Research Article

Improved Bidirectional RRT * Path Planning Method for Smart Vehicle

Qingying Ge ¹, **Aijuan Li** ^{1,2}, **Shaohua Li** ³, **Haiping Du** ⁴, **Xin Huang** ⁵,
and **Chuanhu Niu** ¹

¹School of Automotive Engineering, Shandong Jiaotong University, Jinan 250357, China

²Energy and Power Engineering College, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China

³State Key Laboratory of Mechanical Behavior and System Safety of Traffic Engineering Structures, Shijiazhuang Tiedao University, Shijiazhuang 050043, China

⁴Department of Electrical, Computer & Telecom Engineering, University of Wollongong, Wollongong, New South Wales 2500, Australia

⁵School of Information Science and Electrical Engineering, Shandong Jiaotong University, Jinan 250357, China

Correspondence should be addressed to Aijuan Li; 65826990@qq.com, Shaohua Li; lishaohua@stdu.edu.cn, and Xin Huang; huangzinuo@163.com

Received 28 November 2020; Revised 9 February 2021; Accepted 11 May 2021; Published 27 May 2021

Academic Editor: Yuanchang Liu

Copyright © 2021 Qingying Ge et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, an improved bidirectional RRT * vehicle path planning method for smart vehicle is proposed. In this method, the resultant force of the artificial potential field is used to determine the search direction to improve the search efficiency. Different kinds of constraints are considered in the method, including the vehicle constraints and the vehicle driving environment constraints. The collision detection based on separating axis theorem is used to detect the collision between the vehicle and the obstacles to improve the planning efficiency. The cubic B-spline curve is used to optimize the path to make the path's curvature continuous. Both simulation and experiment are implemented to verify the proposed improved bidirectional RRT * method. In the simulation analysis, this paper's method can generate the smoothest path and takes the shortest time compared with the other two methods and it can be adaptive to the complicated environment. In the real vehicle experiment, we can see from the test results that this paper's method can be applied in practice on the smart electric vehicle platform; compared with others' algorithm, this paper's algorithm can generate shortest and smoothest path.

1. Introduction

Smart vehicle plays an important role in the smart transportation systems. Especially, the path planning problem is an important component of the planning and control system [1]. The path planning refers to planning a path that does not collide with obstacles when the starting position, the ending position of vehicles, and the distribution of obstacles in the environment are known. The path parameters provided by the path planning method are supplied to the controller to control the vehicle drive along the planned path accurately [2].

In recent years, scholars have done a lot of research on path planning algorithms, and new path planning algorithms are constantly emerging and developing [3]. The most

representative and common path planning algorithms in the field are mainly divided into map-based path planning algorithms [4], sampling-based search path planning algorithms [5], interpolation method [6], and artificial potential field method [7]. Sampling-based search path planning algorithm includes probability map algorithms and rapidly-exploring random tree (RRT) algorithms [8, 9]. The RRT algorithm is widely used because of the advantage that it is suitable to solve the path planning problem under dynamic and multiobstacle conditions. However, the basic RRT algorithm has the disadvantage that the convergence speed and the search efficiency are low and the planned path is difficult to be the optimum [10, 11]. Regarding the disadvantage of the RRT algorithm, researchers have carried out a lot of

improvements. The typical improvements are improving the RRT's search style and fusing the RRT with other algorithms effectively. Thus, the generated path is optimized again and the planning effect is further improved [12]. Among them, improving the RRT's search mode is an important improved direction. Researchers conduct target-biased search on RRT to generate an extended mode of nonrandom sampling, thus improving the planning efficiency [13]. At the same time, rapid expansion was carried out to generate PQ-RRT * [14] and bidirectional expansion RRT * [15], and then path optimization was carried out on the existing planning results. Based on the deviation research, the path is optimized using Bezier curves in the literature [16], and the optimized path can meet the requirements of vehicle safety. Considering the vehicle constraints, the quartic spline curve is used to provide a smooth dynamic feasible path to make the vehicle drive safely in the literature [17]. Shi et al. generated new nodes at the same time through cyclic iterative search and bidirectional search and solved the problems of slow convergence speed and large deviation by adding turning constraints and deleting redundant nodes [18]. Yang combined the RRT of biased target position with greedy RRT and then adopted the anytime strategy to improve the calculation time and path quality effectively [19]. For the dynamic driving environment full of uncertainty, Kuwata et al. mainly used the methods of biased heuristic sampling and safety detection to face the situation of uncertainty and incomplete perception conditions [20]. When faced with a large dynamic environment, Fulgenzi et al. [21] calculated the corresponding motion probability by trajectory prediction of objects moving in the driving environment, so as to carry out the dynamic motion planning of the vehicles. However, the accuracy of obstacle motion prediction needs to be improved.

For the operating environment closer to the actual road scene, Bai et al. used model predictive control for partial connection and collaborative motion planning in an automated environment and used a numerical method based on dynamic programming to solve the problem. This method effectively reduces the impact caused by lane changes. Vibration phenomenon and reduced time-consuming have great potential in real-time applications [22]; Zhanbo et al. proposed a solution method based on two-layer dynamic programming based on the mixed scene where human drivers and autonomous vehicles coexist in the ramp scene, which has strong versatility and lays the foundation for traffic analysis in a mixed traffic environment [23]; Wang et al. carried out a coordinated traffic organization method in multi-intersection environment based on networked automated vehicles. In order to adapt to dynamic traffic demands, they proposed a complex path planning strategy, which could effectively improve traffic efficiency, but did not consider the influence of information delay caused by communication uncertainty [24]. Rahmati et al. considered networked autonomous vehicles and driving scenarios with high human participation and established a human-vehicle interactive decision-making framework to achieve human-vehicle coexistence, but did not combine behavior prediction and intelligent body motion planning to develop planning algorithms [25].

In this paper, an improved path planning method based on the bidirectional RRT * is investigated. The improved path planning method has many advantages. Firstly, this method can reduce the time of planning path and improve the planning efficiency, because the artificial potential field method is applied to establish a mechanism of extension, which makes the random tree's extension have more intentionality; secondly, the vehicle's constraints and the obstacle environment constraints are considered so that the method can be applied to the environment with complicated dynamic obstacles; thirdly, the theory-based SAT is used to detect the collision between the vehicle and the obstacles to make the vehicle drive more safely. At last, the cubic B-spline curve is used to optimize the generated path to make the curvature of the path continuous and the vehicle can drive smoothly and steadily. The remainder of the paper is organized as follows: Section 2 presents the proposed vehicle dynamic model; Section 3 discusses the path planning method based on the improved RRT *; the simulation and experimental results are presented in Section 4; finally, Section 5 presents the conclusions of this research.

2. The Vehicle Model

RRT is an incremental path planning algorithm in which the new state $(q_{\text{near}}, u, \Delta t)$ function can use a dynamic model to limit the vehicle's state. In this paper, we use a dynamic vehicle model. The dynamic model of the smart vehicle is shown in Figure 1.

This model is simplified by projecting the front and the rear wheels on two virtual wheels located at the middle of the vehicle. On the premise of the bicycle model assumption, the following assumptions are made: only the characteristics of the purely lateral tires are considered, and the longitudinal and lateral coupling relationships of the tire forces are ignored; the left and right shifts of the load are not considered; the transverse longitudinal aerodynamics is ignored. The state of the vehicle is defined by $X = (x_g, y_g, \theta, v_y, \omega)^T$, where x_g and y_g represent the coordinates of the vehicle gravity center, θ is its orientation, v_y is the lateral speed, and ω is the yaw rate.

As shown in Figure 1, L_f and L_r are the distance from the center of mass to the front and rear axle, respectively, δ_f is the front wheel angle, and f are the forces in X and Y directions of the front wheel, respectively. Using the fundamental law of dynamics, we can get the dynamic formula, which is

$$\begin{cases} m(\dot{v}_x - v_y \dot{\omega}) = -F_{xf} \cos \delta_f - F_{yf} \sin \delta_f - F_{xr}, \\ m(\dot{v}_y + v_x \dot{\omega}) = F_{yf} \cos \delta_f - F_{xf} \sin \delta_f + F_{yr}, \\ I_z \dot{\omega} = L_f (F_{yf} \cos \delta_f - F_{xf} \sin \delta_f) - L_r F_{yr}. \end{cases} \quad (1)$$

A constant longitudinal speed v_x is chosen for the vehicle model. Furthermore, as the aerodynamic resistance is neglected, the longitudinal tire force F_{xf} becomes zero. Considering this, we get the simplified dynamic formula, which is

$$\begin{cases} \dot{y}_y = \frac{F_{yf}}{m} \cos \delta_f + \frac{F_{yr}}{m} - v_x \omega, \\ \dot{\omega} = \frac{L_f}{I_z} F_{yf} \cos \delta_f - \frac{L_r}{I_z} F_{yr}. \end{cases} \quad (2)$$

Thus, as a linear tire model is used, we can have

$$\begin{aligned} F_{yf} &= -C_{\alpha f} \alpha_f, \\ F_{yr} &= -C_{\alpha r} \alpha_r, \end{aligned} \quad (3)$$

where $C_{\alpha f}$ and $C_{\alpha r}$ are the cornering stiffness coefficients for the front and rear tires, respectively and α_f and α_r are the slip angles for the front and rear wheels, respectively. These angles are assumed to be small. So, we can obtain

$$\begin{aligned} \alpha_f &= \frac{v_y + L_f \omega}{v_x} - \delta_f, \\ \alpha_r &= \frac{v_y - L_r \omega}{v_x}. \end{aligned} \quad (4)$$

According to this, we obtain the full nonlinear model, which is

$$\begin{bmatrix} \dot{y}_y \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} A & C \\ B & D \end{bmatrix} \begin{bmatrix} v_y \\ \omega \end{bmatrix} + \begin{bmatrix} E \\ F \end{bmatrix} \delta_f, \quad (5)$$

where

$$\begin{aligned} A &= -\frac{C_{\alpha f} \cos \delta_f + C_{\alpha r}}{m v_x}, \\ B &= \frac{-L_f C_{\alpha f} \cos \delta_f + L_r C_{\alpha r}}{m v_x} - v_x, \\ C &= \frac{-L_f C_{\alpha f} \cos \delta_f + L_r C_{\alpha r}}{I_z v_x}, \\ D &= -\frac{L_f^2 C_{\alpha f} \cos \delta_f + L_r^2 C_{\alpha r}}{I_z v_x}, \\ E &= \frac{C_{\alpha f} \cos \delta_f}{m}, \\ F &= \frac{L_f C_{\alpha f} \cos \delta_f}{I_z}. \end{aligned} \quad (6)$$

By integrating, we can compute the new coordinates (x_g, y_g, θ) of the vehicle, which is

$$\begin{cases} \dot{x}_g = v_x \cos(\theta) - v_y \sin(\theta), \\ \dot{y}_g = v_x \sin(\theta) + v_y \cos(\theta), \\ \dot{\theta} = \omega. \end{cases} \quad (7)$$

We now have nonlinear vehicle model $\dot{x} = f(x, u)$, starting from a given configuration $q_{\text{near}} = x(t)$. We integrate our system (7) using a given input u to obtain $q_{\text{new}} = x(t + \Delta t)$. The configuration will be added to the tree if it lies in free.

FIGURE 1: Simplified vehicle dynamics model.

3. The Improved RRT * Path Planning Method

3.1. The Basic RRT * Algorithm. RRT * is a kind of advanced gradually optimal sampling method based on the RRT algorithm, in which the selected mode of the parent node is optimized. The pseudocode of the RRT * is shown in Algorithm 1. The map environment M is input and the starting point q_{init} and target point q_{goal} are determined. Taking the starting point q_{init} as the root node of the obtained path G , the random point q_{rand} is generated. Finding the nearest node q_{near} from the random point q_{rand} on the tree, the two points q_{rand} and q_{near} are connected. Taking q_{rand} as the center and r as the radius, the nodes are searched on the tree and the potential parent node q_{parent} is found. The cost to be the parent node q_{parent} is calculated, and it is compared with the cost of the original path. If the cost of the parent node q_{parent} is smaller than the original path, then the collision detection is performed. Otherwise, the next potential parent node is searched until the collision detection is passed. The previous edge is deleted from the tree and the new edge is added. Iterative selection of parent nodes is continuously performed. If a more suitable parent node is selected, the nodes on the existing tree are reconnected, and the original connection is removed, so as to ensure that the progressive optimal solution is obtained.

3.2. The Improvement Strategy. The improvement strategy flow chart of this paper is shown in Figure 2. First, determine the vehicle model and driving environment, as well as the vehicle's starting point and target point location, and then proceed with the vehicle path planning and design on this basis. In the sampling process, the sampling method is improved based on the idea of artificial potential field, the sampling nodes are reduced by considering the constraints of the vehicle itself and the driving environment, and the separation axis collision theory is used to detect whether the generating node collides with obstacles; finally, the whole tree is backtracked to generate the feasible path of the vehicle which is smoothed by using the cubic B-spline curve.

ALGORITHM 1: RRT * pseudocode.

3.2.1. Random Sampling Improvement Method Based on the Artificial Potential Field. According to the wide search space, the random mode is used to do the node extension when the original RRT and RRT * algorithms are sampled. Although the probability integrity can be guaranteed, the search efficacy of the algorithm is reduced to some extent. So, the applicability of the original RRT and RRT * algorithms are challenged for the smart vehicle which has high real-time requirement. According to the narrow search space, the deviation sample reduces the probability of obtaining the feasible path as quickly as possible. So, the RRT * algorithm should be improved to improve both the probability integrity and the search efficacy. The goal attraction thought of the artificial potential field is used to determine the search direction. The artificial potential field method path planning is a virtual force method proposed by Khatib. Its basic idea is to design the movement of the vehicle in the surrounding environment into an abstract artificial gravitational field. The target point produces “gravitation” to the vehicle, and the obstacle produces “repulsion” to the vehicle. Finally, the movement of vehicles is controlled by seeking the combined force. When the search range is larger than the length of the vehicle and the vehicle constraints are satisfied, a fast search method biased towards the target area is adopted to improve the efficiency of the search path and the safety performance of the vehicle. Using the idea of artificial potential field, according to the gravity function of the target position, sampling is conducted in the target direction; the formulation is defined as

$$F = F_{att} + F_{rep}. \quad (8)$$

Among them, F_{att} is the target gravity received by the intelligent vehicle, and F_{rep} is the comprehensive obstacle repulsion received by the target vehicle.

The target gravity F_{att} is specifically expressed as

$$F_{att}(M) = -\nabla U_{att}(M) = K \vec{a}_G \rho_g, \quad (9)$$

where $U_{att}(M) = 1/2K(M - M_g)^2$, $M = [x, y]^T$, $M_g = [x_g, y_g]^T$, and $F_{att}(M)$ is the gravity function; it can be obtained by differentiating the negative gradient of the gravitational field function; K is the gravitational field gain, $\vec{a}_G$ is the forward target, ρ_g is the distance between the controlled vehicle and

FIGURE 2: Improvement strategy flow chart.

the target position, M is the current position of the vehicle, and M_g is the target position [26].

Obstacle repulsion forces are based on static obstacles and dynamic obstacles to establish repulsion potential field functions and finally form a comprehensive obstacle repulsion force F_{rep} .

(1) Static Obstacle Restraint. The static obstacle function in the driving environment is expressed as the following formula:

$$U_{rj}(M) = \begin{cases} \frac{1}{2}K_r \left(\frac{1}{d_{ob}} - \frac{1}{d_0} \right)^2, & d_{ob} \leq d_0, \\ 0, & d_{ob} > d_0. \end{cases} \quad (10)$$

In the above formula, K_r is the static gain coefficient of the repulsive force field, d_{ob} is the distance between the controlled object and the obstacle, and d_0 is the boundary distance of the repulsive force set according to the vehicle speed.

Derive the negative gradient of the repulsion field function to obtain the static repulsion function $F_{rj}(M)$, which is the following formula:

$$F_{rj}(M) = \begin{cases} (F_{r1} + F_{r2})^2 \vec{a}_b, & d_{ob} \leq d_0, \\ 0, & d_{ob} > d_0, \end{cases} \quad (11)$$

Among them, $\begin{cases} F_{r1} = K_r [(1/d_{ob}) - (1/d_0)] (1/d_{ob}^2) \\ F_{r2} = (n/2) K_r [(1/d_{ob}) - (1/d_0)]^2 \end{cases}$ and $\vec{a}_b$ refers to the unit vector of the static obstacle pointing to the controlled vehicle.

$$U_{rm}(x, y) = \begin{cases} \frac{1}{2} \eta_s [V_e - V_b \cos(\theta - \beta)]^2 \rho(x, y) \vec{a}_b, & (\theta - \beta) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \\ 0, & \text{others.} \end{cases} \quad (12)$$

In the above formula, η_s is the speed potential field repulsion gain, V_e is the speed of the vehicle, V_b is the speed of the dynamic obstacle, θ is the angle between the direction of the controlled vehicle and the positive direction of the x -axis, and β is the positive direction of the dynamic obstacle and the x -axis. The direction is the angle; $\vec{a}_b$ is the unit

(2) *Dynamic Obstacle Restraint.* The dynamic obstacle function in the driving environment is expressed as the following formula:

vector of the dynamic obstacle pointing to the controlled vehicle.

When $(\theta - \beta) \notin [-(\pi/2), (\pi/2)]$, the dynamic obstacle loses its repulsive force to the controlled vehicle, so the repulsive force is 0 at this time.

Derive formula (12) to obtain dynamic obstacle repulsion, as shown in the following formula:

$$F_{rm} = \begin{cases} \eta_s [V_e - V_b \cos(\theta - \beta)] \rho(x, y) \vec{a}_b, & (\theta - \beta) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \\ 0, & \text{others.} \end{cases} \quad (13)$$

In the sampling area which is less than $0.1L$, where L is the length of the car, the search mode of random sampling is switched. One reason is to ensure the vehicle can approach the target point quickly and the other reason is that it can effectively face the path generation problem in narrow environments.

3.2.2. The Constraints. When the original RRT extends nodes, it is necessary to judge whether all extended nodes are obstacle positions, which reduces the computational efficiency of the algorithm to a certain extent. Therefore, firstly, the surrounding obstacles with a radius of $0.5L$ are detected. If there are no obstacles within this range, the extended node directly skips the obstacle detection, or the obstacle detection is carried out, so as to improve the implementation efficiency of the algorithm.

(1) *The Constraints of the Vehicle.* The vehicle's executing actuator constraints should be met in the path planning algorithm so that the vehicle can drive safely in the complicate environment. The executing actuator of the vehicle includes the vehicle's front wheel steering angle δ , the maximum driving speed $v_{\max}$, and the maximum acceleration $a_{\max}$. These factors should be considered to do the nodes' extension within certain limits. Therefore, in this paper, the comprehensive factor d_c is used to measure the distance between the sampling point s and the parent node $t_i = [x_i, y_i]$, which is expressed as

$$d_c = d_h + d_{ed}, \quad (14)$$

where, $dh = 1 - h_1 h_2 / |h_1| |h_2|$, $h_1 = [x_s - x_i, y_s - y_i]$, $h_2 = [x_i - x_p, y_i - y_p]$, (x_s, y_s) is the location of the current sampling node, (x_i, y_i) is the location of the current parent node, $d_{ed} = \sqrt{(x_s - x_i)^2 + (y_s - y_i)^2}$, d_h is the heading change of the branch to be extended relative to the current branch, $h_1 h_2 / |h_1| |h_2|$ is the direction similarity, h_1 is the direction vector to be extended, h_2 is the brunch's direction vector to be extended, and d_{ed} is the Euclidean distance between the node and the sampling point.

Furthermore, in the cost function calculation of the traditional RRT, the size of the path length is considered usually. The energy consumption and the time consumption are also considered in this paper; the cost function $ccost(n_i, n_e)$ of the extended brunch is defined as

$$C_{cost}(n_i, n_e) = d_h + d_{ed} + N_f + t, \quad (15)$$

where $N_f = (ma_e + mg\mu_r + 0.5cdApv^2)vt_s$, $a_e \geq 0$, $N_f = (rbma_e + mg\mu_r + 0.5cdApv^2)vt_s$, $a_e < 0$, ne is the extended node, d_h is the heading change of the branch to be extended relative to the current branch, d_{ed} is the Euclidean distance between the node and the sampling point, N_f is the energy consumption of the extended brunch, t is the time consumption of the extended brunch, r_b is the brake feedback ratio, m is the vehicle mass, μ_r is the rolling resistance coefficient, cd is the wind resistance coefficient, A is the windward area, and ρ is the air density.

Based on the improved method, the comprehensive distance value of peripheral nodes is calculated and the cost function is taken into consideration comprehensively.

Meanwhile, it is used for subsequent node extension. The selection of candidate nodes makes the extended branch meet the vehicle motion constraints, thus reducing the amount of the subsequent optimization calculation.

(2) *The Constraints of the Vehicle Driving Environment.* In order to make the planned path feasible, vehicles need to meet the constraints of road environment when they drive on the road. The left boundary and right boundary of the road are represented by B_l and B_r , respectively, so the extended nodes need to meet the requirements of the formula:

$$B_l \leq n(e)_y \leq B_r, \quad (16)$$

$$T_s \leq n(e)_x \leq T_g. \quad (17)$$

To sum up, firstly, artificial potential field is used to expand the bidirectional RRT * around the target direction. Then, in the specific expansion, the constraints of the vehicle itself and the driving environment are considered so that the planned path meets the actual driving demand of the vehicle.

3.3. Collision Detection Based on SAT (Separating Axis Theorem). The main contents of SAT (separating axis theorem) are that when two objects need to be detected for collision, the two objects are supposed to be two convex polygons and the convex polygons are projected onto a vector separation axis at the same time firstly. Then, whether there is a gap between the two projections should be judged. If there is a gap, there is no collision; if there is no gap, the collision will happen. If the obstacle in the driving environment is a concave shape, connect the vertices of the two sides of the concave shape whose internal angle is greater than 180° to convert the concave shape into a convex shape. The specific conversion method is shown in Figure 3. If the concave-sided shape has multiple internal angles greater than 180° , the corresponding multiple conversions are performed until it becomes a convex-sided shape.

In order to make the planned path meet the actual driving environment of the vehicle, the actual geometry of the vehicle itself and obstacles need to be considered in the path planning method. In this paper, as to ensure the driving safety of the vehicle, the geometry of the vehicle is supposed as a circle, with the vehicle's center of mass as the center and the length between the center and the front axle as the radius. At the same time, the geometry of the obstacles is assumed as rectangles and the separation axis theorem is used to transform the collision detection problem between vehicles and obstacles into the intersection test problem of different graphs.

The schematic diagram of the SAT is shown in Figure 4. Both graphs are in the Cartesian coordinate system XOY . The circle A and rectangle B represent the vehicle and obstacle, respectively. The unit vectors of the circle A in the local coordinates with A_o as the origin point are A_x and A_y ; the unit vectors of the rectangle B in the local coordinates with B_o as the origin point are B_x and B_y ; the diameter of the circle A is D_a ; the length and width of the rectangle B are D_b and D_w ; N is the distance between the center point of the

FIGURE 3: Concave polygon conversion diagram.

circle A and the center point of rectangle B; β is the vehicle heading angle in global coordinates; α is the angle between the center line of the two graphs and the x -axis of the global coordinate system.

There are four separation axes, including the x axe and y axe of the circle A and the rectangle B, respectively. In order to detect the collision of the two graphics, whether the projection of two figures on all the separation axes simultaneously satisfies the following conditions should be judged; the conditions are as follows:

Condition 1: the project axle is the x axle of the local coordinate system of the circle A:

$$N|\cos \alpha| > \frac{D_a}{2}|\cos \alpha| + \frac{D_a}{2}|\sin \beta|. \quad (18)$$

Condition 2: the project axle is the y axle of the local coordinate system of the circle A:

$$N|\sin \alpha| > \frac{D_a}{2}|\sin \beta| + \frac{D_a}{2}|\cos \beta|. \quad (19)$$

Condition 3: the project axle is the x axle of the local coordinate system of the rectangle B:

$$N|\cos(\alpha - \beta)| > \frac{D_b}{2}|\cos \beta| + \frac{D_w}{2}|\sin \beta|. \quad (20)$$

Condition 4: the project axle is the y axle of the local coordinate system of the rectangle B:

$$N|(\sin \alpha - \beta)| > \frac{D_b}{2}|\sin \beta| + \frac{D_w}{2}|\cos \beta|. \quad (21)$$

If all of the conditions are met simultaneously, then there is no collision between the vehicle and the obstacles and the path can be used, or if the collision will happen, then the path should be abandoned. The SAT is used to do the collision detection. The advantage of the SAT is the detection speed which is very quick and the detection results are accurate; this method can help find the appropriate path as quickly as possible.

3.4. The Path Optimization Method Based on the Cubic B-Spline Curve. Because the curvature of the path planned by the traditional bidirectional RRT * is not continuous, when the vehicle tracks the path, it must stop to change the driving direction such that the vehicle cannot drive smoothly. So, the planned trajectory should be optimized to make the vehicle drive smoothly and steadily [27]. The

FIGURE 4: The schematic diagram of the SAT.

polynomial and the Bezier curve are used to optimize the path; however, different kinds of the constraints such as curvature, accelerated speed, and heading angle cannot be satisfied. Because the curvature is continuous and different kinds of the constraints are considered, the cubic B-spline curve is used to optimize the generated path.

Suppose there are $n + 1$ control vertexes $P_i (i = 0, 1, \dots, n)$, the cubic B-spline curve is expressed as

$$C(u) = \sum_{i=0}^n P_i B_{i,k}(u), \quad (22)$$

where the primary function $B_{i,k} (i = 0, 1, \dots, n)$, the highest degree is k , the basis function is determined by the sequence of node u , and the value range of u is $u_0 \leq u_1 \leq \dots \leq u_{n+k+1}$.

The cubic B-spline curve is shown as

$$C(u) = \sum_{i=0}^3 P_i B_{i,3}(u). \quad (23)$$

where the basis function $B_{i,3}(u)$ is defined as

$$\begin{aligned} B_{0,3}(u) &= \frac{(-u^3 + 3u^2 - 3u + 1)}{3!}, \\ B_{1,3}(u) &= \frac{(3u^3 - 6u^2 + 4)}{3!}, \\ B_{2,3}(u) &= \frac{(-3u^3 + 3u^2 + 3u + 1)}{3!}, \\ B_{3,3}(u) &= \frac{u^3}{3!}. \end{aligned} \quad (24)$$

The B-spline fitting curve is expressed as

$$\begin{cases} x = B_0 * a(1, i) + B_1 * a(1, i + 1) + B_2 * a(1, i + 2) + B_3 * a(1, i + 3), \\ y = B_0 * a(2, i) + B_1 * a(2, i + 1) + B_2 * a(2, i + 2) + B_3 * a(2, i + 3), \end{cases} \quad (25)$$

where a is the fixed coefficient.

4. Simulation Results and Analysis

4.1. The Simulation Test. The improved two-way RRT * method proposed in this paper is simulated and verified on a PC with processor Intel I7 based on MATLAB R2018a. The size of the simulation environment is equivalent to the actual driving area of 30 m * 30 m. The vehicle starts from the lower left corner of the environment as the starting point, and the upper right corner is the end point of the vehicle. Set the vehicle length to 2 m, width to 1 m, wheelbase to 1.2 m, maximum front wheel turning angle to 30°, and minimum turning radius to 4 m. For the randomness of the algorithms, 100 computations are counted, including the sampling number, average path length, and average searching time.

In order to verify the performance of the improved method, it was verified in multiple simulation scenarios. Based on the static obstacle environment in which concave-convex obstacles exist in Figure 5(a) and the complex obstacle environment in which dynamic and static obstacles exist in Figure 5(b) and Figure 5(c), three simulation environments are established, respectively. There are concave-convex obstacles in all three simulated environments. The

black area is the barrier area and the white area is the passable area. In Figure 5(b), the yellow triangle represents the dynamic obstacle in the environment, and the dotted line represents the movement process of the dynamic obstacle. In Figure 5(c), two yellow triangles represent dynamic obstacles, and the dotted line represents the movement process of dynamic obstacles. There are two dynamic obstacles in the driving environment, and a simulation scene is established. There are bumps and obstacles in these three simulated environments. The black area represents the obstacle area, and the white area represents the passable area. In Figure 5(b), the yellow triangle represents dynamic obstacles in the environment, and the dashed straight line represents the movement process of dynamic obstacles; in Figure 5(c), the two yellow triangles represent dynamic obstacles, and the dashed straight line represents the movement process of dynamic obstacles.

In Figure 6, the generated light purple path is the path formed by the original RRT, the light blue and light pink paths are the paths generated by the bidirectional RRT *, and the red path is the path generated by the improved bidirectional RRT * article. It can be seen from Figure 6 that the path generated by RRT is relatively tortuous and relatively close to obstacles, which is not conducive to the safe driving

FIGURE 5: Three kinds of the obstacle environments. (a) Environment a. (b) Environment b. (c) Environment c.

of the vehicle. Two-way RRT * is more optimized than the original RRT, but the effect is not very obvious. The improved two-way RRT * method proposed in this paper takes into account the dynamic conditions of the vehicle, expands the target direction based on the idea of artificial potential field, and considers the influence factors of dynamic obstacle rejection. At the same time, it considers the separation axis theory to detect the vehicle and the collision safety between obstacles, and then the curve smoothing process is performed again according to the B-spline curve, thereby meeting the needs of vehicle driving. Tables 1–3 show the comparison results based on obstacle environments (a), (b), and (c), respectively. In order to further verify the superior performance of the improved method, the planning effect of the corresponding algorithm at a specific time is described in detail. Figure 7(a) shows a comparison effect diagram of the three methods at 0.4 s. The path generated by the improved method in this paper is represented by the red path. The red solid dots and red triangle in Figure 7(a) are the planning effect of the improved method in this paper at 0.4 s for the

two-way RRT *. It can be seen that it successfully avoided the dynamic obstacles, the generated path is relatively smooth. Figure 7(b) shows the effect of the global path diagram generated by the three methods. It can be seen that the improved two-way RRT * method successfully converges, as shown by the red five-pointed star; Figure 8 shows the verification effect in the simulation environment c. In Figure 8(a), the red route at time 0.35 s is the improved two-way RRT * method in this article, and the red solid dots and red solid triangle are the planning effect of the improved method in this article when the two-way RRT * is 0.35 s, and the generated path located between obstacles; there is still a certain safety distance, which effectively avoids the short generation time of dynamic obstacles; when it runs to 0.46 s, as shown in Figure 8(b), only the traditional RRT has not yet completed the planning. It proves the time advantage of the other two methods and it can be seen that the improved method proposed in this paper is better than the traditional RRT and two-way RRT * methods. It can be seen from Tables 2–4 that the path generated by RRT is not smooth,

FIGURE 6: The planned results of three algorithms based on the obstacle environment (a).

FIGURE 7: The planned results of three algorithms based on the obstacle environment (b). (a) Local path at $t = 0.4$ s. (b) Global build path.

and it takes the longest time and it is not easy for vehicles to follow the path. Compared with the RRT algorithm and the two-way RRT* algorithm, the improved two-way RRT* in this paper takes the shortest time to generate a path, and the path is the smoothest path. Obstacle environment (c) is a complex obstacle environment. It can be seen from Figure 8 and Table 3 that compared with the RRT algorithm and the two-way RRT* algorithm, the path of the improved method in this paper takes the shortest time and the path is smooth. The simulation results show that the method can adapt to various obstacle environments including complex dynamic environments.

4.2. The Real Vehicle Test. In this section, we use the real vehicle test method to verify the method employed in this paper. In order to verify the advantages of the method presented in this paper, the vehicle path planning in real obstacle environment was carried out with the method proposed in literature [28]. The smart electric vehicle is used to verify the proposed algorithm. The smart electric vehicle platform is shown in Figure 9. The vehicle has 4-wheel hub motors; the sensors include ultrasonic radar, laser radar, CCD, GPS, and minicomputer. The smart vehicle has the function of high-precision positioning, SLAM mapping, and automatic navigation tracking.

FIGURE 8: The planned results of three algorithms based on the obstacle environment (c). (a) Local path at $t=0.35$ s. (b) Local path at $t=0.46$ s. (c) Global build path.

TABLE 1: Comparison of environment simulation results based on the obstacle environment (a).

Algorithm	RRT	Bidirectional RRT *	Improved bidirectional RRT *
Random sampling points (piece)	53	50	45
Average path length (m)	26.190	28.660	23.071
Average time (s)	0.52	0.47	0.453

Before the test begins, the environment map is built through SLAM (simultaneous localization and mapping). The actual driving environment of the vehicle is shown in Figure 10. Figures 10(a)–10(d) show the motion states of the vehicle and the dynamic obstacle at different times. Figure 10(a) shows the original driving position and vehicle position; Figure 10(b)

shows that when the vehicle moves with two obstacles at the same time and the first dynamic obstacle is detected, the vehicle makes a right turn to avoid obstacles; Figure 10(c) shows that the vehicle continues to move forward, detects the second dynamic obstacle, and avoids it safely; at this time, the first dynamic obstacle has stopped moving; Figure 10(d) shows that

TABLE 2: Comparison of environment simulation results based on the obstacle environment (b).

Algorithm	RRT	Bidirectional RRT *	Improved bidirectional RRT *
Random sampling points (piece)	60	44	41
Average path length (m)	24.191	22.773	21.157
Average time (s)	0.55	0.53	0.45

FIGURE 9: The smart electric vehicle platform.

(a)

(b)

FIGURE 10: Continued.

(c)

(d)

FIGURE 10: The real obstacle environment.

FIGURE 11: The results of the SLAM.

the vehicle successfully avoided the second obstacle and finally reached the designated driving position. The uniform velocity of dynamic obstacles is 1 m/s.

The SLAM obstacle composition and the movement direction of the dynamic obstacle are shown in Figure 11. The green triangle represents the first obstacle moving horizontally to the right, and the red triangle represents the second obstacle moving to the lower left corner; the black rectangle is the static obstacle in the running scene, and the white is the free driving area.

After the SLAM construction is completed, the starting and target points of the smart vehicle are set to observe the effect of the two kinds of the path planning methods:

FIGURE 12: The planning results of the two planning methods.

comparing the improved method in this paper with the method proposed in the document [28]. The average driving speed is 1.5 m/s and the safety distance is 0.5 m. The planning results of the two planning methods are shown in Figure 12. The data pairs are shown in Table 4. In Figure 12, two kinds of the lines represent different planning results of the real vehicle using two different methods. We can see from Figure 12 and Table 4 that two methods can plan the obstacle avoidance trajectory to make the vehicle reach the target point. However, the trajectory length planned by this paper's improved RRT* algorithm is shorter than the other algorithms, and the curvature of the trajectory planned by

TABLE 3: Comparison of environment simulation results based on the obstacle environment (c).

Algorithm	RRT	Bidirectional RRT *	Improved bidirectional RRT *
Random sampling points (piece)	65	57	46
Average path length (m)	27.245	25.535	22.412
Average time (s)	0.615	0.513	0.463

TABLE 4: Comparison of two methods.

Algorithm	Method of the literature [28]	Method of this article
Random sampling points (piece)	56	47
Average path length (m)	14.53	12.73
Average time (s)	9.71	8.47

the improved RRT * algorithm is the smoothest for the actual tracking of the vehicle, which improves the driving stability and safety of the vehicle.

We can see from the test results that in the complex obstacle environment, the algorithm proposed in this paper can be applied in practice on the smart electric vehicle platform and can be used to plan the noncollision path of the vehicle. The length of the planned trajectory and the smoothness degree planned by this paper's method are superior to the others' algorithm.

5. Conclusions

In this paper, an improved bidirectional RRT * path planning method has been proposed and tested. This improved bidirectional RRT * method can generate the planning path with high efficiency and accuracy; the vehicle can track the path smoothly and safely. The contents are summarized as follows:

- (1) The proposed path planning algorithm for the smart vehicle applied the artificial potential field to establish a mechanism of extension, which makes the random tree's extension have more intentionality. Based on the speed factor, the dynamic obstacle repulsion is constructed, and the collision detection is determined according to the distance between the obstacle and the node, thereby improving the search efficiency of the method.
- (2) The vehicle's own constraints and driving environment constraints are considered together to make the vehicle avoid obstacles accurately. First, the concave polygon obstacles are converted to convex polygon obstacles, and then collision detection is performed based on SAT to make the generated path more reasonable and the car can drive safely. The cubic B-spline curve is used to optimize the generated path to make the curvature of the path continuous. Thus, the vehicle can track the path smoothly.
- (3) The simulation and experiment were implemented and both validated the proposed improved bidirectional RRT * algorithm. In the simulation analysis, this paper's method can generate the smoothest

path and takes the shortest time compared with the other two methods and it can adapt to different kinds of obstacle environments including the complicated environment. In the real vehicle experiment, we can see from the test results that in the complex dynamic obstacle environment, compared with others' algorithm, this paper's method can generate the shortest and smoothest path. The improved bidirectional RRT * method can be applied in practice on the smart electric vehicle platform and it has a practical value in engineering application.

Data Availability

The (Figures 1–12 and Tables 1–5) data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Qingying Ge developed methodology and wrote the original draft; Aijuan Li was responsible for software and validation; Haiping Du and Aijuan Li reviewed and edited the article; Shaohua Li conceptualized the study; Xin Huang did formal analysis; Chuanhu Niu did simulation and analysis. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

This project was supported by the National Natural Science Foundation of China (Grant nos. 51505258, 61601265, and 51405272), Natural Science Foundation of Shandong Province, China (Grant nos. ZR2015EL019 and ZR2020ME126), the Youth Science and Technology Plan Project of Colleges and Universities in Shandong Province (Grant no. 2019KJB019), Open project of State Key Laboratory of Mechanical Behavior and System Safety of Traffic Engineering Structures, China (Grant no. 1903), and Open project of Hebei Traffic Safety and Control Key Laboratory, China (Grant no. JTKY2019002).

References

- [1] P. Marin-Pablo, H. Ahmed, M. David, and E. Arturo-dela, "iCab use case for ROS-based architecture," *Robotics and Autonomous Systems*, vol. 18, pp. 251–262, 2019.
- [2] R. Yadollah, F. Iman, and K. Amir, "Autonomous driving motion planning with obstacles prioritization using lexicographic optimization," *Control Engineering Practice*, vol. 77, pp. 235–246, 2018.
- [3] K. Christos, Q. Mohammed, W. H. Chen, and D. Lipika, "Real-time motion planning methods for autonomous on-road driving: state-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, 2015.
- [4] B. Li, Y. M. Zhang, F. Q. Dou, and Y. M. Wang, *Real-Time Trajectory Planning for AGV in the Presence of Moving Obstacles: A First-Search-Then-Optimization Approach*, Cornell University, Ithaca, NY, USA, 2019.
- [5] C. K. Xia, Y. Z. Zhang, and I. M. Chen, "Learning sampling distribution for motion planning with local reconstruction-based self-organizing incremental neural network," *Neural Computing and Applications*, vol. 31, pp. 9185–9205, 2019.
- [6] L. Chen, D. F. Qin, X. Xu, Y. F. Cai, and J. Xie, "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*, vol. 132, pp. 65–73, 2019.
- [7] Y. Dina and R. Pere, "Motion planning survey for autonomous mobile manipulators underwater manipulator case study," *Robotics and Autonomous Systems*, vol. 107, pp. 20–44, 2018.
- [8] A. Viseras, D. Shutin, and L. Merino, "Robotic active information gathering for spatial field reconstruction with rapidly-exploring random trees and online learning of Gaussian processes," *Sensors*, vol. 19, no. 5, pp. 1016–1040, 2019.
- [9] Y. Dong, Y. Zhang, and J. Ai, "Experimental test of unmanned ground vehicle delivering goods using RRT path planning algorithm," *Unmanned Systems*, vol. 05, no. 01, pp. 45–57, 2017.
- [10] D. Q. Zeng, Z. P. Yu, X. Lu et al., "A novel robust lane change trajectory planning method for autonomous vehicle," in *Proceedings of the IEEE Intelligent Vehicle Symposium (IV)*, Paris, France, June 2019.
- [11] H. Liu, X. Zhang, J. Wen, R. H. Wang, and X. Chen, "Goal-biased bidirectional RRT based on curve-smoothing," *IFAC-PapersOnLine*, vol. 52, no. 24, pp. 235–260, 2019.
- [12] X. Wang, X. Luo, B. Han, Y. Chen, G. Liang, and K. Zheng, "Collision-free path planning method for robots based on an improved rapidly-exploring random tree algorithm," *Applied Sciences*, vol. 10, no. 4, pp. 1381–1394, 2020.
- [13] L.-C. Feng, H.-W. Liang, M. B. Du, and B. Yu, "Guiding-area RRT path planning algorithm based on A * for intelligent vehicle," *Computer Systems & Applications*, vol. 26, no. 8, pp. 127–133, 2017.
- [14] Y. J. Li, W. Wu, Y. Gao, D. L. Wang, and Z. Fan, "PQ-RRT *: An improved path planning algorithm for mobile robots," *Expert Systems With Applications*, vol. 152, Article ID 113425, 2020.
- [15] D. Lee and D. H. Shim, "Path planner based on bidirectional spline-RRT * for fixed-wing UAVs," in *Proceedings of the International Conference on Unmanned Aircraft Systems*, pp. 77–86, IEEE, Arlington, VA, USA, June 2016.
- [16] J. Faris, R. Ron, and N. Philipp, "Smooth path-generation around obstacles using quartic splines and RRTs," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9108–9113, 2017.
- [17] H. Q. Ahmed and A. Yasar, "Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments," *Robotics and Autonomous Systems*, vol. 68, pp. 1–11, 2015.
- [18] Y. Shi, Q. Li, S. Bu, J. Yang, and L. Zhu, "Research on intelligent vehicle path planning based on rapidly-exploring random tree," *Mathematical Problems in Engineering*, vol. 2020, Article ID 5910503, 14 pages, 2020.
- [19] K. Yang, "Anytime synchronized-biased-greedy rapidly-exploring random tree path planning in two dimensional complex environments," *International Journal of Control Automation and Systems*, vol. 9, no. 4, pp. 750–758, 2011.
- [20] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. How, "Motion planning for urban driving using RRT," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots & Systems*, pp. 1681–1686, IEEE, San Francisco, CA, USA, August 2011.
- [21] C. Fulgenzi, A. Spalanzani, C. Laugier, and C. Tay, "Risk based motion planning and navigation in uncertain dynamic environment," *Research Report*, pp. 1–14, 2010.
- [22] Y. Bai, Y. Zhang, and J. Hu, "A motion planner enabling cooperative lane changing: reducing congestion under partially connected and automated environment," *Journal of Intelligent Transportation Systems Technology Planning and Operations*, vol. 10, 2020.
- [23] S. Zhanbo, H. Tianyu, and Z. Peitong, "Cooperative decision-making for mixed traffic: a ramp merging example," *Transportation Research Part C: Emerging Technologies*, vol. 120, 2020.
- [24] Y. Wang, P. Cai, and G. Lu, "Cooperative autonomous traffic organization method for connected automated vehicles in multi-intersection road networks," *Transportation Research Part C: Emerging Technologies*, vol. 111, pp. 458–476, 2020.
- [25] Y. Rahmati, A. Talebpour, A. Mittal, and J. Fishelson, "Game theory-based framework for modeling human-vehicle interactions on the road," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 9, pp. 701–713, 2020.
- [26] X. Ting, *Research on the Path Planning and the Tracking Control Algorithm of the Intelligent Vehicle in Complex Environment*, University of Jilin, Jilin, China, 2019.
- [27] A. Hidalgo-Paniagua, J. PedroBandera, M. Ruiz-De-Quintanilla, and A. Bandera, "Quad-RRT: a real-time GPU-based global path planner in large-scale real environments," *Expert Systems with Application*, vol. 99, pp. 141–154, 2018.
- [28] R. Zacccone and M. Martelli, "A collision avoidance algorithm for ship guidance applications," *Journal of Marine Engineering & Technology*, vol. 19, pp. 62–75, 2020.