AM-RRT*: An Automatic Robot Motion Planning Algorithm Based on RRT

Peng Chi[1], Zhenmin Wang[1], Haipeng Liao[1], Ting Li[1], Jiyu Tian[1], Xiangmiao Wu[1], and Qin Zhang[2](✉)

[1] School of Mechanical and Automotive Engineering, South China University of Technology, WuShan Street, Guangzhou 510641, China
{mechipeng,melt}@mail.scut.edu.cn,
{wangzhm,tianjiyu,xmwu}@scut.edu.cn

[2] School of Computer Science and Engineering, South China University of Technology, Xiaoguwei Street, Guangzhou 511400, China
cszhangq@scut.edu.cn

Abstract. Motion planning is a very important part of robot technology, where the quality of planning directly affects the energy consumption and safety of robots. Focusing on the shortcomings of traditional RRT methods such as long, unsmooth paths, and uncoupling with robot control system, an automatic robot motion planning method was proposed based on Rapid Exploring Random Tree called AM-RRT* (automatic motion planning based on RRT*). First, the RRT algorithm was improved by increasing the attractive potential fields of the target points of the environment, making it more directional during the sampling process. Then, a path optimization method based on a dynamic model and cubic B-spline curve was designed to make the planned path coupling with the robot controller. Finally, an RRT speed planning algorithm was added to the planned path to avoid dynamic obstacles in real time. To verify the feasibility of AM-RRT*, a detailed comparison was made between AM-RRT* and the traditional RRT series algorithms. The results showed that AM-RRT* improved the shortcomings of RRT and made it more suitable for robot motion planning in a dynamic environment. The proposal of AM-RRT* can provide a new idea for robots to replace human labor in complex environments such as underwater, nuclear power, and mines.

Keywords: Motion planning · Sampling-based algorithms · RRT · Speed planning

Supported by Guangdong Provincial Science and Technology Plan Project (Grant number 2021B1515420006, Grant number 2021B1515120026); Guangdong Province Marine Economic Development Special Fund Project(Six Major Marine Industries) (GDNRC [2021]46); National Natural Science Foundation of China (Grant number U2141216, Grant number 51875212); Shenzhen Technology Research Project (JSGG20201201100401005, JSGG20201201100400001).

B. Luo et al. (Eds.): ICONIP 2023, LNCS 14447, pp. 91–103, 2024.
https://doi.org/10.1007/978-981-99-8079-6_8

1 Introduction

Motion planning are important components of mobile robots, which can be used to guide robots to work autonomously in complex environments and avoid obstacles [1]. How to plan a short, smooth, and controllable path is a major challenge. The methods of mobile robot motion planning and their improvements can be divided into classic methods and intelligent methods [1].

The classic methods include environment-modeling-based methods [2–6], graph-search-based methods, sampling-based methods, and curve-based methods. There are three most commonly used environmental modeling methods: grid map [2], road map [3–5], and artificial potential field [6]. The grid map can be divided into two parts: obstacle and free, which relies on the accuracy of the grid. The road map used a limited number of nodes to represent the environment and the nodes were connected to form edges. The more representative road map methods are Visibility Graph [3], Voronoi Diagram [4], and Probabilistic maps [5]. The artificial potential field method [6] was designed by a potential function, which included the attractive field of the target point and the repulsive field of the obstacles. The disadvantage is that it is easy to fall into the local minimum. The graph-search-based methods relied on known maps and obstacle information in the map to construct a feasible path from the start point to the end point [6–9]. However, these methods required a large amount of computation and were not suitable for path planning in dynamic scenes. The sampling-based path planning methods performed random sampling in the environmental space [5,10,11]. The advantage of these methods is that it does not need to model the environmental space. But the planning process is random and the optimal path is not obtained. The most representative random sampling methods include Probabilistic Roadmap Method (PRM) [5] and Rapidly Exploring Random Tree (RRT) [10]. The curve-based method referred to the method of constructing or inserting a new data set within a known data set [12,13]. The curve can fit the previous set of path points to a smoother trajectory [14,15]. Overall, the classic methods tend to favor static path planning and have the disadvantage of easily falling into local minimum, making them unsuitable for dynamic path planning.

The path planning problem is not only a search problem but a reasoning problem, so many intelligent methods can also be used in path planning problems. The most typical intelligent algorithm is the neural-network-based algorithm [16], which simulates the behavior of animal neural networks and performs distributed information processing. The genetic algorithm is a search optimal solution algorithm that simulates the evolution of Darwinian organisms [17], which can realize synchronous planning and tracking. The ant colony algorithm is derived from the exploration of ant colony foraging behavior, which is effective and robust [18]. It can find relatively good paths but is suitable for real-time planning. At present, intelligent planning algorithms are not yet mature and require high hardware devices, making them unsuitable for real-time path planning solutions.

Given the above problems of existing planning methods, a new motion planning method suitable for robot control was proposed in this paper called AM-RRT*. The main contributions are as follows:

- A new RRT path planning algorithm based on elliptical region and potential field constraints was proposed, which had the shottest path length compared to traditional RRT algorithms.
- A path optimization method based on robot dynamic model constraints and cubic B-spline curves was proposed to ensure that the planned path was smooth enough and tightly coupled with the robot controller.
- A speed planning method based on RRT was proposed, which endowed robots with the ability to autonomously control speed in dynamic environments to save energy consumption and time.

2 Related Work

The AM-RRT* proposed in this paper is based on the RRT algorithm. The RRT has been proposed for many years, and many researchers have proposed numerous improved algorithms for it.

The RRT was proposed to solve complex obstacle constraints and a high degree of freedom path planning problems [10]. The idea is to gradually build a tree, attempting to quickly and uniformly explore configuration or state space, providing benefits similar to probabilistic roadmap methods, but applicable to a wider range of problems [11]. The RRT* algorithm maintained the tree structure like RRT while maintaining the asymptotic optimality [19]. Its basic idea was the fusion between the sampling-based motion planning algorithm and the random geometric graph theory. Informed RRT* (IRRT*) [20] was proposed for centralized search by directly sampling the subset to address the issue of low efficiency and inconsistency with their single query properties. Q-RRT* [21] considered more vertices during the optimization process, thereby expanding the possible set of parent vertices, resulting in a path with lower cost than RRT*. IB-RRT* [22] utilized the bidirectional tree method and introduced intelligent sample insertion heuristic to quickly converge to the optimal path solution [23]. P-RRT* was proposed to combine the artificial potential field algorithm in RRT*, which allowed for a significant reduction in the number of iterations. PQ-RRT* [24] combined the advantages of P-RRT* and Q-RRT*, further improving operating speed while possessing superiority. These improvement methods are aimed at optimizing the performance of path. However, robots have the characteristic of multiple modules tightly coupled.

Shi et al. [25] improved and optimized the RRT algorithm by establishing a vehicle steering model to increase vehicle steering angle constraints and solve the problems of high randomness, slow convergence speed, and large deviation. However, the applied model is relatively simple and the optimization effect is not significant enough. Chen et al. [26] incorporated mobility reliability into mission planning and proposed a reliability-based path smoothing algorithm to solve the suboptimal problem. The method has high robustness, but low universality.

Therefore, it is necessary to design a RRT-based path planning method that is short in length, smooth, and coupled with the robot motion control, and to form an autonomous robot motion planning system with the speed planning method.

3 Methodology

The overview of AM-RRT* is shown in Fig. 1. The input of the system is from the environment perception of a robot, which includes three parts: map information, static obstacles information, and trajectory of dynamic obstacles. First, an improved RRT algorithm for robot path planning was utilized, which included the elliptical sampling area of IRRT* and the attractive potential field of Voronoi points. Then, the planned path was optimized to make it smoother and more suitable for robot motion control, including robot dynamic constraints and optimization of cubic spline curves. Finally, the improved RRT algorithm mentioned above was applied to the speed planning of robots, where the extension was used in the distance-time (s-t) image to obtain real-time speed.

3.1 Path Planning Based on RRT

According to the shortcomings of the globssal sampling of the RRT algorithm, the elliptical area was defined and restricted to the sampling area to improve its sampling efficiency [20]. An elliptical region C with the shortest path length is defined as:

$$C = \{x \in \chi \,|\|x - x_{init}\| + \|x - x_{goal}\| < length(\sigma_{\min})\} \tag{1}$$

where $\sigma_{\min}$ is the length of the shortest path. x_{init}, x_{goal} is the initial point and the goal point of the path. x is the current point. Since the average planned path of RRT is about 1.3–1.5 times the shortest path, two kinds of attractive potential fields were added to produce a total attractive potential field. The whole attractive potential field is defined as:

$$U_{art}(x) = U_{goal}(x) + U_{vor}(x) \tag{2}$$

where $U_{goal}(x)$ is the attractive potential field of the target point. $U_{vor}(x)$ is the attractive potential field of the Voronoi diagram of the environment.

The $U_{goal}(x)$ is defined as:

$$U_{goal}(x)=\frac{1}{2}k_{goal}(x - x_{goal})^2 \tag{3}$$

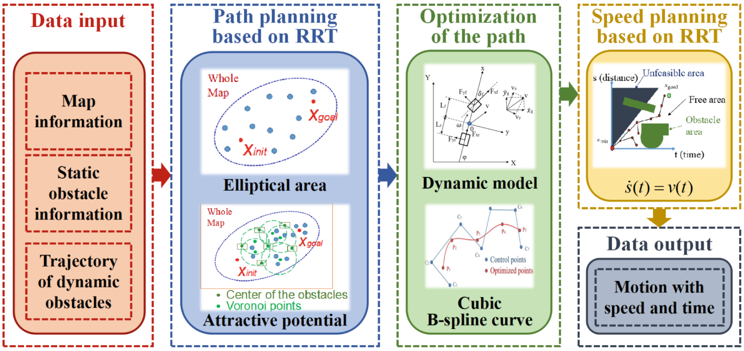


Fig. 1. Overview of the proposed AM-RRT* motion planning system.

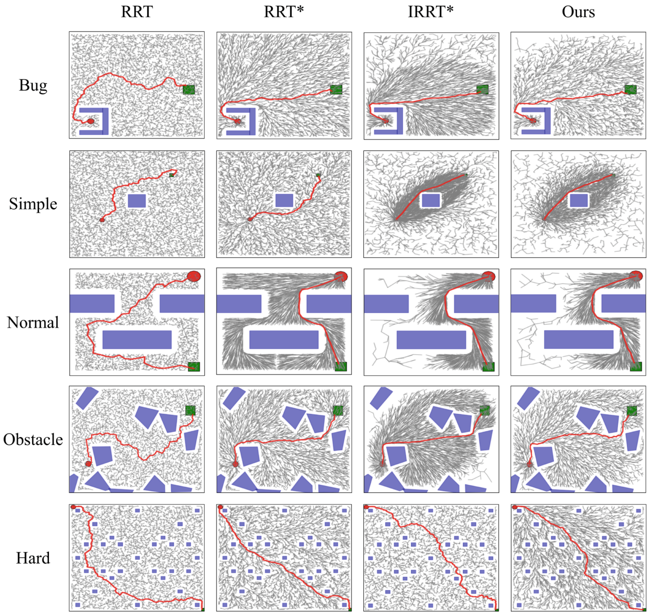


Fig. 2. Path planning results of RRT [10], RRT* [19], IRRT* [20], and our proposed method in five different scenarios.

where k_{goal} is a constant weighting factor. The $U_{vor}(x)$ is defined as:

$$U_{vor}(x)=\frac{1}{2}k_{vor}\sum\nolimits_{x_{vor}\in\chi_{vor}}(x-x_{vor})^2 \tag{4}$$

where k_{vor} is the constant weight coefficient. χ_{vor} is the set of points that make up the Voronoi points. Since Voronoi points are the circumscribed center of the three obstacles, the problem that when the obstacles are too close, random sampling can not find the result can be solved.

3.2 Optimization of the Path

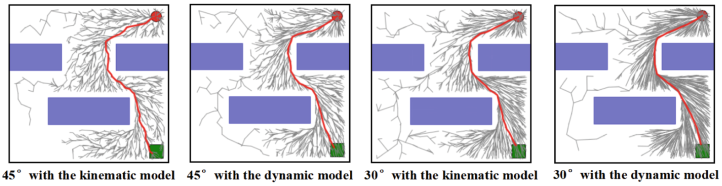
Constraints based on robot kinematics and dynamic model can be used to optimize the path, making the planned path more suitable for robot motion control [25,27]. A 5-DOF vehicle dynamic model was selected for comparative analysis in this paper [27]. The state of the vehicle is defined by:

$$x = (x_g, y_g, \varphi, v_y, \omega)^T \tag{5}$$

where x_g, y_g represent the robot's center of gravity. φ is the yaw angle of the robot. v_y is the speed of lateral and ω is the yaw rate of the robot.

According to the second Newton's law, the fundamental law of dynamics can be defined as:

$$\begin{cases} m(\dot{v}_x - v_y\omega) = -F_{xf}\cos\delta_f - F_{yf}\sin\delta_f - F_{xr} \\ m(\dot{v}_y + v_x\omega) = F_{yf}\cos\delta_f - F_{xf}\sin\delta_f + F_{yr} \\ I_z\dot{\omega} = L_f(F_{yf}\cos\delta_f - F_{xf}\sin\delta_f) - L_rF_{yr} \end{cases} \tag{6}$$



45° with the kinematic model 45° with the dynamic model 30° with the kinematic model 30° with the dynamic model

Fig. 3. Optimization results of RRT path planning based on dynamic models in Normal scene.

where m is the mass of the robot. v_x is the speed of longitude. $F_{xf}, F_{yf}, F_{xr}, F_{yr}$ are the longitudinal and lateral force of the front and rear wheels. δ_f is the front wheel angle. I_z is the moment of inertia around the z-axis. L_f, L_r are the distance from the center of gravity to the front and rear wheel.

The traditional RRT algorithms are not smooth and often require post-processing to make the path smoother. The cubic B-spline curve [15] was selected as the optimization method. A B-spline curve is defined by a series of control points. These points consist of all nodes in the suboptimal path searched by RRT. At the same time, each point of the generated path is only related to the four nearby control points. The planned path can be defined as the x, y coordinates Φ_x, Φ_y:

$$\begin{cases} \Phi_x = [\phi_{x,-1}\phi_{x,0}\cdots\phi_{x,m-1}] \\ \Phi_y = [\phi_{y,-1}\phi_{y,0}\cdots\phi_{y,m-1}] \end{cases} \tag{7}$$

where $\phi_{x,i}$ $\phi_{y,i}$ are the control points. Define the setting $s \in [0, 1]$ is the percentage of the path length that has been traveled, the final path can be expressed as:

$$\begin{cases} x(s) = \sum_{k=0}^{3} f_k(t) \cdot \phi_{x,k+l} \\ y(s) = \sum_{k=0}^{3} f_k(t) \cdot \phi_{y,k+l} \end{cases} \tag{8}$$

where $[\cdot]$ is the round down function. t and l are used to adjust the control points:

$$\begin{cases} t = s \cdot \sigma - [s \cdot \sigma] \\ l = [s \cdot \sigma] - 1 \end{cases} \tag{9}$$

where σ is the length of planned path. $\{f_k(t)\}_{k=[0,3]}$ is the basic function of the B-spline, defined as:

$$\{f_k(t)\}_{k=[0,3]} = \begin{cases} f_0(t) = \frac{1}{6}(1-t)^3 \\ f_1(t) = \frac{1}{6}(3t^3 - 6t^2 + 4) \\ f_2(t) = \frac{1}{6}(-3t^3 + 3t^2 + 3t + 1) \\ f_3(t) = \frac{1}{6}t^3 \end{cases} \tag{10}$$

To optimize the path length and smoothness under the premise of ensuring the safety of the path without collision, the definition of the optimization function consists of two parts: path length and potential field energy along the path. The potential field energy is composed of the obstacle's repulsive potential field:

$$E(\Phi) = L(\Phi) + \lambda \cdot P(\Phi) \tag{11}$$

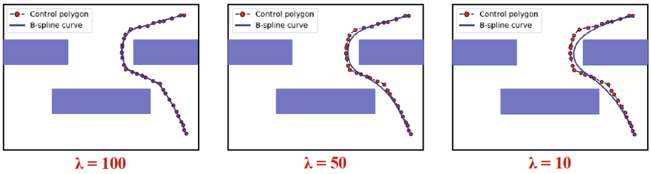


Fig. 4. Optimization results of RRT path planning based on cubic spline curve in Normal scene.

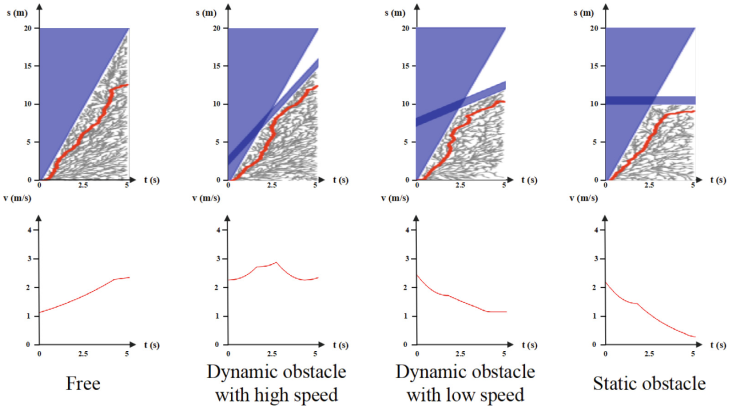
where $L(\Phi)$ is the path length and $P(\Phi)$ is the potential field energy. λ represents the weight between path length and potential. If the value of λ is larger, the final path will be farther away from the obstacles but the path length may be longer.

The cost of the path length is defined as:

$$L = \int_0^1 [\dot{x}^2(s) + \dot{y}^2(s)] ds \tag{12}$$

The cost of the potential field energy is defined as:

$$P = \int_0^1 P(x(s), y(s)) ds \tag{13}$$



Free

Dynamic obstacle with high speed

Dynamic obstacle with low speed

Static obstacle

Fig. 5. Speed planning results for four different scenarios, including free, dynamic obstacle with low speed, dynamic obstacle with high speed, and static obstacle.

3.3 Speed Planning Based on RRT

For speed planning, the RRT-based planning method is still used. Assuming that the robot's pitching motion is not considered, the motion generally includes three parameters: the coordinates of the robot on the path $\rho_x(s)$, $\rho_y(s)$ and the heading angle of the robot along the path $\rho_\theta(s)$:

$$\rho(s) = [\rho_x(s), \rho_y(s), \rho_\theta(s)] \tag{14}$$

The trajectory of the dynamic obstacles is defined as:

$$obs_i(t)_{i \in [1,M]} = [x_{obs,i}(t) y_{obs,i}(t)] \tag{15}$$

where t is time. M is the number of obstacles and $x_{obs,i}(t)$,$y_{obs,i}(t)$ are the coordinates of the obstacle i at time t. Given the reference path $\rho(s)$ and the initial state of the robot, the task of speed planning is to generate a longitudinal motion function $s(t)(t \in [0, T_{forward}])$, where the speed can be expressed as $v(t) = \dot{s}(t)$. The robot status can be described as $x = (s,t)$ and the s-t motion space χ can be described as the state space within the range $x(s \geq s_0, t \geq t_0)$. The s_0 and t_0 represent the initial path length and time of the robot. Assume that the set of states that will collide with obstacles in s-t space is the obstacle space χ_{obs}:

$$\chi_{obs} = \{x \in \chi | \, \|\rho(s) - obs_i(t)\| < d_{danger}\} \tag{16}$$

where d_{danger} is the threshold of the dangerous distance from the robot to the obstacles.

The state in the s-t space should satisfy the following constraints: $|\dot{s}| \leq v_{\max}$, $a_{\min} \leq \ddot{s} \leq a_{\max}$, $|\ddot{s}| \cdot \kappa \leq \omega_{\max}$, where $v_{\max}$ is the maximum speed of the robot. $[a_{\min}, a_{\max}]$ are the acceleration limit of the robot. κ and $\omega_{\max}$ represent the

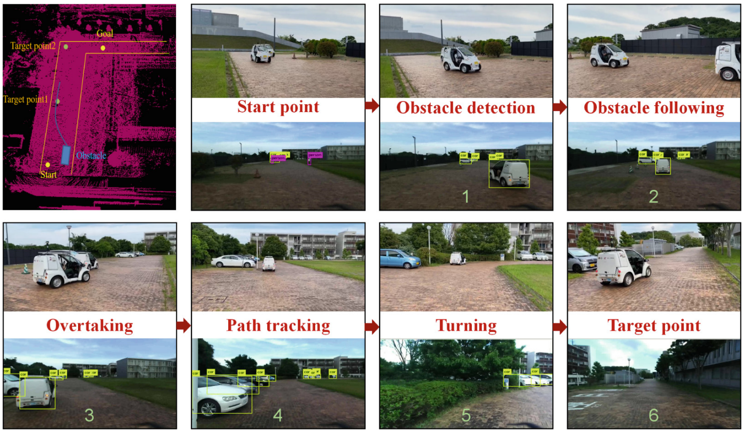


Fig. 6. On-site testing scenarios and distribution of the stages.

curvature of the reference path and the maximum angular velocity of the robot. To narrow the search range and improve the search efficiency, the kinematics unfeasible space χ_{unf}:

$$\chi_{unf}= \{x \in \chi \,|t_0 \le t \le t_{\max} \,, 0 \le s - s_0 \le v_{\max} \cdot (t - t_0)\} \tag{17}$$

The free area χ_{free} can be defined as:

$$\chi_{free}= \{x \in \chi|!(\chi_{obs} \cup \chi_{unf})\} \tag{18}$$

The extension of RRT will be extended in the free area to calculate the speed of the robot.

4 Experimental Results

To verify the effectiveness and robustness of our proposed AM-RRT* method, simulation experiments and on-site tests were conducted. First, AM-RRT* was

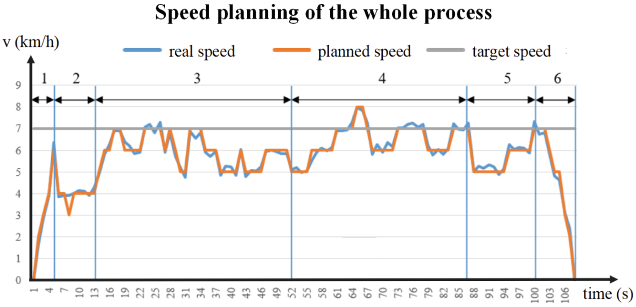


Fig. 7. Speed planning results of the whole on-site testing process.

compared with RRT, RRT*, and IRRT* algorithms. Subsequently, the path optimization method and speed planning method were tested in the normal scene. In the on-site testing section, AM-RRT* was conducted in the actual environment using a modified autonomous driving vehicle. The results showed that our method had good performance in the number of nodes and path length.

First, the proposed path planning algorithm was compared with RRT, RRT*, and IRRT*, and the results are shown in Fig. 2. In the simulation, 5 scenes were designed: Bug, Simple, Normal, Obstacle, and Hard. These scenes comprehensively demonstrate the performance of the algorithm, and our method yields smoother paths compared to IRRT*. For a more rigorous comparison, three perspectives were compared: number of nodes, path length, and time cost. As shown in Tab. 1, Our method has significant advantages in the number of nodes and path length, and with the addition of optimized methods (RRT* and IRRT*), we also have advantages in terms of time cost.

In the path optimization, to verify the effectiveness of the model constraints, a kinematics model [25] was used for comparative verification. As shown in Fig. 3, two models and steering constraints were used for planning results in Normal scenarios, respectively. Under the 30° constraint, the results of both models are better than those under the 45° constraint, but the 5-degree of freedom model we used has a smoother path. Table 2 shows the performance of the four models at 5000 iterations. In addition to time cost, the dynamic model has significant advantages in terms of node number, path length, and maximum angle.

The improved B-spline curve method was also tested under the Normal scene, and the comparison was made by changing the weight value λ. As shown in Fig. 4, the weight values were set to 100, 50, and 10, respectively. As can be seen from the results that when the weight value becomes smaller, the path length will become smaller, but it is much closer to the obstacles. At the same time, due to the use of a fitting B-spline curve, the optimized curve will not pass through all control points. Considering the error caused by the sensor and path tracking in actual situations, it is necessary to set a reasonable weight value.

To verify the feasibility of the speed planning algorithm, four typical scenes were selected for simulation verification, which is shown in Fig. 5. In the simulation process, it is assumed that the maximum speed of the robot is $4\,\mathrm{m/s}$, the expected speed is $2.5\,\mathrm{m/s}$, the acceleration is limited to $\pm 1\,\mathrm{m/s}^2$ and the forward-looking time is $5\,\mathrm{s}$, so the maximum forward-looking distance is $20\,\mathrm{m}$. In the s-t diagram, the blue area is the defined unfeasible area and the dynamic obstacle area, the gray lines are the RRT search tree for speed planning, and the red line is the final selection speed path for the robot.

The on-site test of an automatic driving vehicle was tested to verify its feasibility under vehicle conditions. For the hardware platform, the VLP-16 LIDAR and ZED binocular cameras were used to perceive the surrounding environment and detect the position of obstacles. As shown in Fig. 6, LiDAR-based mapping work was conducted on a closed road and set the starting point and some target points. Another vehicle was used to simulate dynamic obstacles. The speed planning results for the process are shown in Fig. 7, with the target speed set at

Table 1. Comparison of the number of nodes, path length, and time cost of RRT [10], RRT* [19], IRRT* [20], and our proposed method in 5 scenarios.

Environment	methods	No. of nodes	Path length (m)	Time cost (s)
Bug (12*12) 10000 iterations	RRT	8361	17.26	**15.24**
	RRT*	7180	12.88	30.02
	IRRT*	6164	13.03	42.78
	Ours	**5615**	**12.49**	30.83
Simple (16*16) 10000 iterations	RRT	9158	15.67	**16.91**
	RRT*	9166	12.97	37.54
	IRRT*	8738	10.89	81.89
	Ours	**8456**	**10.65**	68.57
Normal (6*6) 10000 iterations	RRT	5001	12.86	**9.85**
	RRT*	5060	6.74	40.69
	IRRT*	2614	6.72	31.31
	Ours	**2324**	**6.47**	25.26
Obstacle (12*12) 10000 iterations	RRT	7102	19.61	**13.64**
	RRT*	7099	14.1	33.89
	IRRT*	7611	13.53	73.19
	Ours	**6333**	**13.02**	39.72
hard (16*16) 10000 iterations	RRT	7893	30.27	**14.95**
	RRT*	7719	24.11	35.12
	IRRT*	7268	27.39	46.02
	Ours	**7141**	**23.89**	30.62

Table 2. Comparison of number of nodes, path length, time cost and maximum angle of kinematics model [25] and dynamics model [27] in Normal scene.

Environment	Methods	Time cost (s)	No. of nodes	Path length (m)	Maximum angle (°)
Normal (6*6) 5000 iterations	45° kinematic	**5.73**	1619	6.95	23.5
	45° dynamic	7.03	1653	6.58	20.2
	30° kinematic	8.38	1307	6.46	14.4
	30° dynamic	12.35	**1194**	**6.36**	**11.6**

7 km/h. First, a dynamic obstacle was detected after the path planning at the start point, as shown in stage 1. Then, in stage 2, the motion planning performed the obstacle following operation and gradually reduced the speed to 4 km/h. In stage 3, with the cooperation of the environment perception, the motion planning executed the overtaking action and increased the speed. After completing the overtaking operation, the action planner re-planned the path and brought the speed closer to the target speed, as shown in stage 4. Finally, as shown in stages 5 and 6, after turning and reaching the goal point, the vehicle slowed down and stopped. From the results, the robot motion planning system AM-RRT* proposed in this paper can be applied to the robot in the real environment through the coupling control of path planning, path optimization, and speed planning.

5 Conclusion and Future Work

This paper proposed an automatic RRT-based motion planning method suitable for robots called AM-RRT*. First, to solve the problem of long paths of traditional RRT, an improved scheme for elliptical regions and attractive potential fields was designed. Then, to solve the problem of insufficient smoothness and large angles in traditional path planning, an optimization method based on a 5-degree-of-freedom dynamic model and a cubic spline curve was proposed. Finally, in response to the shortcomings of traditional path planning not being coordinated with speed control, an RRT-based speed planning method was proposed. Multiple simulation experiments and on-site testing results have demonstrated the effectiveness and robustness of our method. In future work, we will be committed to developing robot motion planning solutions under special conditions, such as magnetic adsorption robots, underwater robots, etc., to enable robots to automatically and intelligently replace humans in hazardous work.

References

1. Hichri, B., Gallala, A., Giovannini, F., Kedziora, S.: Mobile robots path planning and mobile multirobots control: a review. Robotica **40**(12), 4257–4270 (2022)
2. Tae, N., Jae, S., Young, C.: A 2.5d map-based mobile robot localization via cooperation of aerial and ground robots. Sensors **17**(12), 2730 (2017)
3. Lozano-Perez, T., Wesley, M.: An algorithm for planning collision-free paths among polyhedral obstacles. Commun. ACM **22**, 560–570 (1979)
4. Takahashi, O., Schilling, R.: Motion planning in a plane using generalized voronoi diagrams. IEEE Trans. Robot. Autom. **5**, 143–150 (1989)
5. Kavraki, L.E., Svestka, P.: Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Trans. Rob. Autom. **12**(4), 566–580 (1996)
6. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. In: 1985 IEEE International Conference on Robotics and Automation (Cat. No. 85CH2152-7), pp. 500–505 (1984)
7. Hwang, J., Kim, J., Lim, S., Park, K.: A fast path planning by path graph optimization. IEEE Trans. Syst. Man Cybern. Part A-Syst. Humans **33**(1), 121–128 (2003)
8. Kushleyev, A., Likhachev, M.: Time-bounded lattice for efficient planning in dynamic environments. In: ICRA: 2009 IEEE International Conference Robotics and Automation, vol. 1–7. pp. 4303–4309 (2009)
9. Likhachev, M., Gordon, G., Thrun, S.: Ara*: anytime a* with provable bounds on sub-optimality. In: Thrun, S., Saul, K., Scholkopf, B. (eds.) Advances in Neural Information Processing Systems, vol. 16, pp. 767–774 (2004)
10. Lavalle, S.M.: Rapidly-exploring random trees: a new tool for path planning. Research Report (1999)
11. LaValle, S., Kuffner, J.: Rapidly-exploring random trees: progress and prospects. In: Donald, B., Lynch, K., Rus, D. (eds.) Algorithmic and Computational Robotics: New Directions, pp. 293–308 (2001)
12. Reeds, J.A., Shepp, R.A.: Optimal paths for a car that goes both forward and backward. Pac. J. Math. **145** (1991)

13. Brezak, M., Petrovic, I.: Real-time approximation of clothoids with bounded error for path planning applications. IEEE Trans. Rob. **30**(2), 507–515 (2014)
14. Jolly, K.G., Kumar, R.S., Vijayakumar, R.: A bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits. Rob. Auton. Syst. **57**(1), 23–33 (2009)
15. Koyuncu, E., Inalhan, G.: A probabilistic b-spline motion planning algorithm for unmanned helicopters flying in dense 3d environments. In: 2008 IEEE/RSJ International Conference on Robots and Intelligent Systems, Conference Proceedings, vol. 1–3, pp. 815–821 (2008)
16. Luo, M., Hou, X., Yang, S.X.: A multi-scale map method based on bioinspired neural network algorithm for robot path planning. IEEE Access **7**, 142682–142691 (2019)
17. Xin, J., Zhong, J., Sheng, J., Li, P., Cui, Y.: Improved genetic algorithms based on data-driven operators for path planning of unmanned surface vehicle. Int. J. Rob. Autom. **34**(6), 713–722 (2019)
18. Luo, Q., Wang, H., Zheng, Y., He, J.: Research on path planning of mobile robot based on improved ant colony algorithm. Neural Comput. Appl. **32**(6SI), 1555–1566 (2020)
19. Karaman, S., Frazzoli, E.: Incremental sampling-based algorithms for optimal motion planning. Rob. Sci. Syst. **104**, 267–274 (2010)
20. Gammell, J.D., Srinivasa, S.S., Barfoot, T.D.: Informed rrt*: optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), pp. 2997–3004 (2014)
21. Jeong, I.B., Lee, S.J., Kim, J.H.: Quick-rrt*: triangular inequality-based implementation of rrt* with improved initial solution and convergence rate. Expert Syst. Appl. **123**(JUN.), 82–90 (2019)
22. Qureshi, A.H., Ayaz, Y.: Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments. Rob. Auton. Syst. **68**, 1–11 (2015)
23. Jordan, M., Perez, A.: Optimal bidirectional rapidly-exploring random trees (2013)
24. Li, Y., Wei, W., Gao, Y., Wang, D., Fan, Z.: Pq-rrt*: an improved path planning algorithm for mobile robots. Expert Syst. Appl. **152** (2020)
25. Shi, Y., Li, Q., Bu, S., Yang, J., Zhu, L.: Research on intelligent vehicle path planning based on rapidly-exploring random tree. Math. Prob. Eng. **2020** (2020)
26. Jiang, C., et al.: R2-rrt*: reliability-based robust mission planning of off-road autonomous ground vehicle under uncertain terrain environment. IEEE Trans. Autom. Sci. Eng. **19**(2), 1030–1046 (2022)
27. Pepy, R., Lambert, A., Mounier, H.: Path planning using a dynamic vehicle model. In: 2006 2nd International Conference on Information & Communication Technologies, vol. 1, pp. 781–786 (2006)