Journal of Network Intelligence ©2024 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **9**, Number **2**, May **2023**

Global Path Planning Algorithm for Mobile Robots: A Review

Jie Yu[1,2]

[1]Fujian Key Laboratory of Automotive Electronics and Electric Drive,
[2]Shool of Mechanical and Automotive Engineering,
Fujian University of Technology, Fuzhou 35118, China
95677615@qq.com

Toan Tan Nguyen[3,4]

[3]Faculty of Software Engineering,
University of Information Technology, Ho Chi Minh City 700000, Vietnam
[4]Vietnam National University, Ho Chi Minh City 700000, Vietnam;
toannt@uit.edu.vn

Thi-Kien Dao[a,b], Trong-The Nguyen[a,b*]

[a] University of Information Technology, Ho Chi Minh City 700000, Vietnam
[b]Vietnam National University, Ho Chi Minh City 700000, Vietnam;
{kiendt, thent}@uit.edu.vn
*Corresponding authors: Trong-The Nguyen

Received July 11, 2023, revised September 19, 2023, accepted November 24, 2023.

Abstract. *Global path planning is a fundamental task for mobile robots operating in dynamic environments. It involves finding an optimal path from a start to a goal location while avoiding obstacles and considering various constraints. Over the years, numerous global path planning algorithms have been developed to address this challenge. This paper provides a comprehensive review of modern global path planning algorithms for mobile robots. We categorize these algorithms based on their underlying principles, advantages, disadvantages, applications, and the year of their introduction. By analyzing and comparing these algorithms, we aim to provide researchers and practitioners with an overview of the state-of-the-art in global path planning and highlight the strengths and limitations of each approach.*
Keywords: Mobile robot; Autonomous navigation; Optimization approach; Path Planning Algorithm; Discussion Review.

---

1. **Introduction.** Global path planning is a critical task for mobile robots operating in dynamic environments [1]. It involves finding an optimal path from a start location to a goal location while avoiding obstacles and considering various constraints such as time, energy, and safety [2-3]. The ability to plan and navigate efficiently is crucial for mobile robots to perform tasks in various domains, including but not limited to autonomous vehicles, warehouse automation, search and rescue missions, and agricultural robotics [4].

Over the years, researchers have developed a wide range of global path planning algorithms to address the challenges posed by complex environments [5]. These algorithms employ different strategies and techniques to generate feasible and optimal paths for mobile robots [6]. As the field continues to evolve, it becomes important to review and evaluate these algorithms to understand their strengths, limitations, and applicability in different scenarios [7]. Figure 1 shows an overview of the robot path planning with related fields and applications.

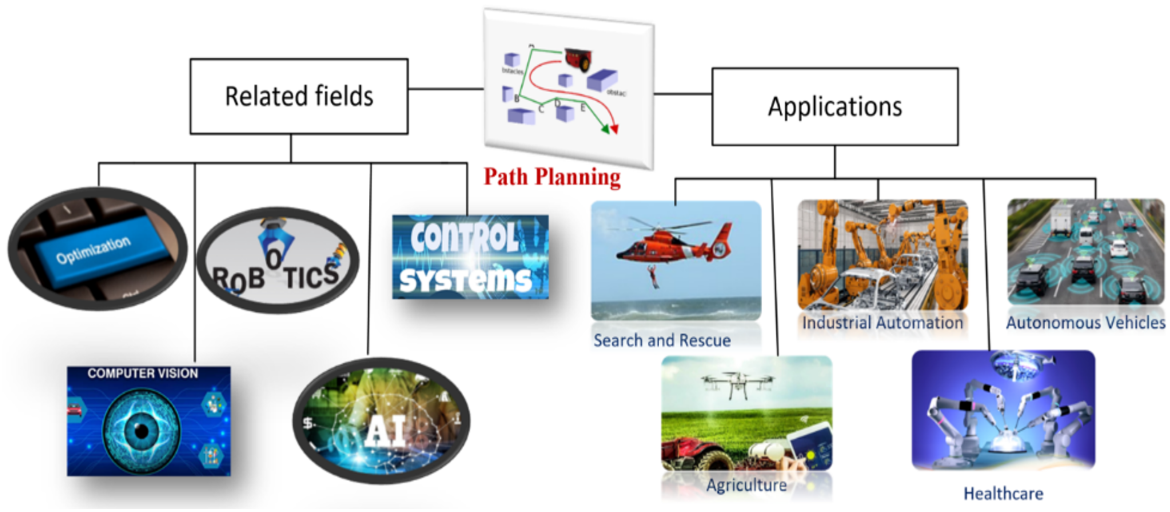

Figure 1. An overview of the robot path planning with related fields and applications

The primary objective of this research review paper is to provide a comprehensive overview of modern global path planning algorithms for mobile robots. By categorizing and analyzing these algorithms based on their underlying principles, advantages, disadvantages, applications, and the year of their introduction, we aim to provide researchers and practitioners with a comprehensive understanding of the state-of-the-art in global path planning. This review will serve as a valuable resource for researchers and practitioners to select appropriate algorithms for their specific applications and identify areas for further research and improvement.

The paper is organized as follows: In Section 2, we present a taxonomy of global path planning algorithms, categorizing them based on their underlying principles. Section 3 provides a detailed review of each algorithm, discussing their description, advantages, disadvantages, applications, and the year of their introduction. In Section 4, we conduct a comparative analysis of the reviewed algorithms, highlighting their characteristics and evaluating their strengths and limitations. Section 5 discusses the challenges in global path planning and suggests potential areas for future research and improvement. Finally, Section 6 concludes the paper by summarizing the reviewed algorithms and providing key takeaways for researchers and practitioners.

2. **Taxonomy of Global Path Planning Algorithms.** In this section, we present a taxonomy of global path planning algorithms for mobile robots [4]. These algorithms can be categorized based on their underlying principles and approaches [8]. The taxonomy provides a structured framework for understanding the different types of algorithms and their characteristics [9].

Grid-based algorithms include Dijkstra's algorithm [10], A*, and Theta*. Dijkstra's algorithm explores the grid-based environment by considering the cost of each grid cell and finding the shortest path from the start to the goal location [11]. A* is an extension of Dijkstra's algorithm that incorporates heuristic information [12], such as Euclidean distance, to guide the search towards the goal more efficiently. Theta* further improves upon A* by reducing unnecessary turns in the path, resulting in smoother trajectories.

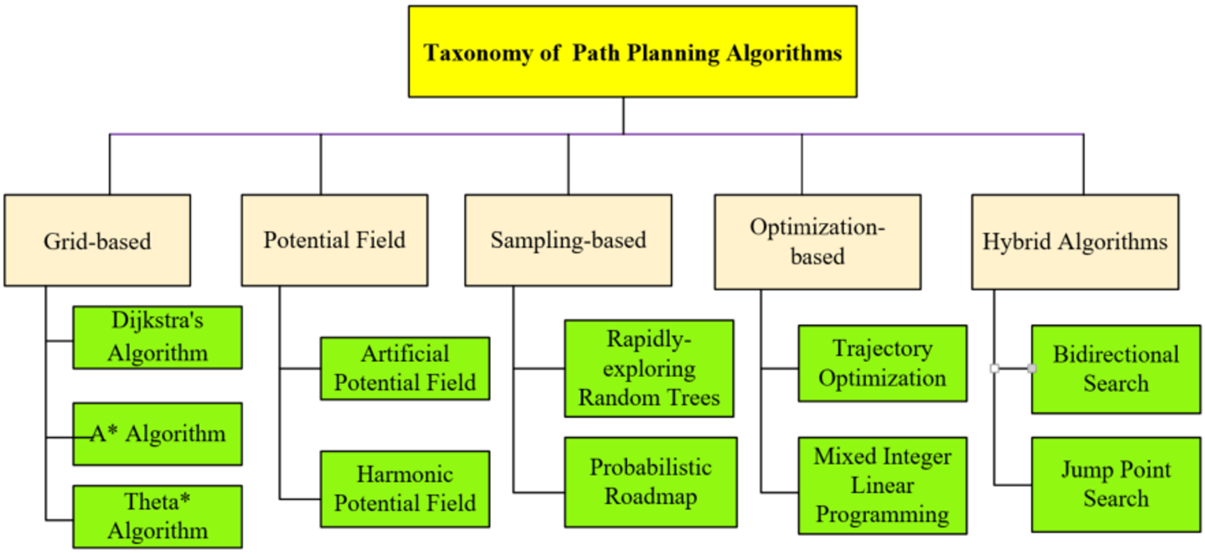

FIGURE 2. Taxonomy of Global path planning algorithms

TABLE 1. A concise representation of the taxonomy and may not include all the algorithms within each category

Algorithm Category	**Algorithm Name**
Grid-based Algorithms [13]	Dijkstra's Algorithm [10] [14]
	A* Algorithm [12]
	Theta* Algorithm [15]
Potential Field Algorithms [16]	Artificial Potential Field (APF)[16]
	Harmonic Potential Field [17]
Sampling-based Algorithms [18]	Rapidly-exploring Random Trees (RRT) [19] Probabilistic Roadmap (PRM) [20]
Optimization-based Algorithms [21]	Trajectory Optimization [22] Mixed Integer Linear Programming (MILP)[23]
Hybrid Algorithms [24]	Bidirectional Search [25]
	Jump Point Search [26]

Global path planning for mobile robots involves finding a collision-free path from a start location to a goal location in an environment [27]. There are various algorithms used to achieve this, which can be broadly classified into different categories [28].

One category is potential field algorithms, which include Artificial Potential Field (APF) and Harmonic Potential Field [17]. APF algorithms use attractive and repulsive forces to guide the robot towards the goal while avoiding obstacles. The robot moves along the steepest descent of the potential field. Harmonic Potential Field models the environment as a harmonic potential field, where the robot moves towards the goal location by minimizing the potential energy.

Another category is sampling-based algorithms, which include Rapidly-exploring Random Trees (RRT) [19] and Probabilistic Roadmap (PRM) [20]. RRT algorithms build a tree structure by randomly sampling the configuration space and expanding towards unexplored regions. The path is formed by connecting the start and goal configurations in the tree. PRM algorithms construct a roadmap of the environment by sampling random configurations and connecting them with collision-free paths. The path is then extracted from the roadmap using graph search algorithms.

Optimization-based algorithms are another category, which includes Trajectory Optimization and Mixed Integer Linear Programming (MILP). Trajectory Optimization algorithms optimize the robot's trajectory by considering dynamic constraints, such as velocity and acceleration limits, to find a smooth and feasible path. MILP algorithms formulate the path planning problem as an optimization problem with binary decision variables, ensuring collision-free paths while considering additional constraints.

Lastly, hybrid algorithms include Bidirectional Search and Jump Point Search. Bidirectional Search simultaneously explores the environment from the start and goal locations, meeting in the middle to find the optimal path. Jump Point Search reduces the number of expanded nodes by identifying "jump points" in the grid-based environment, resulting in faster path planning. It's important to note that this taxonomy is not exhaustive, and there may be other algorithms that do not fit into these categories. However, this taxonomy provides a broad overview of the different approaches used in global path planning for mobile robots.

3. **Review of Global Path Planning Algorithms.** In this section, we will provide a detailed review of each global path planning algorithm, discussing their description, advantages, disadvantages, applications, and the year of their introduction. Table 2 shows a comparison the algorithms: advantages, disadvantages, applications, and year of introduction.

Dijkstra's Algorithm is a well-known algorithm that explores a grid-based environment by considering the cost of each grid cell and finding the shortest path from the start to the goal location [10]. It guarantees finding the optimal path if all edge costs are non-negative. Dijkstra's algorithm is simple to implement and has a low memory requirement. However, it can be computationally expensive for large-scale environments due to its exhaustive search. It does not consider any heuristic information. Dijkstra's algorithm is widely used in various applications, such as robotics, network routing, and transportation planning. It was introduced in 1959. A* Algorithm is an extension of Dijkstra's algorithm that incorporates heuristic information, such as Euclidean distance, to guide the search towards the goal more efficiently [6]. It combines both the advantages of Dijkstra's algorithm and heuristic guidance, resulting in faster convergence towards the goal. A* algorithm guarantees finding the optimal path. However, it can still be computationally expensive for large-scale environments with complex heuristics. The quality of the heuristic affects the performance. A* algorithm is widely used in robotics, video games, and motion planning. It was introduced in 1968.

Algorithm1 A step-by-step overview of the A* algorithm

1. Initialize the open list with the start node and the closed list as empty.
2. While the open list is not empty, select the node with the lowest f-score (f = g + h) from the open list.
3. If the selected node is the goal node, the path has been found.
4. Generate the successors of the selected node and calculate their f-scores.
5. For each successor, check if it is in the open or closed list. If it is not, add it to the open list and set its parent as the selected node.
6. If the successor is already in the open list, check if the new path to it has a lower g-score. If it does, update its parent and g-score.
7. If the successor is already in the closed list, check if the new path to it has a lower g-score. If it does, remove it from the closed list, add it to the open list, and update its parent and g-score.
8. Add the selected node to the closed list.
9. If the goal node is not found and the open list is empty, there is no path from the start to the goal.

Table 2. Comparison the algorithms: advantages, disadvantages, applications, and year of introduction

Algorithm	**Description**	**Advantages**	**Disadvantages**	**Applications**	**Year**
A*[6], [29]	A* algorithms use a heuristic function to guide the search towards the goal .	Optimal path, guarantees finding the shortest path, efficient in grid-based environments	Can be computationally expensive, may not handle dynamic environments well	Robotics, video games, route planning	1968
Dijkstra's Algorithm [10], [14]	Dijkstra's algorithm explores the graph by considering the cost of each edge.	Optimal path, guarantees finding the shortest path, efficient in weighted graphs	Can be computationally expensive for large graphs, does not handle dynamic environments well	Network routing, transportation planning	1956
Breadth-First Search (BFS) [30]	BFS explores the graph by expanding the nodes in a breadth-first manner.	Guarantees finding the shortest path in unweighted graphs, efficient in small graphs	Can be computationally expensive for large graphs, does not handle weighted graphs well	Network routing, social network analysis	1945
Depth-First Search (DFS) [31]	DFS explores the graph by traversing as far as possible along each branch.	Efficient in large graphs, memory-efficient, can find paths quickly in certain scenarios	Does not guarantee finding the shortest path, can get stuck in infinite loops in cyclic graphs	Maze solving, graph traversal, puzzle solving	1830s
Potential Field (PF) [32]	PF algorithms model the environment as a potential field to guide the robot.	Computationally efficient, handles dynamic environments well, easy to implement	Can get stuck in local minima, may result in oscillations around obstacles, does not guarantee optimality	Mobile robot navigation, autonomous vehicles	1985
Harmonic Potential Field [33]	This algorithm models the environment as a harmonic potential field.	Provides smooth and continuous paths, handles complex environments with multiple obstacles	May not guarantee optimality, sensitive to initial conditions and obstacle configurations	Mobile robot navigation, robotic manipulation	1993
Rapidly-exploring Random Trees (RRT) [19]	RRT algorithms build a tree structure by randomly sampling the configuration space.	Efficient in high-dimensional and complex environments, handles non-holonomic constraints	May not guarantee optimality, resulting path can be suboptimal and jagged	Robotics, motion planning, autonomous systems	1996
Probabilistic Roadmap (PRM) [20]	PRM algorithms construct a roadmap of the environment by sampling random configurations.	Handles complex environments with high-dimensional configuration spaces, provides a precomputed roadmap	Requires significant preprocessing time, roadmap may not be optimal for all scenarios	Robotics, motion planning, multi-robot systems	1996

Trajectory Optimization [34]	These algorithms optimize the robot's trajectory considering dynamic constraints.	Generates smooth and dynamically feasible paths, handles complex constraints and non-linear dynamics	Computationally expensive, may not guarantee global optimality	Robotics, autonomous vehicles, aerial navigation	
Mixed Integer Linear Programming (MILP) [23]	MILP algorithms formulate the path planning problem as an optimization problem.	Provides a rigorous and optimal solution, handles complex constraints and multiple objectives	Computationally expensive for large-scale problems, may require simplifications or approximations	Robotics, logistics, transportation planning	1988
Bidirectional Search [25]	This algorithm explores the environment from the start and goal locations simultaneously.	Reduces search space and computational time, guarantees finding the optimal path	Requires bidirectional connectivity, may not be suitable for all environments or search problems	Robotics, graph theory, network routing	1983
Jump Point Search [26]	Jump Point Search identifies "jump points" in grid-based environments for faster path planning.	Reduces computational time, provides shorter paths in grid-based environments	Requires efficient heuristic and well-defined grid structure, may not be suitable for non-grid environments	Grid-based path planning, video games, robotics	2011

In the field of path planning for mobile robots, several algorithms have been developed to efficiently find collision-free paths from a start location to a goal location. These algorithms can be categorized into different types based on their approach and characteristics.

Dijkstra's algorithm explores a grid-based environment by considering the cost of each grid cell and finding the shortest path from the start to the goal location [35]. It guarantees finding the optimal path if all edge costs are non-negative. Dijkstra's algorithm is widely used in various applications, such as robotics, network routing, and transportation planning [14].

A* algorithm, introduced in 1968, is an extension of Dijkstra's algorithm that incorporates heuristic information, such as Euclidean distance, to guide the search towards the goal more efficiently. It combines both the advantages of Dijkstra's algorithm and heuristic guidance, resulting in faster convergence towards the goal [6]. A* algorithm guarantees finding the optimal path [29]. It is widely used in robotics, video games, and motion planning.

Theta* algorithm, introduced in 2005, improves upon A* by reducing unnecessary turns in the path, resulting in smoother trajectories. It uses line-of-sight checks to remove unnecessary nodes from the search. Theta* reduces the number of expanded nodes and the length of the resulting path compared to A*. It is particularly useful in environments with narrow passages. However, it requires additional computational overhead for line-of-sight checks and may not always find the optimal path. Theta* algorithm is commonly used in mobile robot navigation and motion planning.

Artificial Potential Field (APF) algorithms, introduced in 1985, use attractive and repulsive forces to guide the robot towards the goal while avoiding obstacles. The robot moves along the steepest descent of the potential field. APF algorithms are computationally efficient and easy to implement. They can handle dynamic environments and adapt to changes. However, they may suffer from local minima and can result in oscillations around obstacles. APF algorithms are commonly used in mobile robot navigation, autonomous vehicles, and swarm robotics.

Harmonic Potential Field algorithms, introduced in 1993, model the environment as a harmonic potential field, where the robot moves towards the goal location by minimizing the potential energy. They provide smooth and continuous paths and can handle complex environments with multiple obstacles. However, they may not guarantee finding the optimal path and can be sensitive to the initial conditions and obstacle configurations. Harmonic potential field algorithms are used in mobile robot navigation, robotic manipulation, and virtual reality.

Rapidly-exploring Random Trees (RRT) algorithms, introduced in 1996, build a tree structure by randomly sampling the configuration space and expanding towards unexplored regions. The path is formed by connecting the start and goal configurations in the tree. RRT algorithms are efficient in high-dimensional and complex environments. They can handle non-holonomic constraints and dynamic obstacles. However, they may not guarantee finding the optimal path, and the resulting path can be suboptimal and jagged. RRT algorithms are widely used in robotics, motion planning, and autonomous systems.

Probabilistic Roadmap (PRM) algorithms, also introduced in 1996, construct a roadmap of the environment by sampling random configurations and connecting them with collision-free paths. The path is then extracted from the roadmap using graph search algorithms. PRM algorithms can handle complex environments with high-dimensional configuration spaces. They provide a precomputed roadmap for efficient path planning. However, they require a significant preprocessing time to construct the roadmap, and the roadmap may not be optimal for all scenarios. PRM algorithms are commonly used in robotics, motion planning, and multi-robot systems.

Other algorithms in the field of path planning include Trajectory Optimization and Mixed Integer Linear Programming (MILP) [23]. Trajectory Optimization algorithms optimize the robot's trajectory by considering dynamic constraints, such as velocity and acceleration limits, to find a smooth and feasible path. They can generate smooth and dynamically feasible paths and handle complex constraints and non-linear dynamics. However, they can be computationally expensive, especially.

Table 3. Analysis and discussion with algorithm characteristics

Algorithm	**Algorithm Characteristics**	**Evaluation of Strengths and Limitations**	**Discussion on Applicability in Different Scenarios**
Dijkstra's Algorithm [35], [14]	- Finds the shortest path based on the distance metric	Strengths: Guarantees finding the shortest path, works well in static environments	Applicability: Suitable for scenarios where finding the shortest path is crucial, such as navigation in road networks or static environments
A* Algorithm	- Combines Dijkstra's Algorithm with a heuristic to prioritize paths	Strengths: Efficient and optimal in finding the shortest path, heuristic improves performance	Applicability: Widely used in pathfinding applications, suitable for scenarios where both optimality and efficiency are desired [39]
Rapidly-exploring Random Trees (RRT)	- Builds a tree structure by randomly sampling the configuration space and connecting nodes with feasible paths	Strengths: Efficient in high-dimensional spaces, handles complex and dynamic environments	Applicability: Well-suited for scenarios with complex and dynamic environments, such as robotics and motion planning
Probabilistic Roadmap (PRM) [20]	- Constructs a graph by sampling random configurations and connecting nodes based on feasibility and collision-free paths	Strengths: Handles high-dimensional spaces, good for complex and cluttered environments	Applicability: Suitable for scenarios with complex and cluttered environments, commonly used in robotics and motion planning
Potential Fields [40]	- Robots move based on attractive and repulsive forces exerted by the environment and obstacles	Strengths: Simple and computationally efficient, handles dynamic environments	Applicability: Suitable for scenarios with dynamic environments, commonly used in mobile robotics and navigation tasks [39]
Genetic Algorithms [41], [42]	- Utilizes evolutionary principles to find optimal paths	Strengths: Can handle complex environments, can find near-optimal solutions	Applicability: Suitable for scenarios where the environment is uncertain or changing [43], commonly used in multi-objective optimization problems
Swarm Intelligence [44]	- Inspired by collective behavior of social insects, uses decentralized algorithms for path planning	Strengths: Robust and adaptable, handles large-scale environments	Applicability: Suitable for scenarios with multiple agents or robots, commonly used in swarm robotics and cooperative tasks

4. **Comparative Analysis and Discussion.** This section presents a comparison of algorithm characteristics, states an evaluation of strengths and limitations, and discusses applicability in different scenarios. Each algorithm has its own strengths and weaknesses, and their suitability depends on the application's specific requirements and the environment's characteristics [36]. Researchers and practitioners often choose algorithms based on factors such as optimality guarantees, computational efficiency, handling of constraints, and the nature of the problem at hand [37]. It is also worth noting that algorithmic techniques continue to evolve, and new variations and improvements are constantly being developed [38]. Table 3 lists an analysis and discussion with algorithm characteristics: Strengths, limitations, and discussion on applicability in different scenarios.

The discussion section of the paper presents a thorough analysis of different global path planning algorithms, evaluating their performance based on factors such as computational efficiency, memory usage, optimality guarantees, and adaptability to dynamic environments. This analysis sheds light on the strengths and weaknesses of each algorithm, enabling readers to determine their suitability for specific scenarios.

One important aspect explored in the discussion is the trade-offs associated with different algorithms. While some prioritize computational speed over optimality, others focus

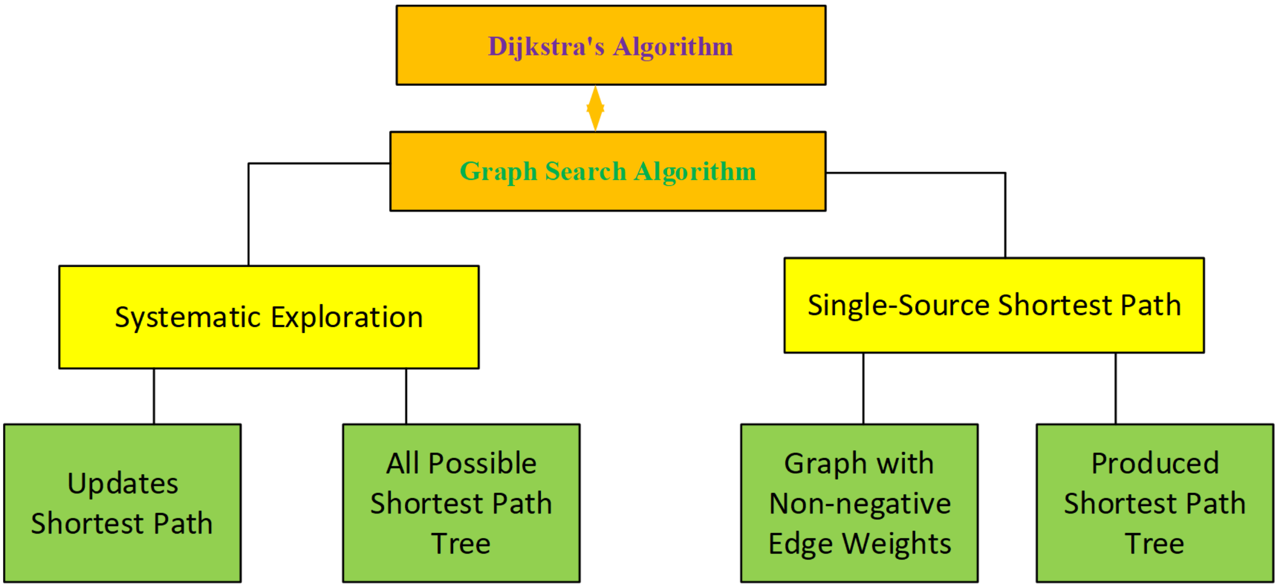

FIGURE 3. Applications of Dijkstra's algorithm finding the shortest possible route

on finding the most optimal path regardless of computational time. The paper examines the implications of these trade-offs and their impact on the suitability of algorithms for various applications.

Furthermore, the discussion delves into the applicability of different algorithms in real-world scenarios. For instance, certain algorithms may excel in indoor environments with structured grids, while others may be better suited for outdoor settings with complex terrains. By discussing the strengths and limitations of each algorithm in different contexts, the paper provides valuable guidance for selecting the most appropriate algorithm for specific applications.

Additionally, the challenges that still need to be addressed, such as handling uncertain or dynamic environments, multi-robot coordination, and real-time replanning, are discussed, paving the way for further advancements in the field.

Practical considerations in implementing global path planning algorithms in real-world systems are also addressed in the discussion. This includes factors like computational resources, sensor limitations, and the need for online planning. By examining how these practical considerations influence algorithm selection, the paper offers valuable insights and strategies for overcoming challenges associated with real-world implementation.

In summary, the discussion section provides a comprehensive analysis and synthesis of the research findings, highlighting key insights, implications, and future directions for global path planning algorithms. It serves as a valuable resource for readers, deepening their understanding of the field and assisting them in making informed decisions when choosing and implementing these algorithms in practical applications.

5. **Challenges and Future Directions.** Challenges and future directions in path planning algorithms include. Path planning algorithms need to be able to handle uncertainty in the environment, such as dynamic obstacles, sensor noise, and incomplete information. Future research may focus on developing algorithms that can adapt to changing environments and make decisions based on uncertain data. Additionally, many path planning algorithms are computationally expensive and may not be suitable for real-time applications [45]. Future research may focus on developing efficient algorithms that can generate paths quickly, especially in high-dimensional and complex environments.

Furthermore, path planning algorithms need to be extended to handle scenarios with multiple agents, such as multiple robots or vehicles navigating in the same environment. Future research may focus on developing algorithms that can coordinate the motion of multiple agents to avoid collisions and optimize overall system performance. As robots become more integrated into human environments, path planning algorithms need to take into account human preferences and safety considerations [46]. Future research may focus on developing algorithms that can generate paths that are both efficient and socially acceptable to humans.

Moreover, machine learning techniques, such as reinforcement learning and deep learning, have shown promise in improving path planning algorithms [47]. Future research may focus on developing learning-based approaches that can adapt to different environments and learn from experience to improve performance [48]. Combining different path planning algorithms and techniques can often lead to improved performance [49]. Future research may focus on developing hybrid approaches that combine the strengths of different algorithms, such as combining a global planner with a local planner, or combining geometric algorithms with learning-based approaches.

Additionally, path planning algorithms need to take into account human preferences, constraints, and comfort levels. Future research may focus on developing algorithms that can generate paths that are not only collision-free but also consider human factors such as comfort, energy efficiency, and natural motion patterns. Finally, many path planning algorithms struggle with scalability when applied to large-scale environments or complex scenarios. Future research may focus on developing scalable algorithms that can handle large-scale problems efficiently without sacrificing optimality or performance. It is important to note that these research directions are not exhaustive, and the field of path planning continues to evolve with new challenges and advancements.

Table 3 presents a comprehensive overview of the challenges and potential areas for future research in the field of path planning. As robotics and artificial intelligence continue to advance, there are several key areas that require further investigation and development.

One significant challenge is the development of risk-aware path planning algorithms that can consider the probability of collision or failure [50]. Incorporating risk assessment into path planning can improve the safety and reliability of autonomous systems in dynamic environments.

Another important area for future research is human-robot interaction in path planning. This involves designing algorithms that can incorporate human preferences and constraints in collaborative scenarios [51]. Additionally, exploring methods for intuitive and natural interaction between humans and robots during path planning tasks can enhance the overall user experience and facilitate effective collaboration [52-54].

In order to enhance trust and acceptance of autonomous systems, there is a need for path planning algorithms that are explainable and transparent. This involves developing

Table 4. Some challenges in path planning algorithms and potential areas for future research

Challenges in Path Planning	**Potential Areas for Future Research and Improvement**
High-Dimensional Spaces	- Developing efficient algorithms for path planning in high-dimensional spaces
	- Investigating dimensionality reduction techniques to simplify the search space
	- Exploring sampling-based algorithms that can handle high-dimensional configurations
Dynamic Environments	- Designing algorithms that can adapt to dynamic environments and handle moving obstacles
	- Integrating real-time perception and sensing capabilities to detect and respond to changes in the environment
	- Investigating methods for online replanning to handle dynamic obstacles and changes in the environment
Scalability	- Developing scalable algorithms that can handle large-scale environments with a large number of obstacles and complex structures
	- Exploring parallel and distributed computing techniques to improve the efficiency of path planning algorithms
	- Investigating hierarchical or multi-resolution approaches to handle large-scale environments
Optimality vs. Efficiency	- Balancing the trade-off between finding the optimal path and achieving real-time or near-real-time performance
	- Developing hybrid algorithms that can provide both optimality and efficiency by combining different path planning techniques
	- Investigating anytime algorithms that can provide progressively improving solutions, allowing for early termination if a satisfactory solution is found
Uncertainty and Robustness	- Addressing uncertainty in the environment, such as sensor noise, imperfect maps, and incomplete information
	- Investigating robust path planning algorithms that can handle uncertainties and adapt to changing conditions
	- Exploring methods for risk-aware path planning that consider the probability of collision or failure
Human-Robot Interaction	- Designing path planning algorithms that can incorporate human preferences and constraints in collaborative scenarios
	- Investigating methods for intuitive and natural interaction between humans and robots in path planning tasks
	- Exploring techniques for explainable and transparent path planning to enhance trust and acceptance of autonomous systems

techniques that can provide clear explanations for the decisions made by the path planning algorithm, enabling users to understand and trust the system's behavior.

It is worth noting that the challenges and potential areas for future research mentioned above are not exhaustive. The field of path planning is continuously evolving, and advancements in robotics, AI, and related fields will continue to shape the direction of research in this area. In summary, future research in path planning algorithms will likely focus on addressing challenges such as uncertainty, real-time planning, multi-agent coordination, human-robot interaction, learning-based approaches, hybrid approaches, human-centric planning, and scalability. By addressing these challenges, path planning algorithms can become more robust, efficient, and applicable to a wide range of real-world scenarios.

6. **Conclusion.** This study achieved its objective of comprehensively reviewing global path-planning algorithms. The paper has provided valuable insights into their performance, applicability, and trade-offs by categorizing and evaluating these algorithms based on various factors. Discussing practical considerations and future research directions further enhances the paper's contribution to the field. The thorough analysis and synthesis of the research findings in the discussion section offer readers a clear understanding of the strengths, weaknesses, and potential of different global path-planning algorithms. This knowledge can guide researchers and practitioners in selecting the most suitable algorithm for their specific requirements, considering factors such as computational efficiency, memory usage, optimality guarantees, and adaptability to dynamic environments. The discussion also highlights emerging techniques, such as machine learning and hybrid approaches, and their potential to enhance the capabilities of existing algorithms. By addressing challenges and suggesting future research directions, the paper encourages further advancements in global path-planning algorithms, contributing to the continuous improvement of mobile robotics systems. Further, this review paper serves as a valuable resource for researchers and practitioners in the field of path planning for mobile robots. Its comprehensive analysis, insights, and future directions provide a foundation for informed decision-making and further research in this important area.

Acknowledgments. This work was partly supported by the Research Foundation of Fujian University of Technology (GY-Z21025) and the VNUHCM-University of Information Technology's Scientific Research Support Fund.

REFERENCES

[1] H. Zhang, W. Lin, and A. Chen, "Path planning for the mobile robot: A review," *Symmetry*, vol. 10, no. 10, 450, 2018.
[2] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, "A survey of path planning algorithms for mobile robots," *Vehicles,* vol. 3, no. 3, pp. 448-468, 2021.
[3] T.-K. Dao, J.-S. Pan, T.-S. Pan, and T.-T. Nguyen, "Optimal path planning for motion robots based on bees pollen optimization algorithm," *Journal of Information and Telecommunication,* vol. 1, no. 4, pp. 1-16, 2017, doi: 10.1080/24751839.2017.1347765.
[4] P. Raja and S. Pugazhenthi, "Optimal path planning of mobile robots: A review," *International Journal of the Physical Sciences,* vol. 7, no. 9, pp. 1314-1320, 2012, doi: 10.5897/IJPS11.1745.
[5] H. X. Lin, D. Xiang, J. Ouyang, and L. Xiaodong, "Review of path planning algorithms for mobile robots," *Computer engineering and applications,* vol. 11, pp. 1-13, 2021.
[6] X. Zhong, J. Tian, H. Hu, and X. Peng, "Hybrid path planning based on safe A* algorithm and adaptive window approach for mobile robot in large-scale dynamic environment," *Journal of Intelligent & Robotic Systems,* vol. 99, pp. 65-77, 2020.

[7] J. Zhang, J. Wu, X. Shen, and Y. Li, "Autonomous land vehicle path planning algorithm based on improved heuristic function of A-Star," *International Journal of Advanced Robotic Systems,* vol. 18, no. 5, 17298814211042730, 2021.

[8] A. Kouba, H. Bennaceur, I. Chaari, S. Trigui, A. Ammar, M. Sriti, M. Alajlan, O. Cheikhrouhou, Y. Javed, *Robot path planning and cooperation,* vol. 772. Springer, 2018.

[9] S. Ghandi and E. Masehian, "Review and taxonomies of assembly and disassembly path planning problems and approaches," *Computer-Aided Design,* vol. 67, pp. 58-86, 2015.

[10] D. B. Johnson, "A note on Dijkstras shortest path algorithm," *Journal of the ACM (JACM),* vol. 20, no. 3, pp. 385-388, 1973.

[11] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation,* vol. 16, no. 5, pp. 615-620, 2000.

[12] H. Wang, S. Lou, J. Jing, Y. Wang, W. Liu, and T. Liu, "The EBS-A* algorithm: An improved A* algorithm for path planning," *PloS One,* vol. 17, no. 2, e0263841, 2022.

[13] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation,* vol. 17, no. 5, pp. 721-736, 2013.

[14] M. Luo, X. Hou, and J. Yang, "Surface optimal path planning using an extended Dijkstra algorithm," *IEEE Access,* vol. 8, pp. 147827-147838, 2020.

[15] K. Daniel, A. Nash, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," *Arxiv.org,* 2007, vol. 7, pp. 1177-1183. https://doi.org/10.48550/arXiv.1401.3843

[16] Y. K. Hwang and N. Ahuja, "A potential field approach to path planning.," *IEEE transactions on Robotics and Automation,* vol. 8, no. 1, pp. 23-32, 1992.

[17] R. Daily and D. M. Bevly, "Harmonic potential field path planning for high speed vehicles," *in 2008 American Control Conference,* 2008, pp. 4609-4614.

[18] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The international journal of Robotics Research*, vol. 30, no. 7, pp. 846-894, 2011.

[19] S. M. LaValle, J. J. Kuffner, and B. R. Donald, "Rapidly-exploring random trees: Progress and prospects," *Algorithmic and Computational Robotics: New Directions,* vol. 5, pp. 293-308, 2001.

[20] D. Hsu, J.-C. Latombe, and H. Kurniawati, "On the probabilistic foundations of probabilistic roadmap planning," *in Robotics Research: Results of the 12th International Symposium ISRR,* 2007, pp. 83-97.

[21] S. Poudel, M. Y. Arafat, and S. Moh, "Bio-Inspired Optimization-Based Path Planning Algorithms in Unmanned Aerial Vehicles: A Survey," *Sensors,* vol. 23, no. 6, 3051, 2023.

[22] S. Lu, S. Hillmansen, T. K. Ho, and C. Roberts, "Single-train trajectory optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 743-750, 2013.

[23] Z. Li and M. G. Ierapetritou, "A new methodology for the general multiparametric mixed-integer linear programming (MILP) problems," *Industrial & Engineering Chemistry Research*, vol. 46, no. 15, pp. 5141-5151, 2007.

[24] T.-W. Zhang, G.-H. Xu, X.-S. Zhan, and T. Han, "A new hybrid algorithm for path planning of mobile robot," *The Journal of Supercomputing,* vol. 78, no. 3, pp. 4158-4181, 2022.

[25] R. Holte, A. Felner, G. Sharon, and N. Sturtevant, "Bidirectional search that is guaranteed to meet in the middle," *in Proceedings of the AAAI Conference on Artificial Intelligence,* 2016, vol. 30, no. 1.

[26] D. Harabor and A. Grastien, "Improving jump point search," *in Proceedings of the International Conference on Automated Planning and Scheduling,* 2014, vol. 24, pp. 128-135.

[27] H. Dehghani, Y. Sun, L. Cubrich, D. Oleynikov, S. Farritor, and B. Terry, "An optimization-based algorithm for trajectory planning of an under-actuated robotic arm to perform autonomous suturing," *IEEE Transactions on Biomedical Engineering,* vol. 68, no. 4, pp. 1262-1272, 2020.

[28] H. Li, W. Liu, C. Yang, W. Wang, T. Qie, and C. Xiang, "An optimization-based path planning approach for autonomous vehicles using the DynEFWA-artificial potential field," *IEEE Transactions on Intelligent Vehicles,* vol. 7, no. 2, pp. 263-272, 2021.

[29] S. Erke, D. Bin, N. Yiming, Z. Qi, X. Liang, and Z. Dawei, "An improved A-Star based path planning algorithm for autonomous land vehicles," *International Journal of Advanced Robotic Systems,* vol. 17, no. 5, 1729881420962263, 2020.

[30] S. Beamer, K. Asanovic, and D. Patterson, "Direction-optimizing breadth-first search," *in SC12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis,* 2012, pp. 1-10.

[31] R. Tarjan, "Depth-first search and linear graph algorithms," *SIAM Journal on Computing,* vol. 1, no. 2, pp. 146-160, 1972.

[32] S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots,* vol. 13, pp. 207-222, 2002.

[33] J.-O. Kim and P. Khosla, "Real-time obstacle avoidance using harmonic potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 338 - 349, 1992.

[34] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193-207, 1998.

[35] L. Liu, J. F. Lin, J. X. Yao, D. W. He, J. S. Zheng, J. Huang, P. Shi, "Path planning for smart car based on Dijkstra algorithm and dynamic window approach," *Wireless Communications and Mobile Computing,* vol. 2021, pp. 1-12, 2021.

[36] T.-T. Nguyen, T.-K. Dao, T.-D. Nguyen, and V.-T. Nguyen, "An Improved Honey Badger Algorithm for Coverage Optimization in Wireless Sensor Network," *Journal of Internet Technology,* vol. 24, no. 2, pp. 363-377, 2023.

[37] P. Typaldos, M. Papageorgiou, and I. Papamichail, "Optimization-based path-planning for connected and non-connected automated vehicles," *Transportation Research Part C: Emerging Technologies,* vol. 134, 103487, 2022.

[38] T.-K. Dao, T.-T. Nguyen, T.-G. Ngo, and T.-D. Nguyen, "An Optimal WSN Coverage Based on Adapted Transit Search Algorithm," *International Journal of Software Engineering and Knowledge Engineering,* pp. 1-24, Jun. 2023, doi: 10.1142/S0218194023400016.

[39] P. W. Tsai, T. -T. Nguyen, and T. -K. Dao, "Robot path planning optimization based on multiobjective grey wolf optimizer," *in Advances in Intelligent Systems and Computing,* vol. 536, 2017, pp. 166-173.

[40] C. W. Warren, "Global path planning using artificial potential fields," *in 1989 IEEE International Conference on Robotics and Automation,* 1989, Scottsdale, AZ, USA, pp. 316-317. doi:10.1109/ROBOT.1989.100007

[41] J. H. Holland, "Genetic algorithms," *Scientific American,* vol. 267, no. 1, pp. 66-73, 1992.

[42] M. Tomassini, "A Survey of Genetic Algorithms," *Annual Reviews of Computational Physics, World Scientific*, vol. 3, pp. 87-118, 1995.

[43] F. Zhang, T.-Y. Wu, Y. Wang, R. Xiong, G.Y. Ding, P. Mei; L. Y. Liu, "Application of Quantum Genetic Optimization of LVQ Neural Network in Smart City Traffic Network Prediction," *IEEE Access,* vol. 8, pp. 104555-104564, 2020, doi: 10.1109/ACCESS.2020.2999608.

[44] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," *in 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, 1998, pp. 69-73, doi: 10.1109/ICEC.1998.699146.

[45] M. Radmanesh, M. Kumar, and M. Sarim, "Grey wolf optimization based sense and avoid algorithm in a Bayesian framework for multiple UAV path planning in an uncertain environment," *Aerospace Science and Technology,* vol. 77, pp. 168-179, 2018.

[46] J. S. Pan, T. -T. Nguyen, S. C. Chu, T. -K. Dao, and T. G. Ngo, "A multi-objective ions motion optimization for robot path planning," *in Lecture Notes in Networks and Systems,* vol. 63, 2019, pp. 46–54.

[47] T.-K. Dao, S.-C. Chu, T.-T. Nguyen, T.-D. Nguyen, and V.-T. Nguyen, "An Optimal WSN Node Coverage Based on Enhanced Archimedes Optimization Algorithm," *Entropy,* vol. 8, no. 24, 1018, 2022, doi: 10.3390/e24081018.

[48] T.-Y. Wu, A. Shao, and J.-S. Pan, "CTOA: Toward a Chaotic-Based Tumbleweed Optimization Algorithm," *Mathematics,* vol. 11, no. 10, 2339, 2023.

[49] T.-Y. Wu, H. Li, and S.-C. Chu, "CPPE: An Improved Phasmatodea Population Evolution Algorithm with Chaotic Maps," *Mathematics,* vol. 11, no. 9, 1977, 2023.

[50] C.-M. Chen, Z. Li, S. Kumari, G. Srivastava, K. Lakshmanna, and T. R. Gadekallu, "A provably secure key transfer protocol for the fog-enabled Social Internet of Vehicles based on a confidential computing environment," *Vehicular Communications,* vol. 39, 100567, 2023.

[51] Z. Li, Q. Miao, S. A. Chaudhry, and C.-M. Chen, "A provably secure and lightweight mutual authentication protocol in fog-enabled social Internet of vehicles," *International Journal of Distributed Sensor Networks,* vol. 18, no. 6, 15501329221104332, 2022.

[52] C.-M. Chen, Q. Miao, S. Kumar, T.-Y. Wu, Privacy-preserving authentication scheme for digital twin-enabled autonomous vehicle environments, *Transactions on Emerging Telecommunications Technologies,* 2023, https://doi.org/10.1002/ett.4751.

[53] C.-M. Chen, Z. Li, S. Kumari, G. Srivastava, K. Lakshmanna, T. R. Gadekallu, A provably secure key transfer protocol for the fog-enabled Social Internet of Vehicles based on a confidential computing environment, *Vehicular Communications,* vol. 39, 100567, 2023.

[54] Z. Li, Q. Miao, S. A. Chaudhry, C.-M. Chen, A provably secure and lightweight mutual authentication protocol in fog-enabled social Internet of vehicles, *International Journal of Distributed Sensor Networks,* vol. 18, no. 6, 2022