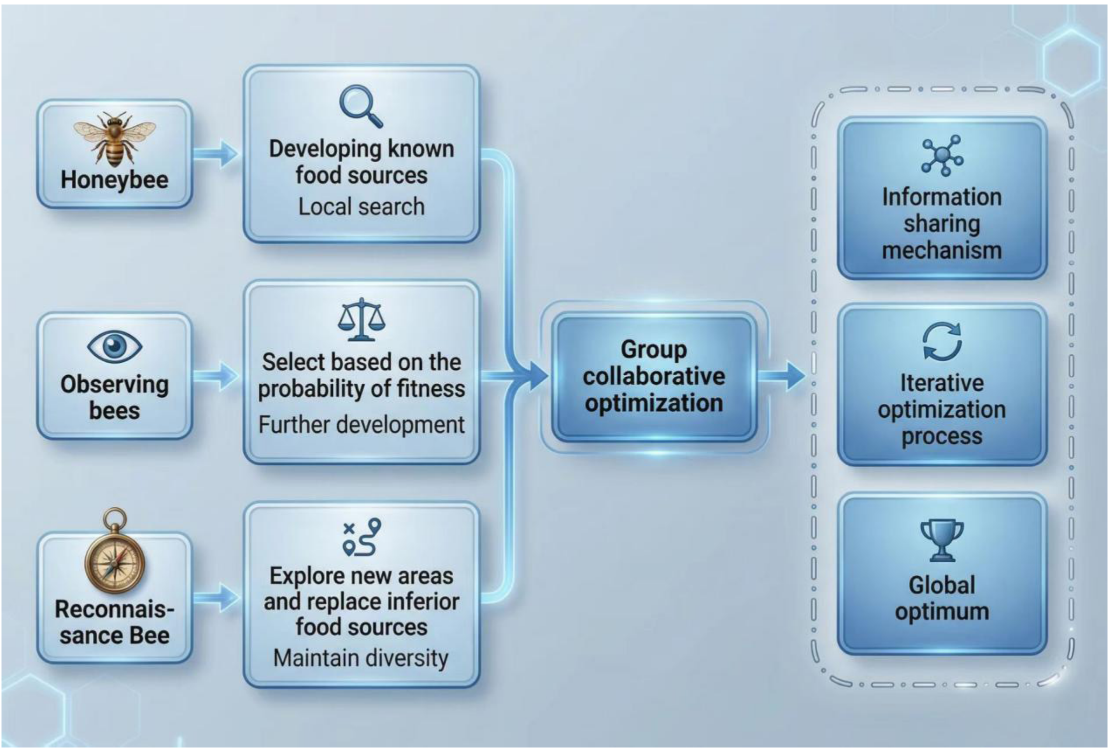


Figure 5. Flowchart of the artificial bee colony algorithm.

To overcome these limitations, Yildirim et al. proposed an improved ABC algorithm by integrating path linearization strategies with local search strategies [49]. This algorithm optimizes the traditional ABC, improves convergence speed, and enhances the algorithm's ability to find global optimal solutions in complex solution spaces, effectively overcoming the tendency of traditional methods to fall into local optima. To solve the multi-objective path planning (PP) problem for mobile robots, Yu et al. proposed an improved ABC algorithm [48]. To optimize the two objectives of path length and path safety, the algorithm designs a well-constructed environment model and path encoding method, and adopts variable neighborhood local search and global search strategies to enhance exploitation and exploration. Experimental results show that the algorithm has significant advantages in HV and SC metrics, with fast convergence speed and better robustness than Non-dominated Sorting Genetic Algorithm II and Multi-objective Particle Swarm Optimization. Ye et al. proposed a learning-based ABC algorithm for solving multi-objective path planning problems [50]. The algorithm effectively addresses the challenges of multi-objective path planning by optimizing three key objectives: path length, safety, and smoothness. Cui et al. developed an improved multi-objective ABC algorithm based on a bionic intelligent optimization mechanism for handling multi-objective path planning problems [47]. The algorithm considers multiple optimization objectives in path planning and improves the adaptability and practicality of the algorithm while ensuring solution quality.

2.3. Artificial intelligence algorithms

2.3.1. Neural networks

A neural network (NN) is a machine learning algorithm that simulates the working mechanism of neurons in the human brain (as shown in **Figure 6**). It serves as the cornerstone of modern deep learning. Its core idea is to construct a network composed of a large number of simple processing units to automatically learn patterns and regularities from data. With its powerful nonlinear fitting capability, it is widely applied in fields such as image recognition and natural language processing [51].

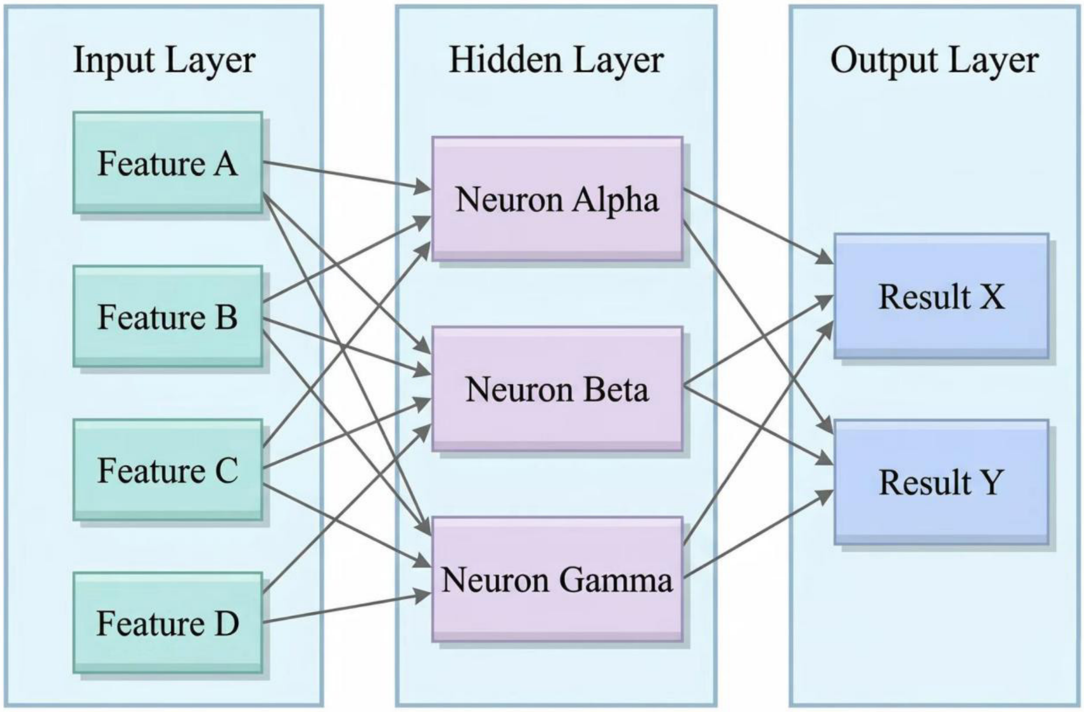
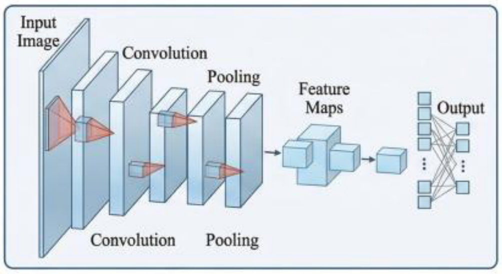


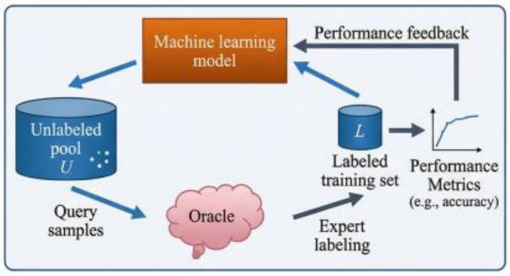
Figure 6. Schematic diagram of neural network structure.

To address the challenges of intelligent path planning for mobile robots in complex road environments, Chen et al. integrated an improved bionic NN with an improved GA to develop a new environment representation and path decision-making model, proposing a hybrid symmetric bionic neural network algorithm [52]. Mulás-Tejeda et al. proposed a Long Short-Term Memory (LSTM) NN model to realize and evaluate the dynamic obstacle avoidance capability of mobile robots. The model acquires information from the robot's LiDAR, enabling it to plan a path from the starting point to the target point while avoiding two dynamic obstacles and multiple static obstacles [53]. As the scale and resolution requirements of the environment increase, the rising computational and time costs of bionic neural networks have attracted scholars' attention. Luo et al. first calculated paths using an improved NN algorithm, then introduced a multi-scale map method into the simulation neural network, and subsequently proposed an improved bionic neural network path planning method based on terrain scaling [54]. This algorithm reduces mathematical complexity and significantly saves time. Molina-Leal et al. proposed an LSTM network with the Adam optimization method for learning trajectory planning strategies for mobile robots in dynamic environments [55]. The algorithm was first validated in a computational simulation environment and then uploaded to GitHub for model training, enabling the robot to find a trajectory from a specified starting point to the target position while avoiding both dynamic and static obstacles. Romano et al. utilized a new extension of the tree-based AutoML software TPOT—TPOT-NN—to integrate and analyze extremely high-performance models generated by automated machine learning and artificial neural networks [56]. They proposed an AutoML + NN estimator to evaluate machine learning results and hoped that AutoML + NN could be used as a free open-source tool in the context of TPOT.

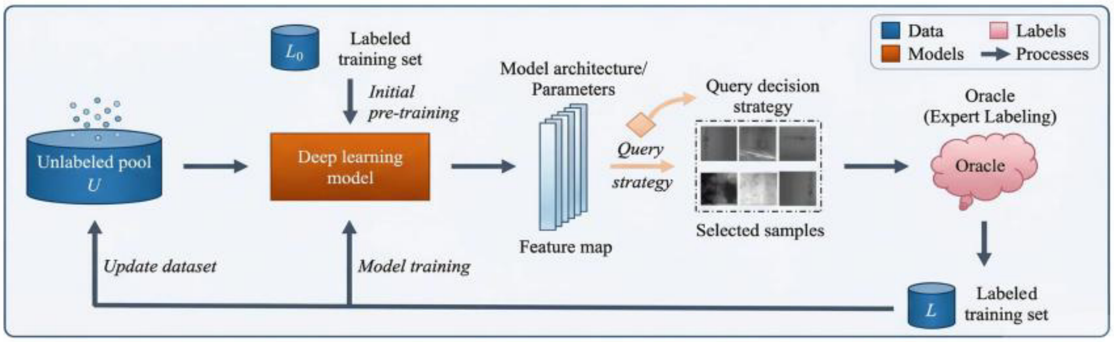
2.3.2. Deep reinforcement learning

Deep Reinforcement Learning (DRL) is the integration of deep learning and reinforcement learning. It leverages the powerful perception capabilities of deep neural networks to process complex inputs and extract critical states. Through the decision-making framework of reinforcement learning, the agent learns the optimal policy through trial-and-error and reward signals during interaction with the environment [57]. Its framework is illustrated in **Figure 7**.





(a) Schematic Diagram of Convolutional Neural Network Structure (b) "Active Learning Loop Based on Pooling"



(c) Classical Example of Deep Active Learning

Figure 7. Framework diagram of deep reinforcement learning.

To address the two deficiencies of slow convergence and excessive randomness in robot path planning, Yang et al. applied the Deep Q-Network (DQN) algorithm within the DRL framework and proposed an improved DQN algorithm [58]. This algorithm combines Q-learning, an experience replay mechanism, and generative neural network techniques. Its convergence speed is faster than that of classical deep reinforcement learning algorithms and improves the efficiency of robot path planning. Li et al. addressed the issues of dimensional disaster and low search efficiency caused by traditional Q-learning in robot path planning by combining the feature extraction capability of deep convolutional neural networks with the decision-making capability of reinforcement learning, proposing an improved algorithm based on DRL [59]. To tackle the overestimation problem, Qi et al. designed a deep double Q-learning strategy [60]. This method uses a replay mechanism to separately calculate the robot's execution actions and decision evaluations, thereby improving algorithm convergence. Yuan et al. introduced an N-step prioritized double DQN algorithm, which integrates double Q-learning and N-step priority strategies to accelerate convergence and improve learning efficiency [61]. Experimental results verified that the algorithm exhibits significant effectiveness and stability in path planning, notably reducing the number of cycles and improving convergence speed.

Currently, DRL is a popular research field, but the problem of dynamic path planning remains prominent. Focusing on DQN, a subfield of DRL, Gök et al. proposed a Dueling Double Deep Q-Network (D3QN) with a prioritized experience replay extension to further optimize DRL models [62]. The algorithm employs both Double Deep Q-Networks (D2QN) and D3QN to train the model and conduct experiments on a test platform. The results show superior performance in path length and time to reach the goal compared to D2QN and D3QN.

As shown in **Table 1**, classical algorithms offer clear principles and relatively reliable planning but are often limited by computational cost, map dependence, or local optima. Intelligent optimization algorithms provide stronger global search capability, although their convergence and performance are sensitive to parameter settings. Artificial intelligence algorithms are more adaptable to dynamic environments but require substantial training data and computational resources. This comparison highlights the need for path-planning methods that balance convergence speed, search stability, and environmental adaptability.

Table 1. Comparative analysis of robot path planning algorithms

Category	Algorithm	Core Idea	Contribution	Optimality	Applicable Environment	Computational Efficiency	Main Features and Limitations
Classical algorithms	Dijkstra	Breadth-first search, gradually expanding the current shortest path nodes	Laid the theoretical foundation for shortest path search	Guaranteed optimal	Static	Relatively Low	Features: Optimal solution, simple principle. Limitations: High computational complexity, unsuitable for large-scale graphs
	A*	Introduces a heuristic function, prioritizes nodes with the smallest comprehensive cost	Pioneering work in heuristic search	Guaranteed optimal	Static	Relatively High	Features: High search efficiency, optimal solution. Limitations: Relies on accurate maps, high computationa l cost in high-dimensional spaces
	RRT	Rapidly constructs a search tree through random sampling to explore space	Solved the planning feasibility problem in high-dimensional spaces	Probabilistically complete	Dynamic	High	Features: Suitable for high-dimensional spaces, fast planning speed. Limitations: Non-optimal paths, instability due to randomness
	PRM	Offline sampling to build a roadmap, online query for feasible paths	Proposed the classical paradigm for multi-query path planning	Probabilistically complete	Static	High	Features: Suitable for multi-query tasks. Limitations: Insensitive to narrow passages; roadmap quality depends on sampling
	APF	Motion guided by the resultant force of attractive force from goal and repulsive forces from obstacles.	Proposed an intuitive physical model for reactive control	No	Dynamic	Very High	Features: Simple computation, good real-time performance. Limitations: Prone to local minima, oscillation in narrow passages
	DWA	Simulates and evaluates trajectories in dynamically feasible velocity space	Deeply integrates robot dynamic constraints with local obstacle avoidance	No	Dynamic	High	Features: Considers dynamic constraints, real-time obstacle avoidance. Limitations: Lacks global view, prone to local optima

Intelligent optimization algorithms	GA	Simulates natural selection, iteratively optimizes population through selection, crossover, and mutation	Classic representative of evolutionary computation	Approximate optimal	Static	Relatively Low	Features: Good generality, strong global search ability. Limitations: Complex parameter tuning, slow convergence
	ACO	Simulates positive feedback mechanism of pheromones, finds optimal path through swarm collaboration	Pioneered the ant colony optimization paradigm	Approximate optimal	Static	Relatively Low	Features: Fast convergence via positive feedback, strong robustness. Limitations: Blind initial search, prone to local optima
	PSO	Simulates social behavior; particles follow individual and global historical best positions.	Typical algorithm of swarm intelligence optimization	Approximate optimal	Static	Medium	Features: Few parameters, fast convergence. Limitations: Prone to premature convergence, weak local search ability
	ABC	Simulates division of labor in a bee colony; optimizes through the cooperation of employed, onlooker, and scout bees.	Proposed an optimization algorithm based on specific foraging behavior	Approximate optimal	Static	Medium	Features: Good global convergence. Limitations: Sensitive to parameter settings; local search ability needs improvement
Artificial intelligence algorithms	NN	Learns mapping from perceptual input to control output through multi-layer network structure	Initiated exploration of end-to-end learning in robot control	No	Dynamic	High	Features: Strong environmental adaptability, fast response. Limitations: “Black-box” model, relies on large amounts of training data
	DL	End-to-end learning of planning strategies from raw data using deep neural networks	Deep Q-Network demonstrates the ability to learn strategies from pixel input	No	Dynamic	High	Features: Strong perception and decision-making ability. Limitations: High demand for data and computing power, poor interpretability
	RL	Agent learns optimal policy to maximize cumulative reward through trial-and-error interaction with the environment.	Marks the paradigm shift from “programming” to “learning”	Approximate optimal	Dynamic	High	Features: Long-term planning capability, adapts to dynamic environments. Limitations: Long training cycles, high cost, difficult reward function design

3. Algorithm fusion mechanisms

3.1. Complementary fusion

Complementary fusion algorithms are computational methods designed to integrate information from multiple sources or modalities. Their core lies in leveraging the complementarity between different information sources to overcome the limitations of a single data source [63]. Through fusion strategies at the data level, feature level, or decision level, information from different dimensions or modalities is synthesized to generate more comprehensive and robust outputs.

To address the shortcomings of traditional path planning algorithms, Liu et al. proposed a method that combines a kinematics-constrained A* algorithm with the DWA algorithm [63]. This method improves the node expansion strategy and heuristic function of the A* algorithm by incorporating robot kinematics to plan the global path. Subsequently, it optimizes the trajectory evaluation function of the DWA algorithm, enabling DWA to plan local paths under the guidance of the global path.

Existing methods face issues such as insufficient obstacle avoidance and low planning efficiency. To tackle these challenges, Zhang et al. proposed a hybrid method combining an improved A* algorithm with an optimized DWA [64]. A tuning factor is introduced to adjust the weight of the heuristic function to enhance the A* algorithm, while the improved DWA algorithm incorporates a path smoothing coefficient and a local target selection strategy, thereby improving the safety and stability of local planning.

3.2. Enhanced fusion

Enhanced fusion algorithms are intelligent processing methods that improve information accuracy and robustness by combining multi-source heterogeneous data. On the basis of traditional fusion, they introduce optimization mechanisms such as adaptive weighting, deep learning, or Kalman filtering to perform deep processing of multi-source data [65]. In addition to integrating information, these algorithms focus on enhancing key features to generate enhanced fusion results that are more suitable for subsequent analysis and decision-making than the original data.

Traditional Artificial Potential Field (APF) is easy to implement but has very limited computational capability. In dense environments filled with obstacles, it frequently encounters problems such as being trapped in local minima and oscillation. In contrast, PSO's global optimization technique is effective in exploring solutions but is relatively slow in large-scale navigation problems. Benmachiche et al. developed a new path planning method by integrating the PSO and APF algorithms, combining the fast response capability of local planners with the extensive search capability of meta-heuristic techniques [66]. This helps mobile robots navigate in areas with both static and dynamic obstacles. The method enables real-time optimization of APF parameters through PSO during navigation, allowing the robot to repeatedly reconfigure the force field according to environmental changes, thereby enhancing responsiveness, robustness, and the ability to escape local minima. Pan et al. proposed an algorithm that uses GA to train DL, combining the advantages of both DL and GA [65]. GA is responsible for collecting states and paths from various scenarios to provide high-quality initial policies or training targets for the DL model, thereby rapidly generating optimized paths. The fusion of "optimization algorithm + learning model" demonstrates strong optimization capability and fast convergence speed.

4. Conclusions and prospects

Path planning for mobile robots, as a core research area in robotics, reflects the profound evolution of artificial intelligence and control science. Classical algorithms form the cornerstone of path planning. Based on rigorous mathematical modeling, they exhibit high computational efficiency and strong interpretability in known environments with static obstacles. These algorithms established the fundamental paradigm of path planning and provided a theoretical reference for subsequent research. The introduction of intelligent optimization algorithms marked the expansion of path planning from precise computation to heuristic search. By simulating self-organizing behaviors in nature, these algorithms demonstrate strong adaptability when dealing with dynamic environments and multi-constraint conditions. They transform path planning problems into multi-objective optimization problems, compensating for the limitations of classical algorithms in handling high-dimensional and nonlinear problems. The surge of artificial intelligence algorithms is triggering a fundamental paradigm shift in path planning. Through end-to-end learning mechanisms, robots can autonomously acquire obstacle avoidance strategies and navigation behaviors in unknown and dynamic scenarios. These algorithms break the absolute dependence on environmental models, enabling robots to possess human-like environmental perception and decision-making capabilities. They particularly demonstrate advantages that traditional algorithms struggle to match in complex semantic scenarios for path understanding.

Looking toward the future, innovation in fusion mechanisms will be a key breakthrough for improving the comprehensive performance of path planning. Specifically, the following directions are worth focused exploration:

(1) Deep fusion at the algorithm level

Constructing hybrid architectures of "knowledge-driven + data-driven" approaches. For example, using classical algorithms such as A* to provide prior knowledge or reward shaping for deep learning, thereby reducing the agent's ineffective exploration space; or incorporating traditional planning algorithms as "differentiable modules" in neural networks to form an end-to-end joint optimization framework that balances global optimality and real-time responsiveness, achieving more efficient and intelligent path planning.

(2) Robust planning in dynamic environments

Combining predictive AI technologies with receding horizon optimization enables robots not only to perceive current obstacles but also to predict the motion trends of dynamic objects. At the same time, introducing federated learning mechanisms allows multiple robots to share obstacle avoidance experience while protecting data privacy, thereby enhancing the level of swarm intelligence.

(3) Fusion perception of semantics and geometry

Breaking through the limitations of traditional two-dimensional or three-dimensional spatial planning by incorporating semantic information into the cost function of path search. Utilizing the scene understanding capabilities of large language models to make path planning more consistent with human behavioral habits and social norms, achieving a transition from "feasible paths" to "reasonable paths".

Acknowledgment

This paper is an output of the science project.

Disclosure statement

The author declares no conflict of interest.

References

[1] Wu J, Yan J, Zhang F, et al., 2026, A Review of Research on Path Planning Algorithms for Mobile Robots. Application Research of Computers, 1–13.

[2] Dias P, Carvalho de Souza J, Solteiro P, et al., 2025, Robot Path Planning: From Analytical to Computer Intelligence Approaches. Journal of Intelligent & Robotic Systems, 111(4): 112.

[3] Niloy M, Shama A, Chakrabortty R, et al., 2021, Critical Design and Control Issues of Indoor Autonomous Mobile Robots: A Review. IEEE Access, 9: 35338–35370.

[4] Liu L, Wang X, Yang X, et al., 2023, Path Planning Techniques for Mobile Robots: Review and Prospect. Expert Systems with Applications, 227: 120254.

[5] Zhang L, Li Y, 2021, Mobile Robot Path Planning Algorithm Based on Improved A*. Journal of Physics: Conference Series, 1848(1): 012013.

[6] Luo W, Zhang C, Zheng W, et al., 2025, Review and Outlook of Path Planning Algorithm System for Mobile Robots. Computer Engineering and Applications, 1–18.

[7] Luo M, Hou X, Yang J, 2020, Surface Optimal Path Planning Using an Extended Dijkstra Algorithm. IEEE Access, 8: 147827–147838.

[8] Li X, 2021, Path Planning of Intelligent Mobile Robot Based on Dijkstra Algorithm. Journal of Physics: Conference Series, 2083(4): 042034.

[9] Gong H, Ni C, Wang P, et al., 2024, Smooth Path Planning Method Based on the Dijkstra Algorithm. Journal of Beijing University of Aeronautics and Astronautics, 50(2): 535–541.

[10] Ahmad J, Nadhir Ab Wahab M, 2025, Enhancing the Safety and Smoothness of Path Planning Through an Integration of Dijkstra's Algorithm and Piecewise Cubic Bézier Optimization. Expert Systems with Applications, 289: 128315.

[11] Zhao J, Zhang Y, Ma Z, et al., 2018, Improvement and Verification of A* Algorithm for AGV Path Planning. Computer Engineering and Applications, 54(21): 217–223.

[12] Ou Y, Fan Y, Zhang X, et al., 2022, Improved A* Path Planning Method Based on the Grid Map. Sensors, 22(16): 6198.

[13] Yin C, Tan C, Wang C, et al., 2024, An Improved A-Star Path Planning Algorithm Based on Mobile Robots in Medical Testing Laboratories. Sensors, 24(6): 1784.

[14] Chen S, Yang G, Cui G, et al., 2025, Improved Path Planning and Controller Design Based on PRM. IEEE Access, 13: 44156–44168.

[15] Liu C, Xie S, Sui X, et al., 2023, PRM-D* Method for Mobile Robot Path Planning. Sensors, 23(7): 3512.

[16] Francis A, Faust A, Chiang H, et al., 2020, Long-Range Indoor Navigation With PRM-RL. IEEE Transactions on Robotics, 36(4): 1115–1134.

[17] Ye L, Chen J, Zhou Y, 2022, Real-Time Path Planning for Robot Using OP-PRM in Complex Dynamic Environment. Frontiers in Neurorobotics, 16: 910859.

[18] Wang K, Zeng G, Lu D, et al., 2019, Mobile Robot Path Planning Algorithm Based on Improved Asymptotically Optimal Bidirectional Rapidly-Exploring Random Tree. Journal of Computer Applications, 39(5): 1312–1317.

[19] Khuat T, Bui D, Nguyen H, et al., 2025, Multi-Goal Rapidly Exploring Random Tree With Safety and Dynamic

Constraints for UAV Cooperative Path Planning. IEEE Transactions on Vehicular Technology, 74(9): 13446–13457.

[20] Yu Z, Xiang L, 2021, NPQ-RRT*: An Improved RRT* Approach to Hybrid Path Planning. Complexity, 2021(1): 6633878.

[21] Bian P, Fan J, Liu Y, et al., 2025, Quad Rapidly-Exploring Random Tree Star Algorithm with Improved Potential Force for Unmanned Aerial Vehicle Path Planning. IEEE Access, 13: 111187–111198.

[22] Bi Y, Fang X, 2025, A Hybrid Path Planning Framework Integrating Deep Reinforcement Learning and Variable-Direction Potential Fields. Mathematics, 13(14): 2312.

[23] Ma B, Ji Y, Fang L, 2025, A Multi-UAV Formation Obstacle Avoidance Method Combined with Improved Simulated Annealing and an Adaptive Artificial Potential Field. Drones, 9(6): 391–421.

[24] Yang J, Zhang H, Ning P, 2023, Path Planning and Trajectory Optimization Based on Improved APF and Multi-Target. IEEE Access, 11: 139121–139132.

[25] Zhai L, Liu C, Zhang X, et al., 2024, Local Trajectory Planning for Obstacle Avoidance of Unmanned Tracked Vehicles Based on Artificial Potential Field Method. IEEE Access, 12: 19665–19681.

[26] Jin F, Ye Z, Li M, et al., 2025, A New Hybrid Reinforcement Learning with Artificial Potential Field Method for UAV Target Search. Sensors, 25(9): 2796.

[27] Kobayashi M, Zushi H, Nakamura T, et al., 2023, Local Path Planning: Dynamic Window Approach With Q-Learning Considering Congestion Environments for Mobile Robot. IEEE Access, 11: 96733–96742.

[28] Chang X, Wang J, Li K, et al., 2025, Research on Multi-UAV Autonomous Obstacle Avoidance Algorithm Integrating Improved Dynamic Window Approach and ORCA. Scientific Reports, 15(1): 14646.

[29] Gong X, Gao Y, Wang F, et al., 2024, A Local Path Planning Algorithm for Robots Based on Improved DWA. Electronics, 13(15): 2965.

[30] Duan Q, 2026, Path Planning for Unmanned Surface Vehicle Based on D and DWA. Control Engineering of China, 33(1): 129–134.

[31] Tao Y, Wen Y, Gao H, et al., 2022, A Path-Planning Method for Wall Surface Inspection Robot Based on Improved Genetic Algorithm. Electronics, 11(8): 1192.

[32] Li J, Hu Y, Yang S, 2025, A Novel Knowledge-Based Genetic Algorithm for Robot Path Planning in Complex Environments. IEEE Transactions on Evolutionary Computation, 29(2): 375–389.

[33] Li K, Hu Q, Liu J, 2021, Path Planning of Mobile Robot Based on Improved Multiobjective Genetic Algorithm. Wireless Communications and Mobile Computing, 2021: 8836615.

[34] Ab Wahab M, Nazir A, Khalil A, et al., 2024, Improved Genetic Algorithm for Mobile Robot Path Planning in Static Environments. Expert Systems with Applications, 249: 123762.

[35] Feng T, Li J, Jiang H, et al., 2024, The Optimal Global Path Planning of Mobile Robot Based on Improved Hybrid Adaptive Genetic Algorithm in Different Tasks and Complex Road Environments. IEEE Access, 12: 18400–18415.

[36] Yang L, Fu L, Li P, et al., 2022, LF-ACO: An Effective Formation Path Planning for Multi-Mobile Robot. Mathematical Biosciences and Engineering, 19(1): 225–252.

[37] Zhang C, Ma J, Wang X, et al., 2025, NMS-EACO: A Novel Multi-Strategy ACO for Mobile Robot Path Planning. Electronics, 14(17): 3440.

[38] Zhao L, Li F, Sun D, et al., 2024, An Improved Ant Colony Algorithm Based on Q-Learning for Route Planning of Autonomous Vehicle. International Journal of Computers Communications & Control, 19(3): 5382.

[39] Li P, Wei L, Wu D, 2025, An Intelligently Enhanced Ant Colony Optimization Algorithm for Global Path Planning of Mobile Robots in Engineering Applications. Sensors, 25(5): 1326.

[40] Shi Y, Zhang H, Li Z, et al., 2023, Path Planning for Mobile Robots in Complex Environments Based on Improved Ant Colony Algorithm. Mathematical Biosciences and Engineering, 20(9): 15568–15602.

[41] Huang C, Zhao Y, Zhang M, et al., 2023, APSO: An A*-PSO Hybrid Algorithm for Mobile Robot Path Planning. IEEE Access, 11: 43238–43256.

[42] Tao B, Kim J, 2024, Mobile Robot Path Planning Based on Bi-Population Particle Swarm Optimization with Random Perturbation Strategy. Journal of King Saud University – Computer and Information Sciences, 36(2): 101974.

[43] Lian J, Yu W, Xiao K, et al., 2020, Cubic Spline Interpolation-Based Robot Path Planning Using a Chaotic Adaptive Particle Swarm Optimization Algorithm. Mathematical Problems in Engineering, 2020: 1–20.

[44] Lu C, Yang J, Leira B, et al., 2023, Three-Dimensional Path Planning of Deep-Sea Mining Vehicle Based on Improved Particle Swarm Optimization. Journal of Marine Science and Engineering, 11(9): 1797.

[45] Wang Y, Wang B, Li Z, et al., 2023, A Novel Particle Swarm Optimization Based on Hybrid-Learning Model. Mathematical Biosciences and Engineering, 20(4): 7056–7087.

[46] Zeng N, Wang Z, Liu W, et al., 2022, A Dynamic Neighborhood-Based Switching Particle Swarm Optimization Algorithm. IEEE Transactions on Cybernetics, 52(9): 9291–9301.

[47] Cui Q, Liu P, Du H, et al., 2023, Improved Multi-Objective Artificial Bee Colony Algorithm-Based Path Planning for Mobile Robots. Frontiers in Neurorobotics, 17: 1196683.

[48] Yu Z, Duan P, Meng L, et al., 2022, Multi-Objective Path Planning for Mobile Robot with an Improved Artificial Bee Colony Algorithm. Mathematical Biosciences and Engineering, 20(2): 2501–2529.

[49] Yildirim M, Akay R, 2025, An Efficient Grid-Based Path Planning Approach Using Improved Artificial Bee Colony Algorithm. Knowledge-Based Systems, 318: 113528.

[50] Ye F, Duan P, Meng L, et al., 2026, A Learning-Driven Artificial Bee Colony Algorithm for Mobile Robot Multi-Objective Path Planning. Applied Soft Computing, 192: 114692.

[51] Mirza N, 2020, Robotic Path Planning and Fuzzy Neural Networks. The International Arab Journal of Information Technology, 17(4A): 615–620.

[52] Chen S, Feng T, Li J, et al., 2025, Research on Intelligent Path Planning of Mobile Robot Based on Hybrid Symmetric Bio-Inspired Neural Network Algorithm in Complex Road Environments. Symmetry, 17(6): 836–863.

[53] Mulás-Tejeda E, Gómez-Espinosa A, Escobedo C, et al., 2024, Implementation of a Long Short-Term Memory Neural Network-Based Algorithm for Dynamic Obstacle Avoidance. Sensors, 24(10): 3004.

[54] Luo M, Hou X, Yang S, 2019, A Multi-Scale Map Method Based on Bioinspired Neural Network Algorithm for Robot Path Planning. IEEE Access, 7: 142682–142691.

[55] Molina-Leal A, Gómez-Espinosa A, Escobedo C, et al., 2021, Trajectory Planning for a Mobile Robot in a Dynamic Environment Using an LSTM Neural Network. Applied Sciences, 11(22): 10689.

[56] Romano J, Le T, Fu W, et al., 2021, TPOT-NN: Augmenting Tree-Based Automated Machine Learning with Neural Network Estimators. Genetic Programming and Evolvable Machines, 22(2): 207–227.

[57] Yu J, Su Y, Liao Y, 2020, The Path Planning of Mobile Robot by Neural Networks and Hierarchical Reinforcement Learning. Frontiers in Neurorobotics, 14: 63.

[58] Yang Y, Juntao L, Lingling P, 2020, Multi-Robot Path Planning Based on a Deep Reinforcement Learning DQN Algorithm. CAAI Transactions on Intelligence Technology, 5(3): 177–183.

[59] Li H, Qi Y, 2020, A Robot Path Planning Method Based on Deep Reinforcement Learning in Complex Environments. Application Research of Computers, 37(S1): 129–131.

[60] Qi R, Wu X, 2022, Robot Path Planning Based on Deep Reinforcement Learning. Manufacturing Automation, 44(12): 177–180.

[61] Yuan G, Zhu B, Hu Y, et al., 2025, Assembly Measurement Path Planning for Mobile Robots Using an Improved Deep Reinforcement Learning. Applied Sciences, 15(23): 12406.

[62] Gök M, 2024, Dynamic Path Planning via Dueling Double Deep Q-Network (D3QN) With Prioritized Experience Replay. Applied Soft Computing, 158: 111503.

[63] Liu Y, Wang C, Wu H, et al., 2023, Mobile Robot Path Planning Based on Kinematically Constrained A-Star Algorithm and DWA Fusion Algorithm. Mathematics, 11(21): 4552.

[64] Zhang J, Ling H, Tang Z, et al., 2025, Path Planning of USV in Confined Waters Based on Improved A* and DWA Fusion Algorithm. Ocean Engineering, 322: 120475.

[65] Pan Y, Yang Y, Li W, 2021, A Deep Learning Trained by Genetic Algorithm to Improve the Efficiency of Path Planning for Data Collection With Multi-UAV. IEEE Access, 9: 7994–8005.

[66] Benmachiche A, Derdour M, Kahil M, et al., 2025, Adaptive Hybrid PSO–APF Algorithm for Advanced Path Planning in Next-Generation Autonomous Robots. Sensors, 25(18): 5742.