

Research Article

Path Planning of Mobile Robot Based on Improved PRM Based on Cubic Spline

Xiaolu Ma ¹, Rui Gong ¹, Yibo Tan ¹, Hong Mei ^{1,2} and Chengcheng Li ¹

¹School of Electrical and Information Engineering, Anhui University of Technology, Maanshan 243002, China

²School of Aeronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210000, China

Correspondence should be addressed to Xiaolu Ma; maxiaolu@ahut.edu.cn

Received 7 April 2022; Revised 2 October 2022; Accepted 28 October 2022; Published 22 November 2022

Academic Editor: Junaid Shuja

Copyright © 2022 Xiaolu Ma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In view of the shortcomings of low search efficiency and many path turning points of Probabilistic Roadmaps (PRM), a bidirectional search PRM global path planning algorithm is proposed. The algorithm improves the search connection rules by using the positive and negative directions to search the path alternately, so that the connection of unnecessary nodes reduces, thereby speeding up the efficiency of path planning. Besides, the algorithm incorporates cubic spline interpolation. That will increase the smoothness of path planning and ensure that the mobile robot can realize the path planning task more smoothly and safely. The simulation results show that the improved algorithm can effectively improve the convergence speed and path smoothness of the algorithm. Finally, the improved algorithm is applied to the actual mobile robot navigation experiment. The experimental results have proven that the path planning strategy was able to a superior advantage over traditional PRM in path quality and computational time.

1. Introduction

Recently, the technological development of mobile robots has made great progress, and the development of path planning technology has become an important research field of mobile robots [1, 2]. Nowadays, path planning technology has been extensively used in industry, service industry, entertainment industry, and so on. Effective and feasible path planning technology can make mobile robots more secure and reliable to complete the tasks required by human beings. The traditional path planning algorithms include Dijkstra algorithm, A* algorithm, and artificial potential field method [3–7]. A series of intelligent bionic algorithms gradually emerge with the increase of the complexity of the working space of mobile robots, such as genetic algorithm, particle swarm algorithm, artificial bee colony algorithm, and so on [8–13].

In the early 1990s, Svestka et al. proposed the Probabilistic Roadmaps (PRM) [14]. The PRM is based on random sampling strategy. The difficulty of constructing effective path graph in high-dimensional space is well solved because

of small calculation amount and good real-time performance of the algorithm [15]. However, search efficiency will decrease significantly in the case of narrow passages. Moreover, it is easy to fall into local optimum. In view of the above problems, a variety of optimization methods is proposed by domestic and foreign scholars. Reference [16] proposes an improved PRM, by optimizing the sampling strategy of traditional PRM, thereby increasing the number of sampling points in narrow passages. The algorithm increases search sampling efficiency. Velagić et al. change the connection distance between sampling points, so that the useless lines in the path network diagram are reduced. This method improves the efficiency of the overall path planning [17]. An improved PRM is proposed, which introduces an artificial potential field method in PRM, so that the point falling within the obstacle moves from the obstacle to the mobile area by the repulsive force of the potential field, thereby increasing the utilization rate of the sampling points [18]. Reference [19] presents Lazy PRM, an improved PRM that minimizes the number of collision checks performed during the planning and therefore minimizes the running

time of the planner. Reference [20] presents improved connectivity of the roadmap by connecting previously generated connected components. PRM* an optimal variant of the original PRM algorithm is presented in [21]. Reference [22] addresses the problem of generating navigation roadmaps for uncertain and cluttered environments represented with probabilistic occupancy maps, proposed the topology-informed growing neural gas algorithm that leverages estimates of probabilistic topological structures computed using persistent homology theory. Reference [23] presents a K-order surrounding roadmap path planner which constructs a roadmap in an incremental manner to produce high-quality roadmaps. Reference [24] proposes method using a layered hybrid Probabilistic Roadmap and the Artificial Potential Field method for global planning. In [25], the main PRM methods are analyzed through a huge set of simulations with a robot into complex static environments. Reference [26] presents a guidance mechanism that encourages the rapid construction of well-connected roadmaps with PRM methods. It leverages a topological skeleton of the workspace to track the algorithm's progress in both covering and connecting distinct neighborhoods, and employs this information to focus computation on the uncovered and unconnected regions. Reference [27] proposes a new approach that combines the technique of particle swarm optimization with PRM to improve both the utilization rate of sampling points and the success rate of narrow passages path planning. The above algorithms can speed up the efficiency of path search. However, path smoothness is also an important index of path planning task. In [28], spline interpolation is applied to robot navigation algorithm, and a fast search random tree algorithm based on spline is proposed, which ensures the continuity of curvature along the path and eliminates the sharp change of control in traditional RRT. In [29], the proposed method combines the classic bidirectional RRT algorithm with the cubic Ferguson's spline technique to generate trajectories that directs smoothness and continuation without any computation. In [30], Cubic Bezier curves are applied to PRMs to shepherd non-holonomic robots to the target while maintaining the whole configuration. In [31], to generate such trajectories, combination of the cubic spline technique based on cubic Ferguson's curve with the PRM method, which eases the production of smoothed curved trajectories for wheeled mobile robots.

In this paper, a bidirectional PRM search algorithm is proposed in dealing with the problem of low search efficiency and excessive number of paths turning points in PRM algorithm. The search strategy of PRM is improved. Firstly, the initial node and the target node of the traditional PRM are searched simultaneously, which reduces the time waste of unnecessary node connection in the search process and effectively improves the search efficiency and safety performance of the algorithm. Moreover, the cubic spline curve smoothing processing is proposed to ensure that the mobile robot can work effectively and realize the path planning task more smoothly and safely.

This paper is structured as follows. Section II explains the basic principle of PRM. In section III, a bidirectional

PRM search algorithm is generated. Section IV introduces the basic principle of the cubic spline interpolation and presents bidirectional PRM algorithm combined with cubic spline interpolation. The simulation and experimental results are demonstrated in Section V. Section VI concludes this paper.

2. Basic Principle of PRM

The traditional PRM is an algorithm based on graph search, which constructs an undirected path network diagram by screening and selecting random sampling points, and then selects the most suitable path. Its basic idea is to use a random network diagram to represent the trajectory planning of mobile robot, and use a probability map to show the undirected cycle property of the diagram. The undirected path network diagram is represented by $G = (X, F)$, where G denotes the optimal path, X is set of nodes, and F denotes a path set between nodes. The path set represents the local feasible path. Then using heuristic search algorithm to find the optimal path, such as Breadth first algorithm, A^* etc.

The PRM can be completed in two phases, namely the offline learning phase and the online query phase. Offline learning phase mainly establishes an undirected path network diagram. Online query phase mainly selects an appropriate path in the undirected path network diagram according to the start node and the target node.

2.1. Offline Learning Phase. Offline learning phase primarily constructs an undirected path network diagram $G = (X, F)$. Firstly, Sampling in the map and there are obviously two possible outcomes. One is that the sampling point is on the obstacles, the point is discarded. The other is that the sampling point is in the free area, the point is constructed into the node set X . After that, the local planner is used to find the next-door node for connection in X , and the path set F is obtained [32, 33]. The specific process is shown in Figure 1.

As can be seen from the above flow chart, in the offline learning phase of PRM, first of all, the initial work is carried out to create the point set X , the path set F , and the uniform random sampling point n_i . Secondly, the validity of the sampling point n_i is judged. If the sampling point n_i is on the obstacle or not in the built map, the sampling point is discarded, otherwise it is the available point, which needs to be added to the X point set to facilitate the connection of the points in the point set. Then, it is judged whether the path of the two-point connection encounters an obstacle. If so, the path is discarded, otherwise the available path is stored in the path set. An undirected network graph $G(X, F)$ is created in this way.

For the undirected path network diagram building, the continuous workspace is transformed into a discrete workspace. Then, executing path planning, so that the complexity of the algorithm only depends on the complexity of the search path, which reduces algorithm complexity. The optimal path is taken by the random sampling points of PRM and the collision detection of these points. When constructing the network map, it starts from the nearest sampling

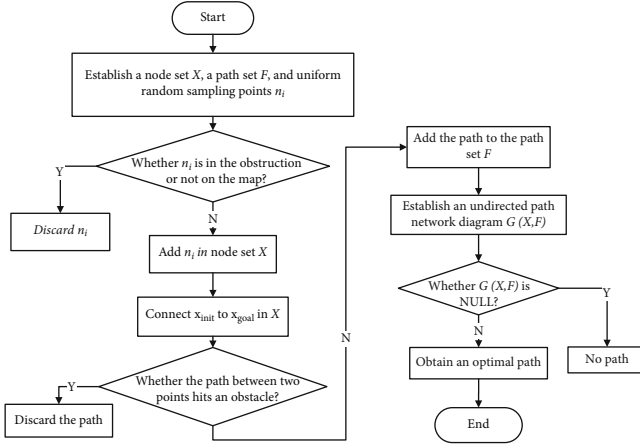


FIGURE 1: Flow diagram of PRM offline learning phase.

point of the start node and spreads to the surrounding sampling points. If the two sampling points can be connected without touching obstacles, then the path is added to the network path set, otherwise, the path is discarded, as shown in Figure 2. Among them, the upper left corner is the start node and the lower right corner is the target node. Figure 2(a) is a sampling diagram. Figure 2(b) is a network path set and the line is the path.

2.2. Online Query Phase. The main objective of the online query phase is to select an appropriate path between the start node and the target node based on the previously generated undirected path network diagram in the offline learning phase. For this purpose, the well-known A* algorithm is used. Specific for the following three steps.

- (1) Connect two points in the map nearest to the start node and the target node, respectively
- (2) Search the path by using A* algorithm
- (3) Smoothen the route and get the final path

The results are shown in Figure 3.

It can be known from the principle of PRM that the algorithm has good real-time performance and small amount of calculation, but there is still shortcoming for its efficiency. Therefore, this paper proposes an optimization algorithm to improve the efficiency of PRM.

3. PRM Optimization

In the offline learning phase, the traditional PRM makes the sampling points evenly distributed in the map and then constructs an undirected path network diagram by connecting each sampling point. Finally, the A* algorithm is used to query the optimal path. In the whole process of PRM, the offline learning phase takes much less time than the online query phase. Therefore, for the efficiency of online query phase, this paper proposes a bidirectional PRM search algorithm.

3.1. The Bidirectional PRM. The basic idea of the bidirectional PRM is that the search strategy is divided into positive and negative directions after obtaining sampling diagram. The forward search starts from the start node and the reverse search starts from the target node. The two paths will meet at the intermediate node, respectively, and their integration is a final path. This method not only improves the path security, but also further accelerates the search efficiency, which greatly decreases the time waste of connection between unnecessary paths. However, there are two completely unrelated paths in the real path planning of mobile robot. They are emitted from the start node and the target node, and end at the target node and the start node, respectively. These two paths may not meet in this process.

Therefore, in order to solve the above problems, a bidirectional PRM proposed in this paper will use the positive and negative alternate search method. The specific process is as follows: The search starts from the start node. When the forward PRM searches from the start node to the next node, the reverse PRM starts the reverse search from the target node. When the reverse search catches a sampling node, the forward search starts again, and then alternately. In this way, the overlapping nodes in the middle are found, and then the optimal path is sought.

The improvement of PRM significantly accelerates the path search efficiency. In Figure 4, there are black obstacles, a start node (s) and a target node (g) after rasterizing the map. In order to prevent confusion, the positive search uses the solid line display, and the negative search uses the dotted line display. The specific search process is shown in Figure 4.

In Figure 4, firstly, the bidirectional PRM searches the sampling points in the positive direction, and then starts to search the next sampling point. Four sampling points, a_1 , a_2 , a_3 , and a_4 will be searched at the same time. If these four paths $\langle s, a_1 \rangle$, $\langle s, a_2 \rangle$, $\langle s, a_3 \rangle$, and $\langle s, a_4 \rangle$ are all effective paths that conform to the search path rules, so these four paths can be stored in the path set F , as shown in Figure 4(a). Then, the feasible sampling points are b_1 , b_2 , b_3 , and b_4 in the reverse direction. Similarly, if the four paths $\langle g, b_1 \rangle$, $\langle g, b_2 \rangle$, $\langle g, b_3 \rangle$, and $\langle g, b_4 \rangle$ meet the requirements of the search path rules, so these four paths are also added to the path set F , as shown in Figure 4(b). Then continue to switch forward search and find nodes c_1 , c_2 , c_3 etc. Determines whether $\langle s, c_1 \rangle$, $\langle s, c_2 \rangle$, $\langle s, c_3 \rangle$ etc. meet the path set requirements. Corresponding paths are stored in the path set F , as shown in Figures 4(c) and 4(d). In this way, if the overlapping node d in the middle is found and the forward PRM finds out overlapping positions as well as the reverse PRM, which means that the bidirectional PRM finds a complete path. As shown in Figures 4(e) and 4(f).

4. Path Generation Using Cubic Spline Curve

The bidirectional PRM meaningfully increases the path search efficiency. However, there are still too many inflection points, which is obviously a shortcoming in the algorithm. Therefore, this paper adopts the cubic spline interpolation method to make the path smoother [34, 35]. The cubic

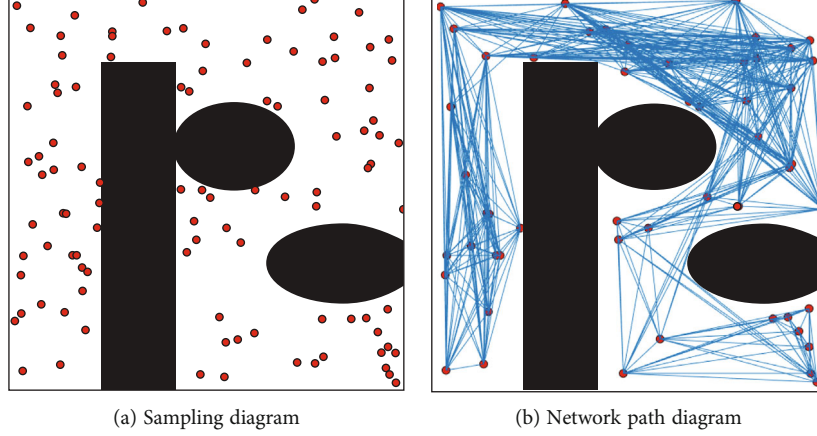


FIGURE 2: Offline learning phase.

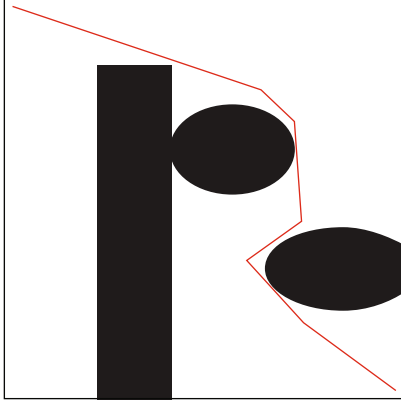


FIGURE 3: Online query phase.

spline interpolation forms a smooth curve through a series of cubic multitype piecewise interpolation. Fitting the path lines of the mobile robot with the cubic spline interpolation method will make the line segments of the path smoother, meanwhile, ensure the high efficiency of the mobile robot and avoid the damage to its hardware caused by the sudden stop or sudden turn when moving.

4.1. Basic Principles. Definition: set $[a, b]$ had equal interpolation nodes, $a = x_0 < x_1 < \dots < x_n = b$, corresponding for the cubic spline function $f(x)$. If function values satisfy the following two conditions, $f(x)$ is called cubic spline interpolation function.

- (1) Requirement of $f(x)$ only in each subinterval $[x_{i-1}, \dots, x_i]$ ($i = 1, 2, \dots, n$) and determine a cubic polynomial, set to

$$f_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3. \quad (1)$$

Among them, a_i , b_i , c_i , and d_i are undetermined coefficients.

- (2) $f(x), f'(x), f''(x)$ is continuous on $[a, b]$ and to satisfy it:

$$f(x_0) = y_0, \dots, f(x_{n+1}) = y_{n+1}, \quad (2)$$

$$f_-(x_i) = f_+(x_i) = y_i, i = 1, 2, 3, \dots, n, \quad (3)$$

$$f'_-(x_i) = f'_+(x_i) = y_i, i = 1, 2, 3, \dots, n, \quad (4)$$

$$f''_-(x_i) = f''_+(x_i) = y_i, i = 1, 2, 3, \dots, n. \quad (5)$$

From the above definition, the cubic polynomial $f(x)$ has $4n + 4$ undetermined coefficients because there are $n + 1$ cells on $[a, b]$, with 4 undetermined coefficients above each interval. Therefore, $4n + 4$ interpolation conditions must be needed in order to determine the cubic polynomial $f(x)$ according to Equation (2), it can be seen that there are $n + 2$ interpolation conditions. According to Equation (3), $f(x)$ is continuous at the interpolation node and has n conditions. Equations (4) and (5) represent that the first and second order derivatives of $f(x)$ are continuous at the interpolation node, with $2n$ interpolation conditions. Therefore, $4n + 2$ interpolation conditions can be obtained from the above definition. However, two boundary conditions are required to deal with the cubic spline interpolation function. The commonly used boundary conditions are as follows:

- (1) Natural boundary conditions. The second order derivative at the two endpoints is zero, namely:

$$f''(x_0) = f''(x_n) = 0. \quad (6)$$

- (2) Clamped boundary conditions. The first order derivative at the first and last node is specified, namely:

$$f'_0(x_0) = A, f'_{n-1}(x_n) = B. \quad (7)$$

- (3) Non-twisting boundary conditions. The third order derivatives of the splines at the first node are the same as at the second node. The third order derivatives at the last

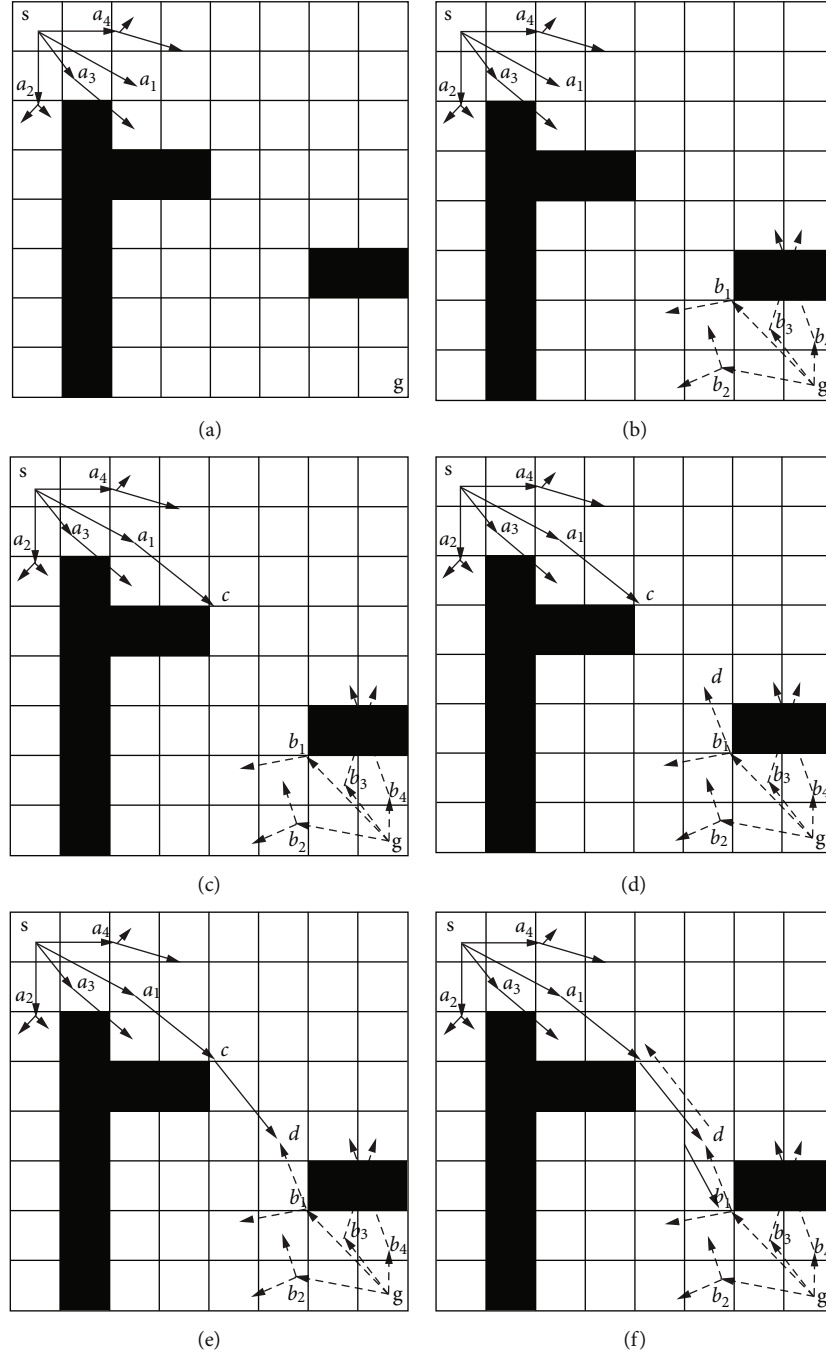


FIGURE 4: Path optimization process of bidirectional PRM.

node are also equal to it at the penultimate node, namely:

$$f_0'''(x_0) = f_1'''(x_1), f_{n-2}'''(x_{n-2}) = f_{n-1}'''(x_{n-1}). \quad (8)$$

4.2. PRM Optimization Algorithm Combined with Cubic Spline Interpolation. On the optimal path searched by the bidirectional PRM algorithm, interpolation algorithm is carried out according to the position of random path nodes. Spline interpolation is distributed interpolation in the

searched node set. The specific algorithm process is as follows:

- (1) Store key nodes. The first step in our algorithm is to build a txt file, which is used to store each inflection point in the PRM path, namely, the key node. Store these key nodes in the form of a file to facilitate later process of spline interpolation algorithm
- (2) Presuppose sampling point. The linspace function is used to generate the linear spacing vector, and the N-

point row vector between x_1 and x_2 is generated. N represents the number of samplings point between two sampling points. In this paper, the sampling point is set to 50 at the beginning. Then, the sampling points are set to 100 and 150, respectively. The optimal combination of PRM algorithm and bidirectional PRM algorithm will be confirmed according to different data information obtained and simulation results

- (3) Ensure $f(x)$ Spline interpolation obtains a cubic polynomial $y = f(x)$ by analyzing the function equation. The cubic polynomial will measure discrete data breakpoints (x, y) in the manner of approximations. According to the theorem, it can be known that only one line can be determined between two points. However, it is not limited to one for the cubic polynomial. Therefore, in this paper, apply non-twisted boundary conditions to obtain the exact expression of the cubic polynomial. The cubic spline interpolation is used to calculate the value of the unitary function $y = f(x)$ determined by vector x and y at the interpolation point xx . Confirm the value of yy after finding xx . Match each column of y matrix with x and get interpolation point function value yy matrix
- (4) Set the return value. Return the coefficient matrix of piecewise spline polynomial determined by vector x and y for the calculation of other commands, and then draw the function curve after cubic spline interpolation

From the above path optimization process, it can be seen that the number of turning points of the smoothed path is less, and the path is smoother, which can reduce the turning times of the mobile robot in navigation, make the mobile robot move more flexible in the working environment, reduce the loss of running time and energy consumption of the mobile robot, and improve work efficiency.

In this paper, the flow diagram is used to show the whole optimization process according to the path optimization method of bidirectional PRM described above and the interpolation method of integrating cubic spline, as shown in Figure 5.

As can be seen from the above flow chart, in the path optimization process of cubic spline bidirectional PRM algorithm, firstly, initial work is carried out to create point set X and path set F . Then, the bidirectional PRM search path is used, that is, the positive direction and the negative direction are alternately searched. The path points are connected in turn, and the path connection points are added to the F path set. When the middle coincident nodes are searched in both positive and negative directions, the optimal path has been found, otherwise continue the cycle. Finally, according to the path set obtained by bidirectional PRM, the four steps of the above interpolation algorithm are used to smoothen the path and obtain the final path.

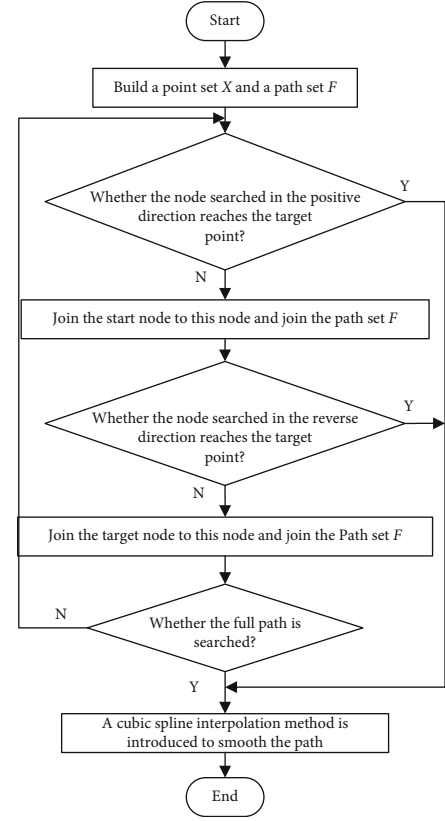


FIGURE 5: Flow diagram of path optimization of cubic spline bidirectional PRM.

5. Simulation and Experimental Verification

5.1. Simulation Verification. In this paper, adopt a computer with a processor of Rui Long R4800H, a display card of RTX2060, and a memory size of 16GB to carry out simulation experiments in order to verify the feasibility of the bidirectional PRM and the effectiveness of the cubic spline smoothing curve. The operating system is Windows 10, and the simulation platform is Matlab2020. The experimental results of PRM algorithm, bidirectional PRM, and bidirectional PRM with cubic spline interpolation are simulated, respectively. In the simulation process, use different grid maps to carry out simulation experiments to ensure that different positions of obstacles can also bring about the comparative effect of experimental data, so as to make the experimental results more convincing.

In this paper, three different map environments are selected for simulation experiments, as shown in Figures 6–8, where the map size is 500×500 . Coordinate (10, 10) is the start node of the mobile robot, and coordinate (490, 490) is the target node. The black part is the obstacle part, and the white part is the free action area of the mobile robot. The broken line is the path diagram without the fusion of cubic spline interpolation. The curve is the path diagram with the introduction of cubic spline interpolation. The sampling point $N = 100$.

The above simulation results show that the search time and path length of three algorithms are different. Therefore,

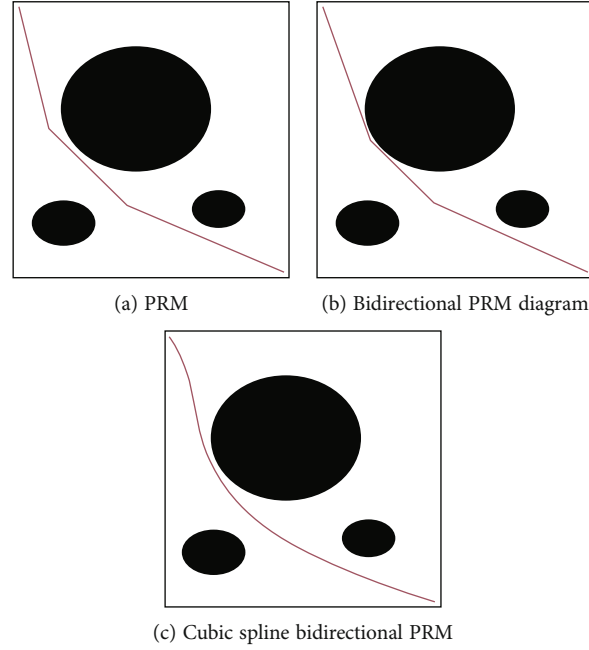


FIGURE 6: Comparison of experimental results of the three algorithms under Map 1.

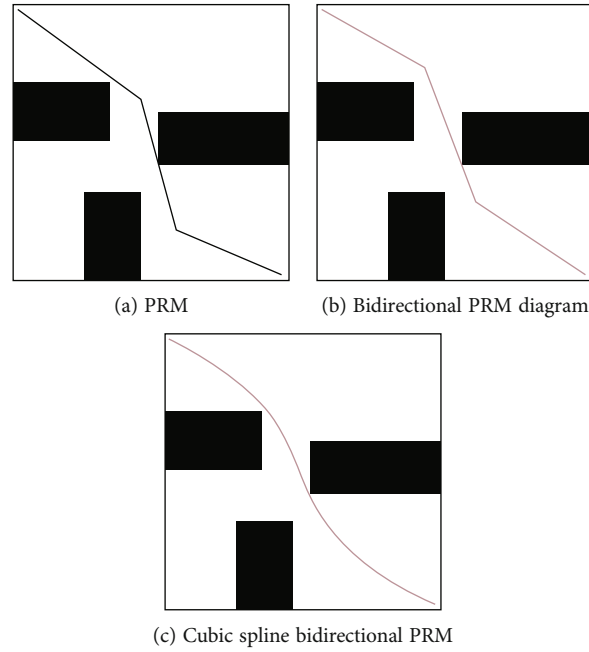


FIGURE 7: Comparison of experimental results of the three algorithms under Map 2.

based on a large number of simulation experiments, this paper makes data statistics on the time and path length obtained by the three algorithms, and takes the average as a reference. The specific data is shown in Table 1.

The search time is set as the running time of the algorithm, which can be determined by time function in MATLAB. Set the path length unit as the map size unit step size. In Table 1, the bidirectional PRM is optimized compared with the traditional PRM in time of the average search and in length of the shortest path. The data show that the

average search time of bidirectional PRM is shortened by 24%, 10%, and 31% compared with traditional PRM, and the shortest path optimization is improved by 2%, 2%, and 4%. That exhibits the improvement of the bidirectional PRM is not obvious in the path length, but the search efficiency has been greatly improved. The time optimization results of 31% and path optimization results of 4% in map 3 show that the bidirectional PRM has more obvious effect with the increase of map complexity. Similarly, the cubic spline bidirectional PRM algorithm has improved the

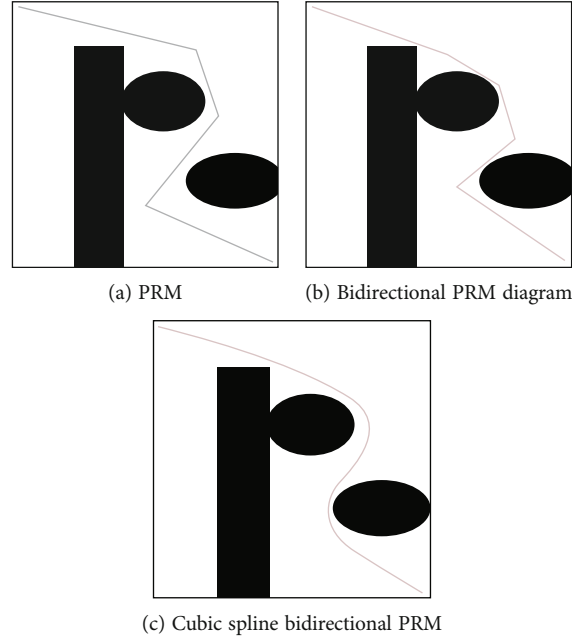


FIGURE 8: Comparison of experimental results of the three algorithms under Map 3.

TABLE 1: Comparison of experimental results of different maps.

Map model	Average search time/s			Average optimal path length		
	PRM	Bidirectional PRM	Cubic spline bidirectional PRM	PRM	Bidirectional PRM	Cubic spline bidirectional PRM
Map 1	1.85	1.41	1.60	742.19	726.56	734.38
Map 2	1.72	1.54	1.56	732.18	718.75	710.94
Map 3	1.73	1.18	1.24	984.38	945.31	867.18

TABLE 2: Comparison of experimental results of different N values.

The value of N	Path search time/s	Path length
50	0.46	823.32
100	1.33	808.90
150	3.07	798.08

average search time and the shortest path length compared with the traditional PRM algorithm. From map 1 to map 3, the average search time is reduced by 14%, 9%, and 28%, and the shortest path is optimized by 1%, 4%, and 12%. Although the bidirectional PRM with cubic spline inevitably loses a certain search time, it has obvious improvement compared with the traditional PRM algorithm, as shown in Figures 6(c), 7(c), and 8(c). Smooth optimization makes the broken route smoother, so that the mobile robot walking will be safer and more stable.

The value of sampling point N is crucial to the search efficiency of the algorithm. At present, the setting of this parameter is basically determined by experimental data and empirical values. In this paper, the values of N are set to 50, 100, and 150, respectively, to obtain the experimental data of the improved algorithm path finding time and path length, so as to determine the optimal value of N . The specific experimental data are shown in Table 2.

In Table 2, the path length decreases with the increase of N , but the corresponding search time increases due to the increase of calculation amount in the search process. After selecting the appropriate N , it can be clearly seen that the original broken line path will be smooth, and the smooth curve is helpful for the path walking of the mobile robot. In this paper, a large number of experimental results show that when $N = 100$, the bidirectional PRM has the highest profit ratio when searching the optimal path.

In summary, the bidirectional PRM has faster convergence rate, higher path planning efficiency than traditional PRM. More, the obtained path quality is the best. After introducing the cubic spline algorithm, the path is smoothed, so that the actual walking of the mobile robot is more stable and safer.

5.2. Experimental Validation. In order to verify the effectiveness of the improved algorithm and smoothing process, the experiment applies the bidirectional PRM and the smoothing processing algorithm after inserting the cubic spline to the actual robot navigation based on ROS. The mobile robot used in the experiment is a four-wheel walking wheeled robot with two rear wheels and two front wheels. The rear wheel is the driving wheel and the front wheel is the passive wheel. Figure 9 shows:

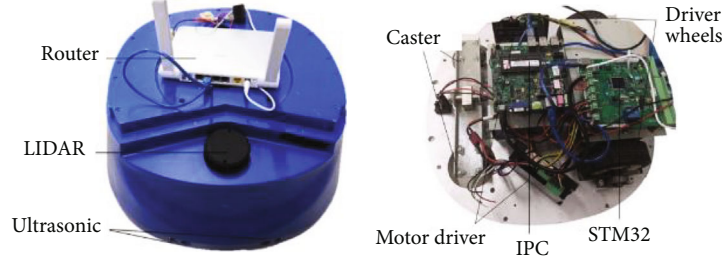


FIGURE 9: External and interior figures of real mobile robot chassis.

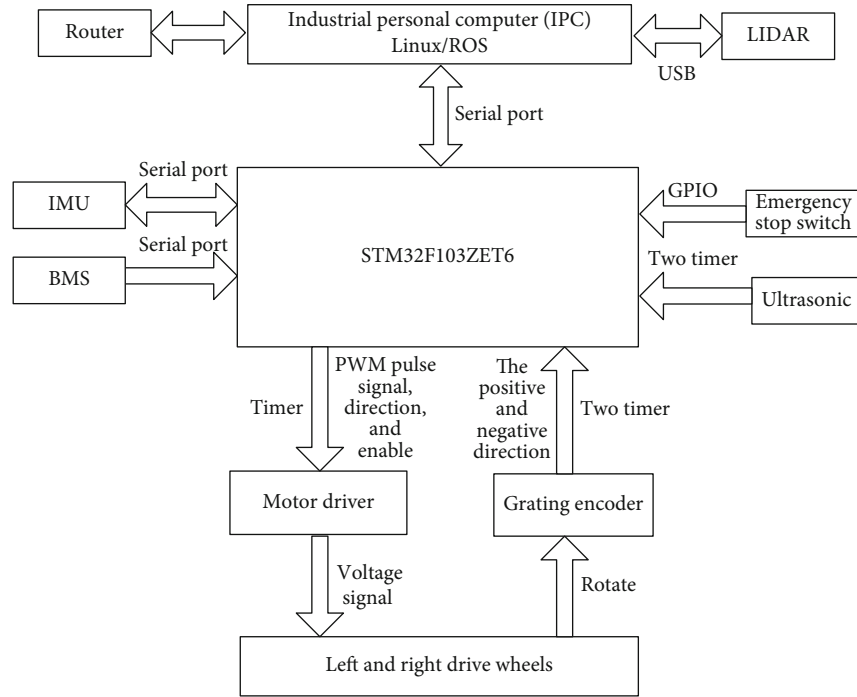


FIGURE 10: Hardware architecture of mobile robot.

In this paper, a complete mobile robot control platform is built, which is mainly composed of laser radar, core control board STM32F103ZET6 (hereinafter referred to as STM32), grating encoder, motor driver, brushless DC motor, battery module, front and rear wheels, and multiple interfaces. ROS sends out the encoder query instruction. The STM32 transmits the encoding information to ROS by serial communication. According to the information, ROS sends the topic of/cmd_vel to help feature pack subscribe topic information, then sends the speed instruction to STM32. After receiving the instruction, STM32 sends it to the DC brushless motor driver in the form of PWM pulse wave. The rotation of the motor is controlled by the voltage of the driver. After the motor rotates, the grating encoder will work. Pulse signal obtained by the information of the rotation is transmitted to STM32, which is integrated into the encoding information for ROS query. This loop realizes autonomous navigation of mobile robot. The overall structure is shown in Figure 10.

In Figure 10, Industrial Personal Computer (IPC) is the decision-making unit. As the upper computer, IPC carries all instructions send and receive by ROS. STM32 is connected to IPC through the USB line. Upper computer sends control and query instructions to STM32 and STM32 can also send encoder values and corresponding messages to upper computer.

In order to ensure the reliability of experimental data, this paper will take the laboratory map as the working environment. The traditional PRM, the bidirectional PRM, and the cubic spline bidirectional PRM are, respectively, carried out the physical experiment. The results are shown in Figures 11 and 12.

The actual path walking data of mobile robots are shown in Tables 3 and 4, where $N = 100$.

Set the path length unit in the table to the map size unit step. The search time is the algorithm running time and can be determined using a time function in the ROS script. The walking time is the time taken by the robot from the starting point to the end point. Tables 3 and 4 are the experimental

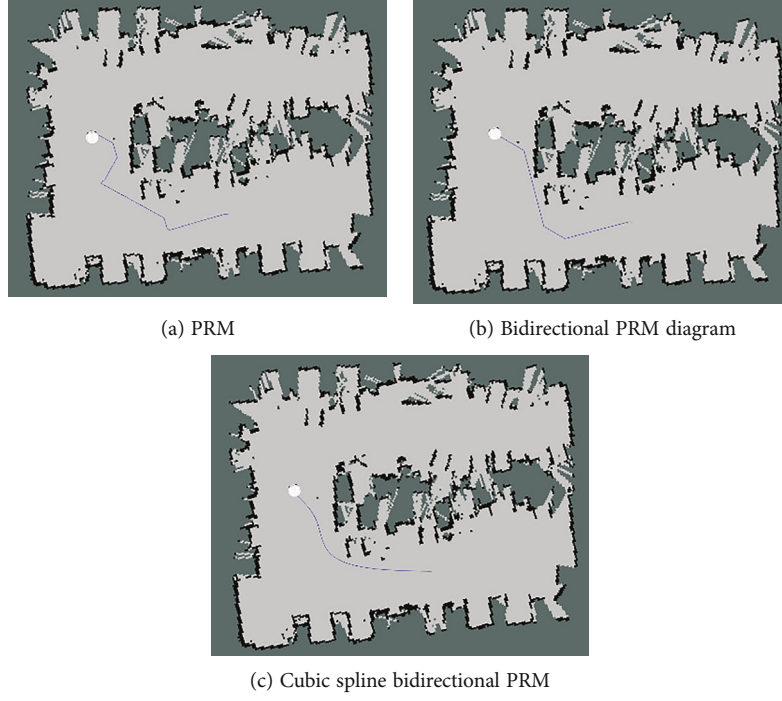


FIGURE 11: Comparison of three algorithms under target node 1.

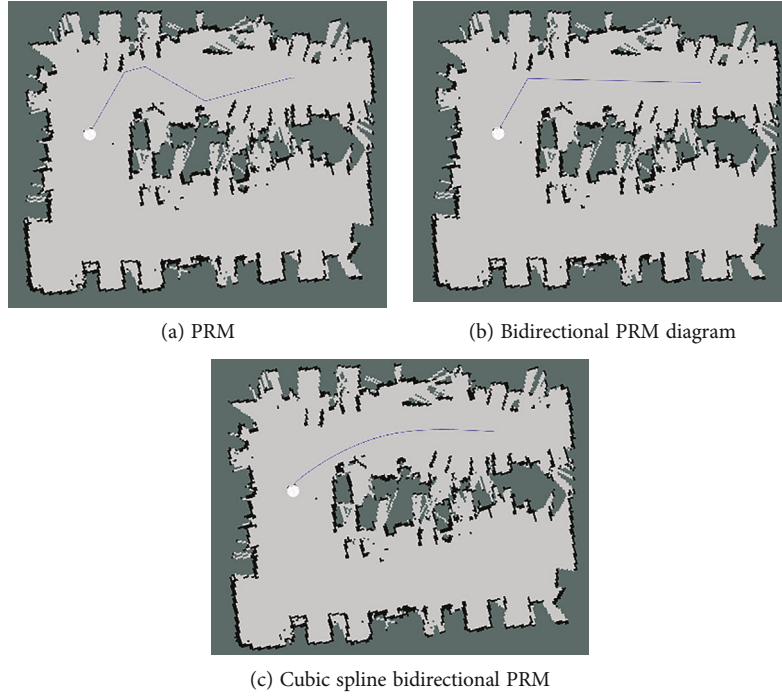


FIGURE 12: Comparison of three algorithms under target node 2.

data comparison of traditional PRM, the bidirectional PRM, and the cubic spline bidirectional PRM in the case of same environment and same conditions. In Table 3, in the case of target node 1, compared with traditional PRM, the path length of the bidirectional PRM is shortened by 8.5%, and the pathfinding time and walking time are reduced by

27.9% and 9.7%, respectively. The path length of the cubic spline bidirectional PRM is shortened by 19.0%, and the path searching time and walking time are reduced by 12.5% and 15.3%, respectively. Table 4 shows that in the case of target node 2, compared with PRM algorithm, the path length of the bidirectional PRM is reduced by 6.3%, and

TABLE 3: Experimental data comparison of three algorithms in target node 1.

	PRM	Bidirectional PRM	Cubic spline bidirectional PRM
Path length/(m)	5.91	5.41	4.79
Path-finding time/(s)	0.391	0.282	0.342
Walking time/(s)	15.21	13.74	12.89

TABLE 4: Experimental data comparison of three algorithms in target node 2.

	PRM	Bidirectional PRM	Cubic spline bidirectional PRM
Path length/(m)	6.70	6.28	5.86
Path-finding time/(s)	0.519	0.365	0.476
Walking time/(s)	34.13	32.51	29.47

the pathfinding time and walking time are reduced by 29.7% and 4.7%, respectively. The path length of cubic spline bidirectional PRM is shortened by 12.5% and the efficiency of path finding time and walking time are improved by 8.3% and 13.7%, respectively.

According to the above analysis, compared with traditional PRM, the path length searched by cubic spline PRM is better, and the path search time and robot walking efficiency are higher. Therefore, the optimization algorithm in this paper is an effective and feasible path planning algorithm.

6. Conclusion

In order to solve the shortcomings of the traditional PRM with slow convergence speed and large number of turning points in path optimization, a cubic spline bidirectional PRM algorithm is proposed in this paper. Firstly, the improved bidirectional PRM uses the alternate search method of positive and negative directions to save time and improve the convergence speed of the algorithm. Secondly, the cubic spline interpolation method is integrated to improve the smoothness of the path and ensure that the mobile robot can work more safely. Finally, through the simulation experiment and the physical experiment, the results show that the cubic spline bidirectional PRM not only has a better execution time, but also has better path length than other methods. Moreover, the improved PRM can effectively solve the path planning task of mobile robot.

Data Availability

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61472282; in part by the University Natural Science Research Project of Anhui Province under Grant KJ2019A0065; in part by Major Special Project of Science and Technology in Anhui Province under Grant 202003a05020028; and in part by Anhui Province Key Laboratory of Special Heavy Load Robot under Grant ZJQR004-2020.

References

- [1] R. S. Pol and M. Murugan, "A review on indoor human aware autonomous mobile robot navigation through a dynamic environment survey of different path planning algorithm and methods," in *2015 International Conference on Industrial Instrumentation and Control (ICIC)*, pp. 1339–1344, Pune, India, May 2015.
- [2] B. Siciliano and O. Khatib, "Robotics and the handbook," in *Springer Handbook of Robotics*, Springer Handbooks, B. Siciliano and O. Khatib, Eds., Springer, Cham, 2016.
- [3] E. W. Dijkstra, "A note on two problems in connection with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [4] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [5] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous Robot Vehicles*, I. J. Cox and G. T. Wilfong, Eds., pp. 396–404, Springer, New York, NY, 1986.
- [6] E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992.
- [7] L. Jiang, S. Liu, Y. Cui, and H. Jiang, "Path planning for robotic manipulator in complex multi-obstacle environment based on improved_RRT," *IEEE/ASME Transactions on Mechatronics*, 2022.
- [8] L. Wenzheng, L. Junjun, and Y. Shunli, "An improved Dijkstra's algorithm for shortest path planning on 2D grid maps," in *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 438–441, Beijing, China, July 2019.

- [9] F. Duchoň, A. Babinec, M. Kajan et al., "Path planning with modified a star algorithm for a mobile robot," *Procedia Engineering*, vol. 96, pp. 59–69, 2014.
- [10] P. Sudhakara, V. Ganapathy, B. Priyadharshini, and K. Sundaran, "Obstacle avoidance and navigation planning of a wheeled mobile robot using amended artificial potential field method," *International Conference on Robotics and Smart Manufacturing*, vol. 133, pp. 998–1004, 2018.
- [11] M. Achite, P. Tsangarators, I. Ilia, and A. K. Toubal, "Applying support vector machines optimized by genetic algorithm for estimating the spatial distribution of mean annual precipitation," *Arabian Journal of Geosciences. Springer*, vol. 14, no. 8, pp. 1–16, 2021.
- [12] M. D. Phung and Q. P. Ha, "Safety-enhanced UAV path planning with spherical vector-based particle swarm optimization," *Applied Soft Computing*, vol. 107, article 107376, 2021.
- [13] W. Sun, W. Zang, C. Liu, T. Guo, Y. Nie, and D. Song, "Motion pattern optimization and energy analysis for underwater glider based on the multi-objective artificial bee colony method," *Journal of Marine Science and Engineering*, vol. 9, no. 3, pp. 327–327, 2021.
- [14] P. Svestka and M. H. Overmars, "Coordinated motion planning for multiple car-like robots using probabilistic roadmaps," in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, Nagoya, Japan, 1995.
- [15] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation (TRA)*, vol. 12, no. 4, pp. 566–580, 1996.
- [16] C. Kai, C. Qian, G. Song, Y. Chen, and C. Chen, "Improved PRM for path planning in narrow passages," in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 45–50, Tianjin, China, August 2019.
- [17] J. Velagić, D. Delimustafić, and D. Osmanković, "Mobile robot navigation system based on probabilistic road map (PRM) with Halton sampling of configuration space," in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pp. 1227–1232, Istanbul, Turkey, June 2014.
- [18] D. Aarno, D. Kragic, and H. I. Christensen, "Artificial potential biased probabilistic roadmap method," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pp. 461–466, New Orleans, LA, USA, April 2004.
- [19] R. Bohlin and L. E. Kavraki, "Path planning using lazy PRM," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, pp. 521–528, San Francisco, CA, USA, April 2000.
- [20] M. Morales, S. Rodriguez, and N. M. Amato, "Improving the connectivity of PRM roadmaps," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, pp. 4427–4432, Taipei, Taiwan, 2003.
- [21] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.
- [22] M. Saroya, G. Best, and G. A. Hollinger, "Roadmap learning for probabilistic occupancy maps with topology-informed growing neural gas," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4805–4812, 2021.
- [23] Y. Li, D. Li, C. Maple, Y. Yue, and J. Oyekan, "K-order surrounding roadmaps path planner for robot path planning," *Journal of Intelligent and Robotic Systems*, vol. 75, no. 3-4, pp. 493–516, 2014.
- [24] A. A. Ravankar, A. Ravankar, T. Emaru, and Y. Kobayashi, "HPPRM: hybrid potential based probabilistic roadmap algorithm for improved dynamic path planning of mobile robots," *IEEE Access*, vol. 8, pp. 221743–221766, 2020.
- [25] B. Palma, C. Lima, W. Caarls, and D. Vettorazzi, "Which probabilistic roadmap method should be used by a robot in an actual environment? An analysis of the main methods through simulations," *IEEE Latin America Transactions*, vol. 14, no. 4, pp. 2020–2025, 2016.
- [26] R. Sandström, D. Uwacu, J. Denny, and N. M. Amato, "Topology-guided roadmap construction with dynamic region sampling," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6161–6168, 2020.
- [27] Q. Chai, Y. Wang, Y. He, C. Xu, and Z. Hong, "Improved PRM path planning in narrow passages based on PSO," in *2022 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 41–46, Guilin, Guangxi, China, 2022.
- [28] K. Yang, S. Moon, S. Yoo et al., "Spline-based RRT path planner for non-holonomic robots," *Journal of Intelligent and Robot Systems*, vol. 73, no. 1-4, pp. 763–782, 2014.
- [29] P. Sudhakara, V. Ganapathy, and K. Sundaran, "Optimal trajectory planning based on bidirectional spline-RRT for wheeled mobile robot," in *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, pp. 65–68, Chennai, India, May 2017.
- [30] A. Krontiris, S. Louis, and K. E. Bekris, "Multi-level formation roadmaps for collision-free dynamic shape changes with non-holonomic teams," in *2012 IEEE International Conference on Robotics and Automation*, pp. 1570–1575, Saint Paul, MN, USA, May 2012.
- [31] P. Sudhakara, V. Ganapathy, and K. Sundaran, "Probabilistic roadmaps-spline based trajectory planning for wheeled mobile robot," in *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pp. 3579–3583, Chennai, India, August 2017.
- [32] L. E. Kavraki, M. N. Kolountzakis, and J.-C. Latombe, "Analysis of probabilistic roadmaps for path planning," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 1, pp. 166–171, 1998.
- [33] D. Delling, P. Sanders, D. Schultes, and D. Wagner, "Engineering route planning algorithms," in *Algorithmics of Large and Complex Networks*, J. Lerner, D. Wagner, and K. A. Zweig, Eds., vol. 5515 of Lecture Notes in Computer Science, pp. 117–139, Springer, Berlin, Heidelberg, 2009.
- [34] L. Xingchen, G. Xifeng, Z. Wei, and L. Hao, "Smooth and collision-free trajectory generation in cluttered environments using cubic B-spline form," *Mechanism and Machine Theory*, vol. 169, article 104606, 2021.
- [35] B. Khaldi, F. Harrou, F. Cherif, and Y. Sun, "Toward emerging cubic-spline patterns with a mobile robotics swarm system," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 14, no. 2, pp. 565–577, 2021.