A Control Strategy for Improving the Accuracy of Lateral Tracking of Autonomous Vehicles

Ping Liu (✉ pingliu@swjtu.edu.cn)

Southwest Jiaotong University School of Mechanics and Engineering

Zibin Liu

Southwest Jiaotong University School of Mechanics and Engineering <https://orcid.org/0000-0003-3534-800X>

Chaohui Wu

Southwest Jiaotong University School of Mechanics and Engineering

Zhisong Wan

Southwest Jiaotong University School of Mechanics and Engineering

Weiping Ding

Southwest Jiaotong University School of Mechanics and Engineering

Haibo Huang

Southwest Jiaotong University School of Mechanics and Engineering

Research Article

Keywords: Accurate lateral tracking, Cubic Bezier curve, Continuous adaptive piecewise fitting method, Improved pure pursuit algorithm, Autonomous vehicles

Posted Date: February 15th, 2022

DOI: <https://doi.org/10.21203/rs.3.rs-1278629/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

A control strategy for improving the accuracy of lateral tracking of autonomous vehicles

Ping Liu^{1,2}, Zibin Liu^{1,2}, Chaohui Wu^{1,2}, Zhisong Wan^{1,2}, Weiping Ding^{1,2*} and Haibo Huang^{1,2*}

Abstract

Lateral tracking is one of the key technologies for autonomous vehicles. It is particularly important when vehicles are traveling at low speeds in narrow and complex scenes, such as parking lots and alleys. In order to improve the effective and accurate control of the traditional Pure Pursuit algorithm (PP) in the lateral tracking of low-speed autonomous vehicles, a continuous adaptive piecewise fitting method based on cubic Bezier curve was proposed to smooth the reference path, and then search the look-ahead point according to the geometric relationship between the vehicle and the reference path: An improved search algorithm based on vehicle speed and road curvature was used to search the look-ahead point. If the reference path curvature changes too much, the preliminarily obtained look-ahead point are modified, and finally the improved control algorithm (Improve Pure Pursuit Method, IPP) was obtained to calculate the steering angle of the front wheel. Subsequently, the Carsim/Simulink co-simulation model and the autonomous vehicle platform were established for verification. Simulation and real Vehicle verification show that, compared with the pure pursuit algorithm based on Vehicle speed control (VPP) and Fuzzy Logic control pure Pursuit (FPP), IPP can perform better in complex low-speed lateral tracking scenes, reduce the problem of cutting-corner when the vehicle enters a corner, and improve the lateral tracking accuracy.

Keywords: Accurate lateral tracking; Cubic Bezier curve; Continuous adaptive piecewise fitting method; Improved pure pursuit algorithm; Autonomous vehicles

1 Introduction

In recent years, as people pay more attention to the safety, environmental protection and convenience of using cars, some of the work has also being done to improve the comfort of cars [1, 2], the very important trend is that Intelligent Traffic System (ITS) has developed rapidly [3, 4, 5, 6, 7]. As one of the important links of ITS, autonomous vehicles have great application potential in many fields, so they are widely concerned by the industry and academia [8, 9, 10].

Path tracking control is one of the key technologies of automatic driving [11], model-based method and geometric method are widely used in vehicle path tracking [12]. Path tracking control is divided into longitudinal tracking control and lateral tracking control, lateral tracking control controls the steering wheel angle to make the vehicle follow the desired path under the premise of satisfying safety, comfort and stability. Large errors in the lateral tracking process will increase

the rate of traffic accidents and delay the efficiency of traffic flow, the tracking error mainly depends on the front wheel angle of the vehicle at the current moment and the smoothness of the reference path. Therefore, it is necessary to give a reasonable reference path curve fitting method, meanwhile, a front wheel steering angle calculation algorithm that can quickly and stably converge to the reference path and can accurately track laterally is designed.

At present, many researches have been carried out on the lateral tracking control algorithm. And lateral tracking control methods mainly include: PID control [13], pure pursuit control [14], sliding mode control [15], LQR(linear quadratic regulator) control [16], model predictive control [17] and fuzzy control [18].

Pure pursuit algorithms mainly use geometric principles [19], people have studied it a long time ago, and it is widely used in the field of robot tracking because of its simplicity and ease of implementation. Related applications are TierIV Autoware [20]and MIT Drake [21]. Among them, the bicycle model based on four-wheel Ackerman steering kinematics [22] simplified form is used to derive the steering angle of pure

*Correspondence: dwp@swjtu.edu.cn; huanghaibo214@my.swjtu.edu.cn

²Engineering Research Center of Advanced Driving Energy-saving Technology, Ministry of Education

Full list of author information is available at the end of the article

pursuit algorithm:

$$\begin{cases} \dot{x}_c = v \cos(\theta) \\ \dot{y}_c = v \sin(\theta) \\ \dot{\theta} = \frac{v \tan(\phi)}{L} \end{cases} \quad (1)$$

where x_c and y_c are the horizontal and vertical coordinates of the center of the rear axle of the vehicle in the global coordinate system, v is the longitudinal speed of the vehicle, θ is the heading angle of the vehicle, L is the wheelbase of the vehicle, ϕ is the front wheel steering angle of the vehicle.

In previous work [14, 19, 23, 24, 25], the traditional pure pursuit algorithm follows the following process (assuming T is a discrete reference path point, the schematic diagram of the traditional pure pursuit algorithm is shown in Figure 1):

- 1) Give the vehicle rear axle center coordinate X_t and heading angle θ_t in the global coordinate system at time t ;
- 2) Find the point P_t closest to the center of the rear axle of the vehicle on the reference path T (the purple discrete point in Figure 1(a));
- 3) Search from the closest point P_t and find the target point G_{pt} on the reference path T (the blue circle in Figure 1(a));
- 4) Calculate the angle α_{pt} , and use the Ackerman steering theorem (shown in Figure 1(b)) to give the expression of the front wheel rotation angle ϕ_t as equation (2);

$$\begin{cases} \alpha_t = \arctan \frac{G_{pt}(2) - X_t(2)}{G_{pt}(1) - X_t(1)} \\ \phi_t = \arctan \frac{2L \sin(\alpha_t)}{\|G_{pt}, X_t\|_2} \end{cases} \quad (2)$$

where $\frac{2 \sin(\alpha_t)}{\|G_{pt}, X_t\|_2}$ is the curvature corresponding to the turning radius of the rear axle center of the vehicle at time t .

- 5) After applying the front-wheel steering angle ϕ_t to the vehicle at time t in 4), it starts from 1) at time $t + 1$ and loops until the vehicle reaches the end of the reference path.

The look-ahead distance (the Euclidean distance from G_{pt} to X_t in Figure 1(a)) is an important parameter in the pure pursuit algorithm, its value has a key influence on obtaining the best real-time front wheel steering angle, and directly affects the tracking effect. The look-ahead distance determined by the traditional method is constant. If the selected look-ahead distance is small, the vehicle tracking reference

path control will be unstable and overshoot will occur, resulting in oscillation (as shown in the upper figure of Figure 2(a)); If a larger look-ahead distance is selected, a larger tracking error will occur when the vehicle is tracking laterally (as shown in the bottom figure of Figure 2(a)). In addition, when the curvature of the current look-ahead point and the curvature of the vehicle steering are greatly different, under the traditional pure tracking method, the vehicle will have a cutting-corner phenomenon, especially when the vehicle enters a curve with a large curvature from a small curvature (as shown in Figure 2(b)).

Many scholars have improved the traditional pure pursuit algorithm, which is mainly reflected in the determination of the look-ahead point. On the one hand, the look-ahead distance is defined as the gain of vehicle speed. For example, Tusng-Ming Hsu et al. [26] defined the look-ahead distance as a quadratic function of vehicle speed, and the coefficient of the function takes into account the vehicle characteristic parameters such as the minimum turning radius. On the other hand, in addition to vehicle speed, some error factors in the tracking process are also taken into consideration in determining the look-ahead point [27], the gain coefficient of the distance is adjusted dynamically. Also, Fu Qiang [28] dynamically adjusts the gain of the look-ahead distance based on conventional vehicle speed adjustment and considering the trend of lateral tracking error, however, the robustness of the adjustment parameter factor to the reference path is poor. In addition, previous studies have shown that if the road curvature is ignored in the autonomous lateral tracking control, it will cause a large lateral tracking error. Therefore, parameterized path fitting methods such as polynomial curves, splines, spirals, and Bezier curves [29] are widely used to obtain reference road curvature information, but how to combine different road curvatures in the autonomous vehicle environment, and the curve fitting method and adaptive segmentation of more suitable order is also a problem worthy of study.

Based on the above problems, this paper proposes a lateral tracking algorithm to improve accuracy in a low-speed environment with the help of the Carsim simulation platform and the autonomous driving car platform, and designs a lateral tracking controller for verification. The main contributions are as follows: 1) Aiming at the problem that the reference path collected by the integrated navigation system of the autonomous driving car platform is not smooth, a continuous adaptive curve fitting method based on cubic Bezier curve is proposed to smooth and obtain the road curvature; 2) An improved pure pursuit algorithm is proposed to search for the look-ahead point according to the relative position of the vehicle and the reference path, the speed of the vehicle and the curvature

of the road. For the cutting-corner phenomenon of the vehicle from a small curvature to a large curvature, the look-ahead point is selected outside the reference path, thereby improve the tracking accuracy of the algorithm; 3) The algorithm proposed in this paper has been verified on Carsim and the autonomous driving car platform, which improves the accuracy and feasibility of the lateral control of the autonomous vehicles. In this paper, the roadmap of the lateral tracking control technology for autonomous driving is designed as shown in Figure 3, the X1 system is the integrated navigation system in this paper.

2 Lateral tracking control strategy

In order to improve the accuracy, firstly, the discrete path points are fitted with a continuous adaptive smooth curve, and then the pure pursuit algorithm is improved from two aspects according to the relationship between the vehicle and the reference path. On this basis, the front wheel steering angle is calculated for precise lateral control of the vehicle.

2.1 Continuous adaptive segmented fitting method based on cubic Bezier curve

In order to improve the accuracy, a continuous adaptive fitting method is proposed. The main ideas are: determine the fitting method; calculate the fitting residual and segment by the residual threshold; adjust the segment control points to smooth the reference path.

The reference path provided by the road network (collected by the integrated navigation system in this article) is often a series of discrete points that do not contain the road curvature, and it's not smooth. In order to better perform accurate lateral tracking control, it is necessary to obtain the reference road curvature. There are many curve parameter fitting methods at present, but how to choose a better curve order to compromise the fitting accuracy, the difficulty of solving and the amount of calculation, and how to appropriately segment the original reference path is a problem worthy of study. In this paper, Bezier curve is selected for road curve fitting, and its curve expression is as in Equation (3). Regarding the order of fitting, a road is built in Prescan/simulink, and a Logitech steering wheel is used to collect a reference path. The first to sixth degree Bezier curves are used for fitting and comparison, and the fitting effect is shown in Figure 4.

$$q(\tau_i) = \sum_{k=0}^m \binom{m}{k} P_k (1 - \tau_i)^{m-k} \tau_i^k, \tau_i \in [0, 1] \quad (3)$$

where m is the order of the Bezier curve, $q(\tau_i)$ is the interpolation point at the parameter τ_i , and P_k is the k -th control point.

It can be seen from Figure 4 that as the order of the Bezier curve increases, the fitting accuracy also increases, but after more than four times, over-fitting occurs, and the six-order Bezier curve fitting error increases significantly. In order to quantitatively analyze the fitting effect, a fitting evaluation standard is proposed: SSE , $RMSE$ and fitting time t .

$$SSE = \sum_{i=1}^n \omega_i (p_i - \hat{p}_i) \quad (4)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n \omega_i (p_i - \hat{p}_i)^2} \quad (5)$$

where p_i and $\hat{p}_i$ is the i -th original path point and the fitted path point, $\omega_i \geq 0$.

A smooth reference path is essential for accurate lateral tracking, from Table 1, while ensuring the accuracy of curve fitting, and considering the amount of calculation, the cubic Bezier curve is selected for fitting to obtain a smooth reference path, which lays the foundation for accurate lateral tracking. The cubic Bezier curve expression as in formula (6):

$$q(\tau_i) = (1 - \tau_i)^3 P_0 + 3\tau_i(1 - \tau_i)^2 P_1 + 3\tau_i^2(1 - \tau_i) P_2 + \tau_i^3 P_3 \quad (6)$$

Subsequently, the continuous adaptive segmented fitting method is used to process the reference path, and the least square method is selected to fit the midpoint of each segment of the reference path. p_i is the discrete waypoint given by the reference path, and the sum of squares of the fitting residual can be expressed as

$$S = \sum_{i=1}^n (p_i - q(\tau))^2 \quad (7)$$

where n is the number of discrete waypoints excluding the first and last two points.

The intermediate control points in the solution process should satisfy $\frac{\partial S}{\partial P_1} = 0$ and $\frac{\partial S}{\partial P_2} = 0$, P_1 and P_2 can be expressed as:

$$\begin{cases} P_1 = \frac{A_2 C_1 - A_{12} C_2}{A_1 A_2 - A_{12}^2} \\ P_2 = \frac{A_1 C_2 - A_{12} C_1}{A_1 A_2 - A_{12}^2} \end{cases} \quad (8)$$

where $A_1 = 9 \sum_{i=1}^n \tau_i^2 (1 - \tau_i)^4$, $C_1 = \sum_{i=1}^n 3\tau_i (1 - \tau_i)^2 [p_i - (1 - \tau_i)^3 P_0 - \tau_i P_3]$, $A_2 = 9 \sum_{i=1}^n \tau_i^4 (1 - \tau_i)^2$,

$$A_{12} = 9 \sum_{i=1}^n \tau_i^3 (1 - \tau_i)^3, C_2 = \sum_{i=1}^n 3\tau_i^2 (1 - \tau_i) [p_i - (1 - \tau_i)^3 P_0 - \tau_i^3 P_3].$$

Substitute the obtained control point parameters into (6), and obtain the residuals between the fitted path points and the original path points (Find the point closest to the original path point for each path point obtained by the fitting, and calculate the Euclidean distance, which is the residual of each fitted path point), if the residual error requirement is not met, segment the reference path and repeat (6) (8) until the residual error threshold requirement is met. After obtaining the adaptive segmentation of the reference path that meets the requirements of the residual threshold, in order to ensure that the curvature of the connection points of the two adjacent segments changes smoothly, the connection points of the adjacent two segments of the curve should meet the G1 continuity (The direction of the tangent vector at the connection point is the same), let $Q_k, P_k (k = 0, 1, 2, 3)$ represent the control points of the two curves respectively. As shown in Figure 5, the control points of the two adjacent curves should satisfy the formula (9):

$$\begin{cases} Q_3 = P_0 \\ P_1 = P_0 + s_b G_b \end{cases} \quad (9)$$

where G_b is the direction vector, s_b is the parameter to be calibrated.

If the two adjacent segments of the curve after adaptive segmentation do not satisfy the formula (9), the middle control point calculated by formula (8) will be continuously corrected by satisfying G_1 at the junction of the two curves:

$$\begin{cases} \tilde{P}_1 = \frac{X_1 - X_2 P_0 - 3X_{31} P_2 - X_4 P_3}{3X_{32}} \\ \tilde{P}_2 = \frac{X_1 - X_2 P_0 - 3X_{32} P_1 - X_4 P_3}{3X_{31}} \end{cases} \quad (10)$$

where $X_1 = \sum_{i=1}^n p_i \tau_i (1 - \tau_i)^2$, $X_2 = \sum_{i=1}^n \tau_i (1 - \tau_i)^5$, $X_{31} = \sum_{i=1}^n \tau_i^3 (1 - \tau_i)^3$, $X_{32} = \sum_{i=1}^n \tau_i^2 (1 - \tau_i)^4$, $X_4 = \sum_{i=1}^n \tau_i^4 (1 - \tau_i)^2$.

The continuous adaptive reference path fitting process based on cubic Bezier curve is shown in Figure 6: First import the reference path collected by the integrated navigation system (X1) and convert it into coordinates in the global coordinate system, and sparse them (The waypoints collected by the high-frequency integrated navigation system are often dense). Then select the start and end points of the original reference path for the first and last two control points of the cubic Bezier curve. In each iterative process of curve fitting, first according to formula (7) and formula (8) solve the position of the middle control point

of each Bezier curve. Then calculate the interpolation value corresponding to the original path point according to formula (6), the circular path needs to consider calculating the cumulative distance for interpolation, and then calculate the residual E_r . If the residual is greater than the artificially set threshold $\varepsilon_r (E_r \geq \varepsilon_r)$, then select the path point corresponding to the maximum residual point to segment the reference path. Repeat this process until all curve fitting points are satisfied $E_r \leq \varepsilon_r$, this article combines references [29], the fitting residual threshold is more stringent as 0.4m. It should be noted that the fitting curve obtained at this time is not continuous at the segmentation. After segmentation, the control points of several segments of Bessel curves after segmentation are adjusted by using Equation (10) considering curvature and residual.

In order to illustrate the effectiveness of the continuous adaptive reference path curve fitting method, the reference path collected by the integrated navigation system in the autonomous driving car in this paper is selected for verification, and the reference path is shown in Figure 7 (the number of waypoints collected was sparse to 1000). Figure 7 is the reference path, original reference path and segmented control points of the continuous adaptive segmented fitting of cubic Bezier curve, the method proposed in this paper can adaptively curve-fit the reference path, divide the original reference path into sixteen segments, and the reference path becomes smoother after fitting. Figures 8 and 9 are the residual graph of the continuous adaptive segmentation of the cubic Bezier curve and the curvature comparison of the final fitting path. It can be seen that the residual between the fitted path and the original path meets the threshold (0.4m) requirement. After continuous adaptive segmentation, the sudden change in the slope at the connection point is reduced and thus becomes smoother (the maximum curvature of the original reference path is 0.3860rad/m, mean curvature is 0.0494rad/m, the maximum curvature of the reference path after adaptive fitting is 0.2887rad/m, mean curvature is 0.0338rad/m).

2.2 Use the geometric relationship between the vehicle and the road to search and correct the look-ahead point
In this section, a method that considers the geometric relationship between the vehicle and the road is proposed to search and correct the look-ahead point. The method is mainly divided into two steps. The first step is to search the look-ahead point p_d on the reference path by improving the search algorithm on the reference path, and the vehicle can quickly converge to the reference path without oscillation. The second step is to correct the look-ahead point outside the reference path (p_l) to reduce the cutting-corner phenomenon of the vehicle entering a curve with a small curvature.

2.2.1 Improved algorithm to search look-ahead point

Generally, the distance between the vehicle and the reference path will not be too large, this article assumes that the distance will not exceed two lanes(7m), the vehicle is within this range from the reference path. The traditional pure pursuit method adopts a fixed look-ahead distance to search for the look-ahead point. The inappropriate choice of the look-ahead distance makes it slower to converge to the reference path during the tracking process. Figure 10(a) takes the center of mass of the vehicle as the center of the circle, and the look-ahead distance is the intersection of the radius and the reference path (black solid line) is the look-ahead point, but there are often two look-ahead points, figure 10(b) calculates the cumulative distance by integrating the path, and there is only one look-ahead point that meets the requirements after selecting the forward direction of the vehicle. In this paper, on the basis of Figure 10(b), the look-ahead point is searched through an improved method, which can solve the problem of slow convergence to the reference path.

Based on the work of Ahn J et al. [30], this paper proposes a improved method to the search look-ahead point. In the work that has been carried out, as shown in Figure 11, p_v is the vehicle location, p_w is the point closest to the vehicle on the reference path, $r_{min}(v)$ is the minimum turning radius of the vehicle at speed v . Find the shortest Dubins path within the search range (the red dotted line in Figure 11), the intersection of the Dubins path and the reference path (the blue dashed line in Figure 11) is the look-ahead point p_d [31].

Specifically, improvements have been made in two aspects: the candidate range of the look-ahead point and the algorithm for finding the closest point of the vehicle from the reference path.

- 1) The candidate range of the look-ahead point is improved from the minimum turning radius $r_{min}(v)$ to the maximum range (12m) at the current vehicle speed to the minimum stable look-ahead distance l_c to $l_c + d_c$ (d_c is the search range).

In previous work, the selection of the starting candidate point is unreasonable when the vehicle speed is high or low. When the vehicle speed is slightly high, the starting candidate point is too far away from the vehicle. When the vehicle speed is too close, the starting candidate point is too small from the vehicle. The minimum stability control look-ahead point is proposed to solve the above problems, the distance between the minimum stability control look-ahead point and the vehicle is the minimum stable look-ahead distance that can control the vehicle to carry out lateral tracking.

The look-ahead distance is related to both vehicle speed and road curvature. Generally, the higher the

vehicle speed, the smaller the road curvature and the greater the look-ahead distance. This paper defines the minimum stable look-ahead distance as:

$$l_c = l_0 + k_1 v + k_2 w \quad (11)$$

where v is the longitudinal speed(km/h), w is the curvature of the road(rad/m), k_1 and k_2 respectively positive and negative coefficients, l_0 is a constant.

The calculation principle of road curvature is shown in Figure 12, at a certain moment, find the path point closest to the preview point (the green circle in Figure 12) on the reference path(x_i, y_i), and the two waypoints behind the closest point ($x_{(i+1)}, y_{(i+1)}$) and ($x_{(i+2)}, y_{(i+2)}$) are selected, these three points are used to calculate the curvature of the road at the moment $w(t)$, as shown in formula (12):

$$w(t) = \frac{4(\bar{x}\hat{y} - \hat{x}\bar{y})}{k^{3/2}} \quad (12)$$

where $\bar{x} = x_{(i+2)} - x_i$, $\bar{y} = y_{(i+2)} - y_i$, $\hat{y} = (y_{(i+2)} + y_i - 2y_{(i+1)})$, $\hat{x} = (x_{(i+2)} + x_i - 2x_{(i+1)})$, $k = (x_{(i+2)} - x_i)^2 + (y_{(i+2)} - y_i)^2$.

Combine the calculation of road curvature in Figure 9, build a typical road with a road curvature of $0 \sim 0.3rad/m$ in Prescan, set the speed range of $6 \sim 50km/h$, Simulates the minimum stable look-ahead distance in the traditional pure pursuit algorithm that prevents the vehicle from oscillating during lateral tracking, as shown in Figure 13, the minimum value of the search range in the search algorithm based on the traditional Dubins curve is effectively defined in combination with vehicle speed and road curvature.

- 2) When looking for the closest point of the reference path to the vehicle, the search has been improved from the global search to the global search and the local search.

In the previous work, every time the reference path was searched for the closest point to the vehicle, it always searched within the global path range, resulting in too long time to search for the look-ahead point. For the specific pseudo code, see References [28]. The improved algorithm searches all reference path points at the initial moment and finds the index of the closest point, marked as $i_{(t-1)}$, from the next moment, search from $i_{(t-1)}$, search within two meters to find the route point closest to the vehicle at that moment(The vehicle position cannot exceed this range in a control period, mark the subscript of this point as i_{pw} , and save iteratively in subsequent searches.)

2.2.2 Correct look-ahead point

In this section, the original look-ahead point p_d is revised to p_l to reduce the phenomenon of cutting-corner

in the corners. As shown in Figure 2(b) and Figure 14, if the vehicle is tracking laterally with p_d , the front wheels will move forward in the direction of $\vec{p}_{wd}$, and $\vec{p}_{wd}$ is the vector from p_w to p_d . Therefore, a deviation from $\vec{p}_{wl}$ is caused during vehicle tracking, and $\vec{p}_{wl}$ is the tangent vector of p_w . In order to solve this problem, the vehicle carries out lateral tracking along the $\vec{p}_{wl}$ direction. As shown in Figure 14, p_w , p_d and p_l form a right-angled triangle, the degree of deviation of p_l from p_d depends on τ , and the calculation of p_l is shown in equation (13)

$$\begin{cases} x_{pl} = x_{pd} + \tau |\vec{p}_{dl}| \cos(\theta_{\vec{p}_{dl}}) \\ y_{pl} = y_{pd} + \tau |\vec{p}_{dl}| \sin(\theta_{\vec{p}_{dl}}) \end{cases} \quad (13)$$

where (x_{p_l}, y_{p_l}) and (x_{p_d}, y_{p_d}) are the coordinates of p_l and p_d respectively. $\vec{p}_{dl}$ is the vector connecting p_d and p_l . The length and direction of $\vec{p}_{dl}$ are represented by $|\vec{p}_{dl}|$ and $\theta_{\vec{p}_{dl}}$ in the global coordinate system respectively.

$$\begin{cases} |\vec{p}_{dl}| = |\vec{p}_{wd}| \sin(\theta_{dl}) \\ \theta_{\vec{p}_{dl}} = \theta_{\vec{p}_{wl}} - \text{sign}(\theta_{dl}) \frac{\pi}{2} \end{cases} \quad (14)$$

where $|\vec{p}_{wd}| = \|p_w - p_d\|$ is the length of $\vec{p}_{wd}$, $\theta_{\vec{p}_{dl}}$ is the angle between the vectors $\vec{p}_{wd}$ and $\vec{p}_{wl}$:

$$\theta_{dl} = \vec{p}_{wl} - \vec{p}_{wd} \quad (15)$$

When the vehicle is tracking laterally, it will generally be closer to the path, and θ_{dl} will not be very large. This article assumes that the value of θ_{dl} is less than 90° .

After fixing the look-ahead point to p_l , τ represents the degree of correction, and it will only be corrected when the vehicle encounters a cutting-corner phenomenon, this phenomenon occurs when the curvature of the waypoint suddenly increases. A typical example is when the vehicle enters a curve along a straight line. When $\tau = 0$, look-ahead point is p_d ; When $\tau = 1$, look-ahead point is p_l . When the vehicle is tracking along the reference path, if the curvature of p_d gradually increases compared to p_l , the value of τ will also increase, and the look-ahead point is selected to gradually approach p_l (outside the reference path).

The calculation of τ can be expressed by the shortest distance α (distance between p_v and p_w) and the curvature difference β (the curvature difference between p_d and p_w):

$$\tau = (1 - \alpha)\beta \quad (16)$$

Figure 15 shows the changing scene of τ , Figure 15 (c) and (d) show that when the curvature difference of

p_d and p_w is small, the value of τ is also very small; When α decreases and β increases, the value of τ increases; when α increases and β also increases, the value of τ decreases.

$\alpha (\in [0, 1])$ is the normalized value of the distance between p_v and p_w :

$$\alpha = \begin{cases} \frac{|p_v - p_w|}{\alpha_{max}} & \|p_v - p_w\| \leq \alpha_{max} \\ 1 & \|p_v - p_w\| \geq \alpha_{max} \end{cases} \quad (17)$$

where α_{max} is a constant. When the value of α is close to 0, it means that the vehicle is very close to the reference path, as shown in Figure 15 (a) and (c).

$\beta (\in [0, 1])$ is the normalized value of the curvature difference between p_d and p_w :

$$\beta = \begin{cases} 0 & \text{if}(cur_{p_w} \geq cur_{p_d}) \\ \frac{cur_{p_d} - cur_{p_w}}{\beta_{max}} & \text{elseif}(cur_{p_d} - cur_{p_w} \leq \beta_{max}) \\ 1 & \text{elseif}(cur_{p_d} - cur_{p_w} \geq \beta_{max}) \end{cases} \quad (18)$$

where cur_{p_w} is the curvature of the point p_w closest to the vehicle on the reference path, and cur_{p_d} is the curvature of the look-ahead point p_d , the value of β is only meaningful when cur_{p_w} is smaller than cur_{p_d} , because the value of β is needed to adjust τ , β_{max} is a constant greater than 0. When $\beta = 1$, it indicates that cur_{p_d} is much larger than cur_{p_w} , and the difference between them is close to β_{max} , as shown in Figure 15(a) and (b).

After obtaining a smooth reference path (the fitted path in Figure 7), the parameters α_{max} and β_{max} need to be analyzed and obtained, combine the curvature range of reference path 1 and lane width, take $\alpha_{max} = 1, 2, 3.5m$ and $\beta_{max} = 0.2, 0.08$ respectively, a total of six combinations. Import the reference path 1 into the Carsim/Simulink environment, set a constant longitudinal speed, and set the simulation time to 65s. The simulation shows the tracking effects of six parameter combinations, as shown in Table 1.

It can be seen from Table 2 that the best performance is when $\alpha_{max} = 1$, $\beta_{max} = 0.2$. It should be noted that only when the curvature of the current look-ahead point is greater than the curvature of the closest point to the vehicle, the look-ahead point will be corrected. These two parameters are meaningful, and α_{max} should not be too small, otherwise the control will be unstable, β_{max} should also be adjusted according to the curvature of the reference path, generally, the greater the curvature of the reference path, the greater the β_{max} .

2.3 Obtain the steering angle of the front wheels based on the look-ahead point

The look-ahead point p_l is obtained in section 2.2, T be the set of a series of reference path points, and P and θ are the positions and heading angles of other vehicles.

It is assumed that the value of the front wheel steering angle α_i depends on the possible angle range $[\alpha_{min}, \alpha_{max}]$ (determined by the limit value of the front wheel steering angle of the vehicle). Then calculate turning radius $\rho_i = -L / \tan(\alpha)$ and the curvature sequence corresponding to the center point C_i , where L is the wheelbase of the vehicle, and the difference between p_l and the circle formed by any C_i is defined as

$$d_s = \|\overrightarrow{C_i p_l} - \rho_i\| \quad (19)$$

Finally, the algorithm selects the steering angle and curvature associated with the minimum value d_s . Algorithm 1 is the process of calculating the optimal angle for a given look-ahead point p_l . The realization method is to select the best steering angle from a set of discrete angles (Figure 16), As in formula (20):

$$\min_{p_l} = \|\overrightarrow{C_i p_l} + \rho_i\| \quad s.t. \alpha_{min} \leq \alpha_i \leq \alpha_{max} \quad (20)$$

Algorithm 1: Choose the best angle α_{best}

```

1  Input:  $p_l$ 
2  Output:  $\alpha_{best}$ 
3   $distpairs \leftarrow$  Empty list 2-tuples
4  For  $\alpha_i$  in  $[\alpha_{min}, \alpha_{max}]$  do
5     $\rho_i := -L / \tan(\alpha_i)$ 
6     $C_i := P - R(\theta)\rho_i$ 
7     $d_j := distance(C_i, p_l)$ 
8     $d_s := d_j - \rho_i$ 
9     $push(d_s, \alpha_i)$  into  $distpairs$ 
10 end
11  $\alpha_{best} \leftarrow$  select  $\alpha_i$  from  $distpairs$  where  $d_s$  is min

```

Regard α_{best} as the front wheel steering angle of the vehicle at that moment, and in order to ensure smooth control, the optimal front wheel steering angle obtained is processed by low-pass filtering (LPF), LPF can reduce the sudden change in the amplitude of the control quantity in a short time, in an autonomous driving environment, inaccurate positioning of the vehicle may cause a sudden change in the control amount. The basic theory of LPF is:

$$y(t) = Ku(t) + (1 - K)y(t - 1) = y(t - 1) + K[u(t) - y(t - 1)] \quad (21)$$

where $K = \frac{dT}{T}$, dT is the running step, T is the time constant, u is the input signal, y is the output signal.

The optimal front wheel steering angle α_{best} is processed by equation (21) to complete the lateral control of the vehicle. The finally designed IPP control strategy is shown in Figure 17. The main principle of the strategy: get the state quantity output by the vehicle object, according to the longitudinal vehicle speed v_x and the road curvature w , the minimum stable look-ahead distance is obtained, and the vehicle state quantity and longitudinal vehicle speed are updated in real time, after the reference path is adaptively fitted to the cubic Bezier curve, the geometric position of the vehicle and the reference path is used to search for the look-ahead point by an improved method, if the curvature of the reference road becomes larger, correct the look-ahead point to reduce the phenomenon of cutting-corner by the vehicle, and finally search in a limited number of front wheel angles to select the optimal steering angle ϕ , the control quantity acts on the vehicle object to complete real-time control, in which the longitudinal vehicle speed is controlled by MPC.

3 Simulation and analysis

This section verifies the effectiveness of the lateral tracking algorithm proposed in this article in the Carsim/Simulink simulation environment. The overall simulation framework is shown in Figure 18, the output state quantity of Carsim is input to the improved lateral controller and longitudinal MPC controller proposed in this paper. The front wheel angle information, accelerator pedal opening and brake master cylinder pressure are obtained and fed back to Carsim to control the movement of the Carsim vehicle. Some parameters of the simulated vehicle are shown in Table 3.

In order to compare and analyze the effectiveness of the algorithm proposed in this article, the other two improved pure pursuit algorithms that adjust the look-ahead distance: Speed control (VPP) [32] and fuzzy logic control (FPP) [27] used for comparison simulation verification. The expression of the look-ahead distance in vehicle speed control is as formula (22):

$$l_d = \frac{1}{2\alpha_{max}} V^2 + BV + C \quad (22)$$

where α_{max} is the maximum braking acceleration of the vehicle, which is taken as $5m/s^2$, B is the driving distance of the vehicle within the reaction time required to take the brake when it encounters a dangerous situation, C is the minimum turning radius of the vehicle, combined with the simulated vehicle, respectively taken as $0.2m$ and $6.3m$.

The look-ahead distance in fuzzy logic control is defined as the gain of the vehicle speed, where the gain K (the fuzzy domain is $[1, 4]$) determined by lateral tracking error E_d (fuzzy domain is $[-15, 15]m$)

and heading angle deviation E_c (fuzzy domain is $[-90, 90]^\circ$).

3.1 Simulation reference path and evaluation index

After setting up the simulation platform, select the reference path to verify the algorithms in sections 2.2.1 and 2.2.2. A total of three reference paths were selected for simulation test, namely Path 1, Path 2 and Path 3. As shown in Figure 19, Path 1 and Path 2 are built by Prescan, path P is the reference path collected by the intelligent driving car in this paper and processed by continuous adaptive curve fitting based on cubic Bezier curve. Path 1 is composed of one section of straight lines. The overall curvature of Path 2 is smaller than that of Path 3, and it consists of two sections of lane changing conditions, two sections of arcs and straight lines. Path 1 verifies that the look-ahead point searched in Section 2.2.1 can quickly converge and track the reference path.

In order to measure the effectiveness of the method proposed in this paper, the average lateral tracking error and maximum lateral tracking error evaluation indicators are proposed. It is assumed that the reference path is known in the process of verifying the lateral tracking strategy and consists of $n(n > 2)$ points.

Lateral tracking is defined as the distance from the vehicle's center of mass position P to the line segment composed of the two nearest points L_1 and L_2 on the center of mass reference, Where the line segment $l(t)$ is expressed as $\vec{L}_u = \vec{L}_1\vec{L}_2$, Then the lateral tracking error from the centroid P to the line segment $l(t)$ is:

$$D_{lat}(P, l(t)) = \frac{\|\vec{PL}_1 \times \vec{L}_u\|}{\|\vec{L}_u\|} = \frac{\|\vec{L}_2\vec{L}_1 \times \vec{PL}_1\|}{\|\vec{L}_2\vec{L}_1\|} \quad (23)$$

The algorithm in this article is to make $D_{lat}(P, l(t))$ as small as possible, the average lateral tracking error can be defined as:

$$D_{avg} = \frac{\sum_{i=0}^n D_{lat}^i(P_i\{L_1, L_2\})}{n} \quad (24)$$

Then the maximum lateral tracking error can also be calculated:

$$D_{max} = \max D_{lat}^i(P_i\{L_1, L_2\}) \quad (25)$$

3.2 Simulation results and analysis

3.2.1 Simulation analysis under Path 1

Set the simulation time of this working condition to 15s, and the longitudinal target speed to 18km/h. The reference path is composed of a straight line with a length of 90, and the starting point is set as the origin

of the absolute coordinate, since this working condition focuses on verifying the effectiveness of searching look-ahead point in section 2.2.1, the initial position of the simulated vehicle (the center of the vehicle's rear axle) and the initial heading angle are set to three combinations. The initial position is set to a certain deviation from the reference path(3m), that is (0, 3) m, the initial heading angle is set to 0, 90, and 270°. The simulation results of IPP, VPP and FPP in Path 1 are shown in Figures 20, 21 and 22.

Figure 20 (a), (b) and (c) are the path comparison diagram, lateral tracking error diagram and heading angle deviation diagram of the three algorithms when the initial deviation is 3m and the initial heading angle is 0°. The three algorithms have similar lateral deviations in the tracking process, but IPP converges to Path 1 the fastest, and the oscillation amplitude is also small. At the same time, the rapid adjustment process also makes IPP's heading deviation larger than the other two algorithms. When the initial heading angle is 90°, FPP shows the worst effect. When the initial heading angle is 270°, the maximum lateral tracking errors of FPP and VPP are equivalent, but the former converges to Path 1 faster. Based on Figures 20 to 22, IPP can enable the vehicle to search for the look-ahead point better under different initial heading angles when the vehicle is a certain distance from the reference path, so as to converge to the reference path more quickly and perform accurate tracking.

3.2.2 Simulation analysis under Path 2

The curvature of Path 2 is slightly smaller than that of Path 3, and the simulation longitudinal target speed of this operating condition is set to 18, 28km/h. The starting point of the reference path is set as the origin of the absolute coordinate system, the initial position of the simulated vehicle (the center of the rear axle of the vehicle) is (0, 0) m, and an initial heading angle of 0° is given. The simulation results of IPP, VPP and FPP in Path2 are shown in Figures 23 and 24.

It is known from Figure 23 and 24 that the improved algorithm shows the best results. In Path 2, both VPP and FPP have a shortcut cutting-corner phenomenon when entering a turn, which is especially obvious in the initial two-stage lane changing conditions after the vehicle speed increases. Figure 23(b) and Figure 24(b) are the lateral tracking error graphs under the action of the algorithm at 18km/h and 28km/h. Table 4 shows quantitatively, as the vehicle speed increases, the lateral tracking error increases, and the lateral tracking error of IPP is within 0.5m at the two vehicle speeds. Combined with Figure 23 (c), which is a comparison diagram of the front wheel steering angle during the 18km/h simulation process, IPP can give the front

wheel steering angle that accurately tracks the reference path, it can be seen from the simulation time of 20 ~ 30s that IPP first rapidly increases the steering wheel angle. After 1 ~ 2s, the steering wheel angle of the three algorithms is equivalent, but the change in lateral tracking error is sufficient to show the superiority of the improved algorithm, Figure 24 (c) can also illustrate the superiority of IPP.

3.2.3 Simulation analysis under Path 3

The curvature of the road in Path 3 is larger than Path 2, and the longitudinal target speeds are set to 8km/h and 18km/h respectively. The starting point of the reference path is set as the origin of the absolute coordinate system, and the initial position of the simulated vehicle (the center of the rear axle of the vehicle) is (0, 0) m, and set the initial heading angle to 89.4°. The simulation results of IPP, VPP and FPP in Path 3 are shown in Figures 25 and 26.

It is known from Figure 25 and 26 that the improved algorithm shows the best results at 8km/h and 18km/h. There are five corners in Path 3, and both VPP and FPP have a cutting-corner phenomenon when entering the corner, the cutting-corner phenomenon of VPP is more obvious when entering a turn, especially when the vehicle speed increases, the cutting-corner phenomenon is also aggravated, and the two algorithms fail to track the path well when exiting a turn. Figure 25(a) and Figure 26(a) are path comparison and partial enlarged views of tracking four bends, which can better observe the above phenomenon. Figures 25(b) and 26(b) quantitatively show the changes in the lateral tracking error of the three algorithms during the simulation process, at 8km/h, FPP is only slightly worse than IPP tracking effect, VPP performance is the worst, its maximum lateral tracking error $D_{max} = 0.9260m$; At 18km/h, the maximum lateral tracking error of IPP is $D_{max} = 0.3040m$, average lateral tracking error is $D_{avg} = 0.0422m$, in FPP, Maximum lateral tracking error is $D_{max} = 0.6789m$, average lateral tracking error is $D_{avg} = 0.1059m$, the maximum lateral tracking error of VPP is $D_{max} = 1.5341m$, the average lateral tracking error is $D_{avg} = 0.3411m$. Simulation shows that the improved algorithm can accurately track Path 3 at the two vehicle speeds in the simulation process, and the lateral tracking error is within 0.35m.

4 Vehicle testing and analysis

This section verifies the lateral tracking algorithm proposed in this article on the autonomous driving car platform. The hardware of the autonomous driving car platform and the structure of the whole vehicle are shown in Figure 27. Vehicle positioning and other

status information are provided by the integrated navigation system X1, steering angle sensor, brake pedal opening sensor and wheel speed sensor. The control algorithm is embedded in MicroAutoBox to generate execution information, and the movement of the car is controlled by the drive, brake and steering motor ECU. Some parameters of the vehicle are shown in Table 5, the integrated navigation system X1 includes a GNSS high-precision receiver and a tactical-level IMU, and X1 consists of two antennas (ANT 1 and ANT 2) and X1 machine, which are connected by coaxial cable, which can provide stable, continuous, and reliable high-precision position and attitude information. In order to facilitate debugging and switching algorithms, and collect the data of the verification process of the car, an interface diagram is established in Dspace ControlDesk. After finding the nearest point on the reference path by initializing the vehicle, adjust different switching values for manual switching between different strategies in the interface.

Since the verification is carried out on campus, the vehicle speed is controlled according to the reference path, pedestrians and vehicles on the road. The test road and vehicle speed changes (L_{cum} is the cumulative distance from each point on the reference path to point A) are shown in Figure 28(a) and (b) respectively. A is the starting point of the test road and B as the end point of the test road, the reference road is a circle, of which a, b, c, d and e are the five curves in the test road. It can be seen that the speed of the vehicle also decreases when encounters a curve, And the test time is 135s. The above lays the foundation for verifying the lateral tracking algorithm in this paper.

The initial lateral error is set to 0.9m, figures 29 and 30 are the lateral tracking error and steering wheel angle diagrams of the intelligent driving car in the process of verifying different strategies. Figure 29 shows that the IPP proposed in this paper can quickly control the steering wheel angle to adjust the lateral tracking error under the same initial lateral error, followed by the VPP response, and the FPP response the slowest. IPP has the best tracking effect, followed by FPP, and VPP is the worst, IPP average lateral tracking error is $D_{avg} = 0.0547m$, average lateral tracking error of VPP is $D_{avg} = 0.1384m$, average lateral tracking error of FPP is $D_{avg} = 0.1199m$. At the same time, it can be seen from Figure 30 that IPP can give a more reasonable steering wheel angle. When there is an initial lateral error, both IPP and FPP give a larger steering wheel angle, which can be adjusted quickly to reduce the lateral tracking error. In addition, the subjective riding experience of the actual vehicle verification also shows that the steering wheel angle changes of the three strategies are relatively stable, and there

will be no sudden changes. Therefore, the autonomous driving vehicle platform verifies the effectiveness of the lateral tracking algorithm proposed in this paper.

5 Concluding

This paper proposes an improved lateral tracking algorithm: First, a continuous adaptive fitting method based on cubic Bezier curve to obtain a smooth reference path and road curvature. Then use an improved search method to search for the look-ahead point according to the relative position of the vehicle and the reference path. Finally, for the phenomenon of cutting-corner when vehicle is entering a curve during the lateral tracking of the vehicle, the look-ahead point is corrected according to the change of the curvature of the reference path. It was verified by the Carsim/Simulink simulation environment and the autonomous driving car platform, and the simulation built three complex path conditions. Compared with VPP and FPP, IPP proposed in this paper can quickly converge to the reference path during the lateral tracking of the vehicle, reduce the cutting-corner phenomenon when the vehicle enters a turn, and greatly improve the accuracy of lateral tracking.

Acknowledgements

The authors acknowledged the anonymous reviewers and editors for their efforts in valuable comments and suggestions.

Funding

This work was supported by the Chinese National Science Foundation Grant (No. 51905408), the Interdisciplinary Basic Research Project of Southwest Jiao-tong University (No. 2682021ZTPY068).

Availability of data and materials

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

Not applicable.

Authors' information

Liu Ping received the MA.Eng degree from the School of vehicle engineering, Chang 'an University, China, in 1995. He also has a Ph.D. degree. He is currently an associate professor in the School of Mechanical Engineering of Southwest Jiaotong University. His main research direction is electric vehicle control and connected vehicle theory and control. He pre-sided over the implementation of more than 20 provincial-ministerial and school-enterprise cooperation projects, published more than 40 papers in domestic and foreign academic journals, and obtained 5 invention patents.

Liu Zibin is currently a master student in the School of Mechanical Engineering of Southwest Jiao-tong University. His main research direction is automobile modern design theory. He has participated in autonomous vehicles following project and the Intelligent Vehicle development and commissioning project.

Ding Weiping (corresponding author) received the Ph.D. degree from the School of Mechanical Engineering, Xi 'an Jiaotong University, China, in 2001. He is currently a professor in the School of Mechanical Engineering of Southwest Jiaotong University. He is an expert member of China Automotive Engineering Society. His main research direction is vehicle NVH and vehicle intelligence theory and technology. He has presided over 7

projects including the development of wheel drive suspension system for pure electric buses. He has published more than 40 papers in domestic and foreign academic journals and obtained 7 invention patents.

Haibo Huang (corresponding author) received the Ph.D. degree from the School of Mechanical Engineering, Southwest Jiaotong University, China, in 2017. He is currently an assistant professor with the School of Mechanical Engineering, Southwest Jiaotong University. He has authored and co-authored more than 30 papers in journals and conference. His research interests mainly include the vehicle optimization and control, vehicle intelligence, vehicle NVH (noise, vibration and harshness).

Author details

¹School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China. ²Engineering Research Center of Advanced Driving Energy-saving Technology, Ministry of Education.

References

- Huang, H. B., Li, R. X., Ding, W. P. et al. (2017). Evaluation of vehicle interior sound quality using a continuous restricted Boltzmann machine-based DBN. *Mechanical Systems and Signal Processing*, 84, 245-267.
- Huang, H., Huang, X., Ding, W. et al. (2022). Uncertainty optimization of pure electric vehicle interior tire/road noise comfort based on data-driven. *Mechanical Systems and Signal Processing*, 165, 108300.
- Li, S., Shu, K., Chen, C. et al. Planning and Decision-making for Connected Autonomous Vehicles at Road Intersections: A Review. *Chin. J. Mech. Eng.* **34**, 133 (2021).
- D. González, J. Pérez, V. Milanés, et al. A Review of Motion Planning Techniques for Automated Vehicles[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2016, 17(4):1135-1145.
- Qin Y, Tang X, Jia T, et al. Noise and vibration suppression in hybrid electric vehicles: State of the art and challenges[J]. *Renewable and Sustainable Energy Reviews*, 2020, 124.
- Xiaolin T, Shanshan L I, Hong W, et al. Research on Energy Control Strategy Based on Hierarchical Model Predictive Control in Connected Environment[J]. *Journal of Mechanical Engineering*, 2020, 56(14):119.
- Liu T, Tang X, Wang H, et al. Adaptive Hierarchical Energy Management Design for a Plug-In Hybrid Electric Vehicle[J]. *IEEE Transactions on Vehicular Technology*, 2019, 68(12):11513-11522.
- Rehrl, K., Zankl, C. Digibus©: results from the first self-driving shuttle trial on a public road in Austria. *Eur. Transp. Res. Rev.* **10**, 51 (2020).
- Gao, R., Li, S., Gao, Y. et al. A lightweight cryptographic algorithm for the transmission of images from road environments in self-driving. *Cybersecur* **4**, 3 (2021).
- Quddus, Mohammed, Chen, et al. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions[J]. *Transportation research, Part C. Emerging technologies*, 2015.
- Yuan Jingxin. Research on Intelligent Vehicle Path Following Control Algorithm Based on CarSim [D]. Jilin University, 2019.
- A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles. *IEEE Transactions on Intelligent Vehicles*, 2016.
- Cai Y F, Li J, Sun X Q, et al. Research on hybrid control strategy for intelligent vehicle path tracking. *China Mech Eng*, 2020, 31(3): 289.
- Jesús Morales, Jorge L. Martínez, María A. Martínez, et al. Pure-Pursuit Reactive Path Tracking for Nonholonomic Mobile Robots with a 2D Laser Scanner[J]. *Eurasip Journal on Advances in Signal Processing*, 2009, 2009:1-10.
- Xu Y, Lu Z F, Shan X, et al. Study on an automatic parking method based on the sliding mode variable structure and fuzzy logical control. *Symmetry*, 2018, 10(10): 523.
- Jond H B, Platos J. A Fast Trajectory Tracking Control Design for Autonomous Driving[C]. // 2019 7th International Conference on Robotics and Mechatronics (ICRoM). 2019.
- Ye H, Jiang H B, Ma S D, et al. Linear model predictive control of automatic parking path tracking with soft constraints. *Int J Adv Rob Syst*, 2019, 16(3): 172988141985220.
- Mohammadzadeh A, Taghavifar H. A robust fuzzy control approach for path-following control of autonomous vehicles. *Soft Comput*, 2020, 24(5): 3223.
- Lv WenJie, Ma Rong, li Suilao, et. Improved path tracking algorithm based on pure tracking model [J]. *Measurement and Control*

- Technology, 2011, 30(7):93-96.
20. [S. Kato, S. Tokunaga, Y. Maruyama, et al. Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems. 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPs), Apr. 2018, pp. 287–296.
 21. R. Tedrake, and t. D. D. Team. Drake: Model-based design and verification for robotics, 2019. Available: <https://drake.mit.edu>.
 22. Coulter, R. C. Implementation of the pure pursuit path tracking algorithm. Carnegie-Mellon UNIV Pittsburgh PA Robotics INST, 1992.
 23. Lal D S, Vivek A, Selvaraj G,. Lateral control of an autonomous vehicle based on Pure Pursuit algorithm[C]. // International Conference on Technological Advancements in Power & Energy. 2017:, 1-8.
 24. Samuel M, Hussein M, Binti M. A Review of some Pure-Pursuit based Path Tracking Techniques for Control of Autonomous Vehicle[J]. International Journal of Computer Applications, 2016, 135(1):35-38.
 25. Paden B, Cap M, Yong S Z, et al. A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles[J]. IEEE Transactions on Intelligent Vehicles, 2016, 1(1):33-55.
 26. Wang W J, Hsu T M, Wu T S. The improved pure pursuit algorithm for autonomous driving advanced system[C]. //2017 IEEE 10th International Workshop on Computational Intelligence and Applications (IWCIA).IEEE 2017:, 33-38.
 27. Guo Bixi, Du XingLe, Tao Xiaosong. Algorithm improvement based on pure tracking model[J]. Car practical technology, 2019(15): 32-34.
 28. F. Qiang, L. Xiang, L. Xueyin, et al. An Improved Pure Pursuit Algorithm for Tractor Automatic Navigation. 2021 IEEE 4th International Conference on Information Systems and Computer Aided Education (ICISCAE), 2021, pp. 201-205.
 29. Gong Jianwei, Jiang Yan, Xu Wei. Model predictive control for unmanned vehicles [M]. Beijing: Beijing Institute of Technology Press, 2014.
 30. Ahn J, Shin S, Kim M, et al. Accurate Path Tracking by Adjusting Look-Ahead Point in Pure Pursuit Method[J]. International Journal of Automotive Technology, 2021, 22(1):119-129.
 31. Dubins L E. On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents[J]. American Journal of Mathematics, 1957, 79(3):497-516.
 32. Duan Jianmin, Xia Tian, Song Zhixue. Research on Path Following Control of Smart Car Based on Co-simulation [J]. Computer simulation, 2017, 34(6):7.

Figures

Figure 1 (a) Principle of traditional pure pursuit algorithm. (b) Ackerman steering kinematics

Figure 2 (a) Improper look-ahead distance. (b) Cutting-corner phenomenon

Figure 3 Lateral tracking control technology diagram

Figure 4 Fitting effect of Bezier curves of different orders: (a) One-order Bessel curve fitting. (b) Quadratic Bessel curve fitting. (c) Cubic Bezier curve fitting. (d) Quartic Bezier curve fitting. (e) Quintic Bezier curve fitting. (f) Six-order Bessel curve fitting.

Figure 5 The continuity of two adjacent cubic Bezier curves

Figure 6 Continuous adaptive reference path curve fitting process

Figure 7 Comparison of adaptive segmented fitting path of cubic Bezier curve

Figure 8 The residual graph of adaptive piecewise fitting of cubic Bezier curve

Figure 9 The curvature comparison of adaptive piecewise fitting of cubic Bezier curve

Figure 10 (a) Search look-ahead point using radius. (b) Search look-ahead point using path integration.

Figure 11 Search algorithm based on Dubins curve

Figure 12 Road curvature calculation

Figure 13 Minimum stable look-ahead distance

Figure 14 Modify the front view point to p_l to reduce the cutting-corner phenomenon

Figure 15 Change scenario of τ

Figure 16 Determining the optimal steering angle from the look-ahead point p_l

Figure 17 Block diagram of pure pursuit control strategy

Figure 18 The overall framework of the simulation

Figure 19 Reference path diagram: (a) Path 1. (b) Path 2. (c) Path 3

Tables

Figure 20 Path 1 tracking effect comparison chart($D_{lat} = 3, \theta = 0^\circ$):(a) Path comparison. (b) Lateral tracking error. (c) Heading angle deviation.

Figure 21 Path 1 tracking effect comparison chart($D_{lat} = 3, \theta = 90^\circ$):(a) Path comparison. (b) Lateral tracking error. (c) Heading angle deviation.

Figure 22 Path 1 tracking effect comparison chart($D_{lat} = 3, \theta = 270^\circ$):(a) Path comparison. (b) Lateral tracking error. (c) Heading angle deviation.

Figure 23 Path 2 lateral tracking effect diagram (18km/h): (a) Path comparison. (b) Lateral tracking error. (c) Front wheel steering angle.

Figure 24 Path 2 lateral tracking effect diagram (28km/h): (a) Path comparison. (b) Lateral tracking error. (c) Front wheel steering angle.

Figure 25 Path 3 lateral tracking effect diagram (8km/h): (a) Path comparison and partial enlargement of the bend. (b) Lateral tracking error

Figure 26 Path 3 lateral tracking effect diagram (18km/h): (a) Path comparison and partial enlargement of the bend. (b) Lateral tracking error

Figure 27 Autonomous driving car hardware and vehicle structure

Figure 28 (a) Test road (b) Changes in vehicle speed

Figure 29 Real vehicle verification lateral tracking error diagram

Figure 30 Real vehicle verification steering wheel angle diagram

Table 1 Comparison of Bezier curve fitting evaluation of different orders

order	SSE	RMSE	t/s
One-order	3739.04	4.75	0.381
Quadratic	148.55	0.95	0.362
Cubic	16.16	0.31	0.445
Quartic	6.62	0.20	0.612
Quintic	58.36	0.60	0.718
Six-order	300073.35	43.17	0.720

Table 2 Simulation results of variable parameters

	D_{avg}	β_{max}	
		0.2	0.08
α_{max}	1	0.0298	0.0302
	2	0.0299	0.0303
	3.5	0.0301	0.0304

Table 3 Some parameters of the simulated vehicle

parameter	value
steering ratio	20
maximum steering wheel angle /°	500
wheelbase L/m	2.65
distance from CG to front axle a/m	1.07
curb quality /kg	1430
centroid height /m	0.65

Table 4 The lateral tracking error of the three methods

Velocity(km/h)	strategy	$D_{avg}(m)$	$D_{max}(m)$
18	IPP	0.0534	0.1980
	FPP	0.1033	0.4172
	VPP	0.2852	1.0706
28	IPP	0.1275	0.4519
	FPP	0.3361	1.3376
	VPP	0.7333	2.8399

Table 5 Some parameters of the simulated vehicle

parameter	value
wheelbase L/m	1.3101
distance from CG to front axle a/m	0.6205
curb quality /kg	190

Figures

Figure 1

(a) Principle of traditional pure pursuit algorithm. (b) Ackerman steering kinematics

(a)

(b)

Figure 2

(a) Improper look-ahead distance. (b) Cutting-corner phenomenon

Figure 3

Lateral tracking control technology diagram

Figure 4

Fitting effect of Bessel curves of different orders: (a) One-order Bessel curve fitting. (b) Quadratic Bessel curve fitting. (c) Cubic Bessel curve fitting. (d) Quartic Bessel curve fitting. (e) Quintic Bessel curve fitting. (f) Six-order Bessel curve fitting.

Figure 5

The continuity of two adjacent cubic Bezier curves

Figure 6

Continuous adaptive reference path curve fitting process

Figure 7

Comparison of adaptive segmented fitting path of cubic Bezier curve

Figure 8

The residual graph of adaptive piecewise fitting of cubic Bezier curve

Figure 9

The curvature comparison of adaptive piecewise fitting of cubic Bezier curve

Figure 10

(a) Search look-ahead point using radius. (b) Search look-ahead point using path integration.

Figure 11

Search algorithm based on Dubins curve

Figure 12

Road curvature calculation

Figure 13

Minimum stable look-ahead distance

Figure 14

Modify the front view point to p_l to reduce the cutting-corner phenomenon

(a)

(b)

(c)

(d)

Figure 15

Change scenario of $\mathcal{T}$

Figure 16

Determining the optimal steering angle from the look-ahead point p_l

Figure 17

Block diagram of pure pursuit control strategy

Figure 18

The overall framework of the simulation

Figure 19

Reference path diagram: (a) Path 1. (b) Path 2. (c) Path 3

Figure 20

Path 1 tracking effect comparison chart($D_{lat} = 3, \theta = 0^\circ$): (a) Path comparison. (b) Lateral tracking error. (c) Heading angle deviation.

Figure 21

Path 1 tracking effect comparison chart($D_{lat} = 3, \theta = 90^\circ$):(a) Path comparison. (b) Lateral tracking error. (c) Heading angle deviation.

Figure 22

Path 1 tracking effect comparison chart($D_{lat} = 3, \theta = 270^\circ$):(a) Path comparison. (b) Lateral tracking error. (c) Heading angle deviation.

Figure 23

Path 2 lateral tracking effect diagram (18km/h) :(a) Path comparison. (b) Lateral tracking error. (c) Front wheel steering angle.

Figure 24

Path 2 lateral tracking effect diagram (28km/h):(a) Path comparison. (b) Lateral tracking error. (c) Front wheel steering angle.

Figure 25

Path 3 lateral tracking effect diagram (8km/h) :(a) Path comparison and partial enlargement of the bend.
 (b) Lateral tracking error

Figure 26

Path 3 lateral tracking effect diagram (18km/h) :(a) Path comparison and partial enlargement of the bend. (b) Lateral tracking error

(a)

(b)

Figure 27

Autonomous driving car hardware and vehicle structure

(a)

(b)

Figure 28

(a) Test road (b) Changes in vehicle speed

Figure 29

Real vehicle verification lateral tracking error diagram

Figure 30

Real vehicle verification steering wheel angle diagram