



Intelligent vehicle obstacle avoidance path-tracking control based on adaptive model predictive control

Baorui Miao^{1,2} and Chao Han^{1,2}

¹School of Electrical Engineering (School of Integrated Circuits), Anhui Polytechnic University, Wuhu, 241000, China

²Key Laboratory of advanced perception and intelligent control of high-end equipment, Ministry of Education, Anhui Polytechnic University, Wuhu, 241000, China

Correspondence: Chao Han (hanchaozh@126.com)

Received: 10 June 2022 – Revised: 20 April 2023 – Accepted: 10 May 2023 – Published: 15 June 2023

Abstract. In order to solve the problems of low path-tracking accuracy, poor safety, and stability of intelligent vehicles with variable speeds and obstacles on the road, a double-layer adaptive model predictive controller (MPC) is designed. A vehicle point mass model is used in an obstacle avoidance planning controller, and the safety collision distance model is established according to the distance relationship between the vehicle and the obstacle to improve the driving safety of the vehicle. The design of the path-tracking controller is based on the three-degrees-of-freedom dynamics model. According to the relationship between the predictive horizon and vehicle speed in the MPC algorithm, an adaptive path-tracking control strategy which can update the prediction horizon in real time is proposed to improve the accuracy of vehicle path tracking. To increase the vehicle stability, a sideslip angle and an acceleration control variable are added to the vehicle dynamics model as soft constraint conditions. The proposed method is simulated based on a CarSim and MATLAB/Simulink co-simulation platform. The simulation results show that the maximum lateral path deviation and the maximum centroid sideslip angle of the designed controller are 0.13 m and 0.4°, respectively. Compared with the traditional MPC, the adaptive MPC maximum lateral path deviation and the maximum centroid sideslip angle are reduced by 0.51 m and 1.57°, respectively, which proves the effectiveness of the proposed method.

1 Introduction

With the rapid development of artificial intelligence and sensor technology, intelligent vehicle technology has made great progress. Obstacle avoidance planning and path-tracking control are the key technologies of intelligent vehicle automatic driving. Obstacle avoidance planning calculates a safe and stable driving path based on the information about obstacles. Path-tracking control ensures that the vehicle can accurately track the reference path. Therefore, the intelligent vehicle obstacle avoidance path-tracking control has become a research hotspot in recent years.

On the path-tracking control problem, Gutjahr et al. (2016) proposed a linear time-varying MPC control method, but the method is based on the kinematic model without considering the vehicle dynamics constraints. Mata et al. (2018) presented a trajectory-tracking control method based on multi-

ple constraints, and the continuous linearization error model and quadratic programming were used to obtain a good trajectory-tracking effect. Liu et al. (2018) proposed a hierarchical vision-based lateral-control scheme, and the controller is designed by a robust H_∞ -based linear quadratic regulator (LQR) algorithm to compensate for sensor-induced delays. Mata et al. (2018) and Liu et al. (2018) did not take into account the view of obstacles in the actual driving process of the vehicle. On the obstacle avoidance planning problem, Karaman et al. (2011) proposed an anytime algorithm based on RRT* to generate the planned path, but the stability of the algorithm was not considered. The stability of the algorithm will affect the convergence time and whether the generated path can converge to the global optimal path. Tomas-Gabarron et al. (2013) converted the obstacle avoidance route of intelligent vehicles into a multitasking objec-

tive problem and computed the safe obstacle avoidance path. Chen et al. (2019) switched the path-planning problem of intelligent vehicles to the problem solved by the Bezier curve and the feasible planning path obtained by a genetic algorithm. The obstacle avoidance planning algorithm proposed by Tomas-Gabarron et al. (2013) and Chen et al. (2019) has a lot of computation, and the model is complex. Considering the real road conditions in the process of intelligent vehicle driving and the large amount of calculation in obstacle avoidance planning and path-tracking control, a simplified vehicle model should be adopted in this paper.

On the joint problem of obstacle avoidance planning and path-tracking control, the sensing module, including vehicle sensors, cameras, and radar modules, detects the position coordinates of obstacles on the road in real time and sends the coordinate information to the obstacle avoidance planning layer. The obstacle avoidance planning layer uses the coordinate information to calculate and give the optimal path, and then the path-tracking control layer controls the vehicle to drive along the path. Li et al. (2018) presented an integrated method of trajectory planning and control based on a nonlinear vehicle model predictive control algorithm and improved the four-wheel dynamics model and nonlinear tire model. Xu et al. (2020) proposed a powerful hierarchical path-planning and trajectory-tracking framework. For path planning, considering vehicle kinematics, a feasible and collision-free trajectory is generated. For path tracking, the front-wheel steering angle is obtained by solving the constraint model predictive control problem. Zuo et al. (2021) proposed a progressive model predictive control scheme in which the cooperative control of local planning and path tracking of intelligent vehicles is considered. The model predictive control is combined with the artificial potential field method. The above two methods have had a lot of research on the obstacle avoidance path-tracking control of intelligent vehicles. However, in actual driving, the vehicle is often in a variable speed state, and there are obstacles in the road. Thus, driving safety and stability are being challenged. We propose an adaptive double-layer model predictive control strategy for intelligent vehicle obstacle avoidance planning and path-tracking control in this paper, and the feasibility of the proposed control strategy is verified by simulation. The innovation and contribution of this paper are as follows.

- I. To ensure the security of the intelligent vehicle obstacle avoidance trajectory-tracking control, an obstacle avoidance function is designed in the controller based on the relationship between obstruction and vehicle location, and a safe collision distance model is built.
- II. A double-layer adaptive variable prediction horizon obstacle avoidance route-tracking controller is created, and an adaptive variable prediction horizon control approach is suggested. The controller can increase the accuracy of intelligent vehicle path tracking by calculat-

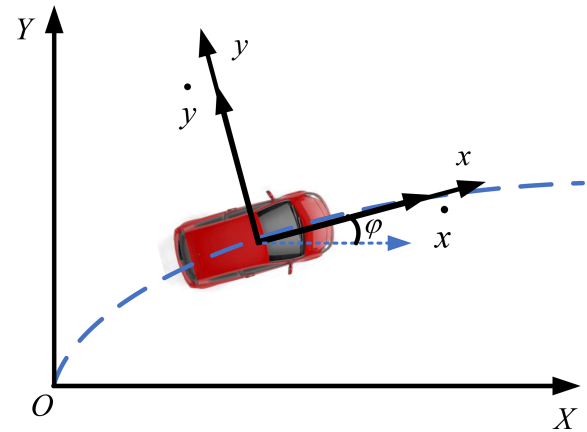


Figure 1. Vehicle point mass model.

ing the best prediction horizon in real time in response to changes in vehicle speed.

- III. Sideslip angle and acceleration control variables are introduced into the vehicle dynamics model as soft constraints to improve vehicle stability for intelligent vehicle driving.

2 Vehicle model

2.1 Point mass model

In order to avoid obstacles on the road, the intelligent vehicle obstacle avoidance planning controller needs to plan a feasible target path based on the position information between the vehicle and the obstacle. In the process of obstacle avoidance planning and path-tracking control of intelligent vehicles, the calculation amount of the algorithm in the obstacle avoidance planning controller is large. In the obstacle avoidance planning controller, the vehicle running on the planning reference path can be regarded as a mass point, so the vehicle point mass model is adopted, which not only simplifies the model, but also reduces the amount of calculation (Li et al., 2020). The vehicle point mass model diagram is shown in Fig. 1.

In the established reference coordinate system, the vehicle point mass model can be expressed as

$$\dot{X} = \dot{x} \cos \psi - \dot{y} \sin \psi, \quad (1)$$

$$\dot{Y} = \dot{x} \sin \psi + \dot{y} \cos \psi, \quad (2)$$

where \dot{X} and \dot{Y} are the longitudinal and transverse velocities in the inertial coordinate system, respectively. \dot{x} and \dot{y} are the longitudinal and transverse velocities in the vehicle coordinate system, respectively. ψ is the vehicle's course angle.

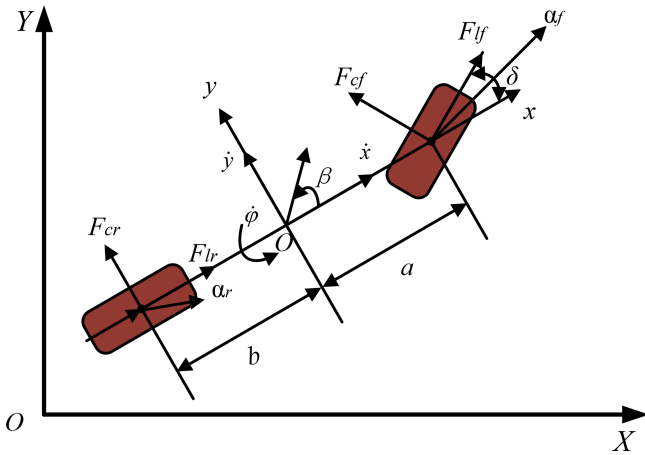


Figure 2. Three-degrees-of-freedom dynamics model.

2.2 Three-degrees-of-freedom dynamics model

Considering the longitudinal, lateral, and yaw motions of the vehicle, the point mass model is no longer suitable for vehicle path-tracking control (Bobier-Tiu et al., 2019; Kucuk, 2017). In order to make the vehicle track the planned path quickly and accurately under high-speed driving conditions, it is necessary to establish a vehicle dynamics model that can accurately describe the vehicle’s motion state in the path-tracking controller. Figure 2 shows a three-degrees-of-freedom dynamics model of a vehicle.

Assuming that the vehicle travels on a horizontal road and ignores the effect of air resistance, the vehicle dynamics equation based on Newton’s second law is as follows (Nan et al., 2021; Woo et al., 2021).

$$m\ddot{x} = m\dot{y}\dot{\varphi} + 2(F_{lf} + F_{cf}\delta + F_{lr}), \tag{3}$$

$$m\ddot{y} = 2(F_{cf} + F_{cr}) - m\dot{x}\dot{\varphi}, \tag{4}$$

$$I_z\ddot{\varphi} = 2(aF_{cf} - bF_{cr}), \tag{5}$$

where m is the vehicle body mass, I_z is the vehicle yaw moment of inertia, $\ddot{\varphi}$ is the yaw angle acceleration, δ is the front wheel angle, $\dot{\varphi}$ is the heading angle acceleration, and a and b are the distances between the vehicle centroid and the front and rear axles, respectively. F_{lf} and F_{lr} are the longitudinal forces of the front and rear tires, respectively. F_{cf} and F_{cr} are the lateral forces of the front and rear tires, respectively. \ddot{x} and \ddot{y} are the longitudinal and transverse accelerations in the vehicle coordinate system, respectively.

3 Design of the obstacle avoidance planning controller

In order to make the vehicle meet the requirements of obstacle avoidance in path tracking (Cheng et al., 2020), as shown in Fig. 3, a double-layer adaptive MPC obstacle avoidance

planning and path-tracking control strategy is designed. The upper controller is based on the vehicle point mass model. The controller avoids obstacles according to the position information of vehicles and obstacles. The under-layer controller is based on the vehicle dynamics model, and the target path is tracked by controlling the front wheel angle and the speed change.

Firstly, after the obstacle avoidance planning layer receives the obstacle information from the sensor module, the obstacle information in the form of discrete points is calculated into a curve by the five-times fitting polynomial method (Li et al., 2019). Secondly, the obstacle avoidance planning layer outputs the curve information to the path-tracking control layer. Also, the path-tracking control layer receives the curve information and constructs a traceable path. Finally, the front wheel angle is used as the control output to control the vehicle to track this executable path in the path-tracking control layer.

3.1 Establishment of the safety collision distance model

The intelligent vehicle environment perception module detects obstacles on the road in real time (these obstacles may suddenly appear), and the perception module sends obstacle information to the obstacle avoidance planning controller. The obstacle avoidance planning controller uses the safe obstacle avoidance model designed in this paper to re-plan a safe driving path to ensure vehicle driving safety. The obstacle is treated with expansion with the aim of preventing vehicles from crossing it. The extension method measures the radius of the outer circle of the obstacle and the vehicle motion center as the extension size and the safe distance, respectively. When the vehicle runs at a constant speed, the distance between the vehicle and the obstacle can be expressed as $L = vt$. If the vehicle speed is too fast or the distance between the vehicle and the obstacle is too close, the safety of the vehicle will be greatly reduced. The safety distance of the vehicle running at a constant speed is designed as

$$L = kv t + l_0, \tag{6}$$

where k is the safety factor, l_0 is the minimum distance between the vehicle and the obstacle, v is the vehicle speed, and t is the vehicle driving time.

In general, the speed of the vehicle changes during driving. In order to improve the vehicle safety, an anti-collision constraint condition needs to be added on the basis of obstacle expansion. The anti-collision constraint is established as follows.

$$(R + r)^2 < [X(t) - x_i]^2 + [Y(t) - y_i]^2, \tag{7}$$

where $X(t)$ and $Y(t)$ are, respectively, the lateral and longitudinal coordinates of the vehicle at time t . The center circumcircle radii of the vehicle and the obstacle are R and r , respectively. (x_i, y_i) is the center coordinate of the obstacle.

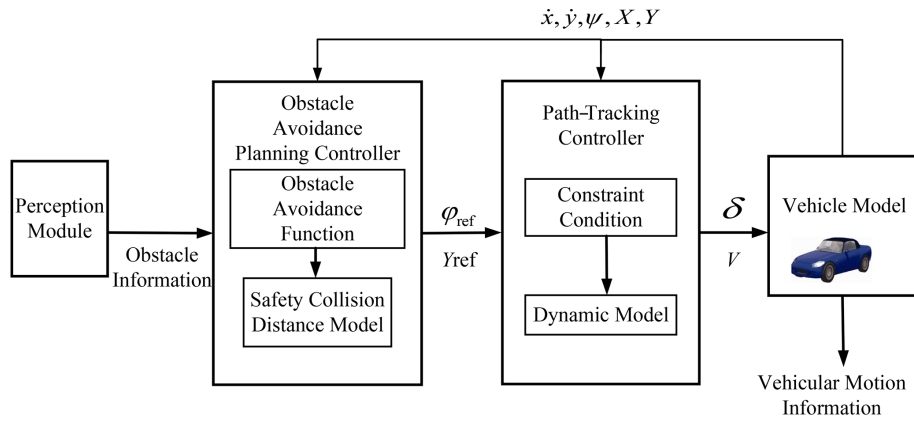


Figure 3. Double-layer adaptive MPC structure.

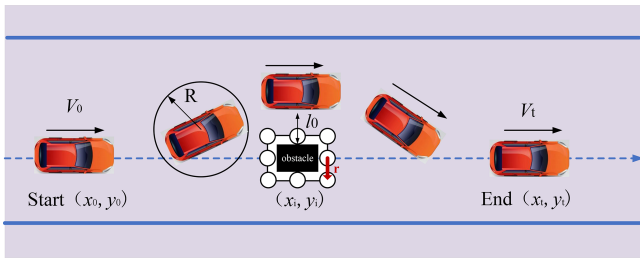


Figure 4. Vehicle safety collision distance model.

Figure 4 is a safe collision distance model, which takes into account both uniform and variable speeds. When the vehicle runs at a constant speed, the controller adopts a safety distance model (Jeong, 2021). When the vehicle speed changes, the controller applies the anti-collision constraint conditions to the vehicle.

(x_0, y_0) is the vehicle starting center point coordinate, (x_t, y_t) is the vehicle ending center point coordinate, and (x_i, y_i) is the center point coordinate of the obstacle.

3.2 Obstacle avoidance function

When there are obstacles in the reference path, the obstacle avoidance function is designed to describe the distance relationship between the vehicle and the obstacle; combined with the influence of vehicle speed, the expression of the obstacle avoidance function is given:

$$J_{obs,i} = \frac{S_{obs} (v_x^2 + v_y^2)}{(x_i - x_0)^2 + (y_i - y_0)^2 + \varepsilon}, \tag{8}$$

where S_{obs} is the weight coefficient, and ε is a constant (in order to prevent the denominator from being zero). It can be seen from Eq. (8) that the vehicle is closer to the obstacle, and the function value is larger. When the vehicle is farther away from the obstacle, the function value is smaller, which means that the vehicle is in a safe position.

3.3 Establishing the target function

According to the requirements of the path-tracking control, the distance between the actual driving path and the reference path of the vehicle should be reduced as much as possible. The re-planning path can be expressed by the following target function:

$$J(t) = \min_{U_i} \sum_{i=1}^{H_p} \left\| \eta(t+1|t) - \eta_{ref}(t+1|t) \right\|_Q^2 + \left\| U_i \right\|_R^2 + S J_{obs,i}, \tag{9}$$

where H_p is the predictive horizon, U_i is the control variable in the prediction horizon, Q is the output weight, R is the input weight, and η_{ref} is the reference path point.

3.4 Five-times polynomial fitting

According to the target function of the obstacle avoidance planning controller, the re-planning path is composed of discrete points. When these points are transmitted to the path-tracking controller, it is necessary to fit these discrete points with a five-times polynomial.

$$y_{ref} = \sum_{i=0}^5 p_i x^i = p_0 + p_1 x^1 + p_2 x^2 + p_3 x^3 + p_4 x^4 + p_5 x^5, \tag{10}$$

where y_{ref} and x are the lateral and longitudinal positions of the vehicle under the reference path, respectively. p is the polynomial coefficient.

4 Design of the path-tracking controller

The path-tracking control problem is essentially a vehicle steering-wheel angle control problem, also known as the vehicle lateral-control problem. Based on the reference path information input by the path-planning layer, according to

the current state of the vehicle, the optimal control quantity is calculated and output to the execution layer. Finally, through the accurate execution of the actuator, the vehicle path-tracking control is completed.

4.1 Linearized system

Combined with the vehicle dynamics model in Eqs. (3)–(5), the design of the path-tracking controller can be expressed as the following state space expressions.

$$\tilde{x} = f(x, u), y = \mathbf{C}x, \tag{11}$$

where x represents the state value, $x = [\dot{y} \dot{x} \phi \dot{\phi} Y X]$, u represents the front-wheel angle control value, y represents the output value, and \mathbf{C} is the output matrix.

In order to establish a linear time-varying model predictive control (LTV-MPC) system, Eq. (11) is linearized and discretized by the Taylor formula and the forward Euler method. The expressions are shown in Eqs. (12) and (13) (Xu et al., 2021).

$$x(k+1) = \mathbf{A}_k x(k) + \mathbf{B}_k u(k), \tag{12}$$

$$y(k) = \mathbf{C}_k x(k), \tag{13}$$

where \mathbf{A} is the state matrix, \mathbf{B} is the control matrix, and \mathbf{C} is the output matrix. $\mathbf{A}_k = \mathbf{I} + T\mathbf{A}_t$, $\mathbf{B}_k = T\mathbf{B}_t$, T is the sampling time, and $\mathbf{C}_k = [\mathbf{C} \ 0]$, \mathbf{I} is the unit matrix. The time step is k .

In order to reduce the static error, the control variable in Eqs. (12) and (13) is rewritten as the control increment.

$$\xi(k+1|t) = \tilde{\mathbf{A}}_k \xi(k|t) + \tilde{\mathbf{B}}_k \Delta u(k|t), \tag{14}$$

$$y(k|t) = \tilde{\mathbf{C}}_k \xi(k|t), \tag{15}$$

where $\tilde{\mathbf{A}}_k = \begin{pmatrix} \mathbf{A}_k & \mathbf{B}_k \\ 0_{1 \times 6} & \mathbf{I} \end{pmatrix}$, $0_{1 \times 6}$ is a 1×6 dimension 0 matrix, and $\tilde{\mathbf{B}}_k = \begin{pmatrix} \mathbf{B}_k \\ \mathbf{I} \end{pmatrix}$, $\tilde{\mathbf{C}}_k = (\mathbf{C}_k \ 0)$, $\Delta u(k|t) = u(k|t) - u(k-1|t)$, $\xi(k|t) = \begin{pmatrix} x(k|t) \\ u(k-1|t) \end{pmatrix}$.

Further, the system output in the prediction horizon can be calculated by Eq. (16) (Xu et al., 2021).

$$Y(k+1|t) = \psi_k \xi(k) + \Theta(k) \Delta U(k), \tag{16}$$

$$\text{where } \psi(k) = \begin{bmatrix} \tilde{\mathbf{C}}\tilde{\mathbf{A}} \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^2 \\ \vdots \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{H_p} \end{bmatrix}, \Delta U(k) = \begin{bmatrix} \Delta u(k|t) \\ \Delta u(k+1|t) \\ \vdots \\ \Delta u(k+H_p|t) \end{bmatrix},$$

$$\Theta(k) = \begin{pmatrix} \tilde{\mathbf{C}}\tilde{\mathbf{B}} & 0 & 0 & 0 \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{B}} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{N_c-1}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{N_c-2}\tilde{\mathbf{B}} & \dots & \tilde{\mathbf{C}}\tilde{\mathbf{B}} \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{N_c}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{N_c-1}\tilde{\mathbf{B}} & \dots & \tilde{\mathbf{C}}\tilde{\mathbf{A}}\tilde{\mathbf{B}} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{H_p-1}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{H_p-2}\tilde{\mathbf{B}} & \dots & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{H_p-N_c-1}\tilde{\mathbf{B}} \end{pmatrix}.$$

4.2 Establishing the target function

The goal of the path-tracking control layer is to ensure that the deviation between the actual driving path and the local obstacle avoidance path input by the planning layer is minimized (Liang et al., 2022; Yu et al., 2018), and the control increment is minimized to ensure the driving stability. The target function of the path-tracking controller is

$$J(k) = \sum_{i=1}^{H_p} \left\| y(k+i|t) - y_{\text{ref}}(k+i|t) \right\|_Q^2 + \sum_{i=1}^{N_c-1} \left\| \Delta u(k+1|t) \right\|_R^2 + \rho \varepsilon^2, \tag{17}$$

where y_{ref} is the output reference value, ρ is the weight coefficient, Q is the output weight, and R is the input weight.

The objective function will be solved by MATLAB software, and the objective function can be transformed into the standard quadratic solution problem.

$$J(\xi(k), u(k), \Delta U(k)) = \frac{1}{2} \Delta U^T H \Delta U + f^T \Delta U = [\Delta U(k)^T, \varepsilon]^T H_k [\Delta U(k), \varepsilon]^T + G_k [\Delta U(k)^T, \varepsilon], \tag{18}$$

where $H_k = \begin{bmatrix} \Theta_k^T Q \Theta_k + R & 0 \\ 0 & \rho \end{bmatrix}$, $G_k = [2\Theta_k^T Q \Theta_k \ 0]$, and \mathbf{T} is the transpose matrix.

4.3 Constraint condition

In order to ensure the accurate tracking of the desired path and the stability of the driving, the following three constraints should be added to the dynamics model (Bai et al., 2019).

1. In the three-degrees-of-freedom vehicle dynamics model established in Sect. 2.2, the tire sideslip angle is not calculated as a state variable (Jazar et al., 2020). Considering that the path-tracking controller needs to be calculated by quadratic programming, the constraint of the front wheel sideslip angle is set as $-2.5^\circ < \alpha_f < 2.5^\circ$.

- The vehicle adhesion condition directly affects the comfort of vehicle riding and the calculation results. When the vehicle adhesion condition is too large, the comfort of vehicle riding will be reduced. When the vehicle adhesion condition is too small, this will lead to the failure of the calculator solution. The constraint vehicle adhesion condition is set as

$$a_{\min} - \varepsilon \leq \sqrt{a_x^2 + a_y^2} \leq a_{\max} + \varepsilon, \tag{19}$$

where α_x and α_y are the transverse and longitudinal accelerations, respectively, ε is the relaxation factor, and α_{\min} and α_{\max} are the minimum and maximum accelerations, respectively.

- The vehicle centroidal sideslip angle can accurately reflect the driving stability (Kim et al., 2020). Considering the good road environment established in CarSim, the vehicle centroidal sideslip angle constraint is set to $-6^\circ < \beta < 6^\circ$.

5 Design of the double-layer adaptive MPC

It can be known from Eq. (17) that other controller parameters are constant, and the larger the prediction horizon (H_p), the more the controller can predict the farther position and obtain more vehicle state information. However, if the prediction horizon is too large, the tracking deviation in the near area will increase. When the prediction horizon is too small, the real-time performance of the controller will be improved; however, the predicted future vehicle state information is too little. Therefore, it is necessary to select the best prediction horizon to improve the vehicle path-tracking control accuracy.

5.1 Selecting the best prediction horizon

In order to ensure the effectiveness of the controller (Choi et al., 2021), the prediction horizon selection criteria are as follows. The set prediction horizon can ensure the controller activation and vehicle driving safety. Then, the real-time performance of the controller also needs an appropriate predictive horizon.

The CarSim and MATLAB/Simulink co-simulation is one of the mainstream simulation platforms in the current intelligent vehicle path-planning and path-tracking control test (Liang et al., 2020). CarSim and MATLAB/Simulink co-simulation results can be easily ported to real vehicles. At present, many scholars have applied the simulation results to the real vehicle. Firstly, the appropriate vehicle model is selected in CarSim, and the perception method used in this paper is the intelligent vehicle environment perception system of the CarSim software. The system includes vehicle sensors, cameras, and radar modules. The driving environment information obtained by the system is integrated to realize the understanding and recognition of the driving environment, and

Table 1. Vehicle model parameters.

Parameter name	Value
Complete vehicle quality (kg)	1769
Moment of inertia (kg m^{-2})	3962
Distance between centroid and front axis (m)	1.36
Distance centroid to rear axis (m)	1.58
Lateral stiffness of front and rear tires (N rad^{-1})	67 400
Longitudinal stiffness of front and rear tires (N rad^{-1})	62 900
Front and rear tire slip ratio	0.21
Computational cost (s)	0.02

the driving environment that matches the real scene is constructed. Secondly, the corresponding program of the MPC algorithm is written by MATLAB. The vehicle model selected in CarSim and the program written by MATLAB are sent to Simulink. Finally, the selected vehicle model in CarSim, the built driving environment, and the program written by MATLAB are called by Simulink (Zhai et al., 2022). Vehicle stability and path-tracking accuracy are analyzed by Simulink. CarSim has a lot of accurate vehicle models to be chosen and can build accurate driving environments, but the program required for the MPC algorithm cannot be written in CarSim. The main function of MATLAB is to write the code needed for simulation, and MATLAB has a special interface to connect with CarSim. By Simulink, an accurate intelligent driving simulation module can be created. A lot of accurate vehicle models in CarSim and the driving environment close to the real scene make the simulation results more accurate. At the same time, the rich interfaces between CarSim and MATLAB/Simulink ensure that they can accurately simulate the effectiveness of the algorithm designed in this paper. Through CarSim and MATLAB/Simulink co-simulation, the path-tracking effects of different H_p at three fixed speeds were tested, and the relationship between the H_p and the speed was analyzed. Some parameters of the vehicle model involved in this simulation are shown in Table 1. Many vehicle models are available in CarSim software, and the software has verified the availability of the models through extensive experiments. The experiment was completed on a computer platform with an Intel Core i5-8400 CPU at 280 GHz (main frequency of 2.8 GHz) and 8 GB memory. The vehicle model parameters used in this paper are provided by the public data set of the CarSim simulation software.

Simulation 1: the vehicle speed is set to 40 km h^{-1} . The prediction horizons are set to 6, 9, 12, and 15, respectively. Figure 5 shows the results of vehicle low-speed simulation.

Figure 5a shows that, when $H_p = 6$, path tracking fails, and when $H_p = 9$, path tracking is more accurate. Figure 5b and c are the results of the yaw angle and sideslip angle, respectively. The value of H_p is 9, and the error of the yaw angle and sideslip angle is the smallest. It can be seen from Fig. 5d that the lateral acceleration of the vehicle differs

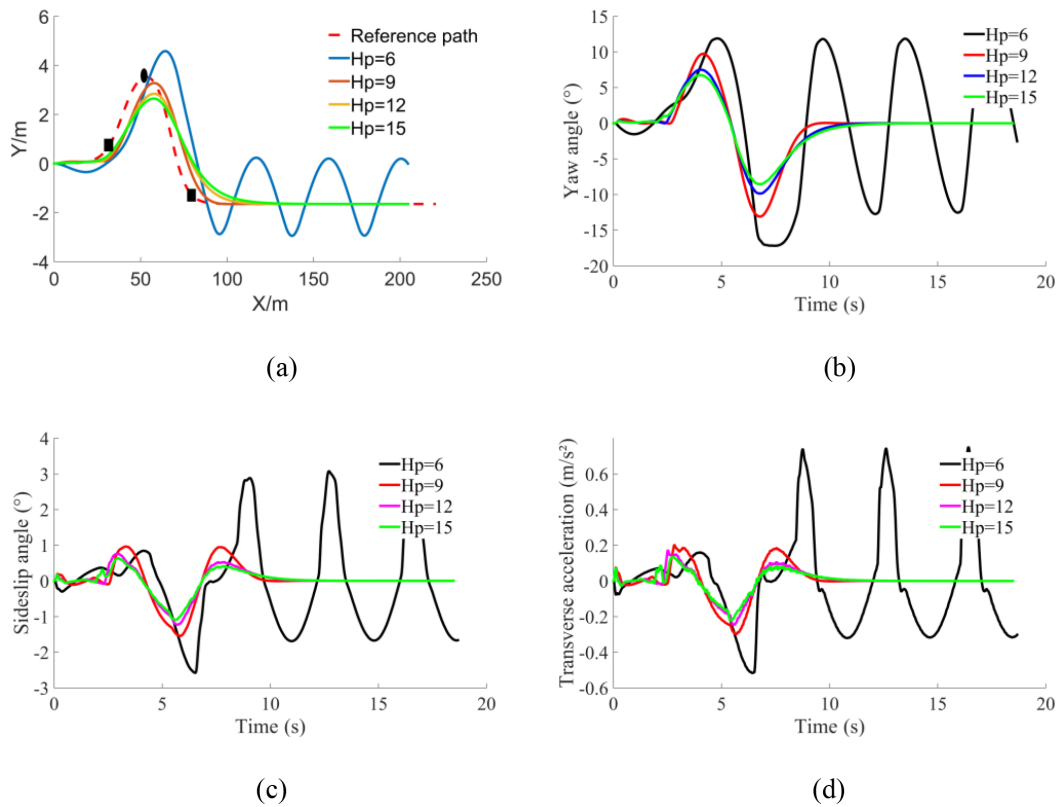


Figure 5. Vehicle low-speed simulation results. (a) Tracking path. (b) Yaw angle. (c) Sideslip angle. (d) Transverse acceleration.

slightly, indicating that the vehicle has good stability at low speed.

Simulation 2: the vehicle speed is set to 70 km h^{-1} . The prediction horizons are set to 9, 12, and 15, respectively. The simulation results of mid-speed vehicles are given in Fig. 6.

The path-tracking curve of Fig. 6a shows that, when $H_p = 12$, the path-tracking accuracy is the highest. Figure 6b and c show that, when $H_p = 12$, the yaw angle and the sideslip angle error are minimum, and when $H_p = 9$, the error is maximum. It can be seen from Fig. 6d that, when $H_p = 9, 12$, and 15 , the transverse acceleration of the vehicle is $0.61, 0.48$, and 0.45 m s^{-2} , respectively. According to the engineering requirements of the Pacejka tire model (Pacejka and Bakker, 2007), when the transverse acceleration range of the vehicle is $0\text{--}3.92 \text{ m s}^{-2}$, the controller has good stability. Thus, the stability of the controller is judged by the smoothness and jitter of the lateral acceleration change. When $H_p = 12$ and $H_p = 15$, the change in the vehicle transverse acceleration curve is not smooth with jitter. When $H_p = 9$, the transverse acceleration curve of the vehicle changes smoothly without obvious jitter, indicating that the controller has better stability when $H_p = 9$.

Simulation 3: the vehicle speed is set to 100 km h^{-1} . The prediction horizon is set to 12, 15, and 18, respectively. Figure 7 is the simulation curve of the vehicle at high speed.

Figure 7a shows that, when $H_p = 12$ or 15 , the path-tracking accuracy is not high, and when $H_p = 18$, the path-tracking accuracy is the highest. Figure 7b and c show that, when $H_p = 15$, the yaw angle and sideslip angle error is minimum. When the vehicle runs at high speed, the stability of the vehicle will be reduced.

5.2 Establishment of the speed-predictive horizon control law

Based on the analysis of Sect. 5.1, this paper performed a lot of simulations to explore the relationship between vehicle speed and prediction horizon. The relationship shows that there is an appropriate prediction horizon for the vehicle at a certain speed, which minimizes the path-tracking deviation of the vehicle. When other parameters remain unchanged, the relationship between speed and prediction horizon is approximately linear. According to the relationship, this paper uses a multiple fitting polynomial to calculate the optimal prediction horizon corresponding to different vehicle speeds.

When the vehicle detects the obstacles, the vehicle slows down, and the obstacle avoidance planning controller adjusts the prediction horizon in real time by the relationship between speed and prediction horizon (Wang et al., 2021; Chen et al., 2020). The expression of the speed-predictive horizon by two-times polynomial fitting for the adaptive obstacle

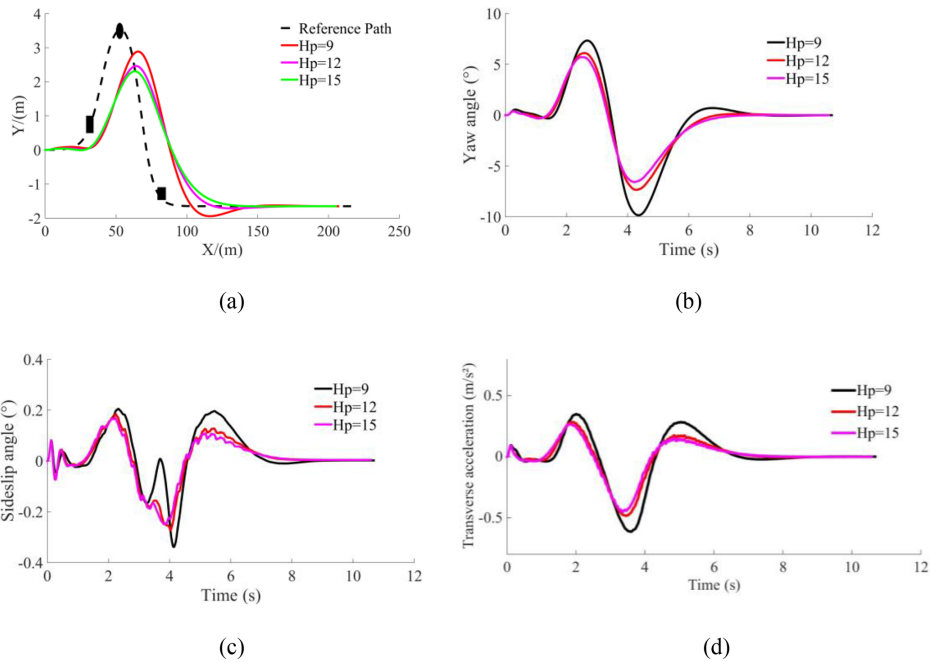


Figure 6. Vehicle mid-speed simulation results. (a) Tracking path. (b) Yaw angle. (c) Sideslip angle. (d) Transverse acceleration.

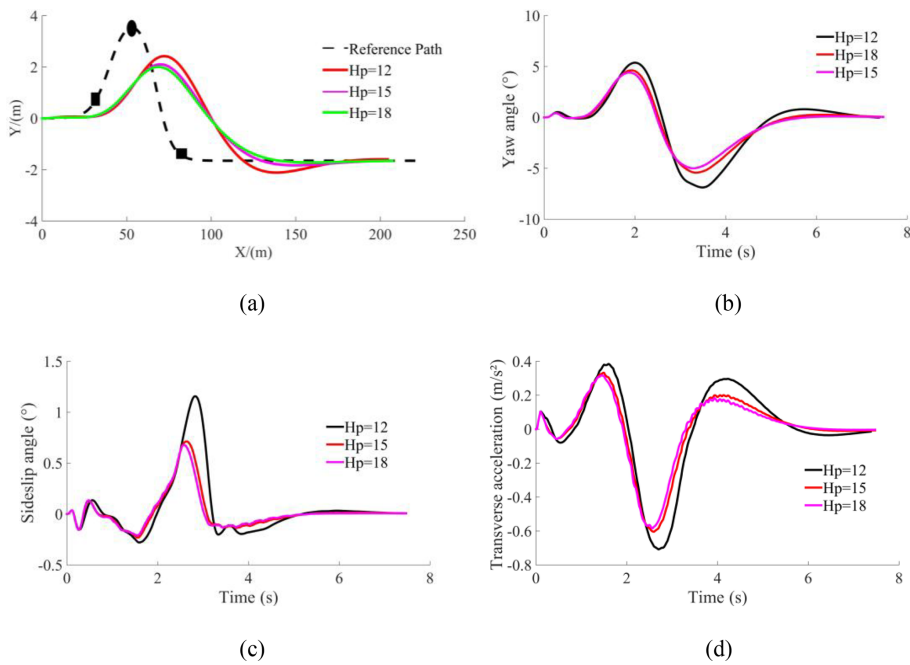


Figure 7. Vehicle high-speed simulation results. (a) Tracking path. (b) Yaw angle. (c) Sideslip angle. (d) Transverse acceleration.

avoidance planning controller is

$$H_{p1} = \begin{cases} 8, & v \leq 30 \\ \text{int}(0.0025v^2 - 0.0529v + 8), & 30 < v < 100 \\ 18, & v \geq 100 \end{cases}, \quad (20)$$

where v is the vehicle speed, int is the full character, and H_{p1} is the predictive horizon of the obstacle avoidance planning controller.

As shown in Fig. 8, the predictive horizon of the obstacle avoidance planning controller evolution diagram is given.

When the vehicle is driving on a road without obstacles, the vehicle will travel according to the reference path. At this

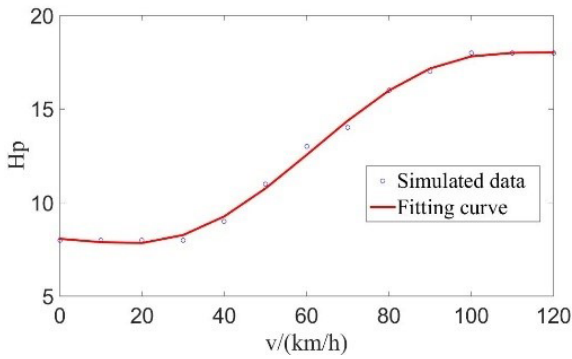


Figure 8. Predicted horizon evolution diagram.

time, the change in vehicle speed will also affect the prediction horizon in the path-tracking controller. Similarly, the expression of the speed–predictive horizon by three-times polynomial fitting for the adaptive path-tracking controller is

$$H_{p2} = \begin{cases} 8, & v \leq 30 \\ \text{int}(-0.000042866v^3 + 0.0116v^2 - 0.6944v + 20), & 30 < v < 100 \\ 20, & v \geq 100 \end{cases}, \quad (21)$$

where H_{p2} is the predictive horizon in the path-tracking controller.

5.3 Convergence analysis of the algorithm

The convergence analysis of the algorithm can be illustrated by Fig. 9 (Taghavifar and Rakheja, 2019). In the whole control process, suppose the current time is k , and curve 1 is the reference path. Curve 2 represents the predicted output path of the controller, and to meet the requirements of path-tracking control, curve 2 should coincide with curve 1 as much as possible. It can be seen from curve 2 that the predicted output path is gradually close to the reference path.

6 Simulation analysis

The Sect. 5.1 analysis shows that the traditional model predictive controller with constraint conditions can safely and stably track the reference path and avoid obstacles on the road under different vehicle speeds (low speed 40 km h^{-1} , medium speed 70 km h^{-1} , and high speed 100 km h^{-1}). The vehicle tracking accuracy and controller stability are analyzed under different vehicle speeds and different H_p values. On the basis of adding constraints, combined with the speed–predictive horizon control law, a double-layer adaptive MPC (AMPC) path-tracking controller is constructed. The control effect of the path-tracking controller under variable speed conditions is analyzed. The double-layer AMPC is compared

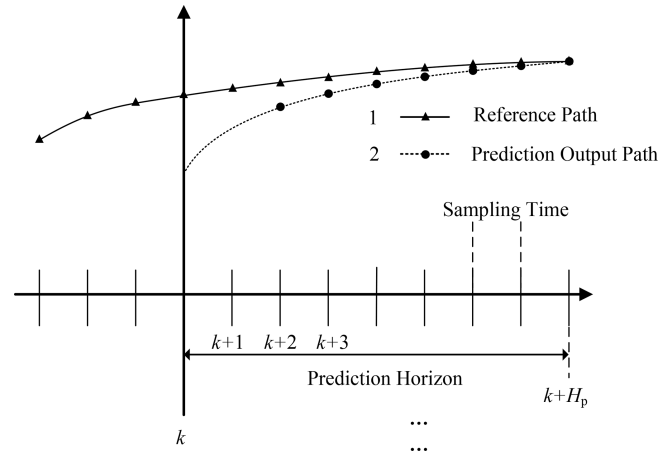


Figure 9. Convergence analysis of the algorithm.

with the traditional MPC and the single-layer AMPC (speed–predictive horizon control law only established in the obstacle avoidance planning layer). The variable speed condition is built in CarSim. In order to improve the calculation speed and control precision of the controller, the simulation is carried out in a single obstacle environment. The simulation results are shown in Fig. 10.

The accuracy of path tracking can be expressed by the lateral deviation between the actual trajectory of the vehicle and the planned one. The smaller the lateral deviation, the higher the path-tracking accuracy. Figure 10a can reflect the vehicle path-tracking accuracy. It can be seen that the maximum lateral deviation of the double-layer adaptive MPC controller is 0.12 m, the maximum lateral deviation of the single-layer adaptive MPC controller is 0.37 m, and the maximum lateral deviation of the traditional MPC controller is 0.63 m. Compared with the algorithm proposed by Dai et al. (2020), the maximum lateral error of the proposed algorithm is reduced by 0.09 m. The deviation of double-layer adaptive MPC controller is minimal, and thus the double-layer adaptive MPC has the highest tracking accuracy. In terms of vehicle driving stability, Fig. 10b and c show that the error of the yaw angle of the double-layer adaptive MPC vehicle is the smallest. From Fig. 10c, it can be seen that the fluctuation range of the yaw angular rate of the control algorithm proposed in this paper is $21.20 \text{ degree s}^{-1}$, the fluctuation range of the yaw angular rate of the single-layer adaptive MPC controller is $21.14 \text{ degree s}^{-1}$, and the fluctuation range of the yaw angular rate of the traditional MPC controller is $31.02 \text{ degree s}^{-1}$. The results show that the fluctuation range of the yaw angular rate of the control algorithm proposed in this paper is small and has no obvious jitter, indicating that the control algorithm proposed in this paper has good stability. In general, inclination angle and transverse acceleration are the indexes used to evaluate vehicle comfort (Xu et al., 2021). Simulation results in Fig. 10d and e show that the maximum inclination angle of the traditional MPC vehicle is 1.53° , the maximum

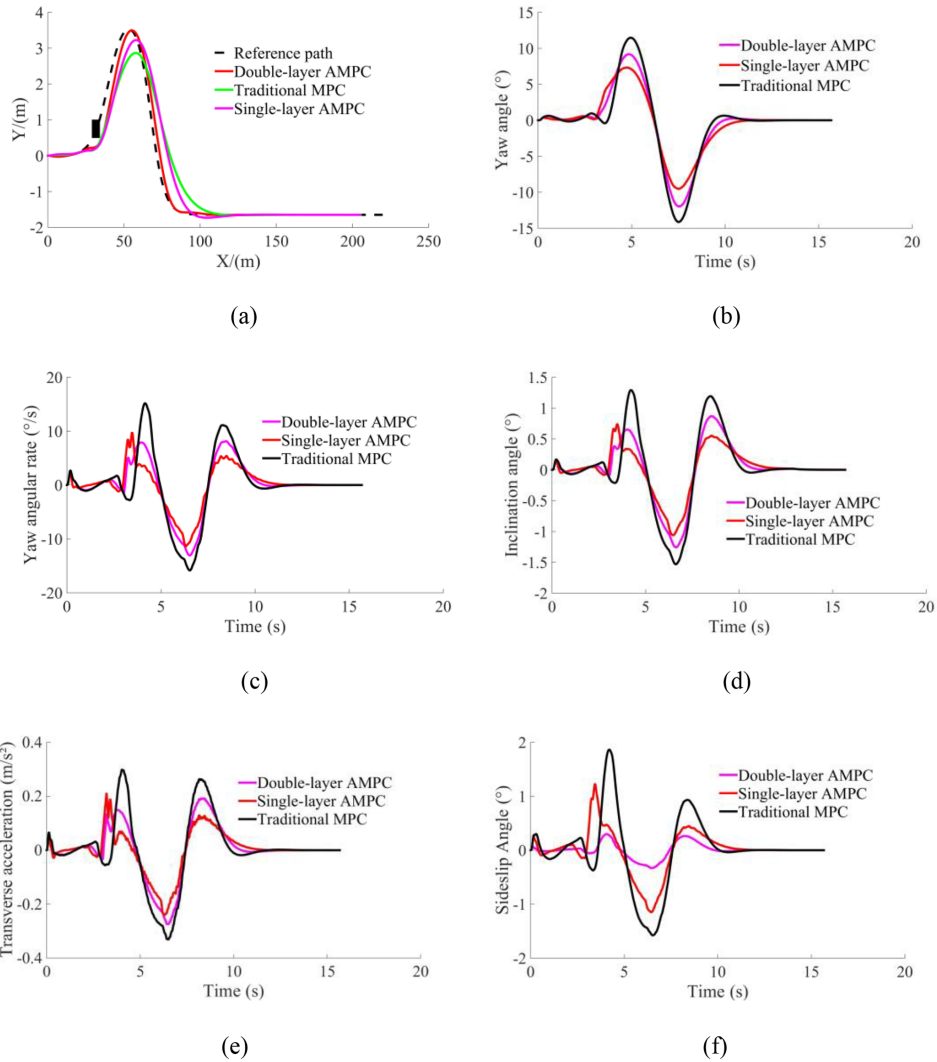


Figure 10. Comparison of simulation results. (a) Tracking path. (b) Yaw angle. (c) Yaw angular rate. (d) Inclination angle. (e) Transverse acceleration. (f) Sideslip angle.

inclination angle of the double-layer adaptive MPC vehicle is 1.25°, the maximum transverse acceleration of the traditional MPC is 0.34 m s⁻², and the maximum transverse acceleration of the double-layer adaptive MPC is 0.27 m s⁻². It can be seen from Fig. 10f that the fluctuation range of the sideslip angle of the control algorithm proposed in this paper is 0.30°, the fluctuation range of the sideslip angle of the single-layer adaptive MPC controller is 1.23°, and the fluctuation range of the sideslip angle of the traditional MPC controller is 1.87°. Compared with the algorithm proposed by Xie et al. (2021), the improved algorithm in this paper shows a smaller fluctuation range of the vehicle sideslip angle, indicating that the driving stability of intelligent vehicles is better. The results show that the fluctuation range of the sideslip angle of the control algorithm proposed in this paper is the smallest, indicating that the control algorithm proposed in this paper has good stability. Combined with the analysis

of the simulation results, the proposed double-layer adaptive MPC control strategy not only has the highest path-tracking control accuracy, but also has good safety and stability.

7 Conclusion

In this paper, a double-layer adaptive model predictive controller is designed. In the obstacle avoidance planning controller, the vehicle point mass model is used to expand the obstacle, and the safe collision distance model is established according to the relationship between obstacle and vehicle position. In the path-tracking controller, the vehicle dynamics model is adopted. Based on the model predictive control algorithm, the vehicle sideslip angle and acceleration constraints are added to transform the path-tracking control into a quadratic programming problem with constraints. The Car-Sim and MATLAB/Simulink simulation results show that the

double-layer adaptive model predictive controller designed in this paper can effectively improve the path-tracking accuracy, safety, and stability of intelligent vehicles.

Data availability. The data are available upon request from the corresponding author.

Author contributions. CH surveyed the literature and wrote the paper. BM constructed the overall framework.

Competing interests. The contact author has declared that neither of the authors has any competing interests.

Disclaimer. Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Financial support. This work is supported by the Open Research Fund of Anhui Key Laboratory of Detection Technology and Energy Saving Devices, Anhui Polytechnic University (grant no. DTESD2020A06), the National Natural Science Foundation of China (grant no. 62203012), the 2021 Graduate Science Research Project of the Department of Education of Anhui Province (grant no. YJS20210447), and the Wuhu City Science and Technology Plan Project (grant no. 2021cg21).

Review statement. This paper was edited by Daniel Condurache and reviewed by Jingang Jiang, Minghui Ou, and three anonymous referees.

References

- Bai, G. X., Meng, Y., Liu, L., Luo, W. D., Gu, Q., and Li, K. L.: A New Path Tracking Method Based on Multilayer Model Predictive Control, *Appl. Sci.*, 9, 2649, <https://doi.org/10.3390/app9132649>, 2019
- Bobier-Tiu, C. G., Beal, C. E., Kegelman, J. C., Hindiyeh, R. Y., and Gerdes, J. C.: Vehicle control synthesis using phase portraits of planar dynamics, *Vehicle Syst. Dyn.*, 57, 1318–1337, <https://doi.org/10.1080/00423114.2018.1502456>, 2019.
- Chen, L., Qin, D. F., Xu, X., Cai, Y. F., and Xie, J.: A path and velocity planning method for lane changing collision avoidance of intelligent vehicle based on cubic 3-D Bezier curve, *Adv. Eng. Softw.*, 132, 65–73, <https://doi.org/10.1016/j.advengsoft.2019.03.007>, 2019.
- Chen, Y., Chen, S. Z., Ren, H. B., Gao, Z. P., and Liu, Z.: Path Tracking and Handling Stability Control Strategy With Collision Avoidance for the Autonomous Vehicle Under Extreme Conditions, *IEEE T. Veh. Technol.*, 69, 14602–14617, <https://doi.org/10.1109/TVT.2020.3031661>, 2020.
- Cheng, S., Li, L., Guo, H. Q., Chen, Z. G., and Song, P.: Longitudinal Collision Avoidance and Lateral Stability Adaptive Control System Based on MPC of Autonomous Vehicles, *IEEE T. Intell. Transp.*, 21, 2376–2385, <https://doi.org/10.1109/TITS.2019.2918176>, 2020.
- Choi, Y., Lee, W., Kim, J., and Yoo, J.: A Variable-Sampling Time Model Predictive Control Algorithm for Improving Path-Tracking Performance of a Vehicle, *Sensors*, 21, 6845, <https://doi.org/10.3390/s21206845>, 2021.
- Dai, C. H., Zong, C. F., and Chen, G. Y.: Path Tracking Control Based on Model Predictive Control With Adaptive Preview Characteristics and Speed-Assisted Constraint, *IEEE Access*, 8, 184697–184709, <https://doi.org/10.1109/ACCESS.2020.3029635>, 2020.
- Gutjahr, B., Gröll, L., and Werling, M.: Lateral Vehicle Trajectory Optimization Using Constrained Linear Time-Varying MPC, *IEEE T. Intell. Transp.*, 18, 1586–1595, <https://doi.org/10.1109/TITS.2016.2614705>, 2016.
- Jazar, R. N., Alam, F., Milani, S., Marzbani, H., and Chowdhury, H.: Mathematical Modelling of Vehicle Drifting, *MIST International Journal of Science and Technology (MIJST)*, 8, 25–29, [https://doi.org/10.47981/j.mijst.08\(02\)2020.187\(25-29\)](https://doi.org/10.47981/j.mijst.08(02)2020.187(25-29)), 2020.
- Jeong, Y.: Self-Adaptive Motion Prediction-Based Proactive Motion Planning for Autonomous Driving in Urban Environments, *IEEE Access*, 9, 105612–105626, <https://doi.org/10.1109/ACCESS.2021.3100590>, 2021.
- Karaman, S., Walter, M. R., Perez, A., Frazzoli, E., and Teller, S.: Anytime Motion Planning using the RRT*, in: 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011, IEEE, 1478–1483, <https://doi.org/10.1109/ICRA.2011.5980479>, 2011.
- Kim, D., Min, K., Kim, H., and Huh, K.: Vehicle sideslip angle estimation using deep ensemble-based adaptive Kalman filter, *Mech. Syst. Signal Pr.*, 144, 106862, <https://doi.org/10.1016/j.ymssp.2020.106862>, 2020.
- Kucuk, S.: Optimal trajectory generation algorithm for serial and parallel manipulators, *Robot. Cim.-Int. Manuf.*, 48, 219–232, <https://doi.org/10.1016/j.rcim.2017.04.006>, 2017.
- Li, B. Y., Du, H. P., Li, W. H., and Zhang, B. J.: Integrated trajectory planning and control for obstacle avoidance manoeuvre using non-linear vehicle MP algorithm, *IET Intell. Transp. Sy.*, 13, 385–397, <https://doi.org/10.1049/iet-its.2018.5002>, 2018.
- Li, S. and Feng, X.: Study of structural optimization design on a certain vehicle body-in-white based on static performance and modal analysis, *Mech. Syst. Signal Pr.*, 135, 106405, <https://doi.org/10.1016/j.ymssp.2019.106405>, 2020.
- Li, S. S., Li, Z., Yu, Z. X., Zhang, B. C., and Zhang, N.: Dynamic Trajectory Planning and Tracking for Autonomous Vehicle With Obstacle Avoidance Based on Model Predictive Control, *IEEE Access*, 7, 132074–132086, <https://doi.org/10.1109/ACCESS.2019.2940758>, 2019.
- Liang, Y., Li, Y. N., Khajepour, A., and Zheng, L.: Holistic Adaptive Multi-Model Predictive Control for the Path Following of 4WID Autonomous Vehicles, *IEEE T. Veh. Technol.*, 70, 69–81, <https://doi.org/10.1109/TVT.2020.3046052>, 2020.
- Liang, Y. X., Li, Y. N., Khajepour, A., Huang, Y. J., Qin, Y. C., and Zheng, L.: A Novel Combined Decision and Control Scheme for Autonomous Vehicle in Structured Road Based on Adaptive

- Model Predictive Control, *IEEE T. Intell. Transp.*, 23, 16083–16097, <https://doi.org/10.1109/TITS.2022.3147972>, 2022.
- Liu, Q., Liu, Y., Liu, C., Chen, B., Zhang, W., Li, L., and Ji, X.: Hierarchical lateral control scheme for autonomous vehicle with uneven time delays induced by vision sensors, *Sensors*, 18, 2544, <https://doi.org/10.3390/s18082544>, 2018.
- Mata, S., Zubizarreta, A., Cabanes, I., Nieva, I., and Pinto, C.: Linear time varying model based model predictive control for lateral path tracking, *Int. J. Vehicle Des.*, 75, 1–22, <https://doi.org/10.1504/IJVD.2018.10011992>, 2018.
- Nan, J. F., Shang, B. X., Deng, W. W., Ren, B. T., and Liu, Y.: MPC-based Path Tracking Control with Forward Compensation for Autonomous Driving, *IFAC-PapersOnLine*, 54, 443–448, <https://doi.org/10.1016/j.ifacol.2021.10.202>, 2021.
- Pacejka, H. B. and Bakker, E.: The Magic Formula Tyre Model, *Vehicle Syst. Dyn.*, 21, 1–18, <https://doi.org/10.1080/00423119208969994>, 2007.
- Taghavifar, H. and Rakheja, S.: Path-tracking of autonomous vehicles using a novel adaptive robust exponential-like-sliding-mode fuzzy type-2 neural network controller, *J. Mech. Syst. Signal Pr.*, 130, 41–55, <https://doi.org/10.1016/j.ymsp.2019.04.060>, 2019.
- Tomas-Gabarron, J., Egea-Lopez, E., and Garcia-Haro, J.: Vehicular Trajectory Optimization for Cooperative Collision Avoidance at High Speeds, *IEEE T. Intell. Transp.*, 14, 1930–1941, <https://doi.org/10.1109/TITS.2013.2270009>, 2013.
- Wang, H. R., Wang, Q. D., Chen, W. W., Zhao, L. F., and Tan, D. K.: Path tracking based on model predictive control with variable predictive horizon, *T. I. Meas. Control.*, 43, 2676–2688, <https://doi.org/10.1177/01423312211003809>, 2021.
- Woo, S., Cha, H., Yi, K., and Jang, S.: Active Differential Control for Improved Handling Performance of Front-Wheel-Drive High-Performance Vehicles, *Int. J. Automot. Techn.*, 22, 537–546, <https://doi.org/10.1007/s12239-021-0050-2>, 2021.
- Xie, Z., Wu, Y., Gao, J., Song, C., Chai, W., and Xi, J.: Emergency obstacle avoidance system of driverless vehicle based on model predictive control, *C. 2021 International Conference on Advanced Mechatronic Systems (ICAMechS)*, Tokyo, Japan, 9–12 December 2021, IEEE, <https://doi.org/10.1109/ICAMechS54019.2021.9661515>, 2021.
- Xu, T. and Wang, X.: Roll stability and path tracking control strategy considering driver in the loop, *IEEE Access*, 9, 46210–46222, <https://doi.org/10.1109/ACCESS.2021.3067649>, 2021.
- Xu, Y. C., Zheng, H. R., Wu, W. M., and Wu, J.: Robust Hierarchical Model Predictive Control for Trajectory Tracking with Obstacle Avoidance, *IFAC-PapersOnLine*, 53, 15745–15750, <https://doi.org/10.1016/j.ifacol.2020.12.056>, 2020.
- Xu, Y., Tang, W. T., Chen, B. Y., Qiu, L., and Yang, R.: A Model Predictive Control with Preview-Follower Theory Algorithm for Trajectory Tracking Control in Autonomous Vehicles, *Symmetry*, 13, 1–16, <https://doi.org/10.3390/sym13030381>, 2021.
- Yu, Z., Zhang, R., Xiong, L., and Fu, Z.: Robust hierarchical controller with conditional integrator based on small gain theorem for reference trajectory tracking of autonomous vehicles, *Vehicle Syst. Dyn.*, 57, 1143–1162, <https://doi.org/10.1080/00423114.2018.1555333>, 2018.
- Zhai, L., Wang, C. P., Hou, Y. H., and Liu, C.: MPC-Based Integrated Control of Trajectory Tracking and Handling Stability for Intelligent Driving Vehicle Driven by Four Hub Motor, *IEEE T. Veh. Technol.*, 3, 2668–2680, <https://doi.org/10.1109/TVT.2022.3140240>, 2022.
- Zuo, Z. Q., Yang, X., Li, Z., Wang, Y. J., Han, Q. N., Wang, L., and Luo, X. Y.: MPC-Based Cooperative Control Strategy of Path Planning and Trajectory Tracking for Intelligent Vehicles, *IEEE Transactions on Intelligent Vehicles*, 6, 513–522, <https://doi.org/10.1109/TIV.2020.3045837>, 2021.