International Journal of Automotive Technology, Vol. 23, No. 5, pp. 1239–1253 (2022)
DOI 10.1007/s12239–022–0109–8
Copyright © 2022 KSAE/ 129–05
pISSN 1229–9138/ eISSN 1976–3832

PATH PLANNING AND PATH TRACKING FOR AUTONOMOUS VEHICLE BASED ON MPC WITH ADAPTIVE DUAL-HORIZON-PARAMETERS

Yaohua Li*, Jikang Fan, Yang Liu and Xiaoyu Wang

School of Automobile, Chang'an University, Xi'an 710064, China

(Received 23 July 2021; Revised 22 November 2021; Accepted 15 December 2021)

ABSTRACT–According to the position relationship between the vehicle and the obstacle, a new obstacle avoidance path planner was designed to solve the limitation of traditional local obstacle avoidance path planner in excessive obstacle avoidance. In order to improve the control accuracy of the path tracking controller and ensure the stability of the vehicle, a comprehensive evaluation index of path tracking performance considering control accuracy and driving stability was established. The optimal prediction time domain and control time domain parameters at different vehicle speeds were obtained, and an adaptive dual time domain parameter path tracking controller was designed. Based on the joint-simulation platform, the integrated structure of the planning layer and the control layer was simulated and verified. Simulation results show that the new obstacle avoidance function can avoid excessive obstacle avoidance while ensuring real-time performance, and improve the driving stability of the vehicle. The adaptive time-domain parameter path tracking controller has better comprehensive control performance and can improve driving safety under extreme conditions. The integrated structure of local obstacle avoidance path planning and path tracking control are beneficial for the vehicle to plan and accurately track the local obstacle avoidance path in multiple static obstacle scenes.

KEY WORDS : Autonomous vehicles, Model predictive control, Obstacle avoidance, Local path planning, Path tracking

1. INTRODUCTION

Path planning refers to planning a collision-free path from the initial position to the target position according to certain standards in a driving environment with obstacles (Liu *et al*., 2017; Li *et al*., 2017). According to the scope of planning, it is generally divided into global path planning and local path planning (Dolgov *et al*., 2010). Global planning is to plan the optimal driving route from the start point to the end point according to the map information. The local path planning is based on the global path and the vehicle surrounding information detected by sensors to plan a safe local path. Path tracking is an important part of automatic driving technology, whose main task is to steer the self-driving vehicle to accurately track the reference path based on the reference path outputted by the planning layer and the current vehicle motion state (Lu *et al*., 2020; Zhang and Zhu, 2019; Huang *et al*., 2020; Amer *et al*., 2017).

Aim at path planning, the rapidly-exploring random tree (RRT) was improved by using spline curve to fit the path generated by RRT to improve the smoothness of the planned path. However, there is a problem of long calculation time and non-optimal solution of the planned path (Jeon *et al*., 2013). A real-time path planning algorithm was presented, which calculated the path based on a set of predefined waypoints and provides the best path for off-road automatic driving to avoid static obstacles. The predefined waypoints provided the basic framework of the curvilinear coordinate system to generate path candidates for autonomous vehicle path planning. The selection of the optimal path is determined by considering the path safety cost, path smoothness, and path consistency. But it just avoids static obstacles (Chu *et al*., 2012). The vehicle shape is taken into consideration during trajectory generation as a convex polygonal region defined by linear constraints rather than a single point. The MPC optimization problem with the vehicle shape is solved as a convex quadratic programming (CQP). The MPC's real-time performance is improved, but the collision-free navigation function may cause unnecessary excessive obstacle avoidance (Wang *et al*., 2018).

Aiming at the problem of path tracking, a linear parameter varying (LPV) multiple-input single-output (MISO) H-infinity controller was developed based on the feedback of the lateral error at the center of gravity and the look-ahead distance. The controller architecture offered a way to cope with the effect of the steering nonlinearities by scheduling one of the control weighting functions (Corno *et al*., 2021). Aimed at the path tracking problem of four-wheel independent steering drive electric vehicles, MPC algorithm was used to design an objective function including heading angle deviation, longitudinal deviation and control amount

**Corresponding author*. e-mail: yaohua.li@chd.edu.cn

increment (Hang *et al*., 2017). MPC has higher path tracking accuracy and better adaptability to changes in vehicle speed and road adhesion coefficient, compared with optimal preview. However, the MPC algorithm needs to solve the nonlinear programming to obtain the optimal solution, which requires a large amount of calculation and poor real-time performance of the algorithm (Li *et al*., 2020). Therefore, the actual engineering application of MPC algorithm is closely related to the chip processor level and software algorithm optimization. With the improvement of chip processing capabilities, model predictive control algorithms are expected to achieve practical engineering applications in the future.

This paper took the autonomous vehicle in the obstacle scene as the research object. Aiming at the local obstacle avoidance path planning problem, based on the vehicle point quality model, the nonlinear MPC algorithm was used for local path planning. A new obstacle avoidance function was designed to solve the problem of excessive obstacle avoidance. For path tracking problem, based on two degree of freedom vehicle dynamics model, the linear time-varying MPC algorithm was used for path tracking control, and the influence of the prediction time domain and the control time domain on the control performance was analyzed. A comprehensive evaluation index of path tracking performance considering control accuracy and driving stability was established and the optimal prediction time domain and control time domain parameters at different vehicle speeds were obtained. An adaptive time domain parameter path tracking controller was designed. The integrated design of the local path planner and the path tracking controller was carried out, and the joint simulation integration verification was carried out through CarSim and Simulink. Simulation results show that the new obstacle avoidance function can avoid excessive obstacle avoidance while ensuring real-time performance, and improve the driving stability of the vehicle. The adaptive time-domain parameter path tracking controller has better comprehensive control performance and can improve driving safety under extreme conditions. The integrated structure of local obstacle avoidance path planning and path tracking control are beneficial for the vehicle to plan and accurately track the local obstacle avoidance path in multiple static obstacle scenes.

2. LOCAL OBSTACLE AVOIDANCE PATH PLANNING

2.1. Vehicle Prediction Model at Planning Level

Compared with conventional path planning methods, the MPC has many advantages such as simple implementation in multi-objectives and multivariate systems and recognition of non-linearities/constraints. But nonlinear MPC algorithm has a relatively large amount of calculation, in order to ensure real-time performance, the model should not be too complicated. Therefore, this paper selected the vehicle point quality model as the planning layer vehicle model, as shown in Equation (1), where (x, y) is the position of the vehicle in the body coordinate system, (X, Y) is the position of the vehicle in the geodetic coordinate system, the origin of the body coordinate system is the center of mass of the vehicle, a_y is the lateral acceleration of the vehicle, and φ is the vehicle yaw angle.

$$\begin{aligned} \ddot{x} &= 0, \\ \ddot{y} &= a_y, \\ \dot{\varphi} &= \frac{a_y}{\dot{x}}, \\ \dot{X} &= \dot{x}\cos\varphi - \dot{y}\sin\varphi, \\ \dot{Y} &= \dot{x}\sin\varphi + \dot{y}\cos\varphi. \end{aligned} \tag{1}$$

Equation (1) can be expressed as a system whose state variable is $\boldsymbol{\xi} = [\dot{x}, \dot{y}, \varphi, X, Y]^{\mathrm{T}}$ and control variable is $\boldsymbol{u} = [a_y]$. Equation (1) can be simplified to Equation (2).

$$\dot{\boldsymbol{\xi}}(t) = f(\boldsymbol{\xi}(t), \boldsymbol{u}(t)). \tag{2}$$

Assuming that the current time is k, the sampling time of the local planning layer is T_{p}, the prediction time domain is N_{p}, the control time domain is N_{c}, and normally $N_{\mathrm{p}} > N_{\mathrm{c}}$. The forward Euler method is used to discretize Equation (2), shown as Equation (3).

$$\begin{aligned} &\boldsymbol{\xi}(k+i+1) = \boldsymbol{\xi}(k+i) + T_{\mathrm{p}} f(\boldsymbol{\xi}(k+i), \boldsymbol{u}(k+i)), \\ &i = 0, 1, \cdots N_{\mathrm{p}} - 1. \end{aligned} \tag{3}$$

Where $\boldsymbol{u}(k+i) = \boldsymbol{u}(k+N_{\mathrm{c}}-1)$, $i = N_{\mathrm{c}}, N_{\mathrm{c}}+1, \ldots, N_{\mathrm{p}}-1$. That means the control amount outside the control time domain in the prediction time domain is kept as the last value in the control time domain.

Define the output of the system as $\boldsymbol{\eta} = [\varphi, Y]^{\mathrm{T}}$, then $\boldsymbol{\eta} = \boldsymbol{C}\boldsymbol{\xi}$, where $\boldsymbol{C} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$. According to formula (3) and the relationship between state quantity and output quantity, the output quantity of the system can be obtained.

2.2. Obstacle Avoidance Function

The obstacle avoidance function describes the closeness between the local path and the obstacle. According to the requirements of the MPC objective function, the closer the distance between the local path and the obstacle, the greater the value of the obstacle avoidance function. And the traditional obstacle avoidance function is shown in formula Equation (4) (Gong *et al*., 2020).

$$J_{\text{obs}} = \sum_{m=1}^{N} \frac{S_{\text{ob}} v_0}{(x_{\text{ob},m} - x_0)^2 + (y_{\text{ob},m} - y_0)^2 + \zeta} \tag{4}$$

where S_{ob} is the weight coefficient of the obstacle avoidance function, v_0 is the speed of the vehicle, ($x_{\text{ob},m}$, $y_{\text{ob},m}$) is the position of the m-th obstacle point in the body coordinate system, and (x_0, y_0) is the center of mass of the vehicle coordinates, ζ is a small positive number to prevent the denominator from being zero, and N is the total number of obstacle points.

The traditional obstacle avoidance function uses the distance between the obstacle point and the center of mass of the vehicle as the standard to describe the possibility of collision between the vehicle and the obstacle. When an obstacle and a vehicle are actually unlikely to collide, the calculated J_{obs} will still interfere with the model predictive control, and unnecessary excessive obstacle avoidance may occur. The position of the obstacle and the vehicle is shown in Figure 1. At this time, there is no possibility that the obstacle 1 will collide with the vehicle. However, the J_{obs} calculated by the traditional obstacle avoidance function will interfere with the objective function, causing unnecessary obstacle avoidance planning on the planned path, which affects driving comfort.

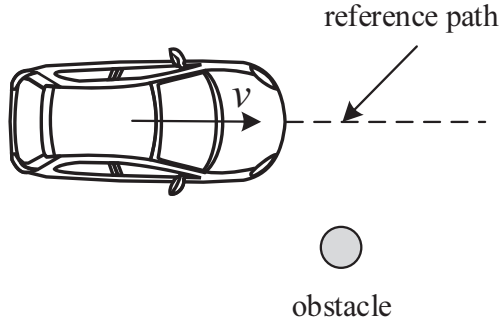


Figure 1. Schematic diagram of the possibility of obstacle collision.

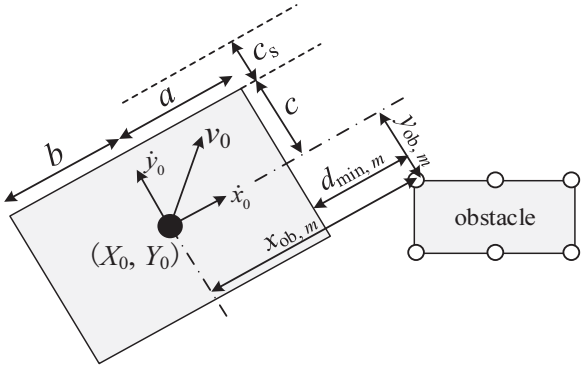


Figure 2. Position relationship between vehicle and obstacle.

In order to solve this problem, this paper defines that when the distance between the obstacle point and the vehicle exceeds the safe distance range, the corresponding distance is directly set to a fixed large constant to eliminate its influence in the objective function. The relationship between vehicle and obstacle is shown in Figure 2, where a, b and c are the body size parameters. The equivalent distance between the vehicle and the obstacle is shown in Equation (5), where $d_{\min,m}$ is the distance between the vehicle and the obstacle point, $d_{\min}$ is the distance between the vehicle and the obstacle, and c_s is the lateral safety distance threshold, which is taken as 0.5 m in this paper. W is a constant, which is taken as 1×10^7 in this paper.

$$\begin{aligned} d_{\min,m} &= \begin{cases} x_{\text{ob},m} - a & x_{\text{ob},m} \in (a,+\infty) \cap y_{\text{ob},m} \in [-c-c_s, c+c_s], \\ 0 & x_{\text{ob},m} \in [-b,a] \cap y_{\text{ob},m} \in [-c-c_s, c+c_s], \\ \text{W} & \text{Others}, \end{cases} \\ d_{\min} &= \min(d_{\min,m}). \end{aligned} \tag{5}$$

From Equations (4) and (5), the new obstacle avoidance function can be obtained as shown in Equation (6).

$$J_{\text{obs}} = \frac{S_{\text{ob}} v_0}{d_{\min} + \zeta} \tag{6}$$

2.3. Optimization Problem Construction and Solution

The goal of the local obstacle avoidance path planning layer is to avoid obstacles safely and ensure that the planned path does not deviate too much from the global reference path. The objective function of the path planning layer is shown in formula (7).

$$\begin{aligned} J(\boldsymbol{\xi}(t), \boldsymbol{\xi}_{\text{ob}}(t), \boldsymbol{U}(t)) &= \sum_{i=t+1}^{t+N_p} \left\| \boldsymbol{\eta}(i|t) - \boldsymbol{\eta}_{\text{ref}}(i|t) \right\|_{\boldsymbol{Q}}^2 \\ &+ \sum_{i=t+1}^{t+N_c} \left\| \boldsymbol{u}(i-1|t) \right\|_{\boldsymbol{R}}^2 + \sum_{i=t+1}^{t+N_p} J_{\text{obs},i} \end{aligned} \tag{7}$$

Where $\boldsymbol{U}(t)$ is the desired control variable sequence, $\boldsymbol{\eta}_{\text{ref}}$ is the local reference path, $\boldsymbol{\eta}_{\text{ref}} = [\varphi_{\text{ref}}, Y_{\text{ref}}]^{\text{T}}$, $\boldsymbol{Q}$ is the output deviation weight matrix, $\boldsymbol{R}$ is the control variable weight matrix, and $J_{\text{obs},i}$ is the value of the obstacle avoidance function at the i-th moment.

In order to make the planned route ensure the driving stability of the vehicle, corresponding constraints are added to the objective function of the planning layer. Therefore, the local obstacle avoidance path planning layer is transformed into solving the optimization problem shown in Equation (8).

By solving Equation (8), the optimal control sequence in the predicted time domain is obtained, and the optimal control sequence is introduced into Equation (3) to obtain the discrete

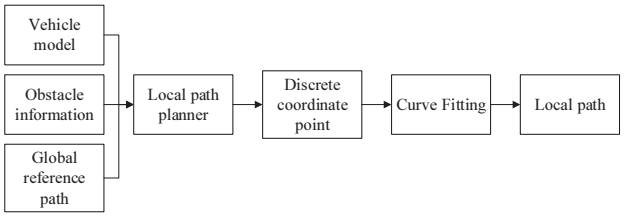


Figure 3. Block diagram of local path planning.

$$\begin{aligned}\min_{\boldsymbol{U}(t)} J = &\sum_{i=t+1}^{t+N_p} \left\| \boldsymbol{\eta}(i|t) - \boldsymbol{\eta}_{\text{ref}}(i|t) \right\|_{\boldsymbol{Q}}^2 + \sum_{i=t+1}^{t+N_c} \left\| \boldsymbol{u}(i-1|t) \right\|_{\boldsymbol{R}}^2 \\ &+ \sum_{i=t+1}^{t+N_p} \frac{S_{\text{ob}} v_{0,i}}{d_{\min,i} + \zeta}, \\ \text{s.t. } &\boldsymbol{\eta}_{\min} \le \boldsymbol{\eta}(i|t) \le \boldsymbol{\eta}_{\max}, i = t+1, t+2, \cdots, t+N_p, \\ &\boldsymbol{u}_{\min} \le \boldsymbol{u}(i|t) \le \boldsymbol{u}_{\max}, i = t+1, t+2, \cdots, t+N_c. \end{aligned} \tag{8}$$

optimal local obstacle avoidance path points in the predicted time domain. The path discrete points are carried out through the fifth-order polynomial curve fitting. and the planned path of local obstacle avoidance is achieved.

The structural block diagram of local path planning is shown in Figure 3.

3. PATH TRACKING CONTROL

The path tracking control is essentially the vehicle steering wheel angle control problem, also known as the vehicle lateral control problem. Based on the reference path information from the path planning layer, the optimal control value is calculated according to the current state of the vehicle and outputted to the execution layer. Finally, through the accurate execution of the actuator, the path tracking control of the vehicle is completed.

3.1. Control Layer Vehicle Model Establishment

In order to take the control accuracy and real-time performance into account, a two-degree-of-freedom vehicle dynamics model is selected as the vehicle model of the control layer, as shown in Equation (9).

$$\begin{aligned} \dot{v}_y &= \left(\frac{2C_f + 2C_r}{m v_x} \right) v_y + \left(\frac{2aC_f - 2bC_r}{m v_x} - v_x \right) \dot{\varphi} + \left(-\frac{2C_f}{m} \right) \delta_f, \\ \dot{\varphi} &= \dot{\varphi}, \\ \ddot{\varphi} &= \left(\frac{2aC_f - 2bC_r}{I_z v_x} \right) v_y + \left(\frac{2a^2 C_f + 2b^2 C_r}{I_z v_x} \right) \dot{\varphi} + \left(-\frac{2aC_f}{I_z} \right) \delta_f, \\ \dot{X} &= v_x \cos\varphi - v_y \sin\varphi, \\ \dot{Y} &= v_x \sin\varphi + v_y \cos\varphi. \end{aligned} \tag{9}$$

where v_x and v_y are the vehicle's x-axis and y-axis speed in the body coordinate system, C_f and C_r are the vehicle front and rear wheel cornering stiffness, m is the vehicle's curb mass, and a and b are the center of mass to the front and rear respectively. δ_f is the front wheel turning angle, φ is the vehicle yaw angle, I_z is the vehicle's moment of inertia around the z axis, and (X, Y) is the position of the vehicle in the geodetic coordinate system.

The above model can be abbreviated as Equation (10).

$$\dot{\boldsymbol{\xi}}(t) = f(\boldsymbol{\xi}(t), \boldsymbol{u}(t)) \tag{10}$$

In the formula, the state quantity is $\xi = [v_y, \varphi, \dot{\varphi}, X, Y]^T$, and the control quantity is $u = [\delta_f]$.

For the path tracking control layer, the vehicle model is more complex and the sampling period is shorter. In order to improve the real-time performance of the control, the Equation (10) is approximately linearized, and a linear time-varying equation can be obtained, as shown in the Equation (11).

$$\begin{aligned} \dot{\tilde{\boldsymbol{\xi}}}(t) &= \boldsymbol{A}(t)\tilde{\boldsymbol{\xi}}(t) + \boldsymbol{B}(t)\tilde{\boldsymbol{u}}(t), \\ \boldsymbol{A}(t) = \left. \frac{\partial f}{\partial \boldsymbol{\xi}} \right|_{\boldsymbol{\xi}_0(t), \boldsymbol{u}_0(t)}&, \ \boldsymbol{B}(t) = \left. \frac{\partial f}{\partial \boldsymbol{u}} \right|_{\boldsymbol{\xi}_0(t), \boldsymbol{u}_0(t)}, \\ \tilde{\boldsymbol{\xi}}(t) = \boldsymbol{\xi}(t) - \boldsymbol{\xi}_0(t),& \ \tilde{\boldsymbol{u}}(t) = \boldsymbol{u}(t) - \boldsymbol{u}_0(t). \end{aligned} \tag{11}$$

where $\boldsymbol{A}(t)$ and $\boldsymbol{B}(t)$ are the Jacobian matrix of f to $\boldsymbol{\xi}$ and $\boldsymbol{u}$ respectively, $\boldsymbol{\xi}_0(t)$is the reference state quantity, and $\boldsymbol{u}_0(t)$ is the reference control quantity.

Using forward Euler method to discretize equation (11), we can get:

$$\tilde{\boldsymbol{\xi}}(k+1) = \boldsymbol{A}_{k,t}\tilde{\boldsymbol{\xi}}(k) + \boldsymbol{B}_{k,t}\tilde{\boldsymbol{u}}(k) \tag{12}$$

where $\boldsymbol{A}_{k,t} = \boldsymbol{I} + T_c \boldsymbol{A}(t)$, $\boldsymbol{B}_{k,t} = T_c \boldsymbol{B}(t)$, T_c is the sampling period of discretization. Normally the control accuracy is higher, the real-time performance is worse. In Equation (12), T_c can be larger than the sampling period of the control layer in order to decrease the computation burden of discretization, but the control performances will be worse. In this paper, T_c is set as the sampling period of the control layer.

3.2. Optimization Problem Construction and Solution

The goal of the system path tracking control layer is to make the deviation between the actual driving path of the control vehicle and the local obstacle avoidance path from the path planning layer as small as possible and ensure driving stability. The objective function of the model predictive controller of the path tracking control layer is shown in Equation (13).

$$J(\boldsymbol{\xi}(t),\boldsymbol{u}(t-1),\Delta\boldsymbol{U}(t),\varepsilon)=\sum_{i=1}^{N_p}\left\|\boldsymbol{\eta}(t+i|t)-\boldsymbol{\eta}_{ref}(t+i|t)\right\|_{\boldsymbol{Q}}^2 +\sum_{i=0}^{N_c-1}\left\|\Delta\boldsymbol{u}(t+i|t)\right\|_{\boldsymbol{R}}^2+\rho\varepsilon^2 \tag{13}$$

where $\Delta\boldsymbol{U}(t)$ is the desired control increment sequence, ε is the relaxation factor, $\boldsymbol{Q}$ is the output deviation weight matrix, $\boldsymbol{R}$ is the control variable weight matrix, and ρ is the relaxation factor weight coefficient.

After establishing the objective function and adding constraints, the path following control is transformed into solving the optimization problem shown in Equation (14).

$$\begin{aligned}&\min_{\Delta\boldsymbol{U}(t),\varepsilon} J(\boldsymbol{\xi}(t),\boldsymbol{u}(t-1),\Delta\boldsymbol{U}(t),\varepsilon)\\ &\text{s.t. } \Delta\boldsymbol{u}_{\min}\le\Delta\boldsymbol{u}(k)\le\Delta\boldsymbol{u}_{\max},\\ &k=t,t+1,\cdots,t+N_c-1\\ &\boldsymbol{u}_{\min}\le\boldsymbol{u}(k)\le\boldsymbol{u}_{\max},\\ &k=t,t+1,\cdots,t+N_c-1\\ &\boldsymbol{\eta}_{\min}-\varepsilon\mathbf{1}_p\le\boldsymbol{\eta}(k)\le\boldsymbol{\eta}_{\max}+\varepsilon\mathbf{1}_p,\\ &k=t+1,t+2,\cdots,t+N_p\\ &0\le\varepsilon\le\varepsilon_{\max}\end{aligned} \tag{14}$$

where $\Delta\boldsymbol{u}_{\min}$ and $\Delta\boldsymbol{u}_{\max}$ are the upper and lower limits of the control increment, $\boldsymbol{u}_{\min}$ and $\boldsymbol{u}_{\max}$ are the upper and lower limits of the control quantity, $\boldsymbol{\eta}_{\min}$ and $\boldsymbol{\eta}_{\max}$ are the upper and lower limits of the output quantity, and $\mathbf{1}_p$ is the p-dimensional column vector.

The Equation (14) is transformed into a quadratic programming problem, and the control increment sequence $\Delta\boldsymbol{U}(t)$ in the control time domain is obtained by solving. The first quantity is taken as the actual control increment, and the input of the controlled system is input according to the previous control cycle. Repeat the above-mentioned solution process in each control cycle to obtain real-time feedback control quantity and realize the path tracking control of the vehicle.

3.3. Optimal Time Domain Parameter Selection

The time domain parameters of the MPC controller are N_p in the predicted time domain and N_c in the control time domain, which have great impact on the control effects.

Equation (14) shows that when other controller parameters remain unchanged, the larger the prediction time domain N_p, the controller can predict further position and obtain more vehicle status information. However, if N_p is too large, it increases the control considering the deviation of the output from the current position of the vehicle, and weakens the control of the deviation of the output closer to the current position of the vehicle, resulting in the control value obtained by solving the objective function is not the optimal control amount, which leads to the increase in the tracking deviation at close distance. When the controller enters the next sampling period to re-solve the optimal control value of the next period, it will continue to increase the tracking deviation, and finally increase the overall tracking deviation through accumulation. And too large N_p will increase the calculation time of the controller. When N_p is too small, the real-time performance of the controller will improve, but the predicted future vehicle state information is too little. Under the premise of the control amount and control increment constraints, the front wheels cannot be turned in time, which will also lead to an increase in tracking deviation. It may even cause the vehicle to lose control, as shown in Figure 4.

When other controller parameters remain unchanged, the greater the N_c, the higher the control sensitivity and accuracy, but the system stability and real-time performance will decrease. When N_c becomes smaller, the stability and real-time performance of the controller are better, but the control accuracy is lowered. For example, when N_c = 1, in the prediction time domain, the controller only performs one control action. In order to ensure the minimum average deviation in the prediction time domain, the accuracy of the output near the current position of the solved control variable will be reduced, as shown in Figure 5.

In order to obtain the optimal time-domain parameters at different vehicle speed, a joint simulation platform was built based on Carsim/Simulink, and the limiting operation condition of double-line shifting was selected as the reference path to conduct simulation tests at different driving speed. The vehicle parameters and controller parameters used in simulation are as shown in Tables 1 and 2.

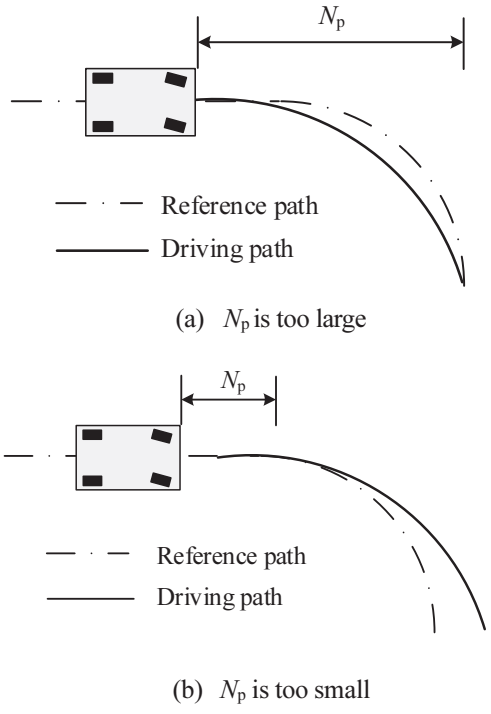
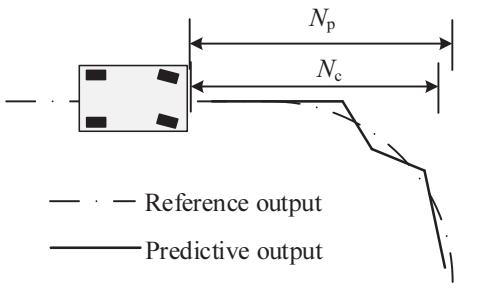
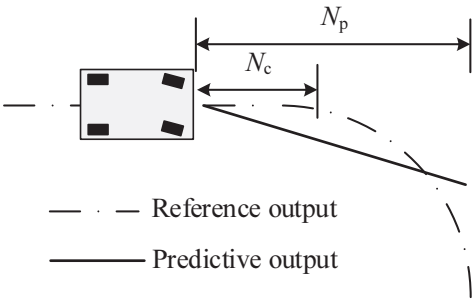


Figure 4. Prediction time domain influence tracking effect.



(a) N_c is too large



(b) N_c is too small

Figure 5. Controlling time domain influence tracking effect.

Table 1. Vehicle parameters.

Vehicle quality, m	1,723 kg
Length × width, $L \times B$	4.893 m × 1.862 m
Distance from front axle to center of mass, a	1.232 m
Distance from rear axle to center of mass, b	1.468 m
Front wheel cornering stiffness, C_f	66,900 N·rad^{-1}
Rear wheel cornering stiffness, C_r	62,700 N·rad^{-1}
Tire type	235/45 R18
Moment of inertia around the z axis, I_z	3,234 kg·m^2

Table 2. Controller parameters.

Sampling period, T_c	20 ms
Weight matrix Q	Diag [2000,10000]
Weight matrix R	[5×10^5]
Weight coefficient, ρ	1000
Control amount constraints	[-10°,10°]
Control incremental constraints	[-0.85°,0.85°]

The maximum lateral position deviation E_{dmax}, the average lateral position deviation E_{dm} and the average yaw angle deviation $E_{\varphi m}$ are used to measure the deviation between the actual tracking path and the reference path, as shown in Equations (15) ~ (17).

$$E_{\mathrm{dmax}} = \max\left[E_{\mathrm{d}}(i)\right],\ i = 1,\ 2,\ \cdots,\ \mathrm{N} \tag{15}$$

$$E_{\mathrm{dm}} = \frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} E_{\mathrm{d}}(i) \tag{16}$$

$$E_{\varphi\mathrm{m}} = \frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}\left[\varphi(i) - \varphi_{\mathrm{r}}(i)\right] \tag{17}$$

where E_d is the lateral deviation of the vehicle from the reference path, N is the total number of measurements, φ is the actual yaw angle, and φ_r is the reference yaw angle.

Path tracking control should meet the requirements of tracking deviation and ensure the driving stability of the vehicle during path tracking. By constraining the vehicle's maximum mass center slip angle and maximum yaw rate, vehicle instability can be prevented. The vehicle's maximum center of mass slip angle and maximum yaw rate constraint conditions are shown in Equations (18) ~ (19) (Chen *et al*., 2012).

$$\left|\beta_{\max}\right| \le \arctan\left(0.02\mu g\right) \tag{18}$$

$$\left|\omega_{\max}\right| \le \frac{0.85\mu g}{v_x} \tag{19}$$

Where β_{max} is the maximum side slip angle of the center of mass, and ω_{max} is the maximum yaw rate.

Taking the driving speed of 30, 40, 50, 60 and 70 km/h as the simulation conditions, the maximum N_p is 40, and the maximum N_c is (N_p-1). Neglecting the combinations which caused out of the control, the total of 3900 combinations are traversed. Simulation results show that the vehicle was out of control with some parameter combinations. In order to ensure the validity of the controller's time domain parameters, the time domain parameter screening criteria are formulated as follows.

1. The set time domain parameters can ensure that the controller can operate normally: the vehicle does not lose control and cause path tracking failure.
2. The set time domain parameters can ensure the driving safety of the vehicle: the maximum value of the parameters such as the center of mass slip angle and the yaw rate does not exceed the upper limit of the constraint.
3. The set time domain parameters can ensure the real-time performance of the controller: the solution time is less than the sampling period.

According to the time-domain parameter selection criteria, after excluding invalid time-domain parameters, simulation results of the effective time-domain parameter combination are shown in Table 3.

Table 3. Simulation results of dual time domain parameters.

v/(km/h)	N_p	N_c	E_{dmax}/m	E_{dm}/m	$E_{\varphi m}$/°	β_{max}/°	ω_{max}/(°/s)
30	17	1	0.072	0.017	0.761	1.380	13.181
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
30	19	16	0.067	0.017	0.746	1.349	10.870
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
50	22	4	0.108	0.030	0.546	0.497	18.890
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
70	33	2	0.215	0.055	0.541	1.337	19.190
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
70	40	1	0.645	0.125	1.210	0.633	13.602

Simulation results show that there is a conflict between the control accuracy of the vehicle and the driving stability. In order to select the optimal prediction time domain parameters and control time domain parameters, the controller's control accuracy and driving stability of the vehicle should be balanced. After comprehensively considering the control accuracy and driving stability, a comprehensive evaluation index of path following performance is proposed, as shown in Equation (20).

$$S_c = Q_1 E_{dmax} + Q_2 E_{dm} + Q_3 E_{\varphi m} + R_1 \beta_{max} + R_2 \omega_{max}. \quad (20)$$

where S_c is the evaluation function, and Q_1, Q_2, Q_3, R_1 and R_2 are the weight coefficients of the maximum lateral deviation, the average lateral deviation, the average yaw angle deviation, the maximum centroid side slip angle and the maximum yaw rate, respectively.

Since the focus of this paper is on obstacle avoidance driving conditions, it focuses on the control accuracy, especially the maximum lateral deviation parameter. According to the different emphasis, and taking into account the different dimensions of the above parameters, through simulation experiments, the weight coefficient values of Q_1, Q_2, Q_3, R_1, R_2 are determined to be 200, 400, 40, 20, and 1, respectively.

After the evaluation function is established, the results of the dual time domain parameter test are substituted into Equation (20) for score calculation. And the lowest S_c value is selected as the optimal prediction time domain parameter and control time domain parameter at the corresponding vehicle speed. The results are shown in Table 4.

Table 4. Optimal dual time domain parameters at different vehicle speeds.

v/(km/h)	30	40	50	60	70
N_p	19	20	22	28	33
N_c	16	8	4	3	2

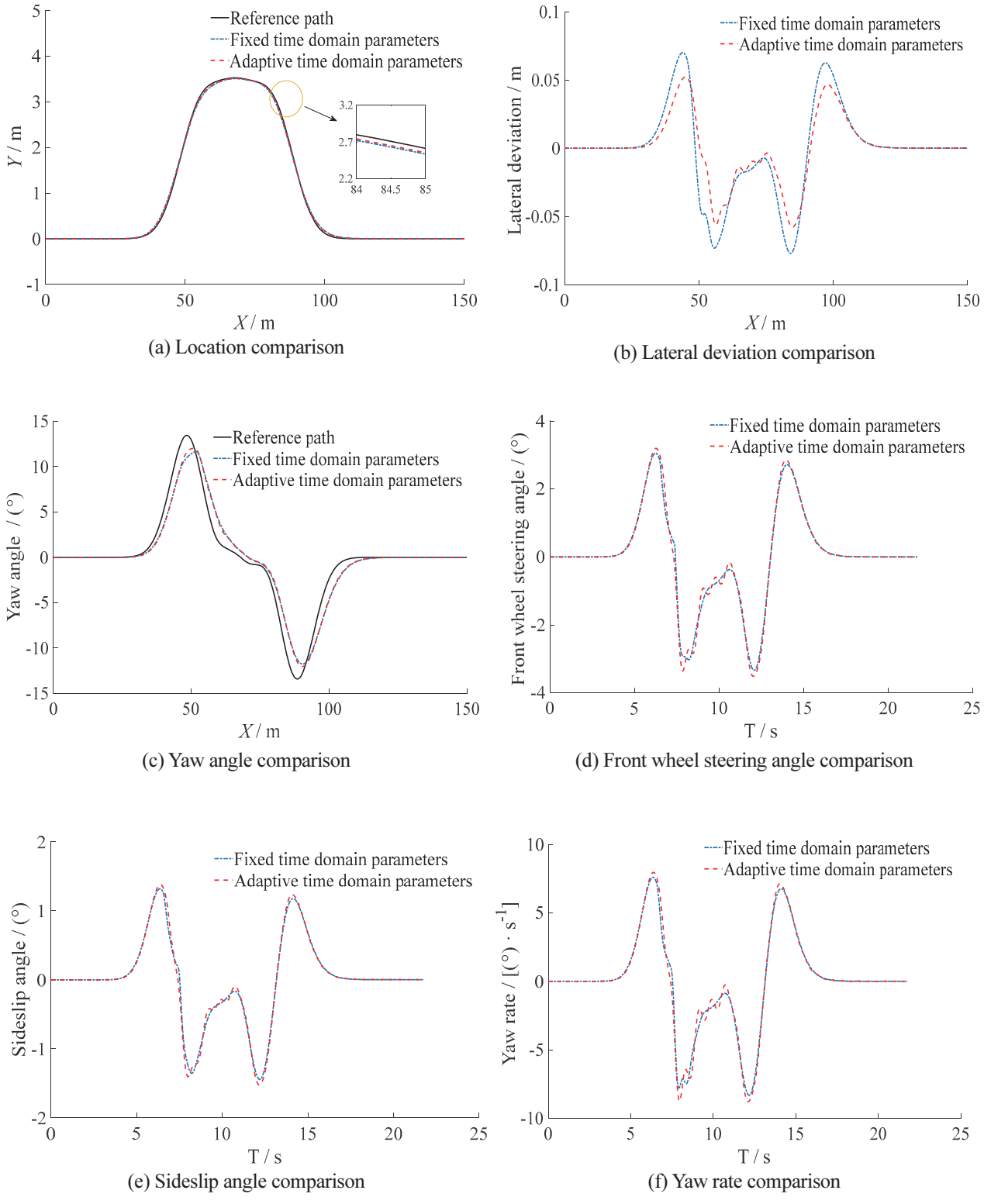
After obtaining the optimal dual time domain parameters at different vehicle speeds, an adaptive dual time domain parameter control strategy is formulated, as shown in Equation (21).

$$[N_p, N_c] = \begin{cases} [19,16] & v \le 30 \\ [20,8] & 30 < v \le 40 \\ [22,4] & 40 < v \le 50 \\ [28,3] & 50 < v \le 60 \\ [33,2] & v > 60 \end{cases} \quad (21)$$

3.4. Controller Performance Verification

In order to verify the optimal dual-time domain parameter selection method, the vehicle speed is set to 25, 35, 45, 55 and 65 km/h, respectively. The fixed dual-time domain parameters controller and the adaptive dual-time domain parameters controller are used to carry out double-line shifting simulation experiments, in which the fixed dual-time domain parameters are set to $N_p = 25$ and $N_c = 1$. Simulation results of vehicle location, lateral deviation, yaw angle, front wheel steering angle, sideslip angle and yaw rate using the fixed time domain parameters controller and adaptive time domain parameters controller at 25 km/h, 45 km/h and 65 km/h are shown in Figures 6 ~ 8. And Table 5 and Figure 9 show the control performance using the fixed time domain parameters controllers and adaptive time domain parameters controllers at different vehicle speeds.

Simulation results show that at different vehicle speeds, the front wheel angle, the front wheel angle increment, the maximum sideslip angle, and the maximum yaw rate of the adaptive time domain parameter controller all meet the constraint conditions. Compared with the fixed time domain parameter controller, the adaptive time domain parameter controller has smaller maximum lateral deviation, smaller average lateral deviation, smaller average yaw angle deviation and smaller comprehensive evaluation index (S_c). As the vehicle speed increases, the difference between the two becomes more obvious. When the vehicle speed reaches 65 km/h, the lateral deviation of the fixed time domain parameter controller is 0.368 m, and the maximum lateral deviation of the adaptive time domain parameter controller is only 0.199 m, with a difference of 0.169 m. The improvement effect is significant. The maximum yaw rate of the fixed time domain parameter controller is 20.591°/s, which is close to the theoretical upper limit of 21°/s, and the driving stability of the vehicle becomes worse, while the maximum yaw rate of the adaptive time domain parameter controller is 17.395°/s, the driving stability has been significantly improved, indicating that the adaptive time domain parameter controller can be used to improve driving safety in extreme conditions.



(a) Location comparison

(b) Lateral deviation comparison

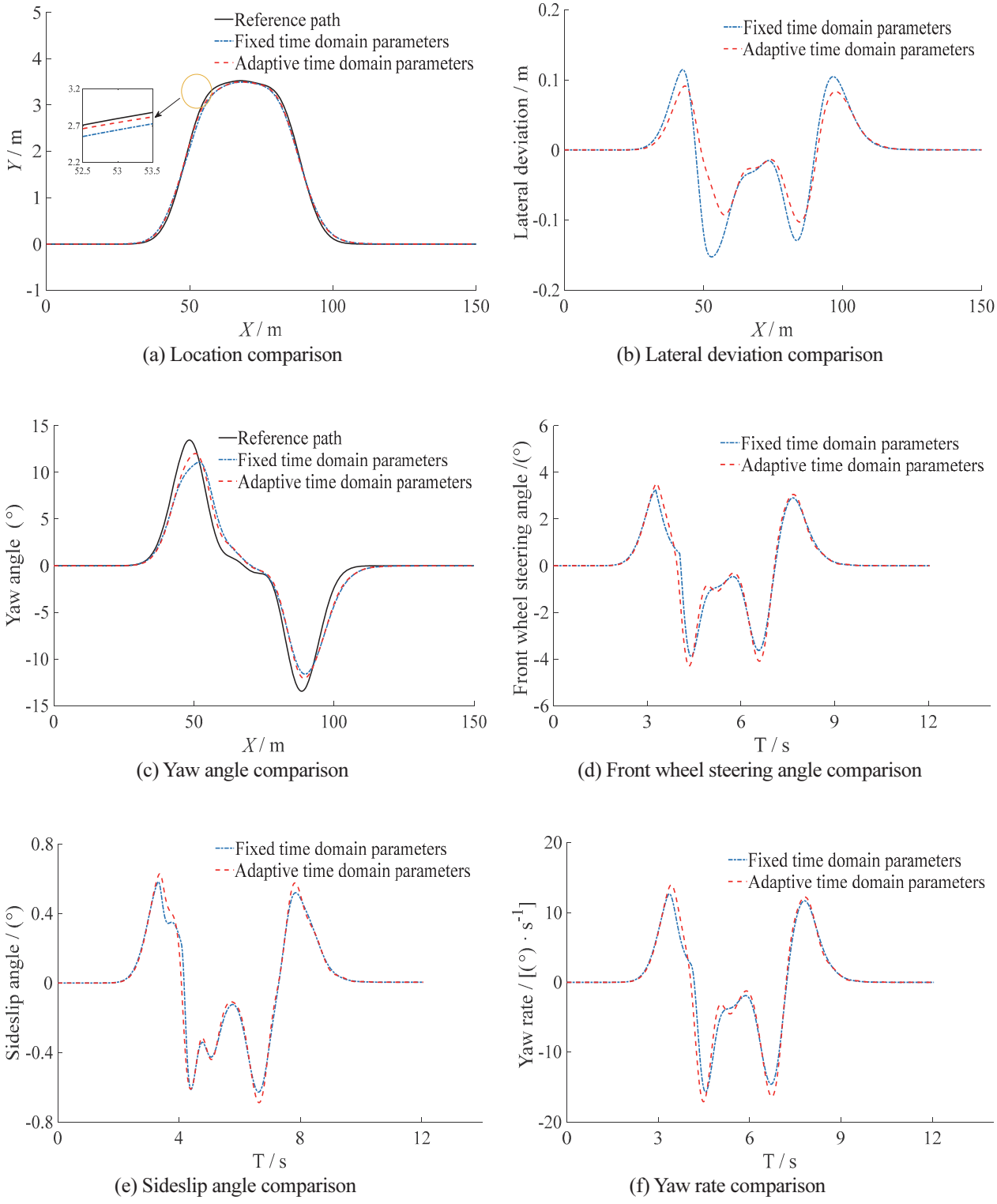
(c) Yaw angle comparison

(d) Front wheel steering angle comparison

(e) Sideslip angle comparison

(f) Yaw rate comparison

Figure 6. Comparison of the 25 km/h path tracking situation.



(a) Location comparison

(b) Lateral deviation comparison

(c) Yaw angle comparison

(d) Front wheel steering angle comparison

(e) Sideslip angle comparison

(f) Yaw rate comparison

Figure 7. Comparison of 45 km/h path tracking situation.

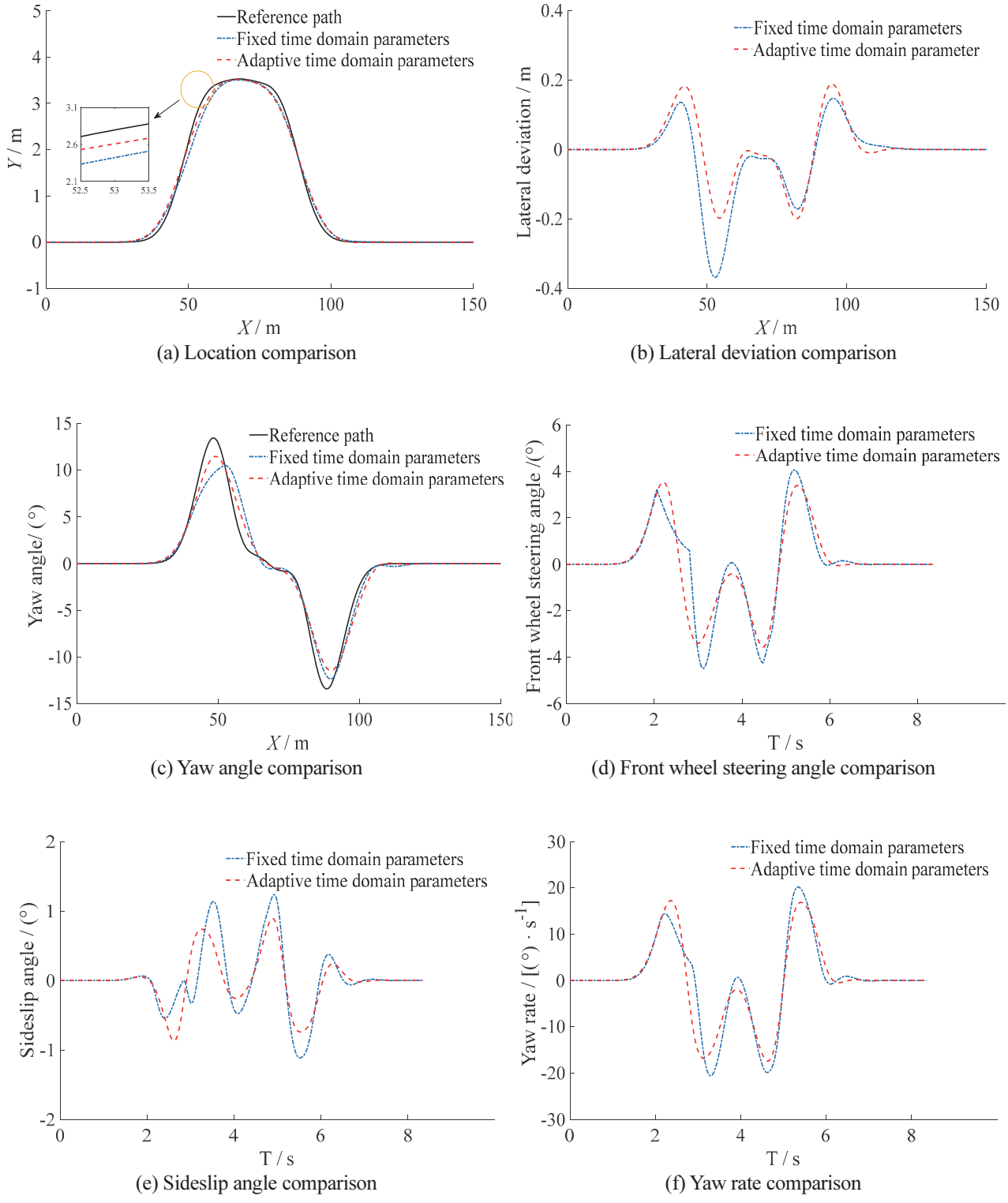
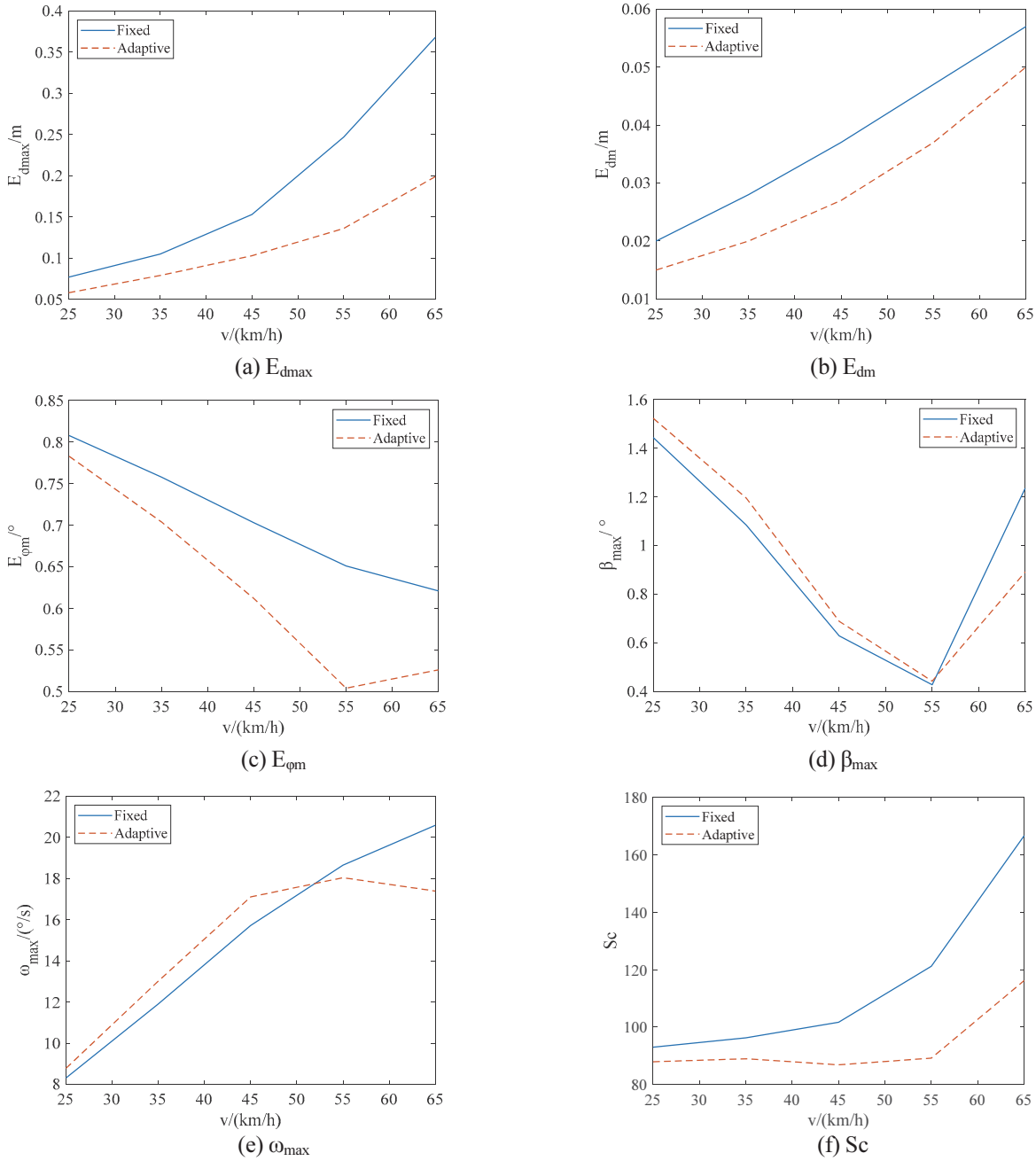


Figure 8. Comparison of the 65 km/h path tracking situation.

Table 5. Control performance of different MPC controllers.

v/(km/h)	E_{dmax}/m		E_{dm}/m		$E_{\varphi m}$/°		β_{max}/°		ω_{max} /(°/s)		Sc		
	Fixed	Adaptive	Fixed	Adaptive	Fixed	Adaptive	Fixed	Adaptive	Fixed	Adaptive	Fixed	Adaptive	Difference
25	0.077	0.058	0.020	0.015	0.808	0.783	1.444	1.523	8.317	8.784	93.029	87.941	5.088
35	0.105	0.079	0.028	0.020	0.758	0.704	1.084	1.195	11.909	13.019	96.254	88.995	7.259
45	0.153	0.103	0.037	0.027	0.703	0.612	0.628	0.688	15.720	17.108	101.719	86.879	14.84
55	0.247	0.136	0.047	0.037	0.651	0.504	0.427	0.442	18.655	18.044	121.202	89.215	31987
65	0.368	0.199	0.057	0.050	0.621	0.526	1.235	0.891	20.591	17.395	166.634	116.193	50.441



(a) E_{dmax} (b) E_{dm}

(c) $E_{\varphi m}$ (d) β_{max}

(e) ω_{max} (f) Sc

Figure 9. Control performance of different MPC controllers.

4. LOCAL OBSTACLE AVOIDANCE PATH PLANNING AND PATH TRACKING INTEGRATED VERIFICATION

The planning layer of the local path planning MPC controller and the MPC controller of the path tracking layer are integrated to form an integrated controller of the planning layer and the control layer, as shown in Figure 10, where the output planning path of the path planning layer is the input of the path tracking layer.

The above-mentioned integrated controller is simulated and verified, and the parameter settings of the path planning MPC controller are shown in Table 6.

Simulation conditions are as follows: a static obstacle with a length of 50 m and a width of 2 m exists on the right side of a straight road with a total length of 100 m and a width of 3.8 m. The vehicle drives in a straight line along the right lane at a speed of 60 km/h. The global reference path is the centerline of the right lane. Simulation results of obstacle avoidance function, objective function and planned path under the traditional and new obstacle avoidance functions are shown in Figures 11 ~ 13, respectively. The simulation results show that due to the setting of the traditional obstacle avoidance function, the obstacle avoidance function is not zero when the vehicle travels near the obstacle, causing the local path planner to plan the inward obstacle avoidance path, which brings a slight lateral deviation. The maximum lateral deviation reaches 0.276 m, and there is excessive obstacle avoidance, which reduces driving stability and driving comfort. If there are vehicles in the left lane at this time, the possibility of collision is increased. The new obstacle avoidance function is always zero, which makes the local path planned by the local path planner coincide with the global reference path when the vehicle travels near the obstacle. The reference path using new obstacle avoidance function coincides with the global reference path, which shows that the local path planner using the new obstacle avoidance function can avoid excessive obstacle avoidance.

Compared with the traditional obstacle avoidance function, the new obstacle avoidance function needs to increase the process of judging the obstacle type. The STM32H743 microcontroller hardware platform is used to verify the real-time responsibility of the two control strategies. Set up real-time verification conditions. There are three obstacles with a size of 4 × 2 m in the detectable range of the on-board sensor, and the body coordinates at the center position are (6, 1), (11, -2) and (2, -4), each obstacle uses 60 obstacle points to simulate the obstacle edge position information obtained by the vehicle-mounted sensor. Calculate the calculation time of the two obstacle avoidance

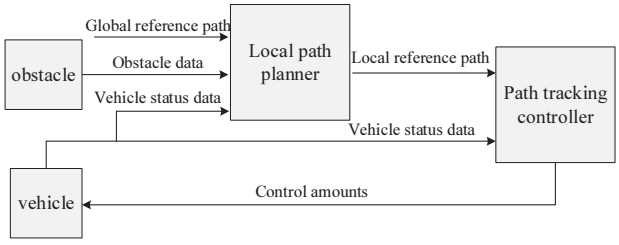


Figure 10. Schematic diagram of the integrated controller structure of the planning layer and control layer.

Table 6. Planner parameters.

Sampling period, T_p	100 ms
Prediction time domain, N_p	25
Control time domain, N_c	1
Weight matrix, Q	Diag [200, 200]
Weight matrix, R	[10]
Obstacle avoidance function weight coefficient, S_{ob}	180
Control incremental constraints	[-0.4 g, 0.4 g]

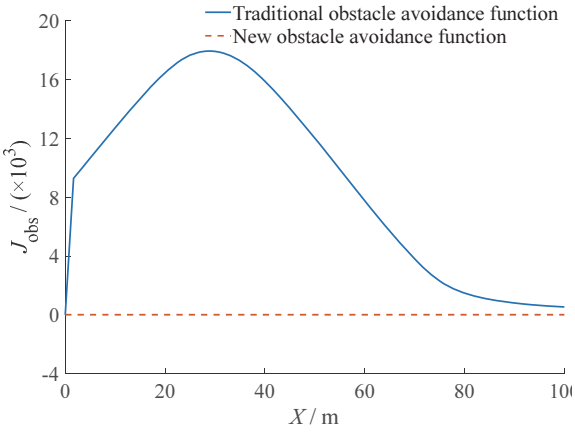


Figure 11. Obstacle avoidance function.

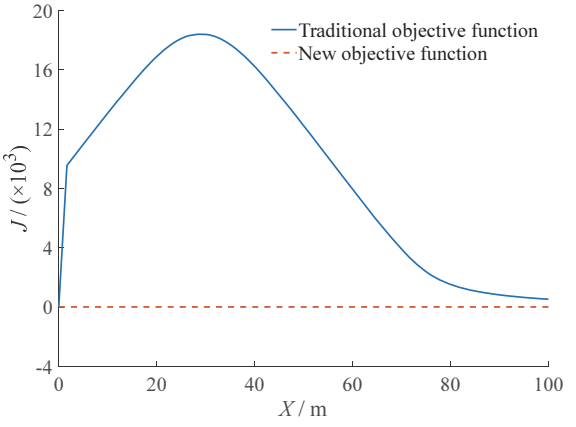


Figure 12. Objective function.

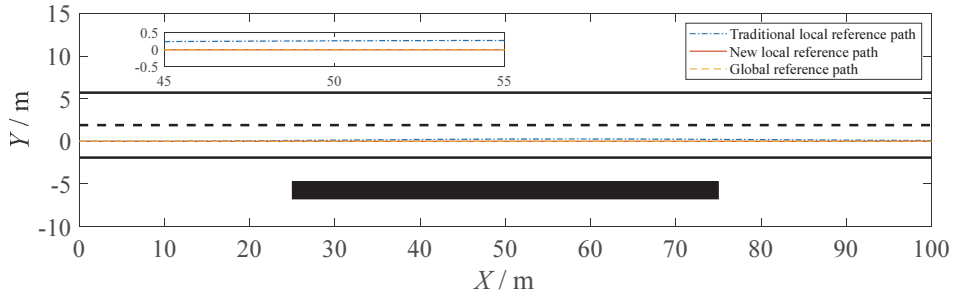


Figure 13. Local plan path.

Table 7. Real-time verification results.

Total number of obstacle points	Calculating time/ms	
	Traditional	New
60 (Case 1)	0.9540	1.0810
120 (Case 2)	1.9030	2.0680
180 (Case 3)	2.8530	3.1470

functions in the three cases respectively. Case 1 assumes that there is only obstacle 1. Case 2 assumes that there are obstacles 1 and 2. Case 3 assumes that there are obstacles 1, 2 and 3. The corresponding total number of obstacle points N is 60, 120 and 180 respectively. In different obstacle avoidance situations, the real-time verification results of calculating 100 times of obstacle avoidance function are shown in Table 7.

It can be seen from Table 7 that in the case of 100 calculations, the calculation time of the new obstacle avoidance function is slightly higher than that of the traditional obstacle avoidance function. When the total number of obstacle points is 180, it is only 0.294 ms more. Since the sampling period of the planning layer is 100 ms, and the obstacle avoidance function in the sampling period is calculated as Np times, which is less than 100 times. Therefore, the new obstacle avoidance function can avoid excessive obstacle avoidance and improve the driving stability of the vehicle while ensuring real-time performance.

In order to verify the performance of the local path planning and the controller integrated controller, this paper carries out the simulation verification of the multi-obstacle avoidance condition. The simulation conditions are set as follows: a straight road with a total length of 150 m and a width of 3.8 m with 4 static obstacles of 5 × 2 m, and their position coordinates are (35, 0), (70, 3.8), (105, 0) and (140, 3.8). The vehicle travels at a constant speed of 30 km/h, and the initial position of the vehicle is the origin of the geodetic coordinate system. The schematic diagram is shown in Figure 14. The planned path of the planning layer and the actual path of the tracking layer are shown in Figure 15, and the side slip angle and yaw rate of the vehicle center of mass

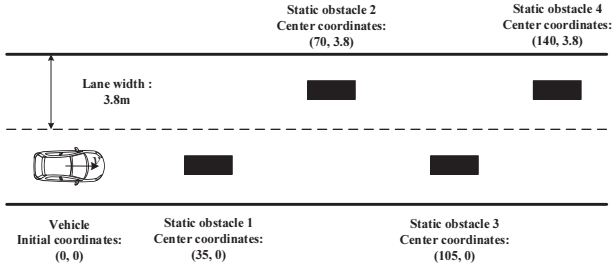


Figure 14. Schematic diagram of obstacle avoidance conditions.

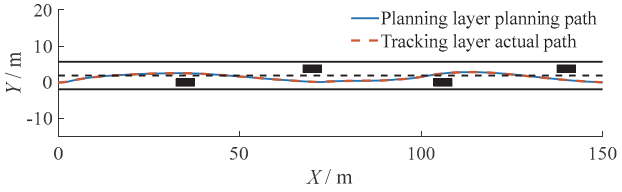


Figure 15. Planned path of planning layer and actual path of tracking layer.

are shown in Figures 16 and 17. Table 8 shows the maximum lateral deviation, average lateral deviation, average yaw angle deviation, maximum center of mass slip angle, and maximum yaw rate between the actual path and the planned path.

The simulation results show that the vehicle can safely avoid all obstacles in the actual driving process, and the driving state of the vehicle is stable. The maximum lateral deviation between the actual path of the vehicle and the reference path is only 0.293 m, and the average lateral deviation and the average yaw angle deviation are both small. Because the initial speed, initial position and heading are set to zero, the vehicle's initial center of mass slip angle and yaw rate fluctuate greatly, but they are within a reasonable range. The simulation results show that the integrated controller of local path planning and control can make the vehicle plan a safe reference path under multi-obstacle conditions, and realize accurate tracking, and realize the function of vehicle obstacle avoidance.

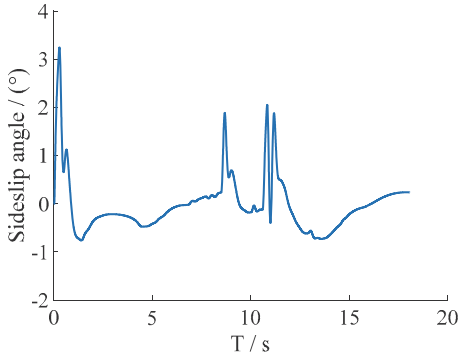


Figure 16. Sideslip angle.

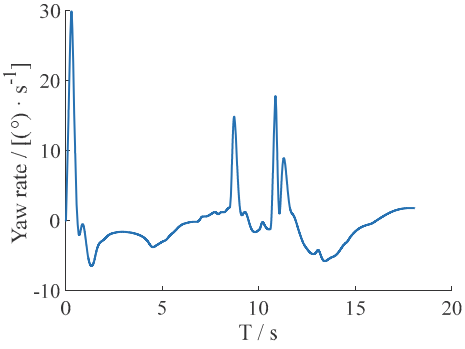


Figure 17. Yaw rate.

Table 8. Simulation results of obstacle avoidance conditions.

E_{dmax}/m	E_{dm}/m	$E_{\varphi m}$/°	β_{max}/°	ω_{max}/(°/s)
0.293	0.017	3.836	3.244	29.946

In order to verify the performance of the integrated controller for local path planning and control under dynamic obstacle conditions, the simulation of dynamic obstacle avoidance conditions is designed in this paper. Simulation conditions are as follows: a linear road with a total length of 100 m and a width of 3.8 m has dynamic obstacles of 10 m × 2 m. The dynamic obstacle is moving in the right lane at 30 km/h, and its initial position center coordinate is (22.5, 0). The vehicle travels at 60 km/h, and the initial position is the origin of the geodetic coordinate system. The planned path at the planning layer, the actual path at the tracking layer, and the real-time location of dynamic obstacles are shown in Figure 18.

It can be seen from the simulation results that the vehicle can safely avoid dynamic obstacles, and the maximum lateral deviation between the actual path of the vehicle and the reference path is 0.088 m. The simulation results show that the local path planning and control integrated controller can make the vehicle plan a safe reference path under dynamic obstacle conditions, and achieve precise tracking, and realize the vehicle's obstacle avoidance function against dynamic obstacles.

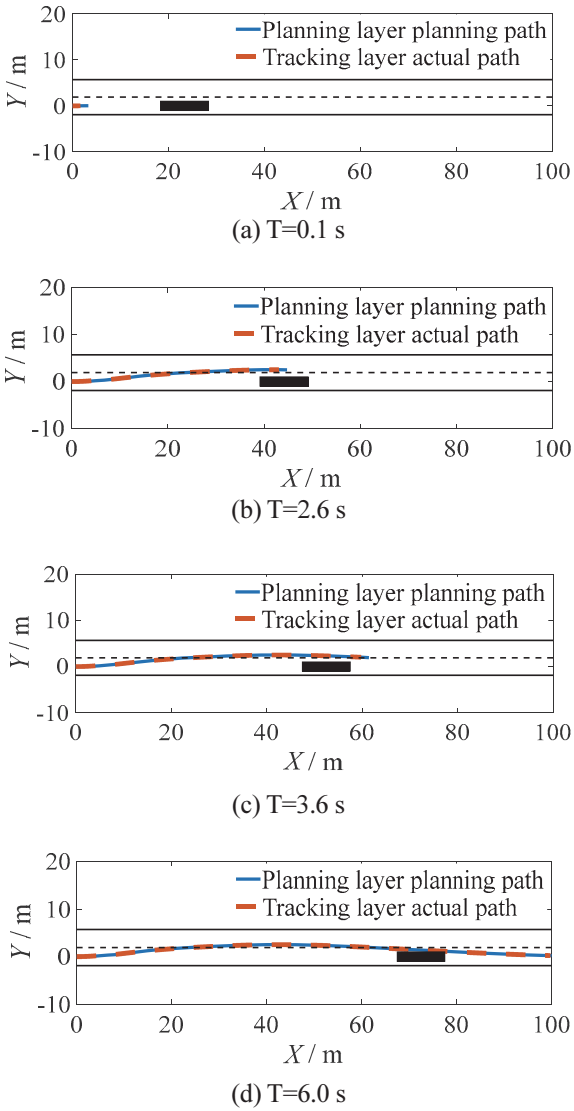


Figure 18. Planning layer planning path and tracking layer actual path.

5. CONCLUSION

The local obstacle avoidance path planner and path tracking controller are designed for the obstacle avoidance problem of vehicles. According to the position relationship between the vehicle and the obstacle, a new obstacle avoidance function is designed, and the local path planner is designed based on the nonlinear model predictive control algorithm to realize the real-time planning of the local path in the obstacle scene. Aiming at the influence of the controller's time

domain parameters on the performance of the controller, the optimal time domain parameters at different vehicle speeds are obtained through theoretical analysis and combined with simulation experiments. Furthermore, an adaptive time-domain parameter path tracking controller is designed to realize the accurate tracking of the reference path by the vehicle. Set up a joint simulation platform to simulate and verify the integrated structure of the planning layer and the control layer. The simulation results show that the new obstacle avoidance function can avoid excessive obstacle avoidance and improve the driving stability of the vehicle while ensuring real-time performance. The adaptive time-domain parameter path tracking controller has better comprehensive control performance and can be used to improve driving safety under extreme conditions. The integrated structure of local obstacle avoidance path planning and path tracking control can control the vehicle to plan and accurately track the local obstacle avoidance path in multiple static obstacle scenes, so as to realize the safe and stable driving of the vehicle.

ACKNOWLEDGEMENT–This work was supported by National Natural Science Foundation of China (51207012) and Natural Science Foundation of Shaanxi Province (2021JM-163).

REFERENCES

Amer, N. H., Zamzuri, H., Hudha, K. and Kadir, Z. A. (2017). Modelling and control strategies in path tracking control for autonomous ground vehicles: A review of state of the art and challenges. *J. Intelligent & Robotic Systems* **86**, **2**, 225–254.

Chen, W., Liu, X., Huang, H. and Yu, H. (2012). Research on side slip angle dynamic boundary control for vehicle stability control considering the impact of road surface. *J. Mechanical Engineering* **48**, **14**, 112–118.

Chu, K., Lee, M. and Sunwoo, M. (2012). Local path planning for off-road autonomous driving with avoidance of static obstacles. *IEEE Trans. Intelligent Transportation Systems* **13**, **4**, 1599–1616.

Corno, M., Panzani, G., Roselli, F., Giorelli, M., Azzolini, D. and Savaresi, S. M. (2021). An LPV approach to autonomous vehicle path tracking in the presence of steering actuation nonlinearities. *IEEE Trans. Control Systems Technology* **29**, **4**, 1766–1774.

Dolgov, D., Thrun, S., Montemerlo, M. and Diebel, J. (2010). Path planning for autonomous vehicles in unknown semi-structured environments. *Int. J. Robotics Research* **29**, **5**, 485–501.

Gong, J., Liu, K. and Qi, J. (2020). *Model Predictive Control for Self-driving Vehicles*. 2nd edn. Beijing Institute of Technology Press, Beijing, China.

Hang, P., Luo, F., Fang, S. and Chen, X. (2017). Path tracking control of a four-wheel-independent-steering electric vehicle based on model predictive control. *36th Chinese Control Conf. (CCC)*, Dalian, China.

Huang, Y., Ding, H., Zhang, Y., Wang, H., Cao, D., Xu, N. and Hu, C. (2020). A motion planning and tracking framework for autonomous vehicles based on artificial potential field elaborated resistance network approach. *IEEE Trans. Industrial Electronics* **67**, **2** ,1376–1386.

Jeon, J. H., Cowlagi, R. V., Peters, S. C., Karaman, S., Frazzoli, E., Tsiotras, P. and Iagnemma, K. (2013). Optimal motion planning with the half-car dynamical model for autonomous high-speed driving. *IEEE American Control Conf. (ACC)*, 188–193.

Li, X., Sun, Z., Cao, D., Liu, D. and He, H. (2017). Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles. *Mechanical Systems and Signal Processing*, **87**, 118–137.

Li, Y., Liu, Y., Feng, Q., Nan, Y., He, J. and Fan, J. (2020). Path tracking control for an intelligent commercial vehicle based on optimal preview and model predictive. *J. Automotive Safety and Energy* **11**, **4**, 462–469.

Liu, J., Yang, J., Liu, H., Tian, X. and Gao, M. (2017). An improved ant colony algorithm for robot path planning. *Soft Computing* **21**, **19**, 5829–5839.

Lu, H., Xing, Y. and Zhuo, G. (2020). Review on motion control of autonomous vehicles. *J. Mechanical Engineering* **56**, **10**, 127–143.

Wang, Z., Li, G., Jiang, H., Chen, Q. and Zhang, H. (2018). Collision-free navigation of autonomous vehicles using convex quadratic programming-based model predictive control. *IEEE/ASME Trans. Mechatronics* **23**, **3**, 1103–1113.

Zhang, X. and Zhu, X. (2019). Autonomous path tracking control of intelligent electric vehicles based on lane detection and optimal preview method. *Expert Systems with Applications*, **121**, 38–48.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.