Hindawi
Journal of Advanced Transportation
Volume 2022, Article ID 1675736, 14 pages
https://doi.org/10.1155/2022/1675736

Research Article

Motion Planning and Tracking Control of Autonomous Vehicle Based on Improved A^* Algorithm

Yunlong Bai,[1] **Gang Li**,[1] **and Ning Li**[2]

[1] *School of Automobile and Traffic Engineering, Liaoning University of Technology, Jinzhou 121001, China*
[2] *School of Electronics & Information Engineering, Liaoning University of Technology, Jinzhou 121001, China*

Correspondence should be addressed to Gang Li; qcxyligang@lnut.edu.cn

Received 17 May 2022; Accepted 6 July 2022; Published 31 August 2022

Academic Editor: Wen Liu

The traditional A^* algorithm, applied to the motion planning of autonomous vehicles, easily causes high computational costs and excessive turning points generated in the planning path. In addition, the vehicle cannot track the path due to the unsmooth inflection point. To overcome these potential limitations, an improved A^* algorithm-based motion planning algorithm and a tracking control strategy based on model predictive control theory were proposed in this work. The method of expanding the search neighborhood is adopted to improve the planning efficiency of A^* algorithm. The artificial potential field method is also incorporated into the proposed A^* algorithm. The resultant force generated by each potential field is further introduced into the evaluation function of A^* algorithm to plan the driving path, which could be suitable for autonomous vehicles. The sharp nodes in the path are smoothed by cubic quasi-uniform B-spline curve. The tracking control strategy is designed based on model predictive control theory to realize the accurate tracking of the planned path. Typical obstacle avoidance conditions were selected for co-simulation test verification. The experimental results show that the proposed motion planning algorithm and tracking control strategy can effectively plan the obstacle avoidance path and accurately track the path in different environments.

1. Introduction

With the rapid growth of car ownership, traffic safety, and urban traffic congestion problems becoming more and more serious, and research and development of safe and reliable autonomous vehicles have become an inevitable development trend. Motion planning and tracking control are two key technologies of autonomous vehicles, which solve the problem of how autonomous vehicles drive. Motion planning is to plan a drivable safe path for the autonomous vehicle while tracking control is to track the planned path on the premise of satisfying the vehicle kinematics and dynamics constraints.

For the motion planning and tracking control of the autonomous vehicle, a large number of scholars have conducted in-depth research. In terms of motion planning, A^* algorithm [1], artificial potential field method [2], genetic algorithm [3], RRT algorithm [4], etc, are commonly used in the current research. Among them, A^* algorithm is the most widely used in the motion planning of autonomous vehicles because of its simple working principle and strong robustness. However, with the increase of the map, A^* algorithm also presents problems such as high computational cost and the inability to find the optimal path. In order to solve this kind of problems, a large number of scholars have carried out research on it. Fu et al. [5] proposed an improved A^* algorithm. When exploring the neighborhood, local paths from the current node to the target point are selected and planned according to the current environment, and the drivable local paths are optimized as part of the global path, improving the smoothness of the path planned by A^* algorithm. Tang et al. [6], aiming at the problem of large storage space of A^* algorithm, by using three methods: bidirectional search, guideline, and key point list, the amount of calculation of the algorithm is optimized and the planning time of the algorithm is reduced. Erke et al. [7] improved the obstacle avoidance performance of the A^* algorithm by setting key points around the obstacle and designed a variable step size A^* algorithm to reduce the calculation time of the algorithm. Liu et al. [8] selected and recorded the nodes in the Open List and Close List through the method of jump point search, optimized the algorithm planning time, and combined the dynamic window method to select the optimal path. Zhong et al. [9] combined the adaptive window method with the A^* algorithm, which simplified the calculation of the risk cost function and the distance cost function, and realized the motion planning of the mobile robot in a dynamic environment. Xiong et al. [10] smoothed the planned path through the Bezier curve, which improved the applicability of the A^* algorithm.

In terms of tracking control, the commonly used tracking control theories include pure tracking control [11], fuzzy control [12], optimal control [13], sliding mode variable structure control [14], model predictive control [15], etc. Since model predictive control (MPC) has a good ability to deal with multiple constraints and has a good control effect on the path tracking of nonlinear systems, a large number of scholars have applied MPC to the tracking control of autonomous vehicles. Novi et al. [16] proposed a hierarchical control method for nonlinear model predictive control (NMPC), in which high-order MPC calculated the curve of optimal speed and low-order NMPC constrained motion curve, thus improving the real-time performance of the control algorithm. Zhang et al. [17] proposed an adaptive MPC control strategy based on the recursive least square method, which can achieve better control effects under different driving conditions. Xu et al. [18] proposed a tracking control strategy based on the preview-follower theory (PFT) and MPC theory, in which PFT updated the reference quantity according to the preview-point and controlled the vehicle movement through MPC, effectively reducing the tracking error of the vehicle. Berntorp et al. [19] proposed an adaptive nonlinear model predictive control with variable tire model, which has a good control effect on path tracking on different roads.

To sum up, scholars have conducted a lot of research on the motion planning and tracking control of autonomous vehicles, but the current research mainly focuses on the optimization of the performance parameters of the algorithm itself, the actual running environment of the vehicle and whether the planned path can meet the requirements of vehicle tracking are less considered. Therefore, this study analyzed the motion planning and tracking control of autonomous vehicles in actual obstacle avoidance conditions, proposes a motion planning algorithm improved by A^* algorithm, and shows a tracking control strategy based on model predictive control theory. The main contributions of this study are as follows:

(i) In this study, the planning time of A^* algorithm is reduced by expanding the search neighborhood method, and the traditional algorithm's $3 * 3$ search neighborhood is expanded to $9 * 9$ search neighborhood, which reduces the included nodes in the Close List and increases the searchable direction of paths, effectively improving the planning efficiency of the algorithm.

(ii) In this study, artificial potential field method is combined with A^* algorithm. In the process of motion planning, the minimum safe distance between the planned path and the obstacle is considered. The resultant force generated by each potential field is introduced into the evaluation function of A^* algorithm for obstacle avoidance, and the sharp nodes in the path are smoothed by a cubic quasi-uniform B-spline curve.

(iii) The tracking control strategy is designed based on the model predictive control theory. The vehicle center of mass slip angle constraint is added to the model predictive control to improve the vehicle driving stability. Through the path tracking test under different vehicle speeds, the feasibility of the improved A^* algorithm for path planning and the effectiveness of the tracking control performance are separately verified.

The rest of the study is organized as follows: In Section 2, the improved motion planning algorithm is introduced, which combines the improved A^* algorithm by expanding the search neighborhood with the artificial potential field method, and uses a cubic quasi-uniform B-spline curve to smooth the path. In Section 3, the path tracking controller is designed based on model predictive control theory. In Section 4, the planning algorithm and control strategy are verified by co-simulation. Finally, the conclusions and future research directions of this study are discussed in Section 5.

2. Motion Planning

A^* algorithm was first applied to the motion planning of mobile robots. Most mobile robots are capable of omnidirectional movement, so the curvature of the planned path is not strictly required. With the rapid development of autonomous vehicles, A^* algorithm has also been widely used in the motion planning of autonomous vehicles. However, due to the constraints of autonomous vehicles such as minimum turning radius and maximum front-wheel Angle, the path planned by the traditional A^* algorithm is difficult to meet the tracking requirements of autonomous vehicles. Therefore, relevant improvements should be made to the A^* algorithm to plan the path suitable for autonomous vehicles. This paper studies the obstacle avoidance conditions as shown in Figure 1.

2.1. Environmental Map Modeling. In the motion planning of autonomous vehicles, it is necessary to establish an appropriate map model to represent the current environmental information, this study selects the commonly used grid map to construct the environment map of obstacle avoidance conditions. In the grid map, a series of squares of the same size are used to represent the current environment, and the binary method is used to assign values to each square to divide the map into occupied and unoccupied areas. 0 means that the current grid is not occupied, that is, the driving area of the vehicle, which is shown in white in the grid map, 1 indicates that the current grid is occupied and vehicles

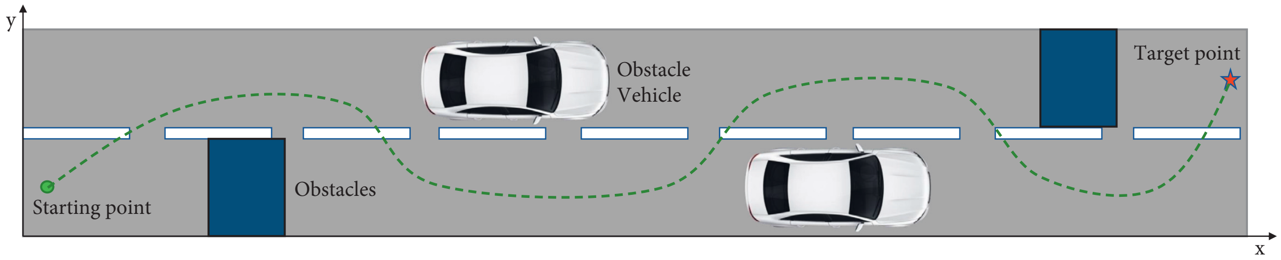


Figure 1: Schematic diagram of obstacle avoidance conditions.

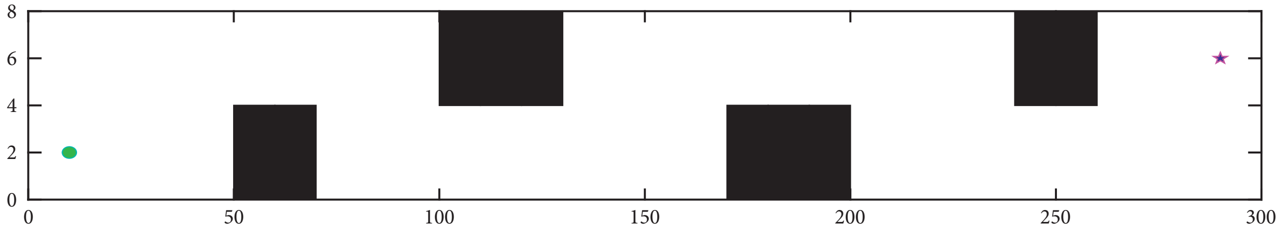


Figure 2: Grid map.

cannot pass through this area, as shown in black on the map. The grid map established in this study according to obstacle avoidance conditions is shown in Figure 2.

2.2. Traditional A^* Algorithm. A^* algorithm is one of the graph search algorithms. First, add the starting point to the Open List, search the grid of the surrounding $3 * 3$ neighborhood from the starting point, and traverse the adjacent nodes. Then the direction to be included is selected by calculating the evaluation function value of each adjacent node, and the node with the smallest evaluation function value is selected as the optimal node for inclusion, and Add the current node to the Close List [20]. Continue to explore the $3 * 3$ neighborhood around the optimal node, and repeat the above process to traverse the map until the target point is included in the Close List. Finally, find the parent node of each included node from the target point to the starting point to form a path. Its evaluation function is

$$f_{(n)} = g_{(n)} + h_{(n)}, \tag{1}$$

where $f_{(n)}$ is the the evaluation function; $g_{(n)}$ is the real cost, and $h_{(n)}$ is the estimated cost. Since autonomous vehicles cannot achieve omnidirectional movement, Manhattan distance is used to represent the estimated cost in order to reduce the planning error, then the estimated cost is

$$h_{(n)} = \left|\left(x_n - x_g\right)\right| + \left|y_n - y_g\right|, \tag{2}$$

where (x_n, y_n) is the coordinate of the current node and (x_g, y_g) is the coordinate of the target point.

It can be seen from the above formula that the traditional A^* algorithm is simple in principle, easy to implement, and can effectively plan the path existing on the map. However, with the expansion of the environmental map, there are also some problems such as long planning time, inability to plan the optimal path, and many sharp nodes in the planning path, so it is necessary to improve the traditional algorithm to plan the optimal path.

2.3. Improved A^* Algorithm

2.3.1. Expand the Search Neighborhood. The number of nodes included in the Close List of A^* algorithm is proportional to the planning time of the algorithm. The more nodes included, the longer the planning time. Therefore, the algorithm planning time can be reduced by reducing the nodes included in the Close List. The traditional A^* algorithm searches the grid of the surrounding $3 * 3$ neighborhood every time, as shown in Figure 3. Then, one of the surrounding eight adjacent nodes is selected as the optimal node for inclusion, that is, one of every eight adjacent nodes will be included in the Close List. In this way, only eight adjacent nodes can be explored at a time, and the planned path direction is limited to an integer multiple of $\pi/4$. The planned path has many turning points and is not smooth enough. To solve this problem, this study adopts the method of expanding search neighborhood to improve the traditional A^* algorithm. The $3 * 3$ search neighborhood of the traditional algorithm is extended to $5 * 5$ search neighborhood to reduce the nodes included in the Close List and increase the searchable direction of the planned path to improve the smoothness of the path, $5 * 5$ search neighborhood is shown in Figure 4.

As can be seen from Figure 4, the number of adjacent nodes searched for $5 * 5$ search neighborhood increases from 8 to 24, that is, one of every 24 nodes is included in the Close List. Compared with the traditional algorithm, it can effectively reduce the nodes included in the Close List. At the same time, the searchable directions of each traversal increase from 8 to 16, and the turning angle of the path within the search step becomes smaller. In order to verify the effectiveness of $5 * 5$ extended neighborhood A^* algorithm,

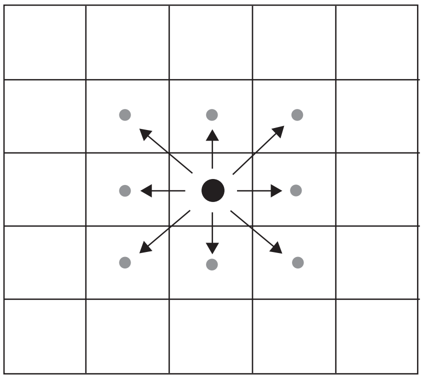


Figure 3: 3 ∗ 3 Search neighborhood.

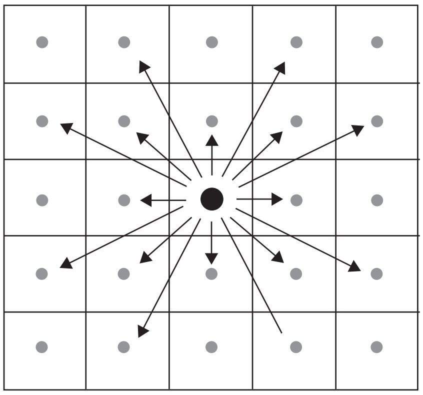


Figure 4: 5 ∗ 5 Search neighborhood.

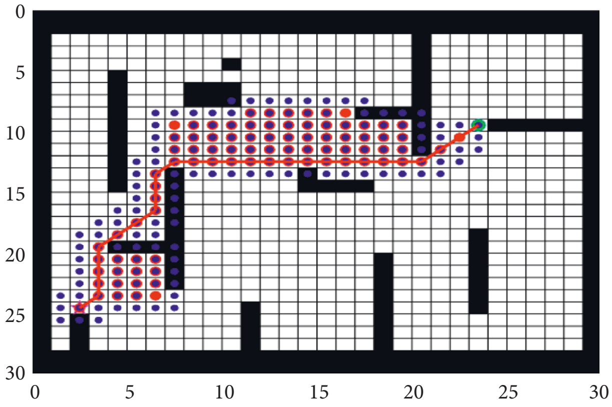


Figure 5: 3 ∗ 3 Search neighborhood planning result.

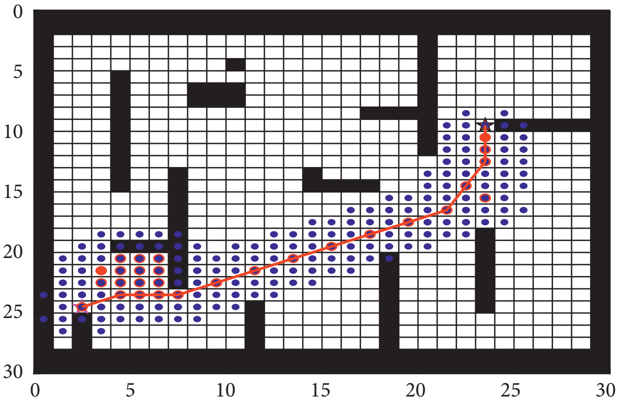


Figure 6: 5 ∗ 5 Search neighborhood planning result.

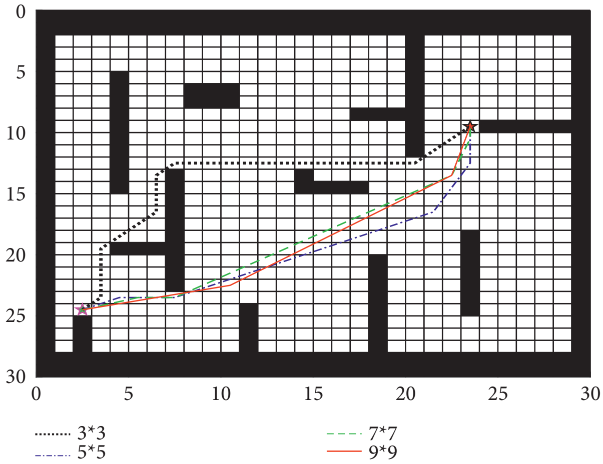


Figure 7: Comparison of paths planned in different search neighborhoods.

MATLAB software was used to verify the algorithm on 30 ∗ 30 grid map, and the planned path is shown in the Figures 5 and 6. The result of 3 ∗ 3 search neighborhood planning is shown in Figure 5, and the result of 5 ∗ 5 search neighborhood planning is shown in Figure 6.

According to the planning results, the planned path length of 3 ∗ 3 search neighborhood is 31.31 m, and the planning time is 3.32 s, while the planned path length of 5 ∗ 5 search neighborhood is 28.36 m, and the planning time is 1.68 s. It can be concluded that the planning effect of 5 ∗ 5 search neighborhood is better, and the planning path of 5 ∗ 5 search neighborhood is smoother than that of 3 ∗ 3 search neighborhood. Therefore, it can be concluded that the method of expanding the search neighborhood can effectively reduce the search time of the algorithm and improve the smoothness of the path. Due to the large driving environment map of autonomous vehicles and the uniform distribution of obstacles, a larger search neighborhood can be expanded to plan the path to reduce the planning time. Similarly, 7 ∗ 7 search neighborhood and 9 ∗ 9 search neighborhood were used to conduct experiments respectively. The planned path is shown in Figure 7, and the experimental results are summarized in Table 1.

According to the simulation results, it can be seen that the length and time of the path planned by the 9 ∗ 9 search neighborhood A^* algorithm are optimal, and the planned path has fewer turning points and is smoother. Therefore, this study adopts the method of expanding the 9 ∗ 9 search neighborhood as an improved A^* algorithm. The improved A^* algorithm is applied to the obstacle avoidance condition in this study, and the path planning is shown in Figure 8.

Table 1: Different search neighborhood planning results.

Search neighborhood	Adjacent node	Search direction	Path length/m	Planning time/s
3 * 3	8	8	31.31	3.32
5 * 5	24	16	28.36	1.68
7 * 7	48	32	27.35	1.51
9 * 9	80	56	27.21	1.49

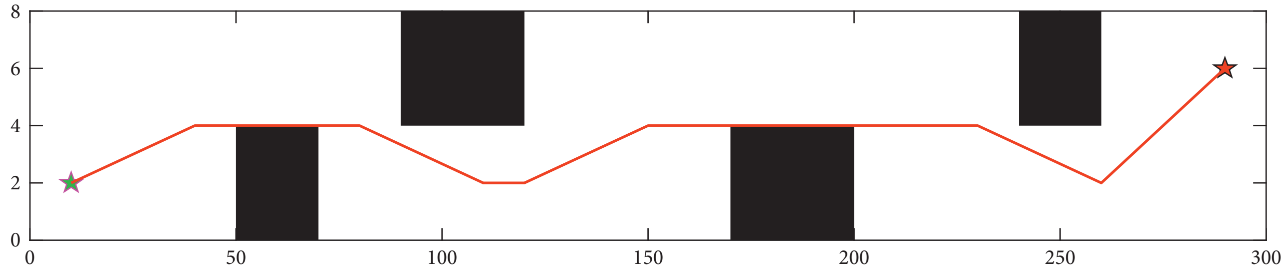


Figure 8: Improved A^* algorithm planning path.

2.3.2. Combined with Artificial Potential Field Method. As can be seen from Figure 8, the improved A^* algorithm of $9 * 9$ search neighborhood only considers the length of the path, resulting in a small distance between the planned path and the obstacle, which cannot meet the tracking requirements of vehicles under normal driving conditions. The normal planned path should consider the minimum safe distance between the obstacle and the vehicle. Therefore, the artificial potential field method is combined with A^* algorithm to construct a repulsive potential field around the obstacle, so that the planned path is far away from the obstacle. At the same time, to prevent the distance between the planned path and the road boundary from being small or beyond the road boundary, repulsive force potential field of the road boundary is created on both sides of the road to ensure that vehicles will not exceed the road boundary. Due to the long road, the gravitational potential field is constructed around the target point in order to make the planned path approach the target point more quickly.

In the traditional artificial potential field method, there is a repulsive force field around the obstacle and a gravitational field around the target point, and autonomous vehicles plan the driving path under the combined action of the repulsive force field and gravitational field [21]. The traditional gravitational potential field changes with the position of the vehicle. In this study, the gravitational potential field of the target point is established based on the traditional gravitational potential field function, and the target point gravitational potential field function is

$$U_{att} = \frac{1}{2} k_{att} \rho^2 (q, q_g), \tag{3}$$

where k_{att} is the action coefficient of gravitational field; $\rho(q, q_g)$ is the euclidean distance between the target point and the vehicle.

The gravitational function can be obtained from the gravitational potential field function, which is the negative derivative of the gravitational potential field with respect to $\rho(q, q_g)$, that is the negative gradient of the gravitational potential field:

$$F_{att} = -\nabla U_{att} = -\frac{1}{2} k_{att} * \nabla \rho^2 (q, q_g) = -k_{att} \rho (q, q_g). \tag{4}$$

The repulsion potential field mainly repulses the vehicle. When the autonomous driving vehicle enters the repulsive potential field, the obstacle will generate repulsive force on the vehicle. When the vehicle is outside the range of the repulsive potential field, the repulsive potential field will not work. The repulsive field function is expressed as

$$\begin{cases} U_{rep} = \frac{1}{2} k_{rep} \left(\frac{1}{\rho (q, q_o)} - \frac{1}{\rho_l} \right)^2, 0 \le \rho (q, q_o) \le \rho_l, \\ 0, \rho (q, q_o) \ge \rho_l, \end{cases} \tag{5}$$

where k_{rep} is the action coefficient of repulsive field, $\rho(q, q_o)$ is the distance between the obstacle and the vehicle, and ρ_l is the influence distance of the potential field.

In the traditional artificial potential field method, the repulsion and attraction may appear equal in magnitude and opposite in direction, resulting in the autonomous vehicle into local optimization, unable to continue to plan the path. Also, when the vehicle is about to reach the target point, the resultant force of the potential field at the target point is not zero, resulting in the unreachable target point and other problems. In this study, the repulsive field function is improved, and the influence factor of distance between vehicle and target point is added into the repulsive field function. At the same time, the influence range of the traditional repulsive field is improved, and the traditional repulsive force field is circular, but for the autonomous vehicle, the length of the lane is much larger than the width, and the longitudinal speed of the vehicle is much larger than the lateral speed of the vehicle. In order to plan a safer path, this study changes the influence range of the repulsive force field to ellipse, the influence range of the improved repulsion field is shown in Figure 9.

In order to plan the path safely, the major axis A and the minor axis B of the ellipse should be appropriately valued. Where the value of A has an important relationship with the

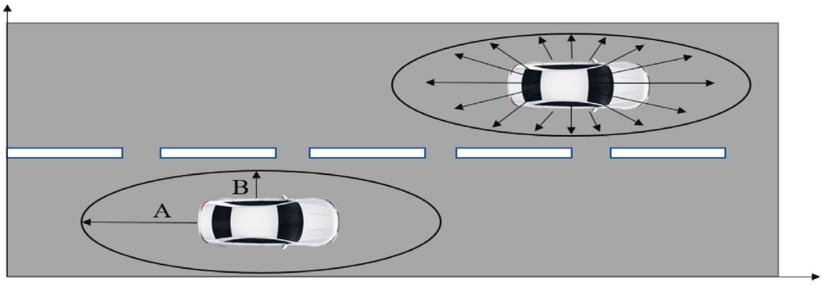


Figure 9: Improved potential field range.

speed, meanwhile, there is a minimum safe distance between the vehicle and the obstacle, so the value of the long axis of the elliptic scope A is

$$A = D_{\min} + \eta v, \tag{6}$$

where $D_{\min}$ is the minimum obstacle avoidance distance between vehicle and obstacles, η is the speed correction coefficient, and v is the speed of the vehicle.

According to the obstacle avoidance safety rules, the value of the short axis B of the ellipse scope is set as

$$B = \frac{\xi L}{2}, \tag{7}$$

where ξ is the safety factor and L is the lane width.

Therefore, the improved repulsive field function is

$$\begin{cases} U_{rep} = \frac{1}{2} k_{rep} \left(\frac{1}{\rho(q, q_o)} - \frac{1}{\rho_l} \right)^2 \rho^m (q, q_g), & q \in \frac{X^2}{A^2} + \frac{Y^2}{B^2} = 1, \\ 0, & q \notin \frac{X^2}{A^2} + \frac{Y^2}{B^2} = 1, \end{cases} \tag{8}$$

where m is the distance regulating factor; $X^2/A^2 + Y^2/B^2 = 1$ is the elliptic region of action.

To prevent the vehicle from falling into the local optimum, the repulsive force generated by the repulsive potential field is decomposed, and the repulsive force of the obstacle on the vehicle is decomposed into two directions. The schematic diagram of the decomposition repulsion force is shown in Figure 10. The repulsive force function after decomposition is

$$F_{rep} = -\nabla U_{rep} = \begin{cases} F_{rep1} + F_{rep2}, & q \in \frac{X^2}{A^2} + \frac{Y^2}{B^2} = 1, \\ 0, & q \notin \frac{X^2}{A^2} + \frac{Y^2}{B^2} = 1, \end{cases} \tag{9}$$

$$F_{rep1} = k_{rep} \left(\frac{1}{\rho(q, q_o)} - \frac{1}{\rho_l} \right) \frac{1}{\rho^2 (q, q_o)} \rho^m (q, q_g),$$

$$F_{rep2} = \frac{m}{2} k_{rep} \left(\frac{1}{\rho(q, q_o)} - \frac{1}{\rho_l} \right)^2 \rho^{m-1} (q, q_g),$$

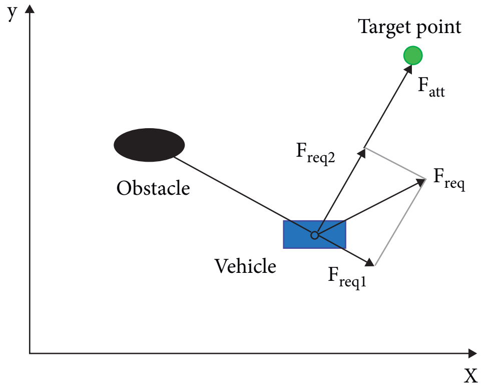


Figure 10: Decomposition repulsion diagram.

where the direction of F_{rep1} is that the obstacle points to the vehicle; the direction of F_{rep2} is that the vehicle points to the target point.

To prevent the planned path from exceeding the road boundary due to the large repulsive force of obstacles on vehicles, the repulsive force potential field is created at the road boundary. The repulsive field function is as follows:

$$\begin{cases} U_r = \frac{1}{2} k_r \left(\frac{1}{\rho(q, q_r)} - \frac{1}{\rho_r} \right)^2, \rho(q, q_r) \leq \rho_r, \\ 0, \rho(q, q_r) \geq \rho_r, \end{cases} \tag{10}$$

where k_r is the boundary repulsive field action coefficient; $\rho(q, q_r)$ is the distance between vehicles and the road boundary; ρ_r is the influence distance.

The repulsive force corresponding to the road boundary is

$$F_r = \begin{cases} k_r \left(\frac{1}{\rho(q, q_r)} - \frac{1}{\rho_r} \right) \frac{1}{\rho^2 (q, q_r)}, & \rho(q, q_r) \leq \rho_r, \\ 0, & \rho(q, q_r) \geq \rho_r. \end{cases} \tag{11}$$

Then the resultant force of the vehicle under the joint action of the target gravitational potential field, the obstacle repulsive potential field and the road boundary repulsive potential field is

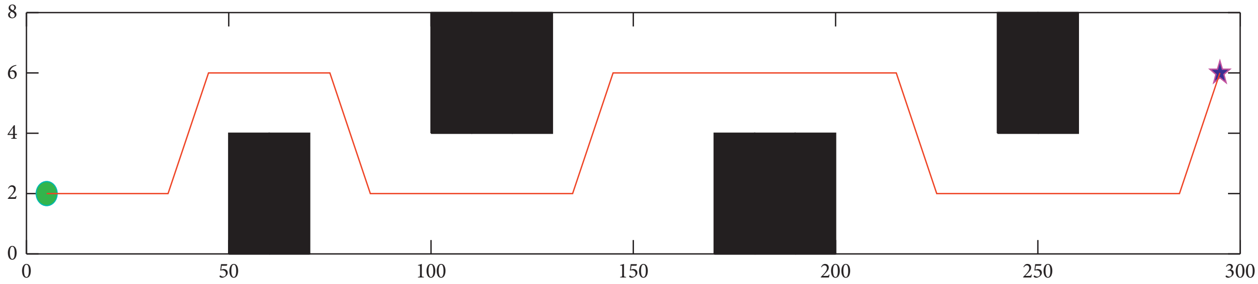


Figure 11: Planning path integrating artificial potential field method.

$$F = F_{att} + F_{rep} + F_r, \tag{12}$$

Combining the artificial potential field method with the improved A^* algorithm, the potential field force of each artificial potential field is introduced into the evaluation function of the A^* algorithm. The improved evaluation function is

$$f_{(n)} = g_{(n)} + h_{(n)} + F. \tag{13}$$

The improved A^* algorithm integrated with the artificial potential field method is applied to the obstacle avoidance condition in this study, and the path planning is shown in Figure 11.

2.3.3. Smoothing with B-Spline Curve. As can be seen from Figure 11, the improved A^* algorithm combined with the artificial potential field method still has problems such as sharp nodes and curvature discontinuity in motion planning. Therefore, a cubic quasi-uniform B-spline curve is used to smooth the sharp nodes in the path to making the planned path smoother.

Suppose there are $n+1$ control points in total, then B-spline curve of order K can be defined as

$$P(u) = [p_0, p_1, \dots, p_n] \begin{bmatrix} B_{0,k}(u) \\ B_{1,k}(u) \\ \cdots \\ B_{n,k}(u) \end{bmatrix} = \sum_{i=0}^{n} P_i B_{i,k}(u), \tag{14}$$

where P is the control points and $B_{i,k}(u)$ is the basis function of the B-spline curve; The B-spline curve basis function is expressed by the de Boor $-$ Cox recursive formula, and the expression is as follows:

$$B_{i,k}(u) = \begin{cases} \begin{cases} 1, & u_i \le u < u_{i+1} \\ & \qquad k = 1, \\ 0, & Other \end{cases} \\ \dfrac{u - u_i}{u_{i+k-1} - u_i} B_{i,k-1}(u) + \dfrac{u_{i+k} - u}{u_{i+k} - u_{i+1}} B_{i+1,k-1}(u), k \ge 2, \\ Define \dfrac{0}{0} = 0. \end{cases} \tag{15}$$

According to the basis function of B-spline curve, we know that B-spline curve of higher order can be obtained by two B-spline curves of lower order. The cubic quasi-uniform B-spline curve retains the properties of the Bessel curve at two endpoints. The tangent directions of the start and end points are tangent to the first and last sides of the characteristic polygon, respectively, and pass through the starting and ending points, which is more suitable for the motion planning of autonomous vehicles. In this study, cubic quasi-uniform B-spline curve were selected to smooth the planned path, and the repeatability of nodes at both ends was set as 3. After derivation, the expression of the cubic quasi-uniform B-spline curve is obtained as:

$$P_{0,3}(t) = \frac{1}{6} \left[t^1 t^2 t^3 \right] \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}, \tag{16}$$

where t is the position parameter, $0 \le t \le 1$; $P_0 - P_3$ are the control points.

The planned path after cubic quasi-uniform B-spline curve processing is shown in Figure 12.

As shown in Figure 12, the path planned by the improved A^* algorithm has no sharp nodes and the path is smooth, which effectively improves the shortcomings of the traditional algorithm. The process of improving A^* algorithm is summarized in Figure 13.

3. Tracking Control

3.1. Vehicle Dynamics Model. In order to reduce the amount of calculation in the process of solving the vehicle dynamics model, a three-degree of freedom vehicle dynamics model considering longitudinal, lateral, and yaw motion is established, as shown in Figure 14. The following simplifications are made: the road surface is assumed to be smooth and the vertical motion of the vehicle is ignored; the vehicle suspension system and aerodynamics are ignored; the lateral load transfer of the tires is ignored.where xoy is the vehicle coordinate system, XOY is the geodetic coordinate system, a, b is the distance from the center of mass to the front and rear axes, $\dot{x}$ and $\dot{y}$ are the speed of the vehicle in the x and y axis direction, $\dot{\varphi}$ is the vehicle yaw rate, δ_f is the front wheel

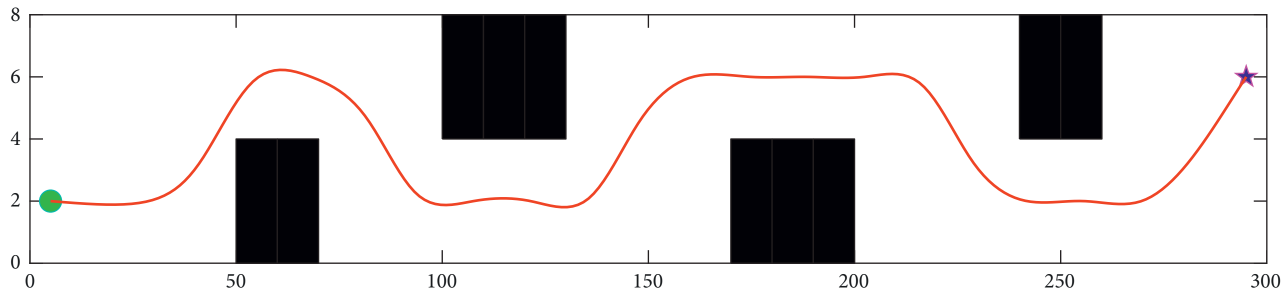


Figure 12: Path after smoothing.

angle, F_{lf} and F_{lr} is the longitudinal force of the front tire and rear tire, F_{cf} and F_{cr} are the lateral force of the front tire and rear tire, and α_f and α_r are the front and rear tire slip angle.

In this study, the simplified "magic formula" tire model is selected to analyze the tire force, and the small angle assumption is adopted. The lateral force and longitudinal force of the front and rear tires are expressed as

$$\begin{cases} F_{lf} = C_{lf}S_f, \\ F_{lr} = C_{lr}S_r, \\ F_{cf} = C_{cf}\alpha_f = C_{cf}\left(\delta_f - \dfrac{\dot{y} + a\dot{\varphi}}{\dot{x}}\right), \\ F_{cr} = C_{cr}\alpha_r = C_{cr}\left(\dfrac{b\dot{\varphi} - \dot{y}}{\dot{x}}\right), \end{cases} \tag{17}$$

where C_{lf} and C_{lr} are the longitudinal stiffness of front and rear tires, C_{cf} and C_{cr} is the lateral stiffness of the front and rear tires, and S_f and S_r are the slip rate of the front and rear tires.

According to Newton's second law, the transformation between vehicle coordinate system and geodetic coordinate system, and the tire force analysis, the vehicle dynamics equation is as follows:

$$\begin{cases} \ddot{x} = \dfrac{2}{m}\left[C_{lf}S_f - C_{cf}\left(\delta_f - \dfrac{\dot{y} + a\dot{\varphi}}{\dot{x}}\right)\delta_f + C_{lr}S_r\right] + \dot{y}\dot{\varphi}, \\ \ddot{y} = \dfrac{2}{m}\left[C_{cf}\left(\delta_f - \dfrac{\dot{y} + a\dot{\varphi}}{\dot{x}}\right) + C_{cr}\left(\dfrac{b\dot{\varphi} - \dot{y}}{\dot{x}}\right)\right] - \dot{x}\dot{\varphi}, \\ \ddot{\varphi} = \dfrac{2}{I_z}\left[a\left(C_{cf}\left(\delta_f - \dfrac{\dot{y} + a\dot{\varphi}}{\dot{x}}\right) - bC_{cr}\left(\dfrac{b\dot{\varphi} - \dot{y}}{\dot{x}}\right)\right)\right], \\ \dot{Y} = \dot{x}\sin\varphi + \dot{y}\cos\varphi, \\ \dot{X} = \dot{x}\cos\varphi - \dot{y}\sin\varphi, \end{cases} \tag{18}$$

where m is the mass of the vehicle, I_z is the moment of inertia of a vehicle, and $\dot{X}$ and $\dot{Y}$ are the speed of the vehicle in the X and Y axis directions of the inertial coordinate system.

3.2. Model Predictive Control.

The tracking control strategy is designed based on MPC theory. The model predictive control can predict the output of the system in the future according to the prediction model, the current state quantity of the system and the future control quantity, and can solve problems with various constraints in a rolling manner [22]. The tracking control strategy designed in this study is shown in Figure 15. The planned path and vehicle state are input into the model predictive controller, and the front wheel angle of the vehicle at the next time can be obtained through rolling optimization. The model predictive control flow is as follows:

Convert the three-degree-of-freedom model of the vehicle to a state-space representation:

$$\begin{cases} \dot{\xi} = f(\xi, u), \\ \lambda = C \cdot \xi. \end{cases} \tag{19}$$

where ξ is the state quantity, $\xi = [\dot{y}, \dot{x}, \varphi, \dot{\varphi}, Y, X]^T$; u is the control quantity, $u = [\delta_f]$; λ is the output, $\lambda = [\varphi, Y]^T$.

The vehicle 3-DOF dynamics model is a nonlinear model, which is linearized. (19) is expanded at point $[\xi_0, u_0]$ using Taylor's formula, retaining the first-order term yields a linear time-varying equation:

$$\dot{\xi} = f(\xi_0, u_0) + A(t)(\xi - \xi_0) + B(t)(u - u_0), \tag{20}$$

where $A(t)$ is the $f(\xi, u)$ Jacobian matrix for ξ; $B(t)$ is the $f(\xi, u)$ Jacobian matrix for u.

Using the first-order quotient difference method to process (19), the discrete state space equation can be obtained:

$$\xi(k+1) = A(k)\xi(k) + B(k)u(k), \tag{21}$$

where $A(k) = I + TA(t)$; $B(k) = TB(t)$; T is the sampling period; t is the sampling time; N_p is the prediction time domain; $K = t, t+1, \ldots\ldots, t+N_p$; I is the unit matrix.

In order to ensure that the vehicle can track the path stably, use the increment of the front wheel angle as the output of the MPC controller, and a new state space expression is obtained:

$$\begin{cases} \tilde{\xi}(k+1|t) = \tilde{A}_{k,t}\tilde{\xi}(k|t) + \tilde{B}_{k,t}\Delta u(k|t), \\ \tilde{\lambda}(k|t) = \tilde{C}_{k,t}\tilde{\xi}(k|t), \end{cases} \tag{22}$$

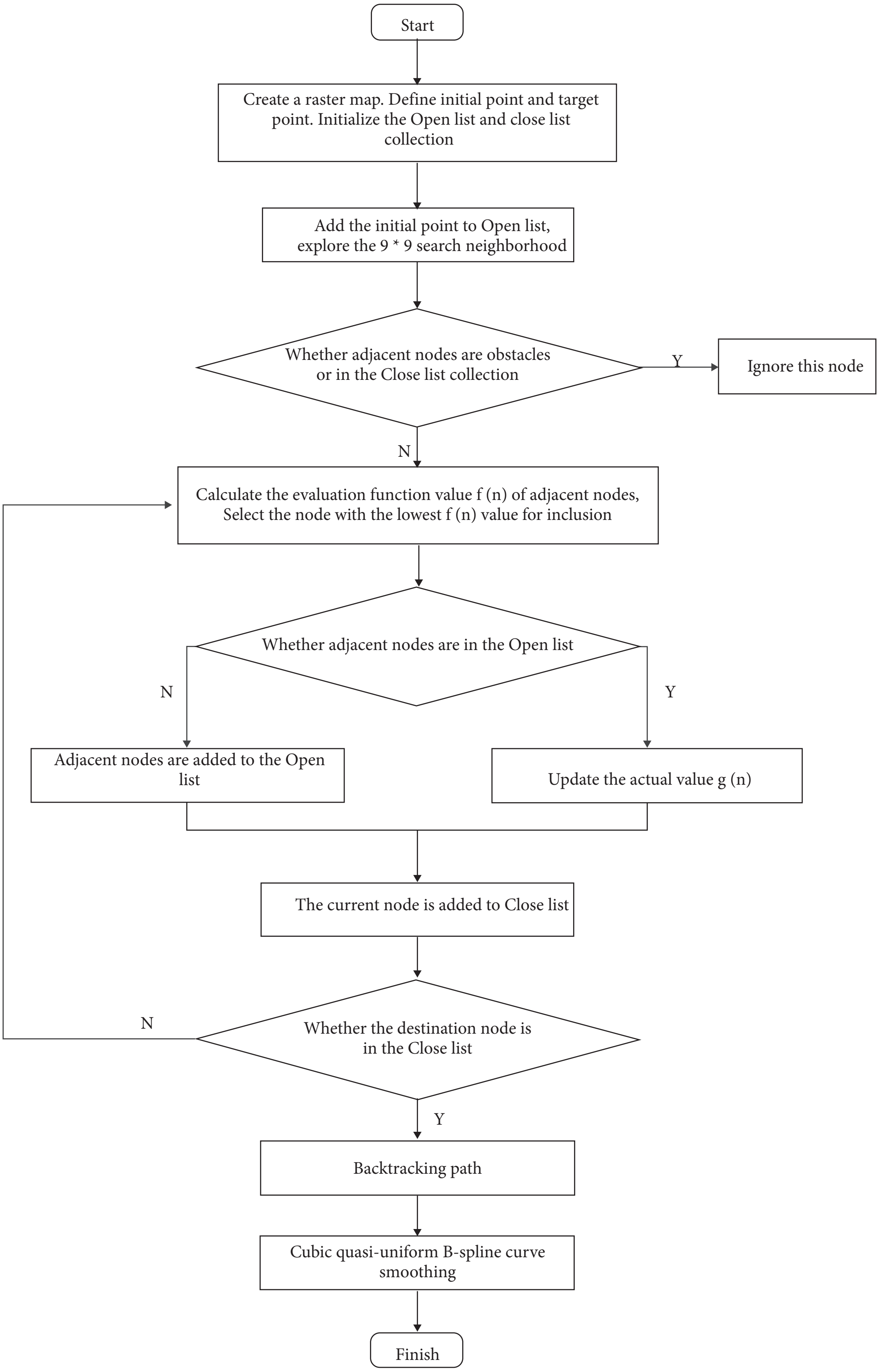

Start
Create a raster map. Define initial point and target point. Initialize the Open list and close list collection
Add the initial point to Open list, explore the 9 * 9 search neighborhood
Whether adjacent nodes are obstacles or in the Close list collection
Y
Ignore this node
N
Calculate the evaluation function value f (n) of adjacent nodes, Select the node with the lowest f (n) value for inclusion
Whether adjacent nodes are in the Open list
N
Y
Adjacent nodes are added to the Open list
Update the actual value g (n)
The current node is added to Close list
N
Whether the destination node is in the Close list
Y
Backtracking path
Cubic quasi-uniform B-spline curve smoothing
Finish

Figure 13: Improve A^* algorithm flow.

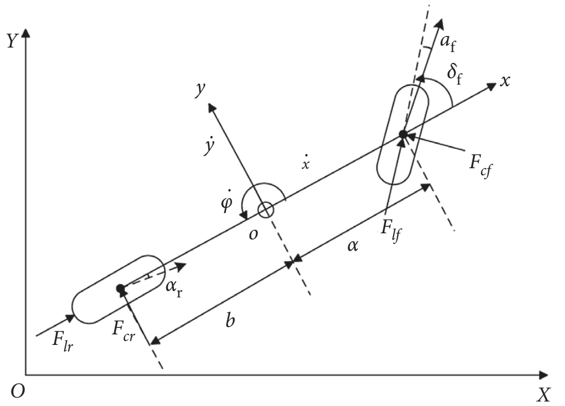


Figure 14: Three degrees of freedom vehicle dynamics model.

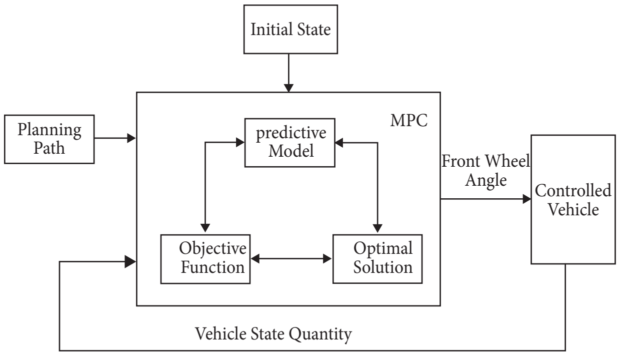


Figure 15: Tracking control strategy.

Table 2: Vehicle parameter list.

Parameters/units	Value
Vehicle mass/Kg	1270
Distance from the center of mass to the front axis/mm	1015
Moment of inertia/Kg·m^2	2875
Front-wheel cornering stiffness/N/rad	−56000
Wheelbase of the front axle/mm	1675
Height of the center of mass/mm	540
Distance from the center of mass to the rear axis/mm	1795
Effective radius of wheel/mm	350
Rear wheel cornering stiffness/N/rad	−34000
Wheelbase of the rear axle/mm	1675

where $\widetilde{\xi}(k)$ is the state matrix of state quantity at k moment and control quantity at *k-1* moment, $\widetilde{\xi}(k|t) = \begin{bmatrix} \xi(k|t) \\ u(k-1|t) \end{bmatrix}$; $\widetilde{A}_{k,t} = \begin{bmatrix} A_{k,t} & B_{k,t} \\ 0 & I \end{bmatrix}$, $\widetilde{B}_{k,t} = \begin{bmatrix} B_{k,t} \\ I \end{bmatrix}$; $\Delta u(k|t)$ is the control increment, $\Delta u(k|t) = u(k|t) - u(k-1|t)$; $\widetilde{\lambda}(k|t)$ is the output of the system at time k.

Assume that the prediction time domain of the MPC controller is *Np*, and the control time domain is *Nc*. The predicted output of the system at time k is obtained as

$$Y(k|t) = \psi_k \widetilde{\xi}(k|t) + \Theta_k \Delta U(k|t), \tag{23}$$

where $Y(k|t)$ is the output matrix in the prediction time domain, $Y(k) = \begin{bmatrix} \widetilde{\lambda}(k+1|t) \\ \widetilde{\lambda}(k+2|t) \\ \vdots \\ \widetilde{\lambda}(k+N_p|t) \end{bmatrix}$; $\Delta U(k)$ is the control increment in the control time domain *Nc*, $\Delta \mathbf{U}(k) = \begin{bmatrix} \Delta \mathbf{u}(k|t) \\ \Delta \mathbf{u}(k+1|t) \\ \vdots \\ \Delta \mathbf{u}(k+N_c-1|t) \end{bmatrix}$; ψ_k and Θ_k are coefficient matrices,

$$\psi_k = \begin{bmatrix} \widetilde{C}_{k,t}\widetilde{A}_{k,t} \\ \widetilde{C}_{k,t}\widetilde{A}_{k,t}^2 \\ \vdots \\ \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_p} \end{bmatrix},$$

$$\Theta_k = \begin{bmatrix} \widetilde{C}_{k,t}\widetilde{B}_{k,t} & 0 & \cdots & 0 \\ \widetilde{C}_{k,t}\widetilde{A}_{k,t}\widetilde{B}_{k,t} & \widetilde{C}_{k,t}\widetilde{B}_{k,t} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_c-1}\widetilde{B}_{k,t} & \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_c-2}\widetilde{B}_{k,t} & \cdots & \widetilde{C}_{k,t}\widetilde{B}_{k,t} \\ \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_c}\widetilde{B}_{k,t} & \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_c-1}\widetilde{B}_{k,t} & \cdots & \widetilde{C}_{k,t}\widetilde{A}_{k,t}\widetilde{B}_{k,t} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_p-1}\widetilde{B}_{k,t} & \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_p-2}\widetilde{B}_{k,t} & \cdots & \widetilde{C}_{k,t}\widetilde{A}_{k,t}^{N_p-N_c-1}\widetilde{B}_{k,t} \end{bmatrix}.$$

The objective function is to ensure that the vehicle tracks the planned path with minimum error. The objective function of the controller is

$$J = \sum_{i=1}^{N_p} \left\| \lambda(k+i|t) - \lambda_{ref}(k+i|t) \right\|_Q^2 + \sum_{i=1}^{N_c-1} \left\| \Delta u(k+i|t) \right\|_R^2 + \rho\varepsilon^2, \tag{24}$$

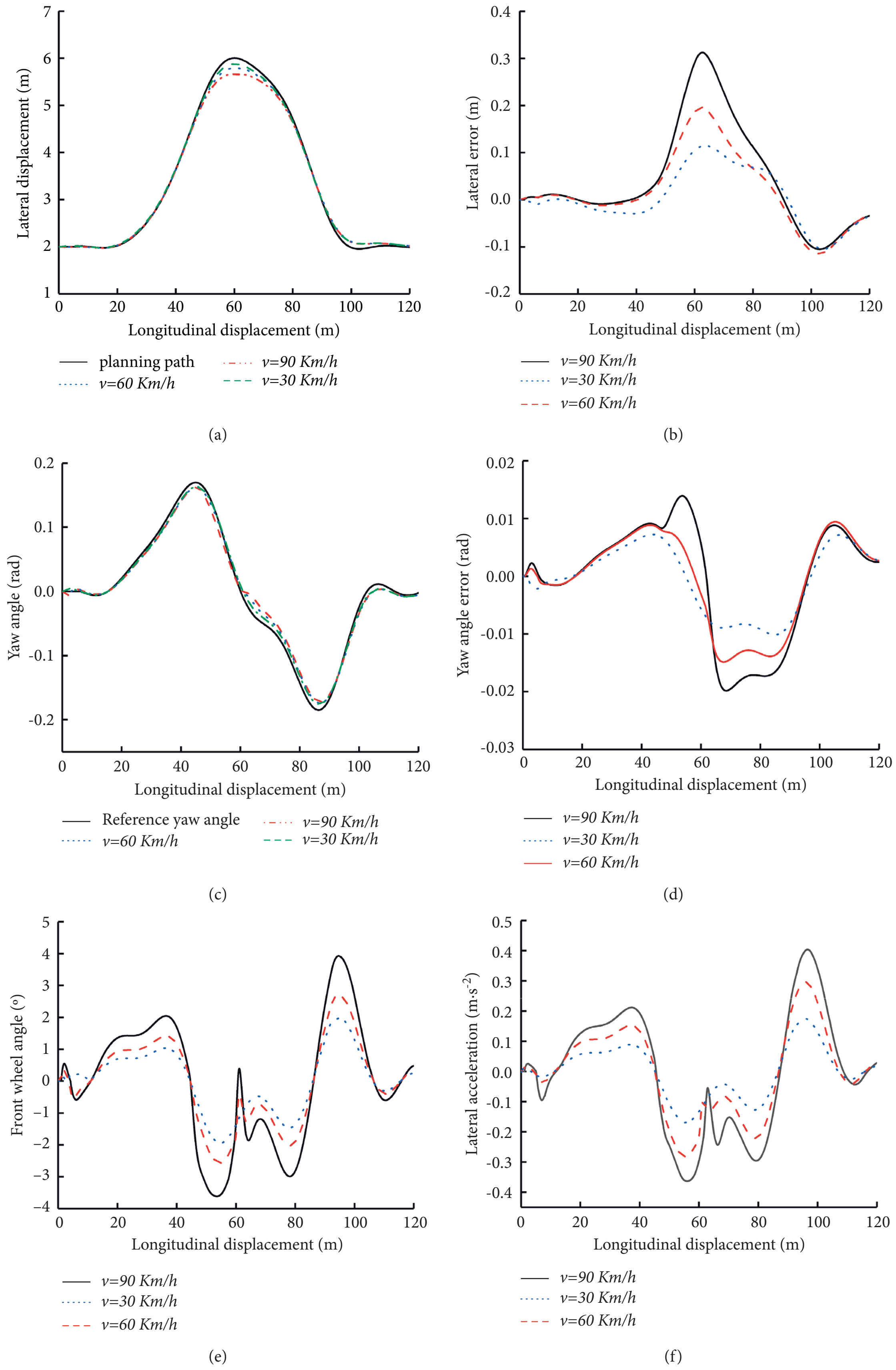
where $\lambda(k+i|t)$ is the actual output, $\lambda_{ref}(k+i|t)$ is the reference output, Q,R is the weight coefficient matrix, ρ is the relaxation factor weight coefficient, and ε is the relaxation factor.

In order to track the planned path more accurately, set the center of mass slip angle constraint as $-8^\circ \le \beta \le 8^\circ$. Set the tire sideslip Angle constraint as $-3^\circ \le \alpha \le 3^\circ$. Set the front wheel Angle range as $-12^\circ \le \delta_f \le 12^\circ$, The angle change is constrained as $-0.8^\circ \le \Delta\delta_f \le 0.8^\circ$. At the same time, the attachment condition is constrained to $\sqrt{a_x^2 + a_y^2} \le \mu g$, in order to prevent the constraint condition from being too small to cause no solution, a relaxation factor can be introduced to define the attachment condition constraint as a soft constraint: $a_{y,\min} - \varepsilon \le a_y \le a_{y,\max} + \varepsilon$.

The quadratic programming method is used to solve the objective function, and the objective function is converted into the standard type:

$$J = \frac{1}{2}[\Delta U(k), \varepsilon]^T H [\Delta U(k), \varepsilon] + G[\Delta U(k), \varepsilon]^T, \tag{25}$$

where *H*, *G* is the coefficient matrix, $H = \begin{bmatrix} 2(\Theta_k^T Q \Theta_K + R)0 \\ 0\,2\rho \end{bmatrix}$, $G = [2E^T(k)\,Q\Theta_k 0]$; $E(k)$ is the output deviation matrix in the prediction time domain.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 16: Continued.

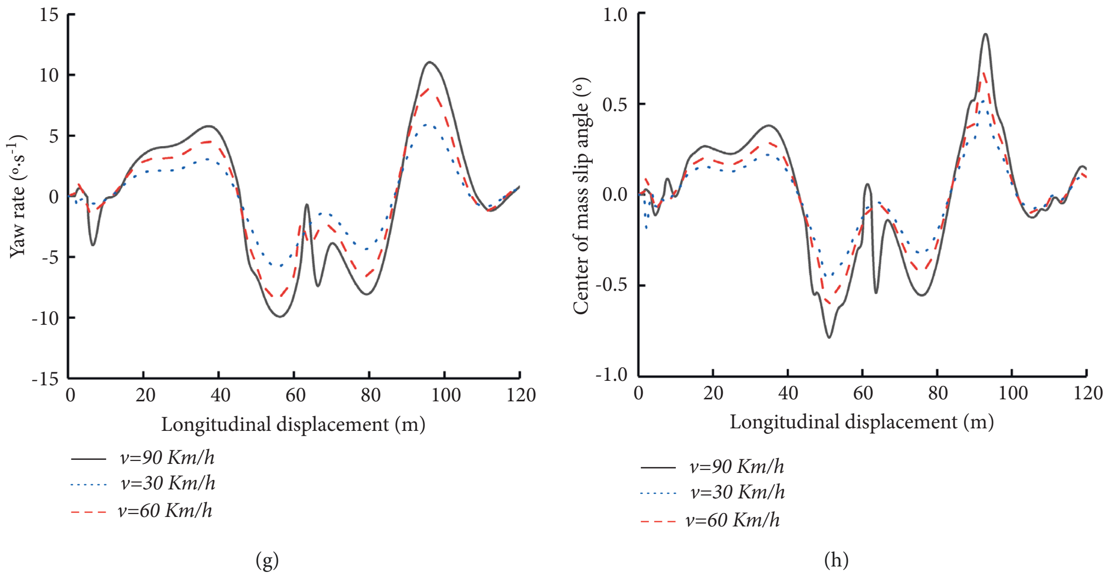


Figure 16: Tracking control renderings. (a) Lateral position tracking comparison. (b) Lateral position tracking error. (c) Yaw angle tracking comparison. (d) Yaw angle tracking error. (e) Front-wheel angle diagram. (f) Lateral acceleration diagram. (g) Yaw rate diagram. (h) Center of mass slip angle diagram.

The optimal control increment of the system is obtained by solving the following constrained problems:

$$\begin{cases} \min\limits_{\Delta u(k),\varepsilon} \dfrac{1}{2}[\Delta U(k), \varepsilon]^T H [\Delta U(k), \varepsilon] + G[\Delta U(k), \varepsilon]^T, \\ \text{s.t.} \Delta U_{\min} \le \Delta U_t \le \Delta U_{\max}, \\ U_{\min} \le U_t \le U_{\max}, \\ y_{hc,\min} \le y_{hc} \le y_{hc,\max}, \\ y_{sc,\min} - \varepsilon \le y_{sc} \le y_{sc,\max} + \varepsilon. \end{cases} \tag{26}$$

To solve (26), the optimal control increment of front wheel rotation angle can be obtained as follows:

$$\Delta U^*(k) = \left[\Delta u^*(k), \Delta u^*(k+1|t), \ldots, \Delta u^*\left(k+N_c-1|t\right)\right]^T, \tag{27}$$

where $\Delta u^*(k), \Delta u^*(k+1|t), \ldots, \Delta u^*(k+N_c-1|t)$ is the system control input at time $k, k+1, \ldots, k+N_c-1$.

The first element of the control sequence is input as the actual control increment to obtain the final control quantity:

$$u(k) = u(k-1) + \Delta u^*(k). \tag{28}$$

4. The Simulation Verification

CarSim and MATLAB/Simulink are used to conduct co-simulation experiments to verify the effectiveness of the planning algorithm and tracking control strategy proposed in this study. The main parameters of the vehicle are shown in Table 2.

In order to verify the feasibility of the planned path and the effectiveness of the tracking control strategy, part of the planned path is selected for the path tracking test. Select the path from the starting point to bypass the first obstacle for the tracking test. The adhesion coefficient of dry and good asphalt pavement can reach 0.7–0.8, in this study, the road surface adhesion coefficient is set as 0.75. The tracking control test of the planned path is carried out at high, medium, and low speed respectively, and the simulation results are shown in Figure 16.

The simulation results show that the planned path can meet the tracking requirements of autonomous vehicles and achieve good tracking effects at different speeds. As can be seen from Figures 16(a) and 16(b), as the vehicle speed increases, the tracking error of the lateral position also increases. When the vehicle speed is 90 Km/h, the maximum lateral position error is 0.31 m. When the vehicle speed is 60 Km/h, the maximum lateral position error is 0.19 m. When the speed is 30 Km/h, the lateral position error of the vehicle can be controlled within 0.11 m, and the errors are all within the allowable range. As can be seen from Figures 16(c) and 16(d), the yaw angle tracking error at different vehicle speeds can be controlled within 0.02 rad and the difference is not large, indicating that the tracking process of the vehicle is relatively stable. When the vehicle speed is 30 Km/h, the yaw angle tracking effect is the best, and the yaw angle error is within 0.01 rad. As can be seen from Figures 16(e) and 16(f), the higher the vehicle speed, the greater the change of the front wheel angle, but there is no step change in the front wheel angle and lateral acceleration at different vehicle speeds, indicating that the tracking process is relatively stable. It can be seen from Figure 16(g) that during the turning process of the vehicle, the yaw rate changes greatly, when the vehicle speed is 90 Km/h, the maximum yaw rate reaches 11°/s, within the permissible limits. It can be seen from Figure 16(h) that after adding the vehicle center of mass slip angle constraint, the change of the center of mass slip angle can be controlled within 1°, and the change is smooth, indicating that the driving stability of the vehicle tracking process is better.

5. Conclusions

To solve the problem of motion planning and tracking control of autonomous vehicles in obstacle avoidance conditions, this study proposes a motion planning algorithm by improved A^* algorithm and a tracking control strategy based on model predictive control theory. Aiming at dealing with the shortcomings of the traditional A^* algorithm applied to the motion planning of autonomous vehicles, the traditional A^* algorithm is improved. By expanding the search neighborhood method, the planning efficiency of the algorithm is improved, and the improved artificial potential field method is combined with the A^* algorithm for obstacle avoidance and improving the driving safety of the vehicle. At the same time, the cubic quasi-uniform B-spline curve is used to smooth the planned path. The tracking control strategy is designed based on the model predictive control theory, and related constraints are considered to realize the accurate tracking of the planned path. By analyzing the simulation results, the following conclusions can be drawn as follows:

(1) The improved A^* algorithm can effectively plan the obstacle avoidance path under corresponding conditions. Compared with the traditional A^* algorithm, the algorithm reduces the planning time and improves the planning efficiency. The planning path considers the minimum safe distance between obstacles and vehicles. The planned paths are smoother and more suitable for autonomous vehicles to track.

(2) The designed tracking control strategy reflect a good control effect on the path tracking at different vehicle speeds. The average lateral position tracking error can be controlled within 0.2 m, the yaw Angle tracking error is controlled within 0.02 rad, and the change of center of mass slip angle is controlled within 1°, which verifies the feasibility of planning path and effectiveness of tracking control strategy.

The simulation results demonstrate great reference value for the subsequent real vehicle test. In the future work, the proposed motion planning algorithm and tracking control strategy will be deployed into the real vehicle testing, and the longitudinal speed of the vehicle will be adaptively adjusted according to the paths, and the tracking error of the vehicle would be reduced through the changeable longitudinal speed.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Science Foundation of China (51675257, 51305190) ,General program of Natural Science Foundation of Liaoning Province in 2022 (2022-MS-376) and ”Liaoning BaiQianWan Talents Program.

References

[1] H. Min, X. Xiong, P. Wang, and Y. Yu, “Autonomous driving path planning algorithm based on improved A^* algorithm in unstructured environment,” *Proceedings of the Institution of Mechanical Engineers - Part D: Journal of Automobile Engineering*, vol. 235, no. 2-3, pp. 513–526, 2021.

[2] Q. Yao, Z. Zheng, L. Qi et al., “Path planning method with improved artificial potential field—a reinforcement learning perspective,” *IEEE Access*, vol. 8, pp. 135513–135523, 2020.

[3] W. Rahmaniar and A. E. Rakhmania, “Mobile robot path planning in a trajectory with multiple obstacles using genetic algorithms,” *Journal of Robotics and Control (JRC)*, vol. 3, no. 1, pp. 1–7, 2021.

[4] H. Wang, G. Li, J. Hou, L. Chen, and N. Hu, “A path planning method for underground intelligent vehicles based on an improved RRT ∗ algorithm,” *Electronics*, vol. 11, no. 3, p. 294, 2022.

[5] B. Fu, L. Chen, Y. Zhou et al., “An improved A algorithm for the industrial robot path planning with high success rate and short length,” *Robotics and Autonomous Systems*, vol. 106, pp. 26–37, 2018.

[6] T. XiangRong, Z. Yukun, and J. XinXin, “Improved A-star algorithm for robot path planning in static environment,” *Journal of Physics: Conference Series*, vol. 2021, Article ID 012067, 1792 pages, 2021.

[7] S. Erke, D. Bin, N. Yiming, Z. Qi, X. Liang, and Z. Dawei, “An improved A-Star based path planning algorithm for autonomous land vehicles,” *International Journal of Advanced Robotic Systems*, vol. 17, no. 5, Article ID 172988142096226, 172988142096617 pages, 2020.

[8] L. Liu, J. Yao, D. He et al., “Global dynamic path planning fusion algorithm combining jump-A algorithm and dynamic window approach,” *IEEE Access*, vol. 9, pp. 19632–19638, 2021.

[9] X. Zhong, J. Tian, H. Hu, and X. Peng, “Hybrid path planning based on safe A algorithm and adaptive window approach for mobile robot in large-scale dynamic environment,” *Journal of Intelligent and Robotic Systems*, vol. 99, no. 1, pp. 65–77, 2020.

[10] X. Xiong, H. Min, Y. Yu, and P. Wang, “Application improvement of A algorithm in intelligent vehicle trajectory planning,” *Mathematical Biosciences and Engineering*, vol. 18, no. 1, pp. 1–21, 2021.

[11] M. Elbanhawi, M. Simic, and R. Jazar, “Receding horizon lateral vehicle control for pure pursuit path tracking,” *Journal of Vibration and Control*, vol. 24, no. 3, pp. 619–642, 2018.

[12] T. S. Dang, D. T. Duong, and V. C. A. Le The, "A combined backstepping and adaptive fuzzy PID approach for trajectory tracking of autonomous mobile robot," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 43, no. 3, pp. 1–13, 2021.

[13] H. Peng, F. Li, J. Liu, and Z. Ju, "A symplectic instantaneous optimal control for robot trajectory tracking with differential-algebraic equation models," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 5, pp. 3819–3829, 2020.

[14] P. Wang, S. Gao, L. Li, S. Cheng, and L. Zhao, "Automatic steering control strategy for unmanned vehicles based on robust backstepping sliding mode control theory," *IEEE Access*, vol. 7, pp. 64984–64992, 2019.

[15] J. Hu, S. Xiong, J. Zha, and C. Fu, "Lane detection and trajectory tracking control of autonomous vehicle based on model predictive control," *International Journal of Automotive Technology*, vol. 21, no. 2, pp. 285–295, 2020.

[16] T. Novi, A. Liniger, R. Capitani, and C. Annicchiarico, "Real-time control for at-limit handling driving on a predefined path," *Vehicle System Dynamics*, vol. 58, no. 7, pp. 1007–1036, 2020.

[17] K. Zhang, Q. Sun, and Y. Shi, "Trajectory tracking control of autonomous ground vehicles using adaptive learning MPC," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 12, pp. 5554–5564, 2021.

[18] Y. Xu, W. Tang, B. Chen, L. Qiu, and R. Yang, "A model predictive control with preview-follower theory algorithm for trajectory tracking control in autonomous vehicles," *Symmetry*, vol. 13, no. 3, p. 381, 2021.

[19] K. Berntorp, R. Quirynen, T. Uno, and S. Di Cairano, "Trajectory tracking for autonomous vehicles on varying road surfaces by friction-adaptive nonlinear model predictive control," *Vehicle System Dynamics*, vol. 58, no. 5, pp. 705–725, 2020.

[20] R. W. Liu, M. Liang, J. Nie, W. Y. B. Lim, Y. Zhang, and M. Guizani, "Deep learning-powered vessel trajectory prediction for improving smart traffic services in maritime internet of things," *IEEE Transactions on Network Science and Engineering*, vol. 2022, Article ID 9674855, 1 page, 2022.

[21] Y. Shin and E. Kim, "Hybrid path planning using positioning risk and artificial potential fields," *Aerospace Science and Technology*, vol. 2021, no. 6, Article ID 106640, 112 pages, 2021.

[22] D. S. Pae, G. H. Kim, T. K. Kang, and M. T. Lim, "Path planning based on obstacle-dependent Gaussian model predictive control for autonomous driving," *Applied Sciences*, vol. 11, no. 8, p. 3703, 2021.