Research on local trajectory planning of autonomous mobile robots

Yang Zhou[1,2,*]

[1]School of Mechanical Engineering, Zhejiang University, Hangzhou Zhejiang, China
[2]Zhejiang Intelligent Robot Research Institute, Hangzhou Zhejiang, China

Abstract. In the process of autonomous motion, the mobile robot needs real-time local trajectory planning based on the global smooth path and nominal local trajectory tracking. In this paper, the local trajectory planning problem of mobile robot is studied, and the local path represented by quintic polynomial is optimized in Frenet coordinate system with the global smooth path as reference path, then, according to the information of the robot's pose, velocity and angular velocity, the robot's velocity is planned according to all kinds of constraints. A LQR closed loop control law is designed to realize the precise tracking of local optimal trajectory, which makes the robot move safely and stably in the dynamic environment. In the "maze" simulation environment, using the Stageros simulator the local trajectory planning algorithm is tested, and the validity and correctness of the algorithm are verified.

Keywords: Mobile robot, Quintic polynomial, Trajectory planning.

1 Introduction

Mobile robots use the sensors to sense the surrounding environment and its own motion state, having the abilities to locate, plan and control independently in environments with static and dynamic obstacles. In order to move to the target safely and complete the navigation task, mobile robots should obtain the obstacles accurately and make the local trajectory plan in real-time. Therefore, the local trajectory planning of mobile robot is a research topic of great significance [1-3].

In recent years, many scholars have studied the global path planning and local trajectory planning of mobile robots [4-8]. Robots that move in the dynamic environment use the environmental information obtained by the sensor to adjust the local trajectory in real-time. Traditional local trajectory planning methods are easy to fall into local optimization and the trajectory may not smooth enough[1]. Trajectory optimization algorithms usually try to minimize the control cost, control error or the running time from the starting point to the target point of the robot. However, obtaining the optimal trajectory requires high computational cost, which limits the use of online real-time trajectory planning. Therefore, many scholars have focused on how to efficiently plan the suboptimal trajectory. Taking

---

[*] Corresponding author: yznudt@126.com

the kinematic constraints of the robot into account, Lau and Sprunk adopt the RPROP algorithm which is not required to compute the gradient to optimize the motion trajectory represented by the spline curves [9-11]. Ratliff et al. proposed an online planning algorithm and conjugate gradient method is used[12]. Rolling window method (DWA) is a widely used robot navigation method[13]. Firstly, multiple groups of speeds are sampled in the speed space, the corresponding motion trajectory is predicted, and then all speeds are exhausted to find the optimal control command according to the cost function. The cost function includes the distance from the target point, the forward speed and the distance from the obstacle. In order to reduce computation time, DWA limits the search space to a subset of all candidate solutions, so it cannot be guaranteed to obtain the optimal solution. Furthermore, due to direct sampling in the speed space, it is easy to lead to unsmooth motion. Roesmann et al. proposed a timed elastic band algorithm (TEB) based on the basic elastic band method. TEB considers the non-holonomic kinematics and dynamics constraints and adds the time information to the path [14-15]. In order to improve efficiency, TEB converts hard constraints into soft constraints. As a result, it may fall into local optimization when dynamic obstacles appear.

Compared to the path planning problem, Trajectory planning adds time information into the path. The existing research on robot navigation shows that it has significant advantages to divide the motion control into two modules: real-time trajectory planning and trajectory tracking control. In this paper the global smooth path is taken as a reference path, the local path represented by quintic polynomial is optimized in the Frenet coordinate system. Based on the current pose, speed and angular velocity information of the robot and considering various constraints, the speed planning aims to generate longitudinal speed profile along the path generated.

2 Formatting the title, authors and affiliations

In order to deal with the complex and dynamic environment and ensure the safety and smoothness, the local trajectory planning algorithm which based on the global smooth path must generative a collision-free and smooth trajectory in real-time. The designed robot motion mode includes three steps: rotation at the start position first, then move forward to the goal, finally rotate at the goal. The flow chart of local trajectory planning is shown in Fig. 1. The input information is the global smooth path, robot state and distance map, and the output information is the optimal local trajectory (the desired pose and speed of the robot at each time). The distance map contains the distance information between each grid and the nearest obstacle. If there exists a global smooth reference path and the robot has not arrived at the goal pose, the local trajectory planning is carried out in real-time. At the beginning of planning, we should judge whether to enter the rotation process. If yes, do rotate speed planning, otherwise do local path planning and speeding planning on the path.

2.1 Path planning

In order to establish the Frenet coordinate, firstly find the nearest point on the global smooth path according to the robot position, and then obtain the local reference path. As shown in Figure 2, the S-axis is the tangent of the reference path, and the L-axis is perpendicular to the S-axis. The longitudinal offset S is the path length from the starting point of the reference line to the projection point, and the horizontal offset L is the distance from the projection point to the robot. The robot's pose, velocity, acceleration and other motion state information in the Cartesian coordinate system can be converted into the lateral offset and the first-order and second-order derivatives of the lateral offset to the arc

length in the Frenet coordinate system. It should be pointed out that the angle between the orientation of the robot and the S-axis cannot be greater than 90 degrees.
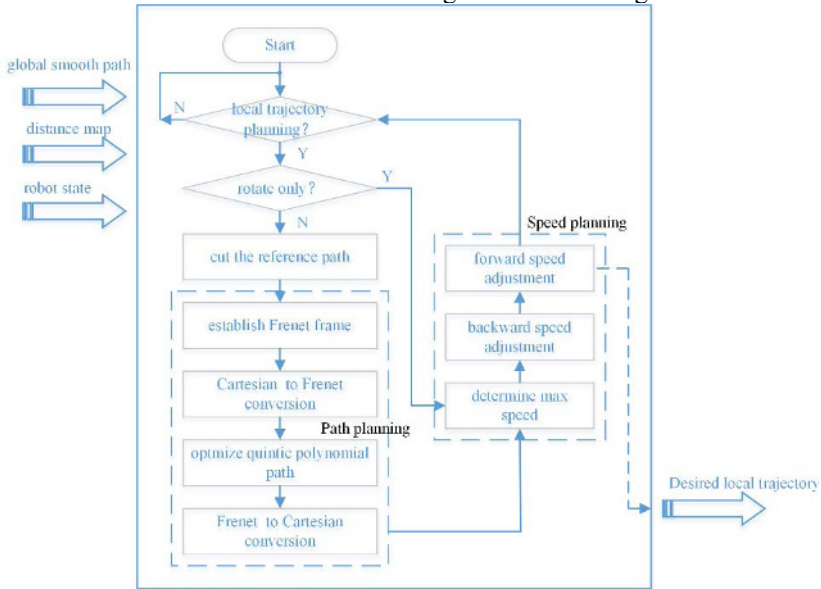

Fig. 1. Local trajectory planning flow chart.

$$l(s) = a_0 + a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4 + a_5 s^5 \qquad (1)$$

The local path planning process is as follows: in order to improve the calculation efficiency, first cut the global smooth curve according to the current position of the robot as the local reference path, then establish the Frenet coordinate system on the local reference path, optimize the quintic polynomial path under the coordinate system, and finally transform the optimal local path to the Cartesian coordinate system.

The optimization process of local quintic polynomial path planning is as follows: firstly, a series of candidate paths are generated according to the current state of the robot and the state of the sampling end, then the safety of the path is detected quickly according to the distance graph, and finally the optimal path is found from the feasible path according to the optimization index. The optimization objective is designed

$$J = w_1 l_1 + w_2 J_{smooth} + w_3 J_{obs} \qquad (2)$$

Where l_1 is horizontal offse, J_{smooth} represents the smoothness, J_{obs} is related to the obstacles, w_1, w_2, w_3 are the weight coefficients.
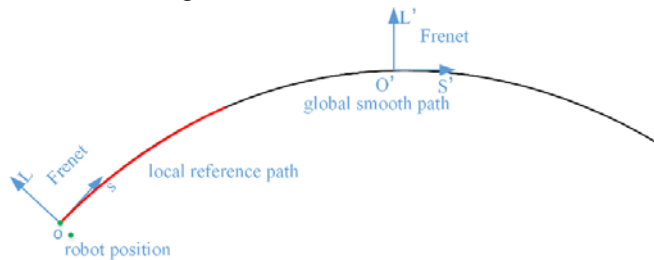

Fig.2. Frenet frame.

2.2 Speed planning

The schematic diagram of speed planning is shown in Figure 3. The speed planning is divided into three steps: First, determine the maximum allowable (angular) speed of each point of the local path; then, starting from the starting point of the local path, adjust the (angular) velocity of the next point according to the (angular) velocity and maximum (angular) acceleration constraints of the current point; at last, similar to step 2, adjust the (angular) speed from the end point of the local path.
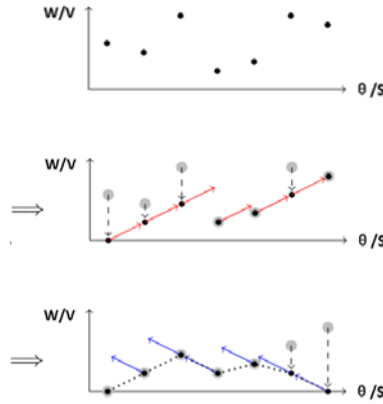

**Fig. 3.** Velocity planning with constant acceleration.

3 Simulation result

In order to verify the effectiveness of local trajectory planning method, we use Stageros simulator to simulate the "maze" scene. Stageros is a 2D robot simulator which can define robots, lidar, obstacles and so on. The applied "maze" scene is shown in Figure 4, and the arrow represents the target point and the circle represents the mobile robot. The mobile robot carries a 2D lidar sensor. The robot will generate a global smooth path when a target point is given, and then the local trajectory planning algorithm is running in real-time to realize obstacle avoidance safely and smoothly. When the robot moves forward, a series of candidate paths are generated according to the current state and the sampling end state. As shown in Figure 4, it can be seen that these candidate paths start from the current position of the robot, and the end is parallel to the global path. These paths meet the smoothness requirements as they are quintic polynomial paths. Then, according to the obstacles sensed by lidar, the safeties of the paths are checked. As shown in Figure 5, infeasible paths are deleted from the figure. Finally, according to the designed optimization objective, the optimal path is chosen from the feasible paths, as shown by the blue arrow in Figure 5, obviously this is a smooth and obstacle-free path. Figure 6 shows the change curve of the robot state x, y and yaw with time. From the figure, it can be seen that the robot motion is relatively stable.

4 Conclusion

In order to handle the dynamic obstacles, a local smooth trajectory optimization method based on Frenet coordinate system with the global smooth path as the reference path is proposed. Firstly, a series of candidate quintic polynomial paths are sampled according to

the robot state, and then check the safety of these paths according to the obstacle information, at last the optimal collision-free path will be chosen. After obtaining the optimal local path, the speed planning with constant acceleration is adopted to obtain a safe and stable local trajectory, so as to realize dynamic obstacle avoidance. Finally, the robot and dynamic environment are simulated in Stageros. The simulation results show that the proposed local trajectory planning algorithm is able to plan a smooth and safe local optimal trajectory in the environment with dynamic obstacles. Therefore, the effectiveness and correctness of the algorithm have been verified.
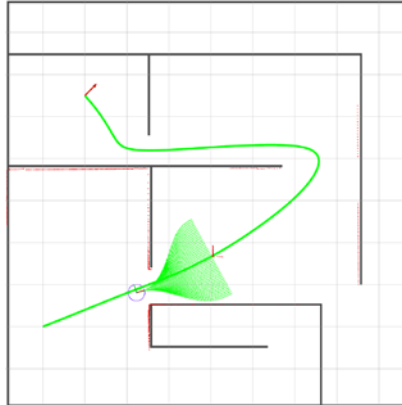
Fig. 4. Local paths generation.
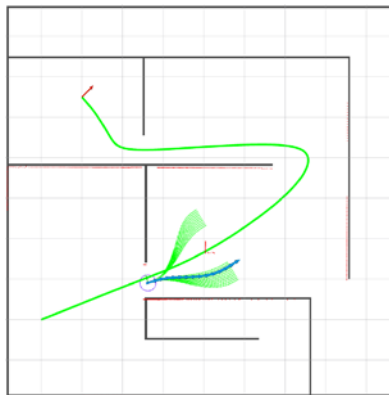
Fig. 5. Choose the optimal path in obstacle-free paths.
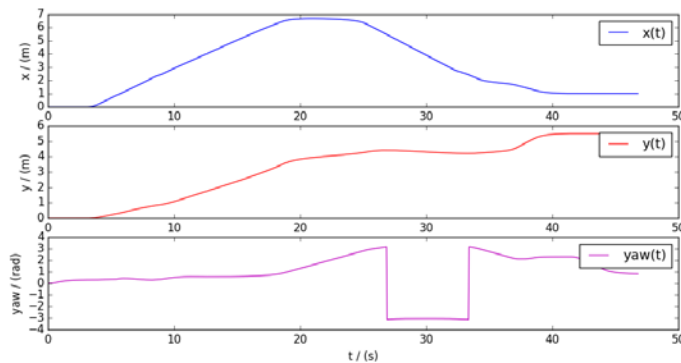
Fig. 6. Time-dependent curve of robot position and orientation.

References

1. Zhang Qi. Research on path planning and location technology of mobile robot [D]. Harbin: Harbin Institute of Technology, 2014.

2. Lv Tai-zhi. Research on path planning and mapping of mobile robot[D]. NanJing: Nanjing University of Science and Technology, 2017.

3. R. Siegwart, I. R. Nourbakhsh. Introduction to autonomous mobile robots[M]. MIT Press, 2004.

4. Ren Guo-hua. Mobile robot trajectory tracking and motion control, 2014 (3):100-102.

5. S. M. LaValle, Planning algorithms[M]. Cambridge University Press, 2006.

6. D. González, J. Pérez, V. Milanés, et al. A review of motion planning techniques for automated vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 17(4): 1135-1145.

7. S. Karaman, M. R. Walter, A. Perez, et al. Anytime motion planning using the RRT*[C]. In Proceedings of IEEE In-ternational Conference on Robotics and Automation(ICRA), 2011.

8. J. D. Gammell , S. S. Srinivasa, T. D. Barfoot. Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic[C]. International Conference on Intelligent Robots and Systems. IEEE, 2014.

9. C. Sprunk, B. Lau, P. Pfaff, et al. An accurate and efficient navigation system for omnidirectional robots in industri-al environments[J]. Autonomous Robots, 2017, 41(2): 473-493.

10. B. Lau, C. Sprunk, W. Burgard. Improved updating of Euclidean distance maps and Voronoi diagrams[C]. IEEE International Conference on Intelligent Robots and Systems, 2010.

11. B. Lau, C. Sprunk, W. Burgard. Kinodynamic motion planning for mobile robots using splines[C]. IEEEInternational Conference on Intelligent Robots and Systems, 2009.

12. N. Ratliff, M. Zucker, J. A. Bagnell, et al. CHOMP: Gradient optimization techniques for efficient motion plan-ning[C]. In Proceedings of IEEE International Conference on Robotics and Automation(ICRA), 2009.

13. D. Fox, W. Burgard, S. Thrun. The dynamic window approach to collision avoidance[J]. IEEE Robotics and Auto-mation Magazine, 1997, 4(1): 23-33.

14. C. Roesmann, F. Hoffmann, T. Bertram. Integrated online trajectory planning and optimization in distinctive to-pologies[J]. Robotics and Autonomous Systems, 2017, 88: 142–153.

15. C. Roesmann, W. Feiten, T. Woesch, et al. Trajectory modification considering dynamic constraints of autonomous robots[C]. In Proceedings of 7th German Conference on Robotics (ROBOTIK), 2012.