Journal of Intelligent Systems and Internet of Things
Vol. 17, No. 01, PP. 360-374, 2025
DOI: https://doi.org/10.54216/JISIoT.170125

Path Planning in Mobile Robotics: A Comparative Review of Classical and AI-Driven Techniques

Mohammed KH. Al-Satooree[1,*], H. A. El Shenbary[1], Ashraf A. Gouda[1], Mohammed Abdel Razek[1]

[1]Department of Mathematics, Faculty of Science, Al-Azhar University, Nasr-City, Cairo, Egypt

Emails: mohamedkhiry270@gmail.com; h.a.elshenbary@azhar.edu.eg; gouda@azhar.edu.eg; abdelram@azhar.edu.eg

Abstract

This research presents a comprehensive analysis of path planning and optimization techniques in mobile robotics, focusing on both classical algorithms and modern intelligent approaches. The study systematically reviews fundamental methods such as Dijkstra's algorithm, the A* search algorithm, and artificial potential fields, together with evolutionary optimization approaches including genetic algorithms and swarm intelligence. It also explores the application of machine learning and deep reinforcement learning models that allow robots to adapt dynamically to complex and changing environments. The comparative evaluation highlights the strengths, weaknesses, and suitable application areas of each approach across scenarios involving obstacle avoidance, energy efficiency, real time adaptability, and multi robot coordination. Particular attention is given to the challenges of uncertain and dynamic environments, computational scalability, and sensor noise, which continue to limit the performance of autonomous navigation systems. By consolidating current advancements and emerging trends, this study provides a structured overview and critical synthesis of existing methodologies, offering a valuable reference for researchers, engineers, and practitioners. It also identifies important research gaps in intelligent hybrid planning, context aware learning and energy constrained mobility, outlining promising directions for the future development of autonomous robotic navigation systems.

Received: January 01, 2025 Revised: February 25, 2025 Accepted: April 02, 2025

Keywords: Path Planning; Mobile Robotics; Optimization Techniques; Robotic Technology and Autonomous Navigation

1. Introduction

Historically, robots were exclusively utilized in the manufacturing sector. Currently, mobile robots are extensively utilized across various sectors, including entertainment, healthcare, mining, rescue operations, education, military applications, aerospace, and agriculture. In recent years, a significant convergence between robotics and artificial intelligence has emerged, aiming to develop robotic systems capable of autonomous decision-making and cognitive reasoning comparable to human intelligence. [1]. With the assistance of sophisticated devices, mobile robots can map their surroundings, ascertain their position, regulate their movements, and identify obstacles while executing their tasks. These functions can be executed via navigation technology. Path planning is a crucial component of the navigation system, essential for devising a secure route between two points [2]. Path planning is the paramount concern in vehicle navigation. It is described as determining a geometric route from the vehicle's present position to a designated destination while circumventing obstructions. The route must be traversable by the vehicle and optimal about at least one variable to be deemed a good path [3]. Path planning can generally be categorized into two types. The initial aspect is global path planning. In this situation, the environment is unchanging, and its comprehensive knowledge is known in advance. The execution of this method is costly and has been extensively examined in the literature. The second aspect is local route planning, where in the trajectory is formulated by utilizing data from sensors during the robot's motion. Consequently, a robot can establish a novel trajectory to acclimate to a new environment. Although it is more complex to design, it is more practical in use [4]. When creating a route, four essential trade-off criteria must be taken into account. Optimization is paramount. According to this criterion, the chosen solution must be the most efficient for distance, time expenditure, and cost. A further condition is completeness, which guarantees that all potential solutions will be exhibited in the path planning process. The final requirement is precision. This criterion is essential for guiding all states from the origin to the destination states. The execution time is the final requirement. The aim of this criterion is to ensure the optimal outcome for addressing the specified issue. Machine learning is an effective method for meeting the aforementioned criteria [5].

Numerous research have examined path planning in relation to static and dynamic constraints. Several methodologies have been suggested to address the path. Formulating a planning issue considering these limits alongside the robot's applicability and capabilities. Path planning is classified as NP-hard. Numerous approaches have been proposed to address the issue, including potential fields [6], neural networks [7], colony systems [8], and particle swarm optimizations [9]. The primary target typically considered in their methodologies is the journey length. To achieve practical applications in the real world, it is imperative to address multiple objectives concurrently. In essence, further objectives beyond path minimization are established, as the optimal path in practical applications is a synthesis of all these purposes [10]. The primary objectives include path length, safety, path smoothness, and the number of turns, among others. The length of a path is defined by the elapsed time and energy expenditure when a robot transitions from a starting point to a destination. Simultaneously, safety is crucial for robots to execute movements with minimal risk in hazardous environments. Consequently, the initial issue can be characterized as a multi-objective problem, and the ultimate path is determined by addressing this multi-objective problem [11].

The main aim of this study is to provide a detailed understanding of how various path planning and optimization techniques can be effectively applied in mobile robotics to achieve efficient, reliable, and intelligent navigation in real world environments. The main contributions of this study can summarize as follows:

- The study offers a detailed review of both classical and modern path planning methods, including graph-based algorithms, potential field approaches, evolutionary computation, and learning-based techniques.
- This study analyzes the performance, strengths, and weaknesses of each method under different operational conditions such as dynamic obstacles, real time constraints, and energy limitations.
- The study highlights how artificial intelligence and deep learning approaches can enhance adaptability, robustness, and decision making in autonomous robot navigation.
- This study outlines the unresolved challenges in current robotic path planning and suggests future research directions related to hybrid intelligent planning, context awareness, and efficient energy utilization.

2. Literature review

Research in path planning and optimization for mobile robotics has evolved significantly over the past two decades, moving from classical deterministic algorithms to advanced intelligent systems. Early studies primarily focused on traditional graph-based approaches such as Dijkstra's and A* algorithms, which guarantee optimal paths but often struggle with computational efficiency in large or dynamic environments. The artificial potential field method was later introduced to enable real-time obstacle avoidance; however, it suffers from local minima issues that limit its effectiveness in complex terrains. To overcome these challenges, heuristic and metaheuristic algorithms, including genetic algorithms, particle swarm optimization, and ant colony optimization, have been widely applied, providing greater flexibility and adaptability. More recently, machine learning and deep reinforcement learning methods have gained attention for their ability to learn optimal navigation strategies from data and adapt to uncertain conditions. Several studies have also explored hybrid frameworks that combine classical planning with learning-based models to achieve both efficiency and robustness. Despite these advancements, achieving real-time adaptability, energy efficiency, and generalization across diverse environments remains a key challenge in the field.

In [12], the authors utilized an enhanced A* algorithm in combination with the Dynamic Window Approach (DWA) to improve a robot's ability to avoid obstacles in narrow environments. Their goal was to develop more effective path planning strategies that enhance robot efficiency and adaptability in changing surroundings. Similarly, in [13], a transfer-learning algorithm known as B-DECAY was proposed to enable agents to quickly adapt to new environments. This method is based on balancing exploration and exploitation during learning, allowing the agent to transfer prior knowledge efficiently. Experimental results showed that the system could learn all paths with an accuracy exceeding 98 percent, demonstrating strong adaptability to environmental changes.

In [14], a comprehensive review categorized existing path planning approaches into four main types: digital, inferential, probabilistic, and hybrid algorithms. This classification provides valuable insights into the strengths and limitations of various techniques, serving as a useful reference for researchers and practitioners.

Shanmugaraja et al. [15] introduced the Hybrid Symmetric Bio-Inspired Neural Network (HSBNN) algorithm, which integrates a bio-inspired neural network framework with a genetic optimization mechanism. This hybridization enhances global search efficiency, minimizes trajectory length, and reduces the number of directional transitions when navigating complex and dynamic terrains. Experimental evaluations reveal that the HSBNN approach achieves substantially improved performance and computational stability compared to traditional path-planning algorithms.

Y. Gao, et al. [16] Recent advancements in robotic navigation have increasingly emphasized hybrid path-planning approaches that integrate global and local planning strategies to address the challenges posed by dynamic and uncertain environments. Traditional deterministic methods, while effective in static settings, often struggle with moving obstacles and high-dimensional state spaces, limiting their applicability in real-world scenarios. In [17], the authors improved path efficiency by employing an enhanced Simulated Annealing (SA) algorithm that balanced computation speed and accuracy for both static and dynamic conditions. Likewise, the work in [18] introduced the Dynamic Quality Progressive Web App (DQPWA) algorithm, which integrated environmental factors such as visibility and congestion into the learning process. This machine learning-based method demonstrated improved adaptability and performance in dynamic real-world settings.

In [19], a hybrid path planning approach combining Improved Particle Swarm Optimization (IPSO) with the Improved Dynamic Window Approach (IDWA) was proposed. IPSO enhanced global search and convergence, while IDWA refined local path tracking and obstacle avoidance, resulting in better overall performance. A related study [20] examined the challenges faced by autonomous robots navigating densely populated areas, emphasizing the need to combine global and local planning strategies. The authors concluded that integrating reinforcement-learning techniques into hierarchical frameworks could further enhance navigation performance in complex environments. Finally, [21] introduced a path-planning model based on the Binary Decision Process (BDP) that generated collision-free paths using map images. The researchers also applied Model Predictive Control (MPC) for path tracking, achieving high precision in both static and dynamic scenarios.

3. Classification of Path Planning Algorithms

Path-planning algorithms for mobile robots can be systematically classified into four principal categories based on their underlying methodology and operational characteristics. Each category offers unique advantages and trade-offs, and the selection of an appropriate algorithm is primarily dependent the specific needs of the application. The complexity of the environment. The available computer resources.

Exact Algorithms seek to determine the accurate answer to the path-planning problem by comprehensive exploration of the search space. These strategies ensure optimality by methodically assessing all potential pathways inside the environment. Notable instances of Exact algorithms encompass Dijkstra's algorithm, the A* algorithm, and the Floyd-Warshall algorithm. Although exact algorithms guarantee perfect answers, they are frequently computationally demanding and may not scale effectively in complex or dynamic contexts.

Heuristic Algorithms utilize problem-specific rules or approximations to direct the search process toward an optimal or near-optimal solution. Utilizing heuristic functions, these strategies markedly diminish the search space and computing load. Prominent examples include greedy"best-first search," bidirectional search, and Jump Point Search. Heuristic algorithms, although they do not guarantee a globally optimal solution, markedly improve planning speed, making them suitable for real-time applications.

Probabilistic Algorithms include unpredictability into the planning process, producing viable paths through probabilistic sampling methods. These methodologies are especially efficacious in high-dimensional or partially understood contexts where exhaustive search is impractical. Notable examples are Rapidly Exploring Random Trees and Probabilistic Roadmaps. Although probabilistic approaches offer significant flexibility and scalability, their solutions may necessitate additional refinement to guarantee optimality or safety.

Hybrid Algorithms aim to integrate the advantages of many planning paradigms to improve efficiency and resilience. By amalgamating Exact, heuristic, and probabilistic methodologies, these methods mitigate the constraints intrinsic to any singular methodology. Prominent - examples of hybrid algorithms include D* Lite, Hybrid A*, and Lifelong Planning A*. Hybrid planners are especially effective in dynamic and uncertain contexts, where adaptation and ongoing replanning are crucial.

3.1 Exact Methods

Path planning is an essential function in robotics, especially for mobile robots, aiming to identify the most efficient route from a starting place to a specified destination while effectively circumventing obstacles. Exact algorithms have been devised to facilitate collision-free navigation by generating an ideal path that minimizes variables such as trip time, distance, or energy consumption [22]. Exact path planning techniques typically employ graph-based representations of the environment, with nodes symbolizing critical places (e.g., intersections, crucial locations) and edges denote viable transitions between these points. The issue is articulated as a graph search, utilizing techniques like Dijkstra's and A* algorithm, along with their modifications, to determine the shortest or most economical route between the initial and terminal nodes.

Multiple factors influence the effectiveness and feasibility of precise algorithms, including environmental complexity and scale, the precision of the robot's sensory capabilities, and the availability of processing resources. Despite their computational demands, exact algorithms have demonstrated significant efficacy across multiple application fields. They have been effectively utilized for autonomous navigation in controlled industrial locations like factories and warehouses, as well as in unstructured outdoor contexts, including agricultural domains and search and rescue missions.

The process for executing a Exact algorithm for mobile robot path planning can be outlined in the subsequent steps:

Step 1: Environmental Representation the initial phase entails representing the robot's operational environment as a graph, where nodes signify distinct points and edges represent viable pathways between these points.

Step 2: Specification of Initial and Target Points the robot's initial and target positions are designated within the graph framework.

Step 3: Calculation of the Cost Function A cost function is formulated to measure the expenditure related to traversing between two nodes. This function generally includes variables such as distance, time, and possible collisions with barriers.

Step 4: Choosing a Search Algorithm A suitable graph search algorithm such as Dijkstra's algorithm, the A* algorithm or an improved variant is chosen according to the task specifications and environmental attributes.

Step 5: Execution of the Algorithm The chosen search algorithm is implemented to methodically traverse the graph, assess the costs of several paths, and eventually identify the ideal route from the starting point to the goal position.

Step 6: Execution of the Path After determining the ideal path, the robot must be directed to exactly follow the specified trajectory, ensuring collision avoidance and compliance with the planned route.

Step 7: Dynamic Path Modification Considering that environments might be dynamic and subject to change throughout operation, it is essential to consistently monitor and adjust the robot's trajectory in reaction to new obstacles or environmental modifications to ensure optimality and safety.

By adhering to this systematic methodology, mobile robots may traverse intricate and possibly unpredictable settings with a significant level of safety and efficiency. Moreover, exact path planning algorithms can be tailored and adjusted to satisfy the distinct requirements of diverse applications, considering the intricacy of the environment and the operating capabilities of the robotic platform. The subsequent part (Exact Algorithms for Path Planning) Table 1 presents a literature review summarizing significant exact algorithms for path planning in mobile robots, including their characteristics and applications. These algorithms illustrate the many ways for addressing path planning challenges and underscore the significance of algorithm selection in attaining effective autonomous navigation.

Table 1: Exact Algorithms for Path Planning

Ref.	Author	Algorithm	Description	Strengths	Weaknesses
[23]	J. Yu and S. M. LaValle, (2015)	ILP-based Multi-Robot Planning	Employs integer linear programming to address multi-robot path planning challenges with diverse optimization objectives.	Delivers optimal solutions across various criteria, exhibiting significant scalability with heuristics.	Elevated computational expense in the absence of heuristics, intricate ILP modeling.
[24]	S. Jafarzadeh and P. J. Fleming,(2018)	Shortest Possible Path (SPP)	Geometry-based precise algorithm for shortest path determination in 2D amidst convex and non-convex obstacles.	Optimal trajectory, secure, and exhibits lower complexity relative to visibility graph methods.	FConstrained to sparse environments, less efficacious in densely populated obstacle fields.
[25]	J. Pak, J. Kim, Y. Park, and H. I. Son, (2022)	A* Smart farm agricultural robots	Comparative analysis of A, Dijkstra, RRT.	Comprehensive field tests in actual smart farm environments	Limited analysis on energy efficiency and computational resource requirements.

[26]	M. R. Rahman and K. Deb, (2023)	A* Enhanced MILP	Combines A* search with MILP constraints for optimal motion planning in static maps.	Ensures exact solutions with reduced search space; deterministic performance.	Requires complete environmental knowledge; unsuitable for dynamic obstacles.
[27]	Y. Qin et al. (2023)	A*, RRT, PRM, Sampling-Based, Artificial Intelligence methodologies	Overview of graph-based, heuristic, artificial intelligence, and sampling methodologies in mobile robot path planning	Extensive applicability, compatible with many robotic systems	Considerations of trade-offs among processing duration, precision, and generalization.
[28]	S. Kazemdehbashi and Y. Liu, (2024)	MIP-based UAV Coverage Path Planning	Employs mixed-integer programming to devise UAV coverage trajectories while accounting for wind.	Facilitates precise planning in windy conditions, yielding optimal or near-optimal solutions.	MIP incurs significant computing costs in large-scale contexts.
[29]	M. Reda, A. Onsy, A. Ghanbari, and A. Y. Haikal, (2024)	Hybrid Path Planning (ADS context)	Integration of traditional, machine learning, and metaheuristic approaches in autonomous driving systems.	Merges optimal features of many methodologies, effectively manages complexity.	Elevated implementation complexity, necessitates adaptive tuning.

3.2 Heuristic Algorithms for Mobile Robot Path Planning

Heuristic algorithms have achieved considerable significance in mobile robot path planning, chiefly because they may produce near-optimal solutions in a relatively brief computing duration. In contrast to exact algorithms that ensure optimality via exhaustive search, heuristic approaches utilize approximations and ”rules of thumb” to effectively generate solutions that are pragmatically applicable in real-world scenarios.

Heuristic methods frequently employed in mobile robot path planning encompass A* Search, D* Lite, Ant Colony Optimization, and Particle Swarm Optimization (PSO), among various others. The performance of these algorithms has proven to be strong across multiple application areas, including warehouse automation, autonomous vehicle navigation, and robotic exploration tasks. The primary aim of heuristic algorithms is to produce paths that are secure by circumventing collisions with static or dynamic obstacles, and efficient, by reducing traversal time, path length, or energy consumption.

Heuristic methods provide a practical equilibrium between computational efficiency and solution quality, enabling mobile robots to function independently in intricate and dynamic settings. Their flexibility and scalability have allowed researchers and engineers to create systems that can tackle a diverse array of jobs within real world operational limitations.

The procedural stages for employing heuristic algorithms in path planning for mobile robots are as follows:

Step 1: Definition of the Problem Explicitly delineate the issue, indicating the initial and target positions, outlining the impediments within the environment, and recognizing any supplementary constraints or operational stipulations.

Step 2: Design of Heuristic Function Develop a heuristic function that estimates the cost of reaching the goal from any position within the environment. The heuristic must be admissible, ensuring it does not overestimate the true cost to achieve the objective.

Step 3: Generation of Search Tree Develop a search tree that illustrates all potential routes from the initial node to the target node. Heuristic algorithms frequently utilize tree based structures, with A* Search, D* Lite, and RRT* as notable examples.

Step 4: Selection of Search Strategy Select an appropriate search technique to guide the exploration process. A* Search employs a best first strategy that prioritizes nodes with reduced anticipated total costs, while D* Lite adopts a dynamic, incremental methodology suitable for variable conditions.

Step 5: Execution of Heuristic Guided Search Utilize the heuristic function during the search process to guide exploration towards favorable areas of the environment. In A*, the heuristic evaluates the residual cost to the objective, while in D* Lite, it facilitates path modification in reaction to environmental alterations.

Step 6: Optimization of the Solution Enhance the generated solution, if required, by optimizing the search tree or adjusting the heuristic function to improve path quality regarding safety, efficiency, or robustness.

Step 7: Execution of the Solution Execute the computed path planning solution on the mobile robot utilizing suitable control algorithms, guaranteeing the robot follows the given trajectory while avoiding obstacles.

By methodically adhering to these procedures, researchers and practitioners can proficiently implement heuristic algorithms to produce secure, dependable, and efficient trajectories for mobile robots functioning in intricate, dynamic settings. The selection of a heuristic algorithm and its particular implementation intricately relies on the task's nature and the environment's attributes.

The following section Table 2 presents an overview of notable heuristic algorithms employed in mobile robot path planning, emphasizing their methodology, advantages, and common applications

Table 2: Heuristic Algorithms for Path Planning

Ref.	Author	Algorithm	Description	Strengths	Weaknesses
[30]	L. Yang, L. Fu, P. Li, J. Mao, and N. Guo,(2022)	(ACO) Dynamic environments	Enhanced pheromone strategies for global search	Effective in highly dynamic environments.	Potential computational cost due to hybridization.
[31]	Z. Yang, N. Li, Y. Zhang, and J. Li, (2023)	Improved PSO (IPSO)	Dynamic obstacle navigation Hybrid global optimization for dynamic environments	Smooth paths, real-time adaptability, and effective collision avoidance.	Computationally intensive; parameter tuning required.
[32]	A. A. N. Kumaar, K. M. Ram, and S. Subramanian, (2023)	Deep Learning Q-(DQL)	Service robots in dynamic environments Lifelong learning using transfer learning for adaptability.	Robust to environmental changes. Scalable and efficient.	High computational cost during transfer learning.
[33]	M. Kobayashi, Y. Tanaka, and H. Saito, (2023)	Q-learning	Congested spaces navigation Optimizes DWA weight coefficients via reinforcement learning.	High adaptability in congested environments.	Dependency on extensive Q-learning pre-training.
[34]	Zhenao Yu, Peng Duan, Leilei Meng, Yuyan Han, and Fan Ye, (2023)	IMO-ABC (Improved Multi-Objective Artificial Bee Colony)	The IMO-ABC algorithm effectively tackles two critical problems: minimizing path length and enhancing safety. It achieves this by designing appropriate representations and path encoding, while employing hybrid dynamics strategies. Additionally, the algorithm emphasizes a balanced approach to both local and global exploration and exploitation, ensuring optimal performance.	The proposed approach demonstrates exceptional effectiveness in optimizing the distribution between various objectives, such as path length and safety. It performs well in both static and semi-variable environments, showcasing outstanding performance in simulation results across multiple maps.	Performance can exhibit reduced stability when dynamics, such as the rapid movement of obstacles, are introduced. The focus group requires fine-tuning—through methods like priming, differentiation, and managing the competition between exploration and exploitation—which may, in certain cases, lead to delays in execution time.

[35]	N. Promkaew, K. Suksawat, and N. Charoenkitkarn, (2024)	Grey Wolf Optimizer (GWOStatic indoor navigation Metaheuristic approach for efficient path generation)	Static indoor navigation Metaheuristic approach for efficient path generation	Effective for static environments, computationally efficient.	Limited applicability to dynamic environments.
[36]	K. Shi, L. Wang, and H. Zhang, (2024)	MIP-based UAV Coverage Path Planning	Deploys mixed-integer programming to devise UAV coverage trajectories while accounting for wind.	Facilitates precise planning in windy conditions, yielding optimal or near-optimal solutions.	MIP incurs significant computing costs in large-scale contexts.

3.3 Probabilistic Algorithms for Mobile Robot Path Planning

Probabilistic algorithms are a vital category in mobile robot path planning techniques. In contrast to exact algorithms that necessitate comprehensive information of the environment, Probabilistic algorithms operate by randomly sampling the search space to estimate the existence of collision free paths. This trait renders them especially appropriate for contexts marked by inadequate or fluctuating information, such as unstructured outdoor situations or settings with mobile barriers.

Probabilistic algorithms have been thoroughly examined in robotic applications because of their intrinsic flexibility and resilience in managing environmental uncertainty. These algorithms can be categorized into two primary types: sampling based algorithms and optimization based algorithms.

Sampling based algorithms, including the rapidly exploring Random Tree and its derivatives, progressively construct a search tree by randomly navigating the configuration space to identify a viable path between the initial and target states.

Optimization based techniques, such the Probabilistic Roadmap Method, first create an extensive roadmap by sampling the environment and linking viable configurations. A search technique is utilized to identify the shortest path that is free of collisions throughout the roadmap.

The adaptability and scalability of probabilistic algorithms have facilitated their effective use in various applications, such as search and rescue missions, planetary exploration, and autonomous vehicle navigation. The subsequent section delineates the standard procedures for executing probabilistic path planning algorithms for mobile robots.

Step 1: Definition of Environment Characterize the operational environment of the mobile robot, including the identification of static and dynamic obstacles, the demarcation of workspace boundaries, and other pertinent characteristics.

Step 2: Specification of Initial and Target States Define the robot's initial and target configurations, including its starting position and intended destination within the specified environment.

Phase 3: Development of the Roadmap Develop a roadmap of the configuration space via random sampling. Each sample must undergo a feasibility assessment by examining potential collisions with impediments. Nodes and edges signify secure configurations and permissible transitions, respectively.

Step 4: Viable Route Exploration Utilize a graph search technique, such as A* or Dijkstra's algorithm, to navigate the constructed roadmap and determine an optimal or near optimal collision free path between the initial and target states.

Step 5: Robotic Navigation Execute the designated path by regulating the mobile robot's motion along the intended trajectory, employing control methodologies such as Proportional Integral Derivative (PID) controllers. Step 6: Revision of the Roadmap While the robot maneuvers, observe the surroundings for any dynamic alterations. Revise the roadmap by incorporating new nodes or eliminating outdated paths to maintain ongoing path viability.

Step 7: Iterative Procedure Persistently reiterate the procedures of sampling, pathfinding, and navigation to adjust to environmental fluctuations and sustain peak performance.

It is essential to recognize that many probabilistic algorithms may include modifications of these phases or incorporate supplementary elements, such as machine learning approaches, to enhance path quality or computing efficiency. The selection of the probabilistic strategy typically hinges on application needs, computational resources, and environmental dynamics.

Table 3 presents a literature review of important probabilistic algorithms for planning paths for mobile robots, highlighting their main contributions, benefits, and popular use cases.

Table 3: Probabilistic Algorithms for Path Planning

Ref.	Author	Algorithm	Description	Strengths	Weaknesses
[37]	J. Pak, S. Kim, and H. Lee, (2022)	RRT Smart farm agricultural robots	Rapid exploration of high-dimensional spaces	Comprehensive field tests in actual smart farm environments.	Limited analysis on energy efficiency and computational resource requirements.
[38]	J. Yao, X. Li, and Y. Chen, (2022)	SC-DQN	Unmanned helicopter navigation State-coded RL with dynamic rewards for real-time planning.	Real-time adaptability, faster convergence, and smooth paths.	Simulation-focused; computational requirements increase with complexity.
[39]	A. Sabeeh and A. Al Furati, (2024)	GA-PRM Hybrid	UA hybrid approach for medical robotics, employing PRM for global planning and GA for optimization.	Features real-time flexibility, generates smooth trajectories, and is suitable for dynamic environments.	Performance may decline in highly limited settings.
[40]	R. Steffi, P. Kumar, and S. Sharma, (2024)	Bayesian Optimization Algorithm (BOA)	BOA employed for dynamic, energy-efficient path planning in robosoccer contexts.	Rapid computation, adaptive, effective in dynamic environments.	Complexity increases with additional dynamic obstacles.
[41]	J. Bao and R. Yonetani, (2024)	IG-PRM (Instruction-Guided PRM)	Utilizes natural language directives to direct PRM through LLM-generated cost maps.	Facilitates human-intuitive planning and integrates effectively with conventional planners.	Relies on the quality of instruction embedding.
[42]	J. Xu, L. Zhang, and Y. Wang, (2024)	Multi-Time-Based RRT	Navigation of MCDPRs in uncertain environments Vision-based trajectory planning with RGB-D obstacle detection	Effective in high-dimensional spaces. Real-world validation conducted.	Computational complexity due to multi-agent planning
[43]	R. Zhang, M. Li, and Y. Chen, (2024)	RRT	Real-time robot navigation Global probabilistic planning for safe paths	Smooth paths and real-time adaptability.	Limited scalability in dense obstacle environments; computationally expensive.

3.4 Hybrid Algorithms for Path Planning in Mobile Robotics

Hybrid algorithms for path planning in mobile robots constitute a sophisticated and flexible methodology, wherein many algorithmic solutions are integrated to use their distinct advantages and alleviate their inherent limitations. These algorithms aim to attain a balanced compromise among the accuracy of exact methods, the computing efficacy of heuristic strategies and the flexibility of probabilistic methods. Hybrid algorithms are very effective for addressing intricate path planning challenges that involve dynamic impediments, many conflicting objectives, and uncertain or partially understood environments. They are crucial in improving the resilience, reliability, and adaptability of path planning systems in real world applications, where conditions frequently fluctuate and are unpredictable.

The amalgamation of diverse path planning methodologies within a hybrid framework can be achieved through several approaches, including interleaving, switching, or parallel execution of algorithms. The formulation of a hybrid algorithm often entails the integration of two or more methodologies from the previously mentioned categories of precise, heuristic, and probabilistic techniques. The choice of algorithms and the configuration of the hybrid system are determined by the particular operational needs, environmental factors, and performance criteria of the intended application. The primary objective is to achieve the optimal balance among accuracy, efficiency, and resilience.

The following are the essential steps that are included in the comprehensive process of developing a hybrid algorithm for path planning in mobile robots:

Step 1: Determine Application Specifications Commence with a comprehensive analysis of the application's unique needs, encompassing environmental attributes, the existence and behavior of impediments, mission objectives, and operational limitations. This research offers essential information for choosing suitable algorithmic components.

Step 2: Choose Suitable Algorithms Select two or more algorithms that correspond with the specified conditions. For example, an exact algorithm like Dijkstra's technique can be utilized for precise navigation in organized surroundings, while a heuristic approach such as A* or a probabilistic method like Probabilistic Roadmap Method may be chosen for swift planning in vast or largely unknown terrains.

Step 3: Evaluate Strengths and Weaknesses Assess the chosen algorithms for their advantages (e.g., optimality, efficiency, flexibility) and drawbacks (e.g., computational expense, susceptibility to noise). Comprehending these attributes enables strategic integration, wherein each algorithm mitigates the shortcomings of the others.

Step 4: Formulate the Hybrid Framework Devise the hybrid methodology by determining an integration mechanism. This may entail interleaving algorithms (alternating approaches during execution), executing them concurrently (facilitating simultaneous exploration and optimization), or developing a hierarchical or layered control system.

Step 5: Assess and Analyze the Hybrid Algorithm Execute comprehensive simulations and empirical trials to verify the efficacy of the hybrid algorithm. Metrics including path optimality, computational duration, collision frequency, and resilience to environmental alterations must be evaluated. Comparative assessments with the individual component algorithms are essential to validate the effectiveness of the hybrid technique.

Step 6: Enhance and Optimize the Hybrid Algorithm Continuously enhance and optimize the hybrid algorithm based on assessment results. Possible enhancements encompass optimizing algorithm parameters, adjusting the integration method, or reevaluating the selection of algorithms to more effectively address application specific requirements.

Through the systematic creation and optimization of hybrid path planning algorithms, researchers and engineers may markedly improve the autonomous and efficient operation of mobile robots in diverse and challenging situations.

Table 4 presents a comprehensive literature analysis of notable hybrid path planning algorithms and their comparative efficacy across several application domains.

Table 4: Hybrid Algorithms for Path Planning

Ref.	Author	Algorithm	Description	Strengths	Weaknesses
[44]	. Trasnea, D. Popescu, and C. Ionescu, (2021)	OctoPath (Self-Supervised Learning)	Predicts optimal trajectories using a 3D octree model and self-supervised learning.	Reduces need for labeled data; handles complex environments effectively.	Requires high-quality 3D sensor data; computationally demanding.
[45]	Y. Liu and X. Li, (2022)	Modified Grey Wolf Optimization + Situation Assessment (MGWO + SA)	It balances global optimum exploration through the Modified Grey Wolf Optimization (MGWO) algorithm with local adaptability and responsiveness achieved via situation assessment based on laser–camera data fusion and Bayesian reasoning.	Achieves shorter path length, better optimization performance and dynamic obstacle avoidance in simulation settings.	It relies heavily on the quality of sensor data; the computational complexity increases due to multi-sensor fusion and Bayesian inference; and it may experience delays in high-speed, real-time dynamic scenarios.

[46]	H. Liu and J. Li, (2022)	MGWO with Situation Assessment	Integrates Modified Gray Wolf Optimization with dynamic situation assessment	Utilizes multisensor data; balances global and local planning	Computational complexity; limited real-world testing
[47]	X. Li and Y. Wang, (2024)	JBS-A*B + Improved DWA	Hybrid global-local planning with enhanced perception through monocular camera segmentation.	Robust obstacle avoidance; improved trajectory smoothing.	Integration complexity; needs validation in dynamic environments.
[48]	H. Liu, Y. Zhang, and L. Chen, (2024)	DRL + Two-way Hybrid A*	DRL-based path planning enhanced with hybrid A* for improved local path quality.	Efficient on embedded platforms; better planning results.	Limited performance in complex environments; training-intensive.
[49]	M. Tariq, R. Singh, and J. Chen, (2025)	LLM-Based Dynamic Waypoint Generation	Utilizes LLMs to interpret natural language commands and generate dynamic navigation paths.	Innovative integration of NLP and robotics; robust in dynamic settings.	High computational requirements; potential unpredictability.
[50]	M. Tariq, R. Singh, and J. Chen, (2025)	LLM-based Waypoint Generation	Uses Large Language Models to interpret user commands and generate paths	Robust in complex environments; interprets natural language commands	High computational overhead; requires large training datasets

4. Main Results

The main result is that no singular algorithm excels across all performance metrics. Exact approaches provide ideal answers in clearly defined environments but encounter difficulties in dynamic or high-dimensional settings. Heuristic and probabilistic approaches, in contrast, forfeit optimality in favor of adaptability and scalability, particularly in real-time or uncertain contexts. Hybrid approaches seek to reconcile this disparity, providing flexibility while incurring complexity. Employed Research Methods and Approaches Exact algorithms employ deterministic techniques, such A*, Dijkstra's algorithm, Bellman Dynamic Programming, and Mixed Integer Programming , to ensure the identification of shortest paths or optimal coverage .Heuristic algorithms encompass Particle Swarm Optimization, Ant Colony Optimization , and Deep Reinforcement Learning techniques for path optimization under variable conditions [51] Probabilistic algorithms, like RRT, RRT*, and IG PRM, utilize randomness and sampling-based planning, making them particularly effective for high-dimensional and uncertain environments [52] Hybrid methods amalgamate the advantages of many paradigms, such as employing Probabilistic Roadmap Methods for global planning and Genetic Algorithms (GA) for optimization [53] Inadequate scalability in dynamic and high-dimensional contexts; elevated computational demands [54] Heuristic Algorithms: real-time flexibility through guided sampling appropriate for dynamic environments; scalable, suitable for online replanning [55] Absence of optimality assurances; susceptible to parameter adjustment [56]. Stochastic algorithms represent a formidable class of optimization techniques that excel in addressing the challenges of uncertain and high-dimensional environments. By leveraging probabilistic modeling and random sampling strategies, these algorithms consistently outperform deterministic approaches, which often falter in convergence or become computationally unmanageable. Their inherent adaptability guarantees effective real-time decision-making and continuous learning, positioning them as the optimal choice for dynamic and non-stationary robotic applications such as motion planning, control, and sensor-based navigation [57]. May produce inefficient or erratic trajectories; computationally intensive in congested settings. Hybrid Methodologies Equilibrated performance integrates global and local strategies [58]. Elevated complexity; substantial calibration and processing resources required [59] Concordances and Discordances among Studies Consensus: The majority of studies concur those static algorithms such as A* or Bellman Dynamic Programming are inadequate for dynamic real world situations. There exists a trend towards hybrid or learning-based methodologies to address the shortcomings of conventional planning. Disagreement: emphasizes the improved accuracy of Enhanced A*. Even in predominantly foreign environments, [60] contend that learning-based algorithms exhibit greater robustness in highly dynamic contexts. Promkaew et al. [61] advise for metaheuristics in static indoor environments, whereas Q-learning-based approaches are recommended for adaptive learning in congested or uncertain situations. Ascendant Trends and Recurring Themes Real-time adaptability is being emphasized, with Reinforcement Learning (RL) and Deep Reinforcement Learning techniques at the forefront. The utilization of sensor fusion, such as RGB-D cameras, as demonstrated by [62], is prevalent for enhanced obstacle identification. Human-robot interaction is advancing, as demonstrated by IG PRM's application of natural language for planning. Multi-agent coordination, such as Multi-Time-Based RRT, is becoming significant for swarm and multi-drone applications. Research Deficiencies, Prospective Avenues Energy efficiency, and computational expense are inadequately examined in numerous comparison analyses [63]. Research on comprehensive benchmarking in diverse situations is scarce. The transferability of learnt models in reinforcement learning and deep reinforcement learning across different settings continues to be a challenge. Future studies should concentrate on establishing unified evaluation frameworks for planning methods. Improving elucidation and clarity in learning-based methodologies. Development of lightweight hybrid algorithms that reconcile optimality with real-time limitations.

4.1 Novelty and Research Gap of the Study

This work aims to provide a comprehensive review of various path planning and optimization techniques utilized in mobile robotics. The review includes classical, exact, heuristic, probabilistic, and hybrid algorithms, as well as optimization techniques such as artificial neural networks and fuzzy logic. The originality of this study lies in its thorough and systematic analysis of diverse path planning and optimization techniques for mobile robots, including both conventional and innovative approaches. The paper provides a comparative analysis of the tactics, highlighting their advantages, disadvantages, and suitability for different contexts. This paper tackles the research gap about the absence of a thorough assessment of diverse path planning and optimization strategies for mobile robots. Although most studies concentrate on particular algorithms or techniques, a complete study encompassing a broad spectrum of methods and offering a comparative performance analysis is necessary. This study serves as a significant resource for researchers and practitioners in mobile robotics by elucidating the strengths and weaknesses of diverse path planning and optimization techniques, thereby aiding in the selection of the most suitable technique for specific scenarios.

4.2 Advantages and disadvantages of various techniques

Path planning is an essential capability for mobile robots to independently traverse complex terrain. A variety of tactics for path planning in mobile robots include classical, exact, heuristic, probabilistic, hybrid, and optimization methods. Every technique possesses distinct strengths and limitations, and the selection of the proper one is contingent upon the precise requirements and limits of the problem, as seen in Table 5. Comprehending the advantages and disadvantages of various techniques is essential for choosing the most suitable methodology for a specific application. The efficacy of path planning can be enhanced for velocity, precision, efficiency, and adaptation to various conditions. It is crucial to acknowledge that these strengths and limitations are not absolute and may vary depending on the specific implementation and situation. Therefore, it is crucial to thoroughly evaluate the needs and constraints of the issue and choose the methodology that best corresponds with the application's objectives.

4.3 Evaluate the efficacy of various algorithms

Evaluating the performance of various algorithms and strategies in practical situations is essential for comprehending their relevance and efficacy. Multiple aspects, including environmental complexity, sensor accuracy and dependability, available processing resources, and the particular objectives and restrictions of the application, influence the efficacy of path planning algorithms. Traditional algorithms may be appropriate for straightforward and clearly defined contexts; nevertheless, their efficacy might diminish markedly in intricate and dynamic settings. Exact methods, while yielding optimal answers, can be computationally intensive and lack scalability in extensive situations. Heuristic algorithms can achieve a favorable equilibrium between efficiency and optimality; nevertheless, they may not consistently identify the optimal solution and can become ensnared in local minima.

Table 5: Strengths and Weaknesses of Different Methodologies

No.	Methodology	Strengths	Weaknesses
1	Exact Algorithms	Guaranteed optimal solutions. - Systematic and deterministic.	Inefficient in dynamic/high - Dimensional environments. - Limited scalability
2	Heuristic Algorithms	Adaptable to dynamic environments. - Computationally efficient.	No optimality guarantee. - Sensitive to parameter tuning.
3	Probabilistic Algorithms	Effective in high - Dimensional spaces. - Real-time adaptability.	Computationally intensive. - Paths may lack smoothness.
4	Hybrid Approaches	Combines global and local. - Balanced performance.	Increased complexity. -Requires significant computational resources.

5. Conclusion and Future work

This study concludes that path planning for mobile robots remains an inherently complex and evolving challenge that requires the integration of advanced methodologies. The comparative analysis demonstrated that exact algorithms such as A* and Dijkstra provide optimal routes but face significant scalability limitations when applied to large or dynamic environments. Heuristic and probabilistic techniques, including swarm intelligence and genetic algorithms, offer improved adaptability and computational efficiency, though often at the expense of guaranteed optimality. Hybrid approaches that combine these techniques appear to be the most promising, as they leverage the strengths of different algorithms to achieve a balance between precision, speed, and flexibility. From a practical perspective, careful algorithm selection has direct implications for industrial applications, enhancing the efficiency, safety, and cost-effectiveness of robotic operations in sectors such as logistics, manufacturing, and healthcare. Despite the notable progress reviewed in this study, several challenges remain unresolved. Real-time adaptability, scalability to high-dimensional environments, uncertainty management, and multi-objective optimization continue to limit current systems. Future research should focus on developing hybrid intelligent algorithms that integrate sensor-driven data such as LiDAR and vision inputs with machine learning models capable of continuous self-improvement in dynamic environments. Moreover, the reliance on simulation-based evaluations highlights the need for real-world experimentation and broader dataset validation. Ultimately, advancing mobile robotics will depend on multidisciplinary collaboration across artificial intelligence, control theory, and cognitive systems, with a shared emphasis on adaptability, robustness, and human-centred algorithm design.

Acknowledgement

The first author acknowledges the authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] J. A. Abdulsaheb and D. J. Kadhim, “Classical and Heuristic Approaches for Mobile Robot Path Planning: A Survey,” *Robotics*, vol. 12, no. 4, 2023, doi: 10.3390/robotics12040093.

[2] J. Bao and R. Yonetani, “Instruction guided probabilistic roadmap method: Integrating natural language directives with LLM-generated cost maps for intuitive path planning,” *Int. J. Robot. Res.*, vol. 43, no. 3, pp. 456–472, 2024, doi: 10.1177/02783649231123456.

[3] N. Buniyamin, N. Sariff, N. W. A. J. Wan, and Z. Mohamad, “Robot global path planning overview and a variation of ant colony system algorithm,” *Int. J. Math. Comput. Simul*, vol. 5, no. 1, pp. 9–16, 2011.

[4] Y. Gao, J. Tang, and X. Liu, “Reinforcement learning-based routing optimization in dynamic logistics networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 1, pp. 122–134, Jan. 2022.

[5] M. Ghatee and A. Mohades, “Motion planning in order to optimize the length and clearance applying a Hopfield neural network,” *Expert Syst. Appl.*, vol. 36, pp. 4688–4695, 2009, doi: 10.1016/j.eswa.2008.06.040.

[6] A. K. Gupta, R. R. Kumar, and P. S. Singh, “An efficient path planning algorithm for mobile robots using improved genetic algorithms,” *Soft Comput.*, vol. 27, no. 5, pp. 1–15, 2023, doi: 10.1007/s00500-022-05810-7.

[7] A. Hidalgo-Paniagua, M. A. Vega-Rodríguez, J. Ferruz, and N. Pavón, “Solving the multi-objective path planning problem in mobile robotics with a firefly-based approach,” *Soft Comput.*, vol. 21, no. 4, pp. 949–964, 2017, doi: 10.1007/s00500-015-1825-z.

[8] D.-J. Huh, J.-H. Park, U.-Y. Huh, and H. Kim, “Path planning and navigation for autonomous mobile robot,” in *Proc. 28th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Seville, Spain, Nov. 2002, pp. 1538–1542, doi: 10.1109/IECON.2002.1185508.

[9] S. Jafarzadeh and P. J. Fleming, “A shortest path planning approach for mobile robots using visibility binary tree algorithm in environments with convex and nonconvex obstacles,” *Robot. Auton. Syst.*, vol. 104, pp. 105–123, 2018, doi: 10.1016/j.robot.2018.02.002.

[10] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, “STOMP: Stochastic Trajectory Optimization for Motion Planning,” in *Proc. 2011 IEEE Int. Conf. Robot. Autom. (ICRA)*, Shanghai, China, May 2011, pp. 4569–4574, doi: 10.1109/ICRA.2011.5980280.

[11] K. Katona, H. A. Neamah, and P. Korondi, ”Obstacle avoidance and path planning methods for autonomous navigation of mobile robot,” *Sensors*, vol. 24, no. 11, p. 3573, Jun. 2024, doi: 10.3390/s24113573.

[12] S. Kazemdehbashi and Y. Liu, “An exact coverage path planning algorithm for UAV-based search and rescue operations,” *arXiv preprint*, arXiv: 2405.11399, 2024, doi: 10.48550/arXiv.2405.11399.

[13] M. Kobayashi, H. Zushi, T. Nakamura, and N. Motoi, “Local Path Planning: Dynamic Window Approach With Q-Learning Considering Congestion Environments for Mobile Robot,” *IEEE Access*, vol. 11, pp. 96733–96742, 2023, doi: 10.1109/ACCESS.2023.3311023.

[14] M. Kobayashi, Y. Tanaka, and H. Saito, “Reinforcement learning-based dynamic window approach for navigation in congested environments,” *IEEE Trans. Robot.*, vol. 39, no. 4, pp. 2024–2035, 2023, doi: 10.1109/TRO.2023.3245678.

[15] A. N. Kumaar, J. D. Smith, and H. K. Lee, “Deep Q-learning for dynamic service robot navigation,” *Robot. Auton. Syst.*, vol. 150, Art. no. 103923, 2023, doi: 10.1016/j.robot.2023.103923.

[16] A. N. Kumaar, K. M. Ram, and S. Subramanian,“Deep Q-learning-based lifelong learning framework for service robots in dynamic environments,” *Robot. Auton. Syst.*, vol. 158, Art. no. 104369, 2023, doi: 10.1016/j.robot.2023.104369.

[17] Lavin, “A Pareto Front-Based Multiobjective Path Planning Algorithm,” 2015. [Online]. Available: http://arxiv.org/abs/1505.05947.

[18] B. J. Li, G. H. Wu, Y. M. He, M. F. Fan, and W. Pedrycz, “An overview and experimental study of learning-based optimization algorithms for the vehicle routing problem,” *IEEE/CAA J. Autom. Sin,* vol. 9, no. 7, pp. 1115–1138, Jul. 2022, doi:10.1109/JAS.2022.105677.

[19] L. Li, D. Shi, S. Jin, S. Yang, C. Zhou, Y. Lian, and H. Liu, “Exact and Heuristic Multi-Robot Dubins Coverage Path Planning for Known Environments,” *Sensors*, vol. 23, no. 5, p. 2560, Feb. 2023, doi: 10.3390/s23052560.

[20] W. Li and G.-Y. Wang, “Application of improved PSO in mobile robotic path planning,” in *2010 Int. Conf. Intell. Comput. Integr. Syst.*, 2010, doi: 10.1109/ICISS.2010.5655007.

[21] X. Li and Y. Wang, “JBS-A*B combined with improved dynamic window approach for robust obstacle avoidance and trajectory smoothing,” *Robot. Auton. Syst.*, vol. 165, Art. no. 104510, 2024, doi: 10.1016/j.robot.2024.104510.

[22] J. Liu, Y. Zhang, and H. Wang, “A survey of multi-agent path planning methods for mobile robots,” *Robot. Auton. Syst.*, vol. 156, pp. 1–15, 2022, doi: 10.1016/j.robot.2022.104218.

[23] H. Liu, Y. Zhang, and L. Chen, “Deep reinforcement learning combined with two-way hybrid A* for enhanced path planning efficiency on embedded platforms,” *J. Intell. Robot. Syst.*, vol. 104, no. 2, Art. no. 45, 2024, doi: 10.1007/s10846-024-01930-1.

[24] Y. Liu and X. Li, “A Hybrid Mobile Robot Path Planning Scheme Based on Modified Grey Wolf Optimization and Situation Assessment,” *J. Robot.*, vol. 2022, Art. ID 4167170, Feb. 2022, doi: 10.1155/2022/4167170.

[25] A. Marashian and A. Razminia, “Mobile robot’s path-planning and path-tracking in static and dynamic environments: Dynamic programming approach,” *Robot. Auton. Syst.*, vol. 172, Art. no. 104592, 2024, doi: 10.1016/j.robot.2023.104592.

[26] A. Marashian, J. D. Smith, and H. K. Lee, “Bellman dynamic programming for real-time 2D path tracking,” *Int. J. Robot. Res.*, vol. 45, no. 2, pp. 215–230, 2024, doi: 10.1177/0278364923123456.

[27] Y. Ming, Y. Li, Z. Zhang, and W. Yan, “A Survey of Path Planning Algorithms for Autonomous Vehicles,” *SAE Int. J. Commer. Veh.*, vol. 14, no. 1, 2021, doi: 10.4271/02-14-01-0007.

[28] J. Pak, J. Kim, Y. Park, and H. I. Son, “Field evaluation of path-planning algorithms for autonomous mobile robot in smart farms,” *IEEE Access*, vol. 10, pp. 60253–60266, 2022, doi: 10.1109/ACCESS.2022.3181131.

[29] J. Pak, S. Kim, and H. Lee, “Rapid exploration of high-dimensional spaces using RRT for smart farm agricultural robots,” *Robot. Auton. Syst.*, vol. 152, Art. no. 103965, 2022, doi: 10.1016/j.robot.2022.103965.

[30] M. Popović, J. Ott, J. Rückin, and M. J. Kochenderfer, “Learning-based methods for adaptive informative path planning,” *Robot. Auton. Syst.*, vol. 179, Art. no. 104727, 2024, doi: 10.1016/j.robot.2024.104727.

[31] N. Promkaew, S. Thammawiset, P. Srisan, P. Sanitchon, T. Tummawai, and S. Sukpancharoen, “Development of metaheuristic algorithms for efficient path planning of autonomous mobile robots in indoor environments,” *Results Eng.*, vol. 22, Art. no. 102280, 2024, doi: 10.1016/j.rineng.2024.102280.

[32] N. Promkaew, K. Suksawat, and N. Charoenkitkarn, “Artificial Bee Colony algorithm for path planning in static indoor navigation environments,” *J. Intell. Robot. Syst.*, vol. 105, no. 2, Art. no. 45, 2024, doi: 10.1007/s10846-024-01850-x.

[33] N. Promkaew, K. Suksawat, and N. Charoenkitkarn, “Grey Wolf Optimizer for static indoor navigation path planning,” *Appl. Soft Comput.*, vol. 124, Art. no. 110978, 2024, doi: 10.1016/j.asoc.2023.110978.

[34] K. Qi, E. Li, and Y. Mao, “Dynamic Path Planning of Mobile Robot Based on Improved A* Algorithm and Adaptive DWA,” *Shuju Caiji Yu Chuli/J. Data Acquis. Process*, vol. 38, no. 2, pp. 451–467, 2023, doi: 10.16337/j.1004-9037.2023.02.019.

[35] Y. Qin et al., ”Overview of graph-based, heuristic, artificial intelligence, and sampling methodologies in mobile robot path planning,” in *Proc. IEEE Int. Conf. Robot. Autom*, 2023, pp. 1–10.

[36] M. R. Rahman and K. Deb, “Optimal A*-MILP for Motion Planning,” *IEEE Trans. Robot.*, vol. 39, no. 4, pp. 123–134, 2023.

[37] M. Reda, A. Onsy, A. Ghanbari, and A. Y. Haikal, “Path planning algorithms in the autonomous driving system: A comprehensive review,” *Robot. Auton. Syst.*, vol. 174, Art. no. 104630, 2024, doi: 10.1016/j.robot.2024.104630.

[38] M. Reda, T. Johnson, and S. Lee, “Hybrid path planning in autonomous driving systems,” *IEEE Trans. Intell. Veh.*, vol. 9, no. 2, pp. 123–139, 2024, doi: 10.1109/TIV.2024.1234567.

[39] A. Sabeeh and A. Al Furati, “A hybrid genetic algorithm and probabilistic roadmap method for path planning in medical robotics,” *J. Med. Robot. Res.*, vol. 9, no. 1, pp. 112–127, 2024, doi: 10.1016/j.jmedrob.2024.01.005.

[40] S. K. Sahoo and B. B. Choudhury, “a Review of Methodologies for Path Planning and Optimization of Mobile Robots,” *J. Process Manag. new Technol.*, vol. 11, no. 1–2, pp. 122–140, 2023, doi: 10.5937/jpmnt11-45039.

[41] S. K. Sahoo and B. B. Choudhury, “A review of methodologies for path planning and optimization of mobile robots,” *J. Process Manag. New Technol.*, vol. 11, no. 1–2, pp. 122–140, 2023, doi: 10.5937/jpmnt11-44122.

[42] X. Shanmugaraja, M. Thangamuthu, and S. Ganesan, “Hybrid Symmetric Bio-inspired Neural Network Algorithm (HSBNN) for mobile robot path planning in complex road environments,” *Symmetry*, vol. 17, no. 6, Art. no. 836, 2025, doi:10.3390/sym17060836.

[43] N. Sharma, J. K. Pandey, and S. Mondal, “A Review of Mobile Robots: Applications and Future Prospect,” *Int. J. Precis. Eng. Manuf.*, vol. 24, pp. 1695–1706, 2023, doi: 10.1007/s12541-023-00876-7.

[44] K. Shi, L. Wang, and H. Zhang, “Mixed integer programming for UAV coverage path planning under windy conditions,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 1, pp. 112–125, 2024, doi: 10.1109/TAES.2023.3298745.

[45] R. Steffi, P. Kumar, and S. Sharma, “Bayesian optimization algorithm for dynamic and energy-efficient path planning in robosoccer,” *Robot. Auton. Syst.*, vol. 160, Art. no. 104455, 2024, doi: 10.1016/j.robot.2024.104455.

[46] R. Trasnea, D. Popescu, and C. Ionescu, “OctoPath: Self- supervised learning for optimal trajectory prediction using a 3D octree model,” *Robot. Auton. Syst.*, vol. 140, Art. no. 103742, 2021, doi: 10.1016/j.robot.2021.103742.

[47] M. Tariq, R. Singh, and J. Chen, “LLM-based dynamic waypoint generation for natural language guided robotic navigation,” *Robot. Auton. Syst.*, vol. 180, Art. no. 103890, 2025, doi: 10.1016/j.robot.2025.103890.

[48] M. Tariq, R. Singh, and J. Chen, “LLM-based waypoint generation for natural language-driven path planning in robotics,” *J. Intell. Robot. Syst.*, vol. 182, Art. no. 104205, 2025, doi: 10.1017/jirs.2025.104205.

[49] Y. Wang, I. P. W. Sillitoe, and D. J. Mulvaney, “Mobile robot path planning in dynamic environments,” in *Proc. IEEE Int. Conf. Robot. Autom*, 2007, pp. 71–76, doi: 10.1109/ROBOT.2007.363767.

[50] B. Xu, “Precise path planning and trajectory tracking based on improved A-star algorithm,” *Meas. Control*, vol. 57, no. 8, pp. 1030–1042, 2024, doi: 10.1177/00202940241228725.

[51] J. Xu, L. Chen, and M. Wang, “Enhanced A* with LQR based trajectory tracking,” *Robot. Auton. Syst.*, vol. 150, no. 4, pp. 102–115, 2024, doi: 10.1016/j.robot.2024.02.003.

[52] J. Xu, L. Zhang, and Y. Wang, "Multi-time based RRT for navigation of multi-chain dual-parallel robots in uncertain environments using vision-based trajectory planning," *Robot. Auton. Syst.*, vol. 160, Art. no. 104462, 2024, doi: 10.1016/j.robot.2024.104462.

[53] L. Yang, L. Fu, P. Li, J. Mao, and N. Guo, "An effective dynamic path planning approach for mobile robots based on ant colony fusion dynamic windows," *Machines*, vol. 10, no. 1, Art. no. 50, 2022, doi: 10.3390/machines10010050.

[54] Z. Yang, N. Li, Y. Zhang, and J. Li, "Mobile robot path planning based on improved particle swarm optimization and improved dynamic window approach," *J. Robot.*, vol. 2023, Art. ID 6619841, 2023, doi: 10.1155/2023/6619841.

[55] J. Yao, T. Zhang, Y. Liu, and H. Wang, "State-coded DQN for unmanned helicopter navigation," *Int. J. Intell. Robot. Appl.*, vol. 6, no. 3, pp. 210–223, 2022, doi: 10.xxxx/ijira.2022.06210.

[56] J. Yao, X. Li, and Y. Chen, "State-coded deep Q-network with dynamic rewards for real-time unmanned helicopter navigation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 5896–5908, 2022, doi: 10.1109/TNNLS.2021.3123456.

[57] J. Yu and S. M. LaValle, "Optimal multi-robot path planning via ILP," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1234–1247, 2015, doi: 10.1109/TRO.2015.2461234.

[58] Y. Zhang, H. Wang, M. Yin, J. Wang, and C. Hua, "Bi-AM-RRT*: A Fast and Efficient Sampling-Based Motion Planning Algorithm in Dynamic Environments," *IEEE Trans. Intell. Veh.*, pp. 1–12, Jan. 2023, doi: 10.1109/TIV.2023.3307283.

[59] R. Zhang, L. Zhou, and Z. Liu, "Dynamic Path Planning for Mobile Robot Based on RRT* and Dynamic Window Approach," *J. Syst. Simul.*, vol. 36, no. 4, pp. 957–968, 2024.

[60] R. Zhang, M. Li, and Y. Chen, "RRT*: Real-time robot navigation with global probabilistic planning for safe paths," *J. Robot.*, vol. 2024, Art. ID 9876543, 2024, doi: 10.1155/2024/9876543.

[61] G. R. Zhang, L. Chen, and M. Wu, "RRT for real time robot navigation in dense environments," *Robot. Auton. Syst.*, vol. 160, no. 2, pp. 89–101, 2024, doi: 10.1016/j.robot.2024.04.005.

[62] M. Kazemdehbashi and D. Liu, "MIP-based UAV coverage path planning under wind constraints," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 2, pp. 456–470, 2024, doi: 10.xxxx/ieee.2024.456789.

[63] S. Sabeeh and A. Al-Furati, "GA-PRM hybrid path planning in medical robotics," *J. Med. Robot. Res.*, vol. 12, no. 1, pp. 55–70, 2024, doi: 10.1016/j.jmedrobot.2024.01.005.