

Robust Path Planning of Obstacle Avoidance for Unmanned Delivery Robots

Mushu Wang, Xingxue Dong, Weigang Pan*, Song Gao, Shuxin Wang

Shandong Jiaotong University, Jinan, 250357

China (Tel: +8617862926613; e-mail: wangmushu@126.com, corresponding author: panweigang1980@163.com).

Abstract: Aiming at complex road conditions and difficult path planning of semi-structured roads in a park, a robust path planning method based on sequential quadratic programming (SQP) algorithm optimization is proposed. Firstly, an improved vector field histogram (VFH) algorithm is used to determine an optimal passable area, and then the determination method of the target state is given based on the optimal passable area; Secondly, according to a starting point and the target state, a description method of an obstacle avoidance path is given based on a piecewise quadratic Bezier curve. Then, the problem of curve parameter optimization is established based on a robot running curvature constraint and a direction change margin constraint of the target point, and the SQP algorithm is used to optimize the curve parameters; Finally, the effectiveness and practicability of the proposed method are verified by simulation experiments and experiments of a real robot. Compared with other methods, the proposed method has the shortest path and better robustness.

Keywords: path planning, obstacle avoidance, navigation, robot, Beziercurve.

1. INTRODUCTION

In recent decades, a large number of scholars have devoted themselves to the research of unmanned driving technology, so unmanned driving technology has made great progress. At present, unmanned driving technology has gradually transitioned from a closed environment to an open scene (González et al., 2015). Based on known information, how to successfully avoid unknown obstacles in a complex distribution on the driving path and finally return to the original driving route is a problem solved. It belongs to a path planning problem in the field of unmanned driving. And commonly used path planning methods include graph-based search, sampling-based methods, interpolating curves, and reaction-based methods, intelligent bionic algorithms, etc. (Zhou et al., 2022; Guo et al., 2021).

Graph-based search, sampling-based methods and intelligent bionic methods were often used for global path generation. The search-based path planning method transformed the planning problem into a graph search problem (Erke et al., 2020; Dakulović et al., 2011). This method divided the solution space into grids, constructed a graph, and then obtained an optimal path. Moreover, A* (Erke et al., 2020) and D* (Dakulović et al., 2011) were common graph search algorithms. A sampling-based method randomly sampled within a fixed workspace to generate suboptimal paths. And it included a rapid exploration random tree (RRT) (Li et al., 2020) and a probabilistic roadmap (PRM) method (Mohanta et al., 2019). Intelligent biomimetic methods simulated biological evolution and insect foraging behaviors in nature. Commonly used intelligent bionic algorithms included a genetic algorithm (GA) (Hao et al., 2020), a particle swarm optimization (PSO) algorithm (Song et al., 2021), etc. In recent years, many studies combined global and local path planning algorithms to complete obstacle avoidance for unmanned vehicles e.g. the Dijkstra algorithm combined with

a dynamic window method (DWA) (Liu et al., 2021), an improved A* algorithm combined with the DWA (Ji et al., 2021), an improved RRT algorithm combined with the VFH (Chen et al., 2020), the PSO combined with an artificial potential field (APF) algorithm (Zhou et al., 2018).

Reaction-based methods were often used for local path planning (Murphy et al., 2019). It made intelligent vehicles or robots reach specified goals without knowing a global map. There were several kinds of reactive navigation algorithms, such as a Bug algorithm, a potential field method and so on. The Bug algorithm was one of the simplest algorithms, and it included Bug 1, Bug 1+2, and a tangent Bug (Babinec et al., 2014). Its basic idea was to avoid obstacles along the edge of an obstacle, so its efficiency was low (McGuire et al., 2019).

A potential field method included an artificial potential field method (Duhé et al., 2021), an extended potential field method (Siegwart et al., 2011), a virtual force field method (Choe et al., 2014). It processed information quickly. However, it was easily affected by an obstacle distribution and produced a local minimum, in which an agent might be trapped. Moreover, it introduced unnecessary oscillations to the agent's motion. In recent years, some studies improved the potential field method. The improved artificial potential field method was applied to the motion planning of humanoid robots (Kumar et al., 2019), the collision-free path generation of autonomous vehicles (Wang et al., 2019), and the dynamic road planning of robots in complex and unknown environments (Park et al., 2020).

The VFH algorithm was evolved from the virtual potential field, and it effectively solved the problem that the virtual potential field method made a robot oscillate and fall into the local minimum. However, the traditional VFH algorithm hardly met vehicle motion constraints, and its threshold was sensitive. An improved VFH algorithm designed an active area according to the minimum left turn radius. And it

planned a path which satisfied vehicle kinematic constraints (Qu et al., 2015). Aiming at the threshold sensitivity problem of the VFH algorithm, an adaptive threshold setting strategy was proposed in (Zhuang et al., 2018). This strategy improved robustness of the VFH algorithm. The VFH family of methods belonged to reactive navigation algorithms (Babinec et al., 2014), but it lacked smoothness of the generated path.

A continuous geometric curve interpolation method was used to generate smooth paths. And it was used for vehicle lane changing and obstacle avoidance path generation, and also used for global path smoothing (Zhou et al., 2022). Geometric curves included polynomial curves (Liu et al., 2019), B-spline curves (Elbanhawi et al., 2015), Bezier curves (Choi et al., 2012), etc. The coefficients of polynomial curves were related to constraints such as a curve curvature, start and end points, and gradients of the start and end segments (González et al., 2015). Aiming at the problem of coefficient determination for polynomial curves, (Glaser et al., 2010) proposed a calculation method, which laid the foundation for the application of polynomial curves. Cubic polynomials were applied to generate safe trajectories for overtaking maneuvers (Petrov et al., 2014). Additionally, quartic and quintic polynomials were also applied to real-time trajectory planning of autonomous vehicles for speed-keeping in highway scenarios in order to describe vehicle motion curves more accurately (Wei et al., 2021; Yue et al., 2019). Moreover, a B-spline-based method was used to plan real-time driving paths with continuous curvature satisfying vehicle kinematic constraints (Zeng et al., 2020). And a particle swarm optimization algorithm was used to generate robot obstacle avoidance path (Lian et al., 2020; Li et al., 2020).

A Bezier curve was mostly used in path planning because of its simple and smooth characteristics (Ding et al., 2019; Chen et al., 2013; Chen et al., 2019; Chen et al., 2015; Zheng et al., 2020; Lattarulo et al., 2018). It satisfied an upper limit of curvature constraint by optimizing the position of the control point. The piecewise quadratic Bezier curve was applied to obstacle avoidance path generation, and the safety and feasibility of the generated trajectory was guaranteed by parameterization of the piecewise Bezier curve (Ding et al., 2019; Chen et al., 2013). A cubic Bezier curve was applied to the planning method of a collision avoidance path and speed of an intelligent vehicle (Chen et al., 2019). In order to improve accuracy of a generated path, a trajectory planning method based on a fourth-order Bezier curve was proposed (Chen et al., 2015). A path planning algorithm based on quartic Bezier curves (Zheng et al., 2020) used a hazard field to determine the end point of a local path, which improved planning efficiency. In addition, parametric Bezier curves were also used for driving trajectory planning in urban road scenarios (Lattarulo et al., 2018). Continuous geometric curve interpolations were also widely used for global path planning and smooth line adjustment (Elbanhawi et al., 2015; Berglund et al., 2009; Deng et al., 2021; Xu et al., 2022). A B-spline curve was exploited to generate continuous smooth paths with maximum curvature constraints (Elbanhawi et al., 2015). And it also was used to plan smooth travel paths of an

autonomous mining vehicle (Berglund et al.). Furthermore, it was used to smooth paths generated by using a D*Lite algorithm (Deng et al., 2021) and an improved PSO algorithm (Xu et al., 2022).

In this paper, a robust path planning method is proposed based on the SQP algorithm. Firstly, based on the VFH algorithm, a screening strategy of candidate passable areas is proposed, and then the optimal passable area is determined by an improved feasible direction loss function, which considers a global path as traction. Meanwhile, a determination method of a target state in an optimal passable area is given. Then, according to the starting point and the target state, a piecewise quadratic Bezier curve is used to construct a path between the starting point and the target, and the determination method of a control point and a piecewise point of the Bezier curve is given. The direction variation margin of the target point is constrained, and the constrained optimization problem of the curve parameters is established. Its optimization objective is the shortest path length and the smallest change in the direction of the target point, and the SQP algorithm is used to solve the optimization problem. Finally, simulation and avoidance experiments of a real robot are designed to verify the effectiveness and practicability of the proposed algorithm.

The innovations of the algorithm proposed in this paper are as follows:

- (1) Aiming at the problem of determining an optimal passable area, an improved VFH algorithm is given and used for screening strategy for candidate passable areas, and a feasibility direction loss function is improved to determine the optimal passable area. This work improves the safety and effectiveness of the optimal passable area.
- (2) Aiming at robustness of the path planning, a new calculation method for the upper limit of curvature is proposed, and it increases flexibility in the direction of the end point. This work improves a success rate of path optimization.
- (3) Aiming at speed of path planning, this paper constructs an optimal planning path described by a piecewise quadratic Bezier curve. Under the premise of ensuring the smoothness of the path, the proposed method takes less optimization time to obtain the shortest path, and it reduces the energy consumption for obstacle avoidance.
- (4) Aiming at the application strategy of the path planning algorithm, this paper proposes a path monitoring strategy, which monitors safety of the planned path in real-time to determine whether a new obstacle avoidance path is needed to be planned or not. This work improves the flexibility of path planning, and reduces time cost of the path planning.

This paper is organized as follows. Section 1 lists recent researches related to path planning. Section 2 introduces the problem to be solved according to obstacle avoidance process of unmanned delivery robots, and introduces a structure of the proposed obstacle avoidance path planning algorithm. Section 3 details steps of the proposed obstacle avoidance path planning algorithm. Furthermore, the monitoring strategy of the obstacle avoidance path planning algorithm is

designed. Section 4 provides some discussion. The last section includes conclusions and some suggestions for future works.

2. ANALYSIS AND DESCRIPTION OF PROPOSED METHOD

2.1 Problem Description

This paper studies the path planning of an unmanned delivery robot traveling on a known global path in a park when it encounters an unknown obstacle. The environment in the park may be structural paths with boundary constraints, such as single-lane or multi-lanes, or non-structural environments with unclear boundaries, such as squares, dispatching stations paths. The purpose of this paper is to plan a path that makes a robot quickly, gently, and safely bypass obstacles and return to the original path. Taking a structured path as an example, the process of obstacle avoidance is shown in Fig1.

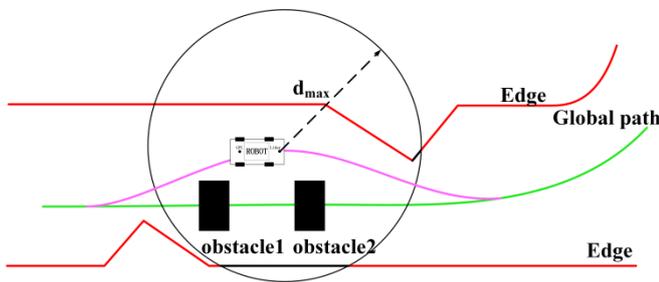


Fig. 1. Obstacle avoidance process.

As shown in Fig.1, according to the positional relationship between the robot and the global path, an obstacle avoidance process is divided into three stages: global path deviation, obstacle avoidance, and global path regression. The problems to be solved are as follows.

- (1) It is necessary to analyze the environmental information to determine a direction that ensures a safe passage of the robot, and to determine an endpoint state of a local obstacle avoidance path (LOAP).
- (2) After determining the travel direction and the endpoint state of the LOAP, it is necessary to describe the obstacle avoidance path.
- (3) It is necessary to use a solution method to quickly obtain a smooth and shorter LOAP that satisfies the kinematic constraints of the robot.

2.2 Algorithm structure

This paper proposes a robust obstacle avoidance path planning method with an unknown distribution of obstacles for a robot in a park. As shown in Fig. 2, the method includes three parts: determination of an optimal traversable area, path construction based on a piecewise quadratic Bezier curve, and path robustness planning.

First of all, the optimal passable area is determined based on the VFH algorithm. According to the detected environment information, this work put forward an improved screening method for a candidate passable area. And an improved feasible direction loss function is used to determine the optimal feasible direction, and then the optimal passable area

is obtained. Based on the obstacle information in the optimal passable area, the endpoint state of the LOAP is determined.

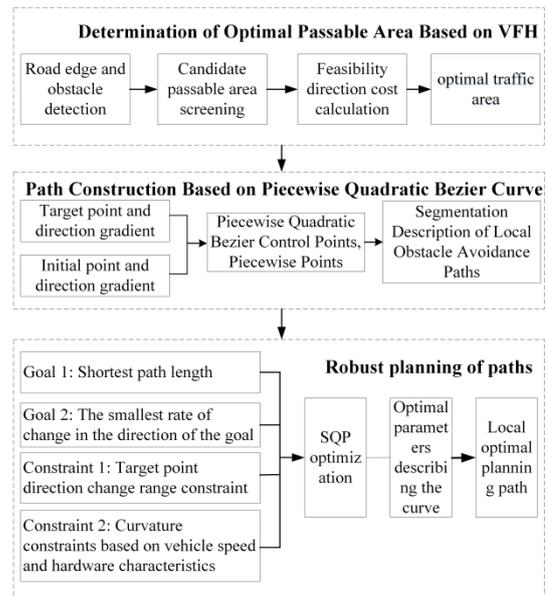


Fig. 2. Robust obstacle avoidance path planning.

Second, a current LOAP is described by using a piecewise quadratic Bezier curve. This work is divided into three steps. The first step combines the current position of the robot to determine the starting and ending points of the LOAP and their directions. The second step is to calculate control and segment points of two Bezier curves, which are used for description of the path. The third step completes the LOAP by using the two Bezier curves.

Finally, the robust path planning is carried out. This work presents a constrained optimization problem based on a Bezier curve piecewise description of the LOAP. The constraints of the optimization problem are a curvature constraint and a direction change margin constraint of the target point, respectively. And the optimization objectives are the shortest path and the minimum target direction change. The SQP method is used to optimize the curve parameters, and then the optimal LOAP is obtained.

3. PATH ROBUST PLANNING ALGORITHMS

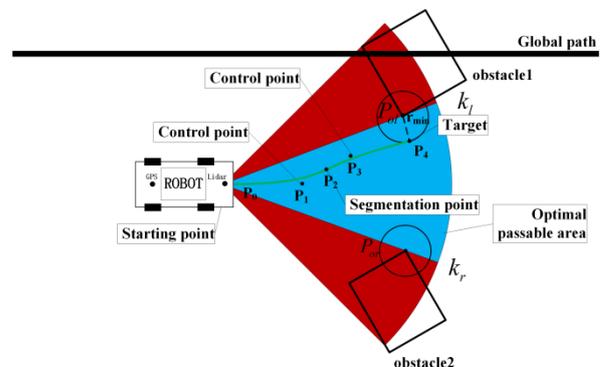


Fig. 3. Determination of the optimal passable area.

For the obstacle avoidance problem in a park, it is first necessary to determine the optimal passable area for obstacle avoidance when the obstacle information and environment

information are known. The blue area in Fig.3 is the optimal passable area; then, determine the target point P_4 of the LOAP and its running direction, and combine the starting point and its direction constraints to study a determination method of control points P_1, P_3 . Then the construction of the local path Bezier curve is realized. Finally, under the constraints of a robot running curvature and the direction of the target point, the parameters of the Bezier curve are optimized to make the obstacle avoidance path shortest and let the change of the target point direction minimum.

3.1. Determination of a passable area based on VFH

In this paper, an adaptive threshold VFH algorithm considering robot width is designed to solve the candidate passable area after obtaining obstacle information. At the same time, by modifying the candidate direction loss function, the traditional VFH algorithm is improved from the target point traction to the global path, to obtain an optimal passable area.

The VFH algorithm divides the environment around the robot into grids. A lidar is used to continuously update the two-dimension grid, and defines a range around the robot as an active window. Each grid acts as an active unit. Then the active window is divided into sectors to calculate the obstacle strength m_n of each sector, and the obstacle strength of each sector is expressed by using equation (1) (Zhuang et al., 2018)

$$m_n = c_v^2 (a - bd_n^2) \quad (1)$$

where c_v is a determined value of each active unit, and depends on an application situation; d is the distance from the active unit to the robot; a and b are both constants satisfying $a = bd_{\max}^2$; d_{\max} is the maximum exploration distance of the lidar. The minimum distance threshold and maximum threshold distance are set according to the inherent characteristics of the robot and the target environment. The maximum distance threshold is denoted by d_{th}^{\max} (Zhuang et al., 2018), and the minimum distance threshold is denoted by d_{th}^{\min} , $d_{th}^{\max} = d_{\max}$. The interval $[d_{th}^{\min}, d_{th}^{\max}]$ is divided into intervals with a certain step size Δd_{th} , and the set of distance thresholds is $\Omega_d = \{d_{th}^{\max} - j \cdot \Delta d_{th} \mid j = 0, 1, 2, \dots\}$. The set of obstacle strength thresholds is $\Omega_r = \{c_v^2 \cdot (a - b \cdot (d_{th}^{\max} - j \cdot \Delta d_{th})^2) \mid j = 0, 1, 2, \dots\}$. An element of the Ω_r is denoted by Ω_r^j . The binarization rule is shown in equation (2)

$$H_n^j = \begin{cases} 1 & \text{if } m_n \geq \Omega_r^j \\ 0 & \text{other} \end{cases} \quad (2)$$

According to the rule, a binarized histogram under the current threshold is established, and it is used to divide the robot environment into a passable area and an impassable area, and then determine the upper and lower angle limits k_l and k_r of the passable area (see Fig. 3).

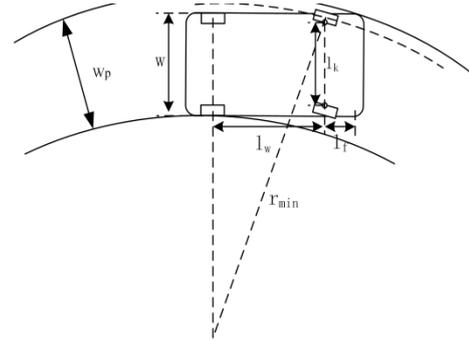


Fig. 4. Minimum passing widths of a robot.

In order to determine safety of the passable area, the angle threshold s_{\min}^j for selecting a passable area is determined based on Ω_d^j . The angle range $[k_l, k_r]$ of the passable area is given according to the current distance threshold Ω_d^j . If $k_r - k_l > s_{\min}^j$, the passable area will be determined as the direction of a candidate passable area. The threshold s_{\min}^j is obtained by

$$s_{\min}^j = 2 \arcsin\left(\frac{w_{\min}}{2\Omega_d^j}\right) \quad (3)$$

where w_{\min} is the minimum passing width to ensure the safe passage of the robot, as shown in equation (4)

$$w_{\min} = w_p + r_{\min} \quad (4)$$

where r_{\min} is the minimum turning radius. w_p is the width of the narrowest running channel, as shown in Fig.4.

And r_{\min} is solved by using equation (5)

$$r_{\min} = \sqrt{(l_w + l_f)^2 + \left(\frac{l_w}{\tan \theta_{\max}} + \frac{w - l_k}{2}\right)^2} \quad (5)$$

where l_w is the wheelbase of the robot; l_f is the length of the front overhang; l_k is the kingpin center distance. w_p is obtained by using equation (6)

$$w_p = w + 2r_{\min}(1 - \cos \theta_{\max}) \quad (6)$$

where w is the width of the robot, and θ_{\max} is the maximum angle of the front wheel of the robot.

When the current obstacle strength threshold is Ω_r^j , $n(n \geq 1)$ passable areas are obtained. And their angle ranges are denoted by $[k_r^{ij}, k_l^{ij}]$, $i = 1, \dots, n$, and the ranges satisfy $k_r^{ij} - k_l^{ij} > s_{\min}^j$.

Candidate feasible directions in this passable area are given by equation (7)

$$c_{ij} = \frac{k_r^{ij} + k_l^{ij}}{2} \quad (7)$$

Substitute the candidate direction c_{ij} into (8) to get an optimal feasible direction under the threshold Ω_r^j

$$c_*^j = \arg \min_{c_{ij}} \mu_1 \square \theta_i + \mu_2 \square \varphi_i + \mu_3 \square \phi_i \quad (8)$$

where $\square \theta_i$ is an angle difference between the candidate direction and the tangent direction of the nearest global path point, $\square \varphi_i$ is an angle difference between the candidate feasible direction and the current movement direction of the robot, and $\square \phi_i$ is an angle difference between the candidate feasible direction and the last movement direction, $\mu_1 > 0$, $\mu_2 > 0$, $\mu_3 > 0$, $\mu_1 \geq \mu_2 + \mu_3$.

Remark 1: The traditional VFH algorithm uses the endpoint as the traction of the movement direction. In order to adapt to complex routes in the park, this paper adopts the global path to pull the movement direction.

Substitute the result of (8) into (9) to obtain an **optimal passable direction** c_* , its corresponding distance threshold is Ω_d^j , and its angle range is $[k_r^*, k_l^*]$ which is regarded as an **optimal passable area**.

$$(c_*, \Omega_d^j) = \arg \min_{c_d^j, \Omega_d^j} \eta(d_{th}^{\max} - \Omega_d^j) + \Delta \theta_*^j \quad (9)$$

where $\eta(\eta > 0)$ is a threshold weight coefficient, and $\Delta \theta_*^j$ is the angle between the path direction and the direction c_*^j .

3.2. Path construction based on piecewise quadratic Bezier curve

In this section, a segmented quadratic Bezier curve is used to describe the local path. For different stages of obstacle avoidance, the determination method of the curve control point and the segment point is given. In addition, higher order Bezier curves require a lot of computation. On the other hand, in order to plan the shortest local path, this paper takes the shortest path length as the optimization objective, and the length of the higher-order Bessel curve is difficult to be accurately solved.

3.2.1. Path target state determination

As shown in Fig.5, to ensure the safety of path planning, this paper further expands the obstacle endpoints at the edge of an optimal passable area, firstly. The expansion radius is the minimum turning radius of the robot, and a safer optimal passable area is obtained, which is called the **optimal inner passable area (OIPA)**.

According to the different external environments, the angle range of the inner passable area is expressed as $[\theta_r^*, \theta_l^*]$, $[k_r^*, k_l^*]$, $[\theta_r^*, k_l^*]$, where θ_r^* is the direction of the vector $P_{or}^j - P_0$, θ_l^* is the direction of the vector $P_{ol}^j - P_0$, the point P_{ol}^j is the tangent point on the expanding circle at the left endpoint, the point P_{or}^j is the tangent point on the expansion circle of the right endpoint.

In this paper, the target state is determined in two cases, i.e.

Case 1: The OIPA has no intersection with the global path

Case 2: The OIPA intersects with the global path

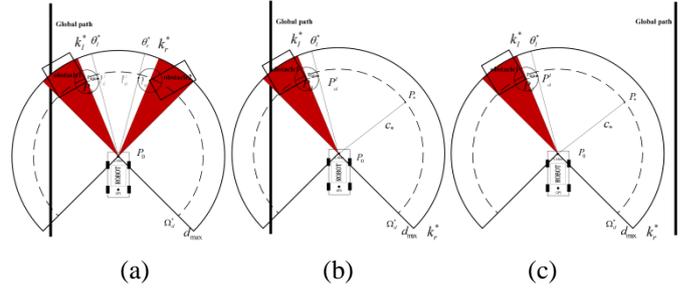


Fig. 5. Three cases of the passable area.

(1) Case 1

As shown in Fig5(a), there is no intersection between the global path and the OIPA, and the OIPA angle range is $[\theta_r^*, \theta_l^*]$, then the local path target point is determined by using equation (10)

$$P_4 = \begin{cases} P_{ol}^j & d_l \leq d_r \\ P_{or}^j & d_l > d_r \end{cases} \quad (10)$$

where d_l and d_r are the shortest distances from points P_{ol}^j and P_{or}^j to the global path, respectively.

As shown in Fig.5(b), (c), when there is an obstacle on one side of the passable area, the global path does not intersect with the OIPA, and the OIPA angle range is $[k_r^*, \theta_l^*]$. As shown in the figures, P_* is the farthest point within the distance threshold range in the optimal feasible direction. Meanwhile, the local path target point is obtained from equation (11)

$$P_4 = \begin{cases} P_{ol}^j & d_l \leq d_* \\ P_* & d_l > d_* \end{cases} \quad (11)$$

where d_l and d_* are the shortest distances from P_{ol}^j and P_* to the global path, respectively.

In this case, the direction, in which the target point is obtained, is the vector $P_4 - P_0$, which is expressed as

$$G_4 = [G_{41}, G_{42}]^T = \frac{P_4 - P_0}{\|P_4 - P_0\|} \quad (12)$$

(2) Case 2

As shown in Fig.6, if there is a global path point in the OIPA, the detection distance of the OIPA is d_{\max} , and the intersection point between the farthest boundary of the OIPA and the global path is P_4 , and P_4 is called a **regression point**.

When the target point is on the global path, its direction is the tangent direction G_{\tan} of the point, and the angle between G_{\tan} and the vector $P_4 - P_0$ is denoted by θ_{\tan} . And G_4 is given by equation (13)

$$G_4 = \begin{cases} G_{\tan} & \text{if } \theta_{\tan} \leq 90^\circ \\ -G_{\tan} & \text{else} \end{cases} \quad (13)$$

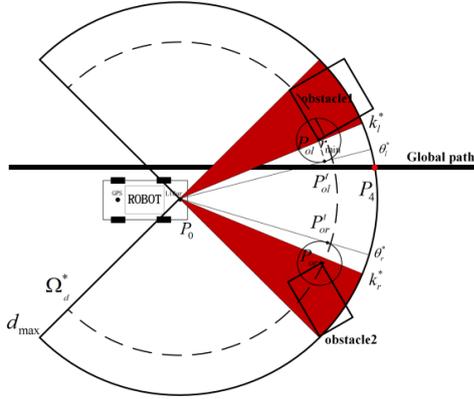


Fig. 6. Determining the regression point.

3.2.2. Control point and segment point determination

The LOAP described by a segmented Bezier curve is shown in Fig.7. P_0 is the initial point of the first quadratic Bezier curve, P_4 is the endpoint of the second quadratic Bezier curve, P_1 and P_3 are control points of the two Bezier curves, respectively. P_2 is the segment point of the two-segment curves. The path is described as

$$\begin{aligned} B_1(t) &= (1-t^2)P_0 + 2t(1-t)P_1 + t^2P_2, 0 \leq t \leq 1 \\ B_2(t) &= (1-t^2)P_2 + 2t(1-t)P_3 + t^2P_4, 0 \leq t \leq 1 \end{aligned} \quad (14)$$

where P_1 , P_2 , P_3 and P_4 are unknown parameters.

For the constraint of path smoothness, it is generally required that the tangent direction of the starting point of the Bezier curve is the same as the starting point heading of the robot, and the tangent direction of the ending point is the same as the ending point heading. The first point P_0 is the coordinate of the robot lidar. G_1 is the unit vector of the current robot heading. Suppose that the distance between P_0 and P_1 is l , and the first control point is obtained by equation (15)

$$P_1 = [P_{01} + G_{11}l, P_{02} + G_{12}l]^T \quad (15)$$

In the previous section, the point P_4 and its direction vector G_4 have been obtained.

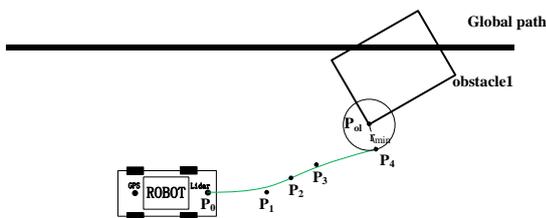


Fig. 7. A piecewise quadratic Bezier curve.

To improve a success rate of path planning, two parameters α and β are introduced to fine-tune G_4 , as (16):

$$\tilde{G}_4 = [G_{41}\alpha, G_{42}\beta]^T \quad (16)$$

And the point P_3 is obtained by using equation (17)

$$P_3 = P_4 - \tilde{G}_4 = [P_{41} - G_{41}\alpha, P_{42} - G_{42}\beta]^T \quad (17)$$

The direction and length of the vector $P_4 - P_3$ are adjusted. And it increases the endpoint-direction flexibility, and improves the success rate and robustness of path planning.

Moreover, P_2 is the midpoint of P_1 and P_3 connections [40], that is

$$\begin{aligned} P_2 &= \left[\frac{P_{11} + P_{31}}{2}, \frac{P_{12} + P_{32}}{2} \right]^T \\ &= \left[\frac{P_{01} + G_{11}l + P_{41} - G_{41}\alpha}{2}, \frac{P_{02} + G_{12}l + P_{42} - G_{42}\beta}{2} \right]^T \end{aligned} \quad (18)$$

As mentioned above, after obtaining the control points $P_0 \sim P_4$, a smooth LOAP described by a piecewise quadratic Bezier curve is built.

Proposition: The LOAP described based on the piecewise quadratic Bezier curve is continuous and smooth.

3.3. Robust planning of paths

As described above, the optimal passable area has been obtained, and the path description is completed by using of the piecewise quadratic Bezier curve. In this section, the objective function of the shortest path and the smallest target-direction change is constructed, and constraints of the variable margin and curvatures are given. The SQP algorithm is used to solve the optimal path parameters.

3.3.1. Target Point Direction Margin Constraint

To improve robustness of the path planning, parameters α , β are introduced so that the original target direction has a certain margin, and the constraint is as follows

$$\left| \frac{\left(\frac{G_{22}\beta}{G_{21}\alpha} - \frac{G_{22}}{G_{21}} \right)}{\frac{G_{22}}{G_{21}}} \right| \leq \delta \quad (19)$$

If $\frac{G_{22}}{G_{21}} > 0$, it is obtained

$$\frac{G_{22}}{G_{21}}(1-\delta) \leq \frac{G_{22}\beta}{G_{21}\alpha} \leq (1+\delta)\frac{G_{22}}{G_{21}} \quad (20)$$

If $\frac{G_{22}}{G_{21}} < 0$, we have

$$\frac{G_{22}}{G_{21}}(1+\delta) \leq \frac{G_{22}\beta}{G_{21}\alpha} \leq (1-\delta)\frac{G_{22}}{G_{21}} \quad (21)$$

Therefore, the margin constraint of the target direction is concluded that

$$(1-\delta) \leq \frac{\beta}{\alpha} \leq (1+\delta) \quad \delta \in [0,1] \quad (22)$$

Remark 2: In order to avoid division by zero, in practical applications, let $\frac{G_{22}}{G_{21}} = \frac{G_{22}}{G_{21}} + \varepsilon$, $G_{21} = G_{21} + \varepsilon$, $\varepsilon > 0$ approaches zero.

3.3.2. Curvature constraint

The curvature for any point on a curve is:

$$k = (x'y'' - y'x'') / ((x')^2 + (y')^2)^{\frac{3}{2}} \quad (23)$$

From equation (24), we can get a new upper limit of the curve curvature, i.e.

$$\begin{aligned} \|k\| &= \left\| \frac{x'y'' - y'x''}{((x')^2 + (y')^2)^{\frac{3}{2}}} \right\| \leq \left\| \frac{(x', y')}{\sqrt{(x')^2 + (y')^2}} \right\| \cdot \left\| \frac{(-x'', y'')}{((x')^2 + (y')^2)} \right\| \\ &= \frac{\sqrt{(-x'')^2 + (y'')^2}}{(x')^2 + (y')^2} \end{aligned} \quad (24)$$

From equation (24), it can be known that the denominator of the rightmost equation is a quadratic function about t , and its minimum value is expressed as $D(l, \alpha, \beta) > 0$, whose numerator is a constant about the parameter l, α, β to be optimized, denoted as $F(l, \alpha, \beta)$. Thus, the upper limit of $\|k\|$ is

$$k_{\max} = \frac{F(l, \alpha, \beta)}{D(l, \alpha, \beta)} \quad (25)$$

Then, the upper bound of the curvature is obtained, that is,

$$k_{\max} \leq 1/r_{\min} \quad (26)$$

3.3.3 Path optimization

In this section, in order to solve the unknown parameters of the Bezier curves, an objective function is given as shown in (27), which not only requires the shortest path length, but also requires the smallest change in the path target direction.

$$\begin{aligned} J(l, \alpha, \beta, \delta) \\ = \int_0^1 (\sqrt{B_1(t, l, \alpha, \beta)} + \sqrt{B_2(t, l, \alpha, \beta)}) dt + \omega \delta \end{aligned} \quad (27)$$

It can be seen from (27) that the optimization objective is a function of parameters α, β, δ . And ω is a weight parameter. The main purpose of this objective function is to encourage the minimization of steering maneuvers and driving risks for the robot.

To sum up, an objective optimization problem with constraints is given by equation (28).

$$\begin{aligned} \text{minimize: } & J(l, \alpha, \beta, \delta) \\ \text{s.t. } & k_{\max} \leq \frac{1}{r_{\min}}, l \geq 0, \alpha \geq 0, \beta \geq 0, \\ & (1 - \delta) \leq \frac{\beta}{\alpha} \leq (1 + \delta), 0 \leq \delta \leq 1 \end{aligned} \quad (28)$$

In this paper, the SQP algorithm is used to solve the problem of path parameters, and an optimal obstacle avoidance path with the shortest length and the smallest change in the target direction is obtained, which satisfies the kinematic constraints of the robot.

3.4 Path planning monitoring strategy

The previous section discusses the problem of how to plan an obstacle avoidance path. However, when applying obstacle avoidance algorithms to the obstacle avoidance process of the robot, the issue of when to plan a new path needs to be addressed. Therefore, this section proposes a path monitoring strategy that decides whether to re-plan the obstacle avoidance path.

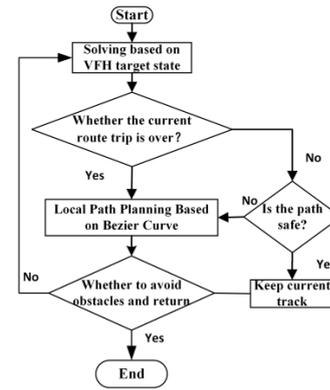


Fig. 8. Path planning monitoring strategy.

The path planning monitoring strategy, as shown in Figure 8, consists of a planning mechanism and a safety detection mechanism. It determines whether to re-plan the LOAP by checking whether the robot has reached the previously planned LOAP target point. In addition, it determines whether to re-plan the LOAP by examining whether the environment around the detected target point has significantly changed due to the robot's movement. As the robot moves and new obstacles are detected, the previously planned LOAP may no longer be safe. Therefore, this section provides an angle change threshold θ_T and a distance change threshold d_T between the real-time planning target point and the current LOAP target point. As mentioned earlier, according to the LOAP planning strategy mentioned, the changes of the target point indicate differences of the surrounding environment. Re-planning conditions of the LOAP are shown in (29).

$$\Delta d_T \geq d_T \ \& \ \Delta \theta_T \geq \theta_T \quad (29)$$

And Δd_T and $\Delta \theta_T$ are defined by using equation (30), i.e.

$$\begin{cases} \Delta d_T = \|P'_k - P'_{k-1}\| \\ \Delta \theta_T = \arccos\left(\frac{[P'_k - P'_c]^T \cdot [P'_{k-1} - P'_c]}{\|P'_k - P'_c\| \cdot \|P'_{k-1} - P'_c\|}\right) \end{cases} \quad (30)$$

where P_c is the current position of the robot.

And if P_c and the target point satisfy equation (29), the current planned obstacle avoidance path will be not safe, and it will be re-planned.

4. EXPERIMENTAL VERIFICATION

Under the same obstacle scenario and conditions, the robust path planning algorithm (RPPA) proposed in this paper compares the obstacle avoidance effect with the Frenet-coordinate system-based polynomial planning algorithm (PAFS) (Wei et al., 2021) and improved VFH algorithm (Qu et al., 2015). The 10 obstacle avoidance scenarios in Fig 9 include open scenarios, narrow passages, and multi-obstacle scenarios, which can satisfy the effect verification of the obstacle avoidance algorithm in driving scenarios with structured, unstructured, and irregular boundary constraints in a park. In addition, this paper also designs irregular boundary scene experiments to further verify the effectiveness of the algorithm proposed in this paper. Finally, experiments of a real robot are conducted on a campus to verify the algorithm.

The computer CPU used in the experiment is Intel Core i5-5200U, and the memory is 8GB. The algorithm is performed in Matlab2018b under Windows 10 (64-bit). The three planning algorithms are run in MATLAB, and the specific parameters are shown in Table 1.

Table 1. Experimental parameters.

Parameter	RPPA	PAFS	VFH
r_{\min}	1.5m	1.5m	1.5m
d_{\max}	20m	—	20m
d_{in}^{\max}	20m	—	20m
d_{in}^{\min}	16m	—	16m
ΔT	0.3s	0.3s	0.3s
\bar{w}_r	—	8m	—
\underline{w}_r	—	-8m	—
Δw_r	—	1m	—
v	10km/h	10km/h	10km/h

In Table 1, \bar{w}_r represents the maximum path width, \underline{w}_r represents the maximum path width, \underline{w}_r represents the minimum path width, Δw_r represents a sampling length of the path width.

4.1 Planning success rate comparison

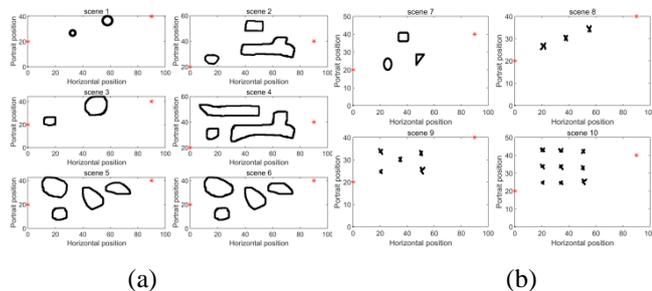


Fig. 9. Obstacle avoidance experiment scenes.

In this paper, a comparative simulation experiment is established to verify the obstacle avoidance effect of RPPA in a complex environment. In this experiment, the starting point is (0, 20), the endpoint is (90, 40), and a total of 10 scenarios with different obstacles are set, as shown in Fig.9. Three

algorithms are used for obstacle avoidance planning, and the planning success rate is shown in Table 2.

Table 2. The success rate of a path planning algorithm.

	RPPA	PAFS	VFH
Number of success	10	7	9
Number of failures	0	3	1
Total	10	10	10

As shown in Fig.10, four scenarios are taken to display the effect of the planned path. The RPPA path is in blue. The PAFS path is pink. The VFH path is in red.

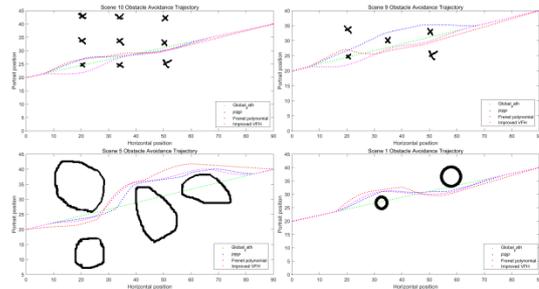


Fig. 10. Schematic diagram of an obstacle avoidance path.

The PAFS path uses the global path as the reference to generate the path, and the generated path is constrained by the global path. The algorithm performed poorly in obstacle avoidance planning in the complex environments, and successfully planned 7 times in 10 scenarios; The VFH for the robot motion planning problem adopts a "dual threshold" strategy based on perception data, which alleviates the VFH threshold sensitivity problem to a certain extent. However, for more complex obstacle scenarios, the fixed threshold cannot adapt to all obstacle avoidance scenarios, so there are planning failures. Therefore, the planning fails in scenario 4, and the planning is successful 9 times; Due to the setting of degrees of freedom at the target point, the method proposed in this paper adopts a candidate passable area screening method based on the adaptive threshold and the narrowest robot passage constraint, which makes the selection of passable directions and the planning of paths more flexible. Therefore, the RPPA has a higher planning success rate for a more complex external environment and successfully avoids obstacles in 10 scenarios.

4.2 Time cost comparison

In the scenarios of 10 different obstacle environments, the path planning cost time comparison with the same starting and ending points is shown in Fig.11. The cost time is the overall time spent for route planning from the start point to the endpoint, in seconds. As shown in Fig.11, the RPPA has the smallest time cost compared with the other two algorithms.

The VFH for the robot generates path points according to the optimal direction in each control cycle and finally obtains the obstacle avoidance path. Therefore, all planning processes need to be executed in each control cycle. In the same way, the PAFS re-plans an obstacle avoidance path in each control cycle. Therefore, the cost time of the two algorithms is relatively long. The RPPA proposes a path monitoring

decision, which re-plans a new path when the current path is unsafe. The cost time of the proposed algorithm is small.

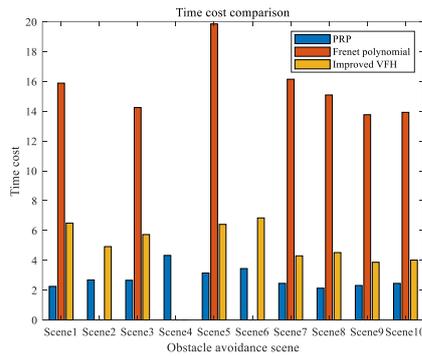


Fig. 11. Cost time of obstacle avoidance planning.

4.3 Path length comparison

The path lengths of the RPPA, the PAFS, and the VFH are calculated in 10 scenarios under the same conditions of starting and ending points and obstacles. Simulations are carried out at a low and constant speed. And results are shown in Fig. 12.

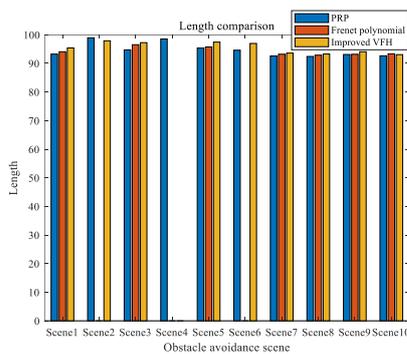


Fig. 12. Lengths of obstacle avoidance paths.

Compared with the other two algorithms, the path length of the RPPA is the shortest. Since the RPPA optimizes the minimum path when generating the path, and sets the curvature constraint according to the minimum turning radius. And the path is the shortest while ensuring that the path is feasible. Compared with the other two algorithms, the path is shorter.

4.4 Path heading angle comparison

In this experiment, the local path is discretized according to a fixed time scale, and the heading angle of each waypoint is obtained. In this paper, stability and smoothness of the obstacle avoidance path are evaluated by comparing the mean square error of the path heading angle.

Fig.13 shows the mean square error of the path heading angles of the three algorithms for obstacle avoidance planning in 10 scenarios. As shown in Fig.13, there are 7 scenarios in which the three algorithms are successfully planned, and there are 4 scenarios in which the mean square error and amplitude of the path heading angle of the RPPA are the smallest. Therefore, the RPPA is relatively stable.

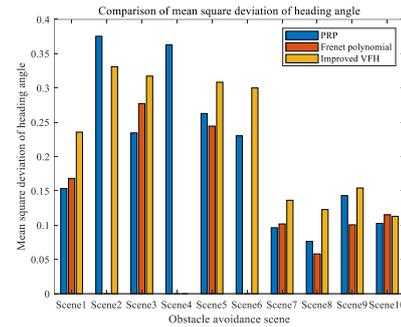


Fig. 13. Mean square errors of heading angle of an obstacle avoidance path.

4.5 Complex boundary scene simulation experiment

This paper simulates a typical scene of the robot in a park. The obstacle avoidance process is shown in Fig.14. The green curve is a known global path, and the red curve is the boundary of the current running scene. And they are obtained from a constructed electronic map. The blue curve is the planned path, and the black curve is an obstacle.

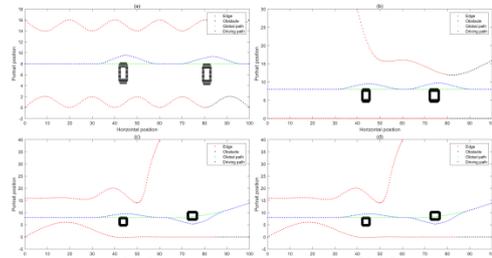


Fig. 14. Obstacle avoidance effect in irregular boundary scenes.

Fig.14(a) shows the uneven roadside scene in the park. Fig.14(b) is a scene where the park enters a narrow environment from a spacious environment. Fig.14(c) is a scene in the park from a narrow environment into a spacious environment. Fig.14(d) is the scene of the "S"-shaped path in the park. In the four typical park scenarios, the robot successfully avoids obstacles and reaches the endpoint. Experiments show that the RPPA can adapt to various complex boundary constraints in the park, and has high robustness.

4.6 Obstacle avoidance experiments of a real robot

The scene of the obstacle avoidance is a school square, where waypoints are collected around a huge flower bed to map out the global path. The test of the real robot uses the Ackerman steering platform JD-01. The in-vehicle Hi-Target GNSS receiver is used for positioning and orientation of the robot, and the SICK16 line lidar is used for obstacle information detection.

First, collect path points around the flower bed, and use cubic polynomial fitting to get the global path, which is the green curve in the figure. The blue path is the running path of the robot, and the pink curve is the planned local path during the running procedure of the robot. The effect of the robot bypassing the flower bed is shown in Fig.15 and Fig. 16.

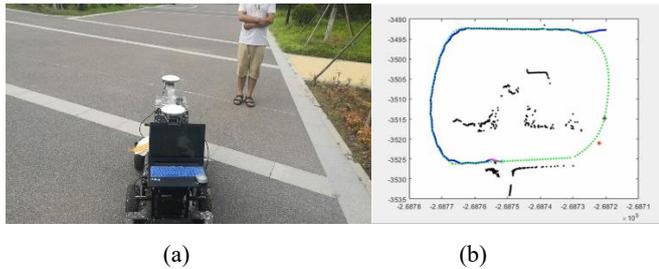


Fig. 15. Obstacle avoidance scene 1 and experiment results.

In Fig.15(a), during the tracking of the global path, the robot encounters an obstacle avoidance. Fig.15(b) shows a real-time running information of the RPPA when the robot encounters obstacles. As shown in the figure, the robot senses the pedestrian ahead successfully, and plans a local path to avoid obstacles.

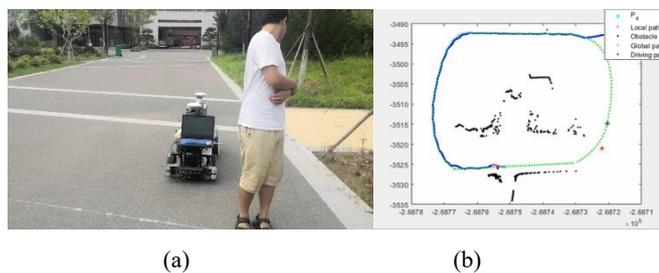


Fig. 16. Obstacle avoidance scene 2 and experiment results.

In Fig.16(a), during the tracking of the global path, the robot bypasses the obstacle and returns to the global path. Fig.16(b) shows the real-time running information of the algorithm when the robot bypasses obstacles and returns to the global path. As shown in the figure, the robot successfully plans a local path returning to the global path.

From the real robot experiment, it is obtained that the RPPA can make the robot successfully avoid obstacles and return to the global path.

In summary, compared with the other methods, advantages of the RPPA are as follows.

- (1) The planning method is more flexible and robust, and it can deal with the problem of more complex obstacle avoidance. And the planning success rate is higher.
- (2) The obstacle avoidance path is safe and the overall planning time cost is less.
- (3) The obstacle avoidance path satisfies the kinematic constraints of the robot, the length is smaller and the stability is better.

5. CONCLUSIONS

In this paper, a robust path planning method based on SQP algorithm is proposed for obstacle avoidance. Firstly, a strategy for screening candidate passable areas is proposed, which improves the accuracy and practicability of the VFH. And the method to determine the target state is given based on the improved VFH. Secondly, a segmented quadratic Bezier curve is used to describe the path used for obstacle avoidance, and its control points and segment points are determined. Then, the problem of curve parameter

optimization is established based on the curvature constraint and the direction change margin constraint of the target point. The SQP algorithm is used to optimize the curve parameters. Furthermore, a path monitoring strategy is proposed, and monitors the security of the planned path in real-time. And it improves flexibility and robustness of path planning.

Compared with other methods (Qu et al., 2015; Wei et al., 2021), this method has better robustness in the face of complex unknown obstacle distribution environment, and the path length is the shortest. Simulation experiments verify the effectiveness and practicability of the algorithm. And the method can successfully complete an obstacle avoidance task when a robot encounters unknown obstacles in a park.

However, the parameter optimization time of the RPPA is still longer. In the future work, the optimization method will be improved. And description ability of the piecewise quadratic curve will be enhanced to improve robustness of the proposed algorithm.

ACKNOWLEDGEMENTS

This work was supported by was supported by the Major Science and Technology Innovation Project of Shandong Province (No.2020CXGC010110) and Shandong Provincial Natural Science Foundation (No. ZR2022MF345). This work was supported by Support Plan for Shandong Key Laboratory of Transportation Industry. This work was supported by Key Science and Technology Projects of the Ministry of Transport (No. 2022-MS6-159)

REFERENCES

- Babinec, A., Duchoň, F., Dekan, M., Pászto, P., & Kelemen, M. (2014). VFH* TDT (VFH* with Time Dependent Tree): A new laser rangefinder based obstacle avoidance method designed for environment with non-static obstacles. *Robotics and autonomous systems*, 62(8), 1098-1115.
- Berglund, T., Brodnik, A., Jonsson, H., Staffanson, M., & Soderkvist, I. (2009). Planning smooth and obstacle-avoiding B-spline paths for autonomous mining vehicles. *IEEE transactions on automation science and engineering*, 7(1), 167-172.
- Chen, J., Zhang, R., Han, W., Jiang, W., Hu, J., Lu, X., ... & Zhao, P. (2020). Path planning for autonomous vehicle based on a Two-Layered planning model in complex environment. *Journal of Advanced Transportation*, 2020, 1-14.
- Chen, C., He, Y. Q., Bu, C. G., & Han, J. (2015). Feasible trajectory generation for autonomous vehicles based on quartic Bézier curve. *Acta Automatica Sinica*, 41(3), 486-496.
- Chen, J., Zhao, P., Mei, T., & Liang, H. (2013, July). Lane change path planning based on piecewise bezier curve for autonomous vehicle. In *Proceedings of 2013 IEEE International Conference on Vehicular Electronics and Safety* (pp. 17-22). IEEE.
- Chen, L., Qin, D., Xu, X., Cai, Y., & Xie, J. (2019). A path and velocity planning method for lane changing collision avoidance of intelligent vehicle based on cubic

- 3-D Bezier curve. *Advances in Engineering Software*, 132, 65-73.
- Choe, T. S., Park, J. B., Joo, S. H., & Park, Y. W. (2014). 1D virtual force field algorithm for reflexive local path planning of mobile robots. *Electronics Letters*, 50(20), 1429-1430.
- Choi, J. W., Curry, R. E., & Elkaim, G. H. (2012). Minimizing the maximum curvature of quadratic Bézier curves with a tetragonal concave polygonal boundary constraint. *Computer-Aided Design*, 44(4), 311-319.
- Dakulović, M., & Petrović, I. (2011). Two-way D* algorithm for path planning and replanning. *Robotics and autonomous systems*, 59(5), 329-342.
- Deng, X., Li, R., Zhao, L., Wang, K., & Gui, X. (2021). Multi-obstacle path planning and optimization for mobile robot. *Expert Systems with Applications*, 183, 115445.
- Ding, W., Zhang, L., Chen, J., & Shen, S. (2019). Safe trajectory generation for complex urban environments using spatio-temporal semantic corridor. *IEEE Robotics and Automation Letters*, 4(3), 2997-3004.
- Duhé, J. F., Victor, S., & Melchior, P. (2021). Contributions on artificial potential field method for effective obstacle avoidance. *Fractional Calculus and Applied Analysis*, 24, 421-446.
- Erke, S., Bin, D., Yiming, N., Qi, Z., Liang, X., & Dawei, Z. (2020). An improved A-Star based path planning algorithm for autonomous land vehicles. *International Journal of Advanced Robotic Systems*, 17(5), 1729881420962263.
- Elbanhawi, M., Simic, M., & Jazar, R. N. (2015). Continuous path smoothing for car-like robots using B-spline curves. *Journal of Intelligent & Robotic Systems*, 80, 23-56.
- Elbanhawi, M., Simic, M., & Jazar, R. (2015). Randomized bidirectional B-spline parameterization motion planning. *IEEE Transactions on intelligent transportation systems*, 17(2), 406-419.
- Glaser, S., Vanholme, B., Mammar, S., Gruyer, D., & Nouveliere, L. (2010). Maneuver-based trajectory planning for highly autonomous vehicles on real road with traffic and driver interaction. *IEEE Transactions on intelligent transportation systems*, 11(3), 589-606.
- González, D., Pérez, J., Milanés, V., & Nashashibi, F. (2015). A review of motion planning techniques for automated vehicles. *IEEE Transactions on intelligent transportation systems*, 17(4), 1135-1145.
- Guo, Y., Liu, H., Fan, X., & Lyu, W. (2021). Research progress of path planning methods for autonomous underwater vehicle. *Mathematical Problems in Engineering*, 2021, 1-25.
- Hao, K., Zhao, J., Yu, K., Li, C., & Wang, C. (2020). Path planning of mobile robots based on a multi-population migration genetic algorithm. *Sensors*, 20(20), 5873.
- Ji, X., Feng, S., Han, Q., Yin, H., & Yu, S. (2021). Improvement and fusion of A* algorithm and dynamic window approach considering complex environmental information. *Arabian Journal for Science and Engineering*, 46, 7445-7459.
- Kumar, P. B., Rawat, H., & Parhi, D. R. (2019). Path planning of humanoids based on artificial potential field method in unknown environments. *Expert Systems*, 36(2), e12360.
- Lattarulo, R., González, L., Martí, E., Matute, J., Marcano, M., & Pérez, J. (2018). Urban motion planning framework based on n-bézier curves considering comfort and safety. *Journal of Advanced Transportation*. doi:10.1155/2018/6060924.
- Lian, J., Yu, W., Xiao, K., & Liu, W. (2020). Cubic spline interpolation-based robot path planning using a chaotic adaptive particle swarm optimization algorithm. *Mathematical problems in engineering*, 2020, 1-20.
- Liu, L. S., Lin, J. F., Yao, J. X., He, D. W., Zheng, J. S., Huang, J., & Shi, P. (2021). Path planning for smart car based on Dijkstra algorithm and dynamic window approach. *Wireless Communications and Mobile Computing*, 2021, 1-12.
- Liu, Z. Q., Zhang, T., & Wang, Y. F. (2019). Research on local dynamic path planning method for intelligent vehicle lane-changing. *Journal of Advanced Transportation*, 2019, 1-10.
- Li, W., Tan, M., Wang, L., & Wang, Q. (2020). A cubic spline method combing improved particle swarm optimization for robot path planning in dynamic uncertain environment. *International Journal of Advanced Robotic Systems*, 17(1), 1729881419891661.
- Li, Y., Wei, W., Gao, Y., Wang, D., & Fan, Z. (2020). PQ-RRT*: An improved path planning algorithm for mobile robots. *Expert systems with applications*, 152, 113425.
- McGuire, K. N., de Croon, G. C., & Tuyls, K. (2019). A comparative study of bug algorithms for robot navigation. *Robotics and Autonomous Systems*, 121, 103261.
- Mohanta, J. C., & Keshari, A. (2019). A knowledge based fuzzy-probabilistic roadmap method for mobile robot navigation. *Applied Soft Computing*, 79, 391-409.
- Murphy, R. R. (2019). Introduction to AI robotics. *MIT press*.
- Park, S. O., Lee, M. C., & Kim, J. (2020). Trajectory planning with collision avoidance for redundant robots using jacobian and artificial potential field-based real-time inverse kinematics. *International Journal of Control, Automation and Systems*, 18(8), 2095-2107.
- Petrov, P., & Nashashibi, F. (2014). Modeling and nonlinear adaptive control for autonomous vehicle overtaking. *IEEE Transactions on Intelligent Transportation Systems*, 15(4), 1643-1656.
- Qu, P., Xue, J., Ma, L., & Ma, C. (2015). A constrained VFH algorithm for motion planning of autonomous vehicles. In *2015 IEEE Intelligent Vehicles Symposium (IV)* (pp. 700-705). IEEE.
- Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). Introduction to autonomous mobile robots. *MIT press*.
- Song, B., Wang, Z., & Zou, L. (2021). An improved PSO algorithm for smooth path planning of mobile robots using continuous high-degree Bezier curve. *Applied Soft Computing*, 100, 106960.

- Thanh, D. N., Le Van, H., & Tan, L. N. (2022). LiDAR-Based Online Navigation Algorithm For An Autonomous Agricultural Robot. *CONTROL ENGINEERING AND APPLIED INFORMATICS*, 24(2), 90-100.
- Wang, P., Gao, S., Li, L., Sun, B., & Cheng, S. (2019). Obstacle avoidance path planning design for autonomous driving vehicles based on an improved artificial potential field algorithm. *Energies*, 12(12), 2342.
- Wei, M., Teng, D., and Wu, S.(2021). Automatic driving trajectory planning and optimization algorithm based on Frenet coordinate system. *Control and Decision*, 36(4), 815-824.
- Xu, L., Cao, M., & Song, B. (2022). A new approach to smooth path planning of mobile robot based on quartic Bezier transition curve and improved PSO algorithm. *Neurocomputing*, 473, 98-106.
- Yue, M., Wu, X., Guo, L., & Gao, J. (2019). Quintic polynomial-based obstacle avoidance trajectory planning and tracking control framework for tractor-trailer system. *International Journal of Control, Automation and Systems*, 17(10), 2634-2646.
- Zeng, D., Yu, Z., Xiong, L., Zhao, J., Zhang, P., Li, Z., & Zhou, Y. (2019, June). A novel robust lane change trajectory planning method for autonomous vehicle. In 2019 *IEEE Intelligent Vehicles Symposium (IV)* (pp. 486-493). IEEE.
- Zheng, L., Zeng, P., Yang, W., Li, Y., & Zhan, Z. (2020). Bézier curve-based trajectory planning for autonomous vehicles with collision avoidance. *IET Intelligent Transport Systems*, 14(13), 1882-1891.
- Zhou, C., Huang, B., & Fränti, P. (2022). A review of motion planning algorithms for intelligent robots. *Journal of Intelligent Manufacturing*, 33(2), 387-424.
- Zhou, Z., Wang, J., Zhu, Z., Yang, D., & Wu, J. (2018). Tangent navigated robot path planning strategy using particle swarm optimized artificial potential field. *Optik*, 158, 639-651.
- Zhuang, Y., Zhao, C., and Yan, H. (2018). An improved strategy for VFH threshold sensitive problem. *J Sichuan Univ: Nat Sci Ed*, 55(5),985-992.