

Article

Lane-Changing Trajectory Tracking and Simulation of Autonomous Vehicles Based on Model Predictive Control

Hui Song , Dayi Qu ^{*} , Haibing Guo, Kekun Zhang and Tao Wang 

School of Mechanical and Automotive Engineering, Qingdao University of Technology, Qingdao 266520, China
* Correspondence: dayiqu@qtech.edu.cn

Abstract: In order to realize the lane-changing maneuver of Connected Autonomous Vehicles (CAV), a lateral controller based on model predictive control is developed with the three degrees of freedom vehicle dynamic model. Then the controller is synthesized to track the reference trajectory fitted by the quintic Bézier curve. The controller is validated by MATLAB/CarSim under different road adhesion conditions and driving speeds. Results show that for different road adhesion conditions and driving speeds, the controller does not need to adjust the control parameters and can continuously correct the deviation from the expected trajectory. During the tracking process, the front wheel angle, front wheel angle increment, centroid side deflection angle, and front wheel side deflection angle are kept within the limited constraint range. The established control algorithm has good control robustness and tracking driving stability. The research can provide a theoretical basis and technical support for lane-changing safety and control of CAV.

Keywords: automotive engineering; lane-changing trajectory tracking; model predictive control; autonomous driving vehicle; dynamic model



Citation: Song, H.; Qu, D.; Guo, H.; Zhang, K.; Wang, T. Lane-Changing Trajectory Tracking and Simulation of Autonomous Vehicles Based on Model Predictive Control. *Sustainability* **2022**, *14*, 13272. <https://doi.org/10.3390/su142013272>

Academic Editor: Michael S. Carolan

Received: 26 August 2022

Accepted: 6 October 2022

Published: 15 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Under the environment of the internet of vehicles, autonomous driving vehicles need four phases to complete autonomous lane-changing behaviors, namely, environment perception, decision making, path planning, and trajectory control. Lane changing is one of the high-frequency behaviors of vehicles to meet driving demands. On the other hand, inappropriate lane-changing behaviors can easily induce traffic accidents. Therefore, the safety of lane-changing behaviors of autonomous driving vehicles is a key issue in the commercialization of autonomous driving vehicles. It has become one of the hot issues studied by Chinese and foreign scholars. The main recent research results are below. In order to study the behavior of vehicles avoiding collisions with obstacles in front of the lane, Petrov et al. [1] designed a double-layer adaptive steering controller based on the reference trajectory of the lane-changing cycloid. Bayar [2] designed a PID tracking controller to study the problem of vehicle motion-tracking control, but the tracking accuracy was not high. Cao et al. [3] proposed the model predictive control algorithm that optimizes the confluence path and confluence point of the highway confluence area and simulates the driving stability of the confluence vehicle and the main line coordinated vehicle. Yang et al. [4] designed a lane-changing controller based on a model predictive control algorithm for lane-changing tracking control using the autonomous driving vehicle's coordinated lane-changing polynomial function as the reference trajectory. Cai [5] designed a Backstepping controller based on the sliding mode control algorithm to track and control the lane-changing trajectory of the vehicle. You et al. [6] established a rectangular vehicle model to study the safety control of the lane changing of autonomous driving vehicles, analyzed the constraint conditions of the safe distance between the lane-changing vehicle and the vehicle in front of the original lane, and created the lane-changing trajectory planning model based on a polynomial function. Ji et al. [7] built a multi-constraint model predictive

control system based on the two degrees of freedom dynamic model to track the lane-change trajectory. Huang et al. [8] designed a quadratic feedback controller based on the three degrees of freedom dynamic model and analyzed the lane-change trajectory-tracking problems of different styles of drivers by treating the front wheel angle as the feedforward control quantity. Wang [9] used the sliding mode variable structure control theory of the underactuated system to design the position control rate and attitude control rate for lane-changing trajectory-tracking control. Shao et al. [10] designed a vehicle lateral fuzzy controller based on the two degrees of freedom dynamic model and fuzzy control theory, which can make the vehicle accurately track the centerline of the lane. Leng [11] used model predictive control theory to design an explicit tracking controller, which takes into account tracking accuracy and driving stability. Wang et al. [12] proposed the mpc control algorithm based on weight coefficients to track the sixth-order polynomial trajectory curve, but in engineering applications, the timeliness of the hardware system is required. Bae et al. [13] determined the desired trajectory by evaluating the lateral acceleration value of the Bezier curve in the local path-planning stage to achieve comfortable and stable movement of the vehicle. However, it does not consider the driving stability of different road adhesion.

In the above research, the PID controller is a classic control algorithm. It is adjusted and controlled by considering the deviation between input and output. The advantage is that it is simple to use and convenient to adjust. The disadvantage is that the adjustment accuracy of the nonlinear system is not high. SMC and MPC are the control methods for nonlinear systems. Among them, SMC is a method to find the control law based on the state trajectory described by the phase plane, which has the advantage of robustness, but bucket vibration will occur when it is stable in the sliding film area. MPC is a model-based optimal control method. Compared with PID, which only considers input and output, MPC is a multivariable control method, considering the influence of variables such as space state. Therefore, MPC control has the advantages of good dynamic optimization performance and robustness of the system.

Because of the interaction of multiple lanes and multiple modes in the actual traffic environment, the driver is required to adjust the vehicle speed and acceleration in real time to adapt to the conditions of a complex traffic environment, as shown in Figure 1. A represents the model control vehicle, and B represents the vehicle in front of the current lane. G represents the vehicle in front of the right lane, F represents the vehicle behind the right lane, C represents the vehicle in front of the left lane, and D represents the vehicle behind the left lane.

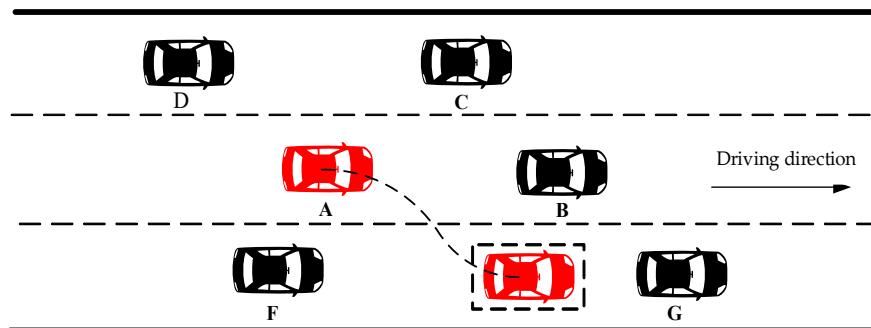


Figure 1. The lane-changing scene of the vehicle.

The novelty and a key contribution of this paper is the design of an MPC controller with three degrees of freedom, establishing the trajectory-tracking control objective function by considering the quintic Bézier curve and converting the objective function into a quadratic planning problem so as to achieve the optimal control increment in the next cycle. In addition, the output of the model has a great relationship with the selection of parameters. Through constant parameter adjustment in the simulation, the output of the model is more consistent with the actual operation trajectory. Finally, trajectory-tracking stability and driving safety are improved.

2. Nonlinear Dynamic Model

In order to reduce the computational complexity of model predictive control and establish a simplified three degrees of freedom nonlinear dynamic model, the following assumptions are made:

- (1) Autonomous driving vehicles only make longitudinal, lateral, and yaw movements on flat roads;
- (2) Ignore the coupling relationship between the longitudinal and lateral directions of the tire and only consider the pure cornering characteristics of the tire;
- (3) Ignore the influence of aerodynamics on vehicle dynamics;
- (4) The vehicle and suspension system are rigid bodies with the same rotation angle as the two front wheels;
- (5) Autonomous driving vehicles only make pure rolling during the steering process, ignoring tire load deviation;
- (6) During lane changes, the vehicle yaw angle and front wheel steering angle are both small angles, and the tire lateral force has an approximately linear relationship with the side deflection angle and the longitudinal force and slip rate.

According to the hypothesis, the dynamic model of the “bicycle” is established as shown in Figure 2:

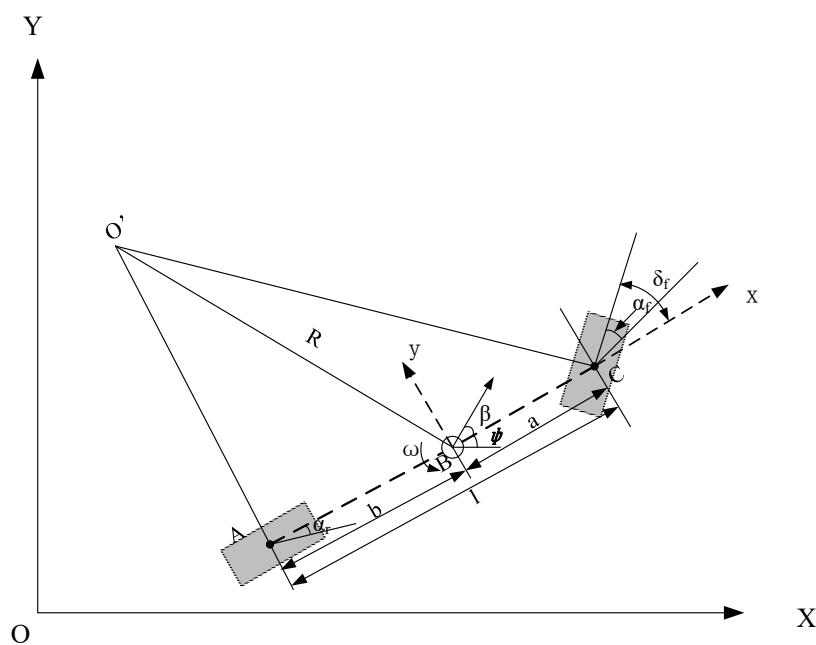


Figure 2. Vehicle dynamics model.

The vehicle nonlinear dynamics model based on the above assumptions is:

$$\begin{cases} \dot{v}_x = \frac{2}{m} [C_{lf}s_f + C_{lr}s_r + C_{cf}(\delta_f - \frac{v_y + a\omega}{v_x})\delta_f] + v_y\omega \\ \dot{v}_y = \frac{2}{m} [C_{cf}(\delta_f - \frac{v_y + a\omega}{v_x}) + C_{cr}\frac{b\omega - v_y}{v_x}] - v_x\omega \\ \dot{\phi} = \omega \\ \dot{\omega} = \frac{2}{I_z} [a \cdot C_{cf}(\delta_f - \frac{v_y + a\omega}{v_x}) - b \cdot C_{cr} \cdot \frac{b\omega - v_y}{v_x}] \\ \dot{X} = v_x \cdot \cos(\phi) - v_y \cdot \sin(\phi) \\ \dot{Y} = v_y \cdot \sin(\phi) + v_x \cdot \cos(\phi) \end{cases} \quad (1)$$

In Figure 2, XOY represents the geodetic coordinate system, and xoy represents the vehicle coordinate system. The symbols and some index values in the model formula are shown in Table 1.

Table 1. Parameter values of vehicle dynamics model.

Symbol	Parameter	Symbol	Parameter	Index Value
v_x	Longitudinal speed	C_{cf}	Front-wheel side deflection stiffness/[N·(°) ⁻¹]	66,900
v_y	Lateral speed	C_{cr}	Rear-wheel side deflection stiffness/[N·(°) ⁻¹]	66,700
ϕ	Vehicle yaw angle	C_{lf}	Longitudinal stiffness of front wheel/N	66,900
ω	Vehicle yaw rate	C_{lr}	Longitudinal stiffness of rear wheel/N	66,700
X	Longitudinal position	S_f	Front-wheel slip rate	0.2
Y	Lateral position	S_r	Rear-wheel slip rate	0.2
δ_f	front wheel angle	m	Vehicle quality/kg	1723
		a	Distance from the centroid of the vehicle to the front axle/m	1.232
		b	Distance from the centroid of the vehicle to the rear axle/m	1.468
		I_z	Yaw moment of inertia/(kg·m ²)	4175

3. Dynamic Constraints

When the model predictive control algorithm is used for lateral control of the vehicle lane changing based on the dynamic model, the vehicle's established three degrees of freedom dynamic model assumes that the vehicle yaw angle and the front wheel angle are small (<5°) during the lane-changing process. Under this assumption, according to the magic tire formula [9], it can be seen that the tire lateral force and the side deflection angle, the longitudinal force, and the slip rate are approximately linear. Therefore, the designed model predictive controller needs to set dynamic constraint conditions, mainly including the conditions addressed below.

3.1. The Constraints of Tire Side Deflection Angle

According to the tire side deflection characteristics, when the tire side deflection angle is about 5°, the side deflection force and the side deflection angle are approximately linear. According to the literature [14–18], the constraint range of the front wheel side deflection angle is set as

$$-2.5^\circ < \alpha_f < 2.5^\circ \quad (2)$$

3.2. The Constraints of Centroid Side Deflection Angle

According to the vehicle stability research results of Bosch Corporation [19], in order to maintain the stability of the vehicle during driving, the centroid side deflection angle must be limited within a certain constraint range. The centroid side deflection angle of the vehicle maintains a stable driving range on dry asphalt pavement between –12 and 12°. The constraint range for vehicles on icy and slippery roads to maintain stability is between –2 and 2°. In this paper, the constraint conditions for the centroid side deflection angle are set as follows:

$$\beta = \begin{cases} (-12^\circ, 12^\circ) & \mu = 1 \\ (-2^\circ, 2^\circ) & \mu = 0.2 \end{cases} \quad (3)$$

3.3. The Constraints of Vehicle Ground Adhesion Conditions

As the vehicle dynamics are restricted by the adhesion conditions of the tires and the ground, it is necessary to increase the constraints of vehicle ground adhesion conditions. The vehicle motion acceleration has the following relationship with the ground adhesion:

$$ma \leq F_f = \mu mg \quad (4)$$

The vehicle acceleration can be expressed as $a = \sqrt{a_x^2 + a_y^2}$, where a_x and a_y are the longitudinal and lateral acceleration of the vehicle, respectively, so the above formula can be simplified as

$$\sqrt{a_x^2 + a_y^2} \leq \mu g \quad (5)$$

In the process of lane changing, excessive lateral and longitudinal acceleration affects comfort, and the range of constraint is too small to calculate a feasible solution. Therefore, the vehicle ground adhesion conditions are softened through a relaxation factor:

$$a_{\min} - \varepsilon < a < a_{\max} + \varepsilon \quad (6)$$

4. The Design of the Lateral Controller

We constructed the mpc lateral controller on the basis of the above vehicle dynamics model and the Bessel lane-change trajectory and further defined the spatial state of the control system $\dot{X} = f(X, u)$, system state vector $X = [\dot{x}, \dot{y}, \phi, \dot{\phi}, X, Y]^T$, system input control vector $u = \delta_f$, and system output $\eta = [\phi, Y]^T$.

4.1. Model Linearization Processing

Taylor expands the system state space expression at the reference point $[X_0, u_0]$ and takes the first-order term to obtain

$$\begin{aligned} \dot{\tilde{X}} &= f(X, u) - f(X_0, u_0) = \frac{\partial f(X, u)}{\partial X} \Bigg|_{\substack{X=X_0 \\ u=u_0}} \bullet (X - X_0) + \frac{\partial f(X, u)}{\partial u} \Bigg|_{\substack{X=X_0 \\ u=u_0}} \bullet (u - u_0) \\ &= \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & A_{36} \\ A_{41} & A_{42} & A_{43} & A_{44} & A_{45} & A_{46} \\ A_{51} & A_{52} & A_{53} & A_{54} & A_{55} & A_{56} \\ A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & A_{66} \end{bmatrix} \cdot (X - X_0) + \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \\ B_6 \end{bmatrix} \cdot (u - u_0) \\ &= A(t)(X - X_0) + B(t)(u - u_0) = A(t)\tilde{X}(t) + B(t)\tilde{u}(t) \end{aligned} \quad (7)$$

In the above equations, the coefficient matrix can be described as

$$A(t) = \begin{bmatrix} A_{11} & \dot{\phi} - \frac{2C_{cf}\delta_f}{mv_x} & 0 & v_y - \frac{2aC_{cf}\delta_f}{mv_x} & 0 & 0 \\ A_{21} & -\frac{2(C_{cf}+C_{cr})}{mv_x} & 0 & -\dot{x} + \frac{2(bC_{cr}-aC_{cf})}{mv_x} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ A_{41} & \frac{2(bC_{cr}-aC_{cf})}{I_z v_x} & 0 & -\frac{2(a^2 C_{cf} + b^2 C_{cr})}{I_z v_x} & 0 & 0 \\ \cos \phi & -\sin \phi & -v_x \sin \phi - v_y \cos \phi & 0 & 0 & 0 \\ \sin \phi & \cos \phi & v_x \cos \phi - v_y \sin \phi & 0 & 0 & 0 \end{bmatrix}$$

In the matrix,

$$A_{11} = \frac{2C_{cf}\delta_f(v_y + a\omega)}{mv_x^2},$$

$$A_{21} = -\frac{2[C_{cf}(v_y + a\omega) - C_{cr}(v_y - b\omega)]}{mv_x^2},$$

$$A_{41} = \frac{2[aC_{cf}(v_y + a\omega) - bC_{cr}(v_y - b\omega)]}{I_z v_x^2}$$

and

$$B(t) = \left[\frac{2C_{cf}}{m} \left(2\delta_f - \frac{v_y + a\omega}{v_x} \right) \quad \frac{2C_{cf}}{m} \quad 0 \quad \frac{2aC_{cf}}{I_z} \quad 0 \quad 0 \right]^T$$

4.2. Model Discretization

According to the literature [20–23], apply the forward Euler method to discretize the above linear model, then we can obtain the following:

$$\tilde{X}(k+1) = A_k \tilde{X}(k) + B_k \tilde{u}(k) \quad (8)$$

In Equation (8), $A_k = I + TA(t)$, $B_k = TB(t)$, let $d_k = X_0(k+1) - A_k X_0(k) - B_k u_0(k)$, denote the deviation of the state quantity between this moment and the previous moment, then (8) is further simplified to obtain

$$X(k+1) = A_k X(k) + B_k u(k) + d(k) \quad (9)$$

At the same time, the state output vector is $\eta(k) = CX(k)$, and the coefficient matrix is

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

4.3. Build a Predictive Model

Establish a new state vector $\xi(k) = \begin{bmatrix} X(k) \\ u(k-1) \end{bmatrix}$, and derive the state vector at the next moment from the state vector to obtain:

$$\begin{aligned} \xi(k+1) &= \begin{bmatrix} X(k+1) \\ u(k) \end{bmatrix} = \begin{bmatrix} A_k X(k) + B_k u(k) + d(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} A_k X(k) + B_k [u(k-1) + \Delta u(k)] + d(k) \\ u(k-1) + \Delta u(k) \end{bmatrix} \\ &= \begin{bmatrix} A_k X(k) + B_k u(k-1) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} B_k \Delta u(k) \\ \Delta u(k) \end{bmatrix} + \begin{bmatrix} d(k) \\ 0 \end{bmatrix} = \begin{bmatrix} A_k & B_k \\ 0_{1 \times 6} & 1 \end{bmatrix} \begin{bmatrix} X(k) \\ u(k-1) \end{bmatrix} + \\ &\quad \begin{bmatrix} B_k \\ 1 \end{bmatrix} \Delta u(k) + \begin{bmatrix} d(k) \\ 0 \end{bmatrix} = \tilde{A}_k \xi(k) + \tilde{B}_k \Delta u(k) + \tilde{d}(k) \end{aligned} \quad (10)$$

From this, the new output is $\tilde{\eta}(k) = \tilde{C} \xi(k)$, and the coefficient matrix is $\tilde{C} = [C, 0_{2 \times 1}]$. According to the above formula, the state vector at the time in the future is further derived as

$$\begin{aligned} \xi(k+N_p) &= \tilde{A}_k \xi(k+N_p-1) + \tilde{B}_k \Delta u(k+N_p-1) + \tilde{d}(k+N_p-1) = \tilde{A}_k^{N_p} \xi(k) + \tilde{A}_k^{N_p-1} \tilde{B}_k \Delta u(k) \\ &\quad + \tilde{A}_k^{N_p-2} \tilde{B}_k \Delta u(k+1) + \cdots + \tilde{B}_k \Delta u(k+N_p-1) + \tilde{A}_k^{N_p-1} \tilde{d}(k) + \tilde{A}_k^{N_p-2} \tilde{d}(k+1) \\ &\quad + \cdots + \tilde{d}(k+N_p-1) \end{aligned} \quad (11)$$

The output at the time N_p in the future is also obtained as

$$\begin{aligned} \tilde{\eta}(k+N_p) &= \tilde{C} \xi(k+N_p) = \tilde{C} \tilde{A}_k^{N_p} \xi(k) + \tilde{C} \tilde{A}_k^{N_p-1} \tilde{B}_k \Delta u(k) + \tilde{C} \tilde{A}_k^{N_p-2} \tilde{B}_k \Delta u(k+1) + \cdots \\ &\quad + \tilde{C} \tilde{B}_k \Delta u(k+N_p-1) + \tilde{C} \tilde{A}_k^{N_p-1} \tilde{d}(k) + \tilde{C} \tilde{A}_k^{N_p-2} \tilde{d}(k+1) + \cdots + \tilde{C} \tilde{d}(k+N_p-1) \end{aligned} \quad (12)$$

In summary, the system state output equation is derived:

$$Y(k) = \Psi_k \xi(k) + \Theta_k \Delta U(k) + \Gamma_k \varphi(k) \quad (13)$$

In the above formula, the status output is

$$Y(k) = \begin{bmatrix} \tilde{\eta}(k+1) \\ \tilde{\eta}(k+2) \\ \vdots \\ \tilde{\eta}(k+N_c) \\ \vdots \\ \tilde{\eta}(k+N_p) \end{bmatrix},$$

state input coefficient matrix is

$$\Psi_k = \begin{bmatrix} \tilde{C}\tilde{A}_k \\ \tilde{C}\tilde{A}_k^2 \\ \tilde{C}\tilde{A}_k^3 \\ \vdots \\ \tilde{C}\tilde{A}_k^{N_p} \end{bmatrix},$$

input control amount increment is

$$\Delta U(k) = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+N_c - 1) \end{bmatrix},$$

control increment matrix is

$$\Theta_k = \begin{bmatrix} \tilde{C}\tilde{B}_k & 0 & \cdots & 0 \\ \tilde{C}\tilde{A}_k\tilde{B}_k & \tilde{C}\tilde{B}_k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{C}\tilde{A}_k^{N_p-1}\tilde{B}_k & \tilde{C}\tilde{A}_k^{N_p-2}\tilde{B}_k & \cdots & \tilde{C}\tilde{A}_k^{N_p-N_c}\tilde{B}_k \end{bmatrix}_{N_p \times N_c},$$

state variable deviation coefficient matrix is

$$\Gamma_k = \begin{bmatrix} \tilde{C} & 0 & \cdots & 0 \\ \tilde{C}\tilde{A}_k & \tilde{C} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{C}\tilde{A}_k^{N_p-1} & \tilde{C}\tilde{A}_k^{N_p-2} & \cdots & \tilde{C} \end{bmatrix}_{N_p \times N_p},$$

and state deviation is

$$\varphi(k) = \begin{bmatrix} \tilde{d}(k) \\ \tilde{d}(k+1) \\ \tilde{d}(k+2) \\ \vdots \\ \tilde{d}(k+N_p - 1) \end{bmatrix}.$$

4.4. Trajectory-Tracking Control

For trajectory-tracking control, we establish the objective function of the trajectory-tracking control and use the linear quadratic programming method in MATLAB to solve the control amount increment at each sampling time.

4.4.1. Establish Objective Function

Formula:

$$J = \sum_{i=1}^{N_p} \|\tilde{\eta}(k+i) - \eta_r(k+i)\|_Q^2 + \sum_{i=0}^{N_c-1} \|\Delta U(k+i)\|_R^2 + \rho \varepsilon^2 \quad (14)$$

In the above formula, N_p represents the prediction horizon, N_c represents the control horizon, Q represents the output weighting matrix, R represents the control increment weight matrix, ε represents the relaxation factor, and ρ represents the relaxation factor weight. The first term in the formula represents the system's ability to track the reference trajectory, the second term represents the control stability constraint, and the third term represents the soft constraint added to avoid infeasible solutions.

The objective function is transformed into a quadratic programming form. In the above formula

$$Y_r = \begin{bmatrix} \tilde{\eta}_r(k+1) \\ \tilde{\eta}_r(k+2) \\ \vdots \\ \tilde{\eta}_r(k+N_c) \\ \vdots \\ \tilde{\eta}_r(k+N_p) \end{bmatrix},$$

let $E(k) = \Psi_k \xi(k) + \Gamma_k \varphi(k) - Y_r$, $\tilde{Q} = I_{N_p} \otimes Q$, $\tilde{R} = I_{N_c} \otimes R$, and the objective function is simplified as

$$\begin{aligned} J = & (Y - Y_r)^T \tilde{Q} (Y - Y_r) + \Delta U^T \tilde{R} \Delta U + \varepsilon^T \rho \varepsilon = (E + \Theta \Delta U)^T \tilde{Q} (E + \Theta \Delta U) + \Delta U^T \tilde{R} \Delta U \\ & + \varepsilon^T \rho \varepsilon = \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix}^T \begin{bmatrix} \Theta^T \tilde{Q} \Theta + \tilde{R} & 0 \\ 0 & \rho \end{bmatrix} \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix} + [2E^T \tilde{Q} \Theta \Delta U, 0] \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix} + E^T \tilde{Q} E \end{aligned} \quad (15)$$

In the above formula, $X = \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix}$, $H = 2 \begin{bmatrix} \Theta^T \tilde{Q} \Theta + R & 0 \\ 0 & \rho \end{bmatrix}$, $f^T = [2E^T \tilde{Q} \Theta, 0]$.

4.4.2. Constraint Analysis

Derive the control increment constraint conditions as

$$\begin{bmatrix} A_I & 0 \\ -A_I & 0 \end{bmatrix} \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix} \leq \begin{bmatrix} U_{\max} - U_t \\ -U_{\min} + U_t \end{bmatrix} \quad (16)$$

Derive the output constraint conditions as

$$\begin{bmatrix} \Theta_k & 0 \\ -\Theta_k & 0 \end{bmatrix} \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix} \leq \begin{bmatrix} Y_{\max} - \Psi_k \xi(k) - \Gamma_k \varphi(k) \\ -Y_{\min} + \Psi_k \xi(k) + \Gamma_k \varphi(k) \end{bmatrix} \quad (17)$$

Combine Equations (16) and (17) to obtain the linear inequality constraints of quadratic programming:

$$\begin{bmatrix} A_I & 0 \\ -A_I & 0 \\ \Theta_k & 0 \\ -\Theta_k & 0 \end{bmatrix} \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix} \leq \begin{bmatrix} U_{\max} - U_t \\ -U_{\min} + U_t \\ Y_{\max} - \Psi_k \xi(k) - \Gamma_k \varphi(k) \\ -Y_{\min} + \Psi_k \xi(k) + \Gamma_k \varphi(k) \end{bmatrix} \quad (18)$$

Derive the upper and lower limit constraint conditions of the control quantity increment of the quadratic programming as

$$\begin{bmatrix} \Delta U_{\min} \\ 0 \end{bmatrix} \leq \begin{bmatrix} \Delta U \\ \varepsilon \end{bmatrix} \leq \begin{bmatrix} \Delta U_{\max} \\ M \end{bmatrix} \quad (19)$$

Combining the above constraints and objective function, the increment of the control quantity for each sampling period can be obtained. The control quantity at the current time

can be obtained by using the first value of the control sequence and the control quantity at the previous time, and the system enters the next control cycle to repeat the calculation of the increment of the control quantity at the next moment, and then obtain the control quantity at the next moment. According to this cycle, all the control quantities in the control time domain are calculated to realize the tracking control of the desired trajectory.

5. The Joint Simulation Analysis

In order to verify the tracking effect of the established control algorithm on the motion trajectory, we build a joint simulation framework based on MATLAB/Carsim (Thomas D. Gillespie, Michael Sayers and Steve Hann, Mechanical Simulation Corporation, Michigan, The United States). We use B-Class Hatchback 2012 model (Carsim software's built-in model) to provide vehicle position, speed, and yaw angle state. In addition, we use the established discrete vehicle dynamics model to write a control program in MATLAB and then output the front wheel steering angle to the Carsim vehicle model. The joint simulation framework constructed is shown in Figure 3.

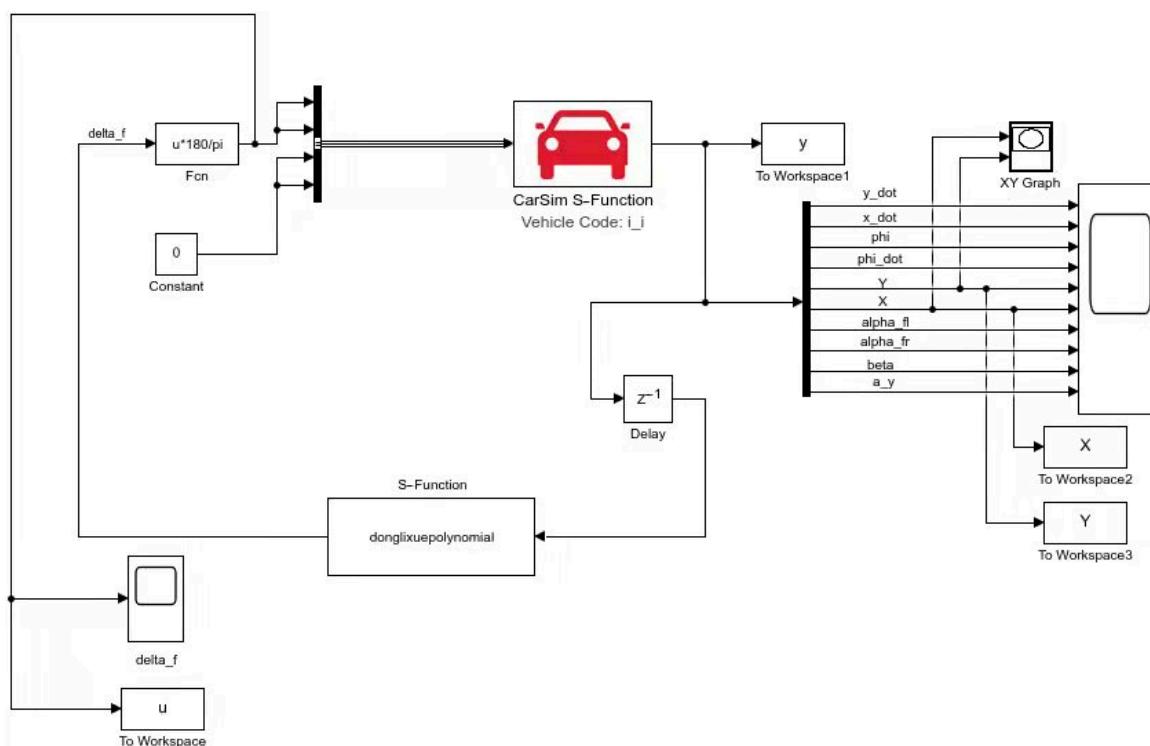


Figure 3. Joint simulation framework constructed based on MATLAB/Carsim.

We use the planned quintic polynomial virtual trajectory model of free lane changing to make the model predictive control for reference trajectory. The control parameters are set as follows: prediction time domain $N_p = 20$, control time domain $N_c = 5$, state output weight matrix $Q = [2000, 0; 0, 10,000]$, control increment $R = 5 \times 10^5$, relaxation factor weight $\rho = 1000$, relaxation factor upper limit $M = 10$, sampling time $T = 0.05$ s, simulation time is 20 s, $-10^\circ \leq$ front wheel angle $\delta f \leq 10^\circ$, and $-0.85^\circ \leq$ increment $\Delta \delta f \leq 0.85^\circ$.

5.1. The Simulation under Different Speed Conditions

The simulation scene is that the same vehicle changes lanes at 10 m/s, 20 m/s, and 30 m/s on a dry asphalt pavement with $\mu = 1$, and the control parameters remain unchanged. The simulation results of trajectory-tracking control are shown in Figures 4–7.

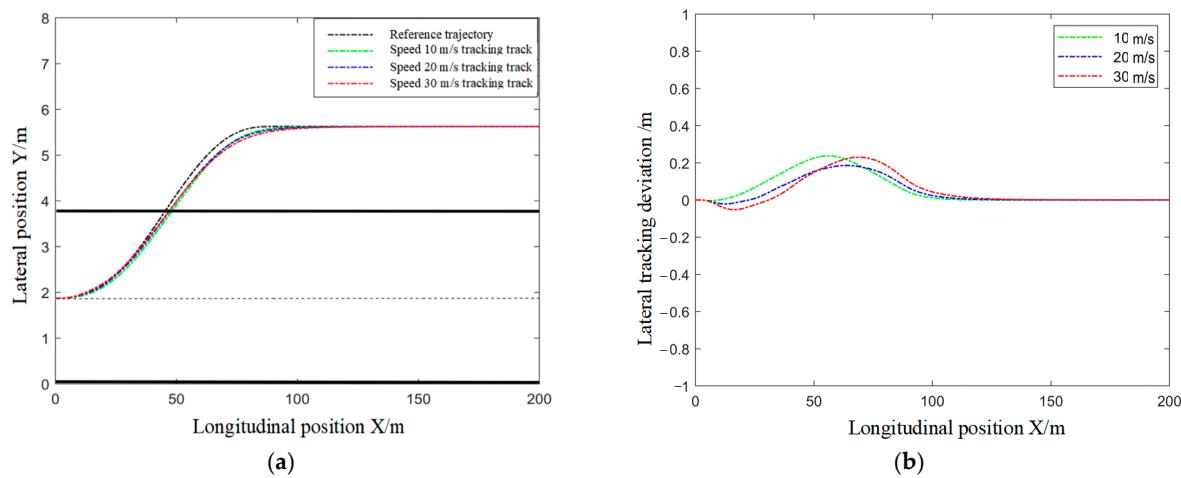


Figure 4. Tracking the lane-changing trajectory: (a) reference trajectory and actual trajectory; (b) tracking lateral deviation.

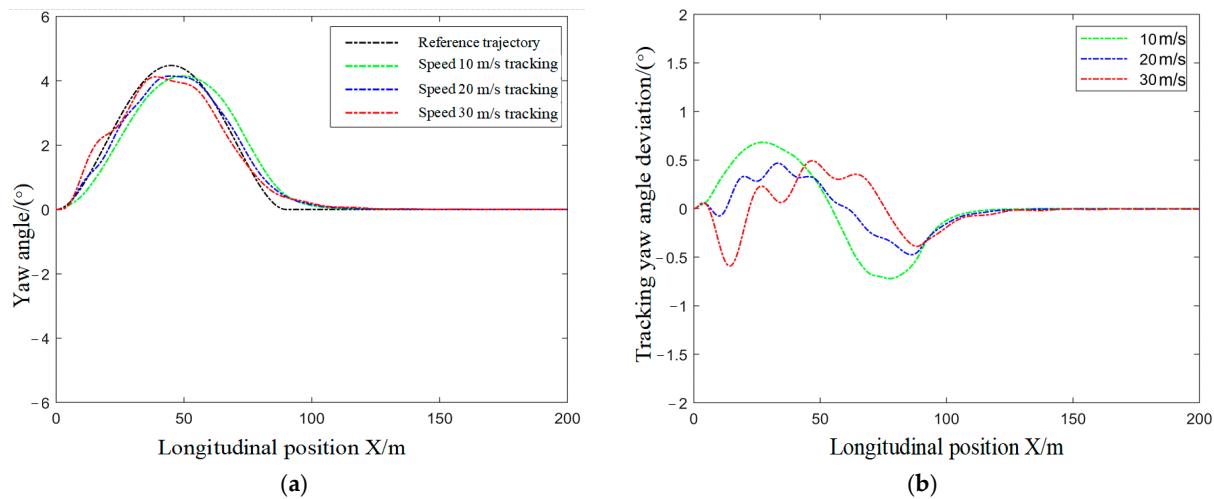


Figure 5. Tracking yaw angle changing: (a) reference yaw angle and actual yaw angle, and (b) tracking yaw angle deviation.

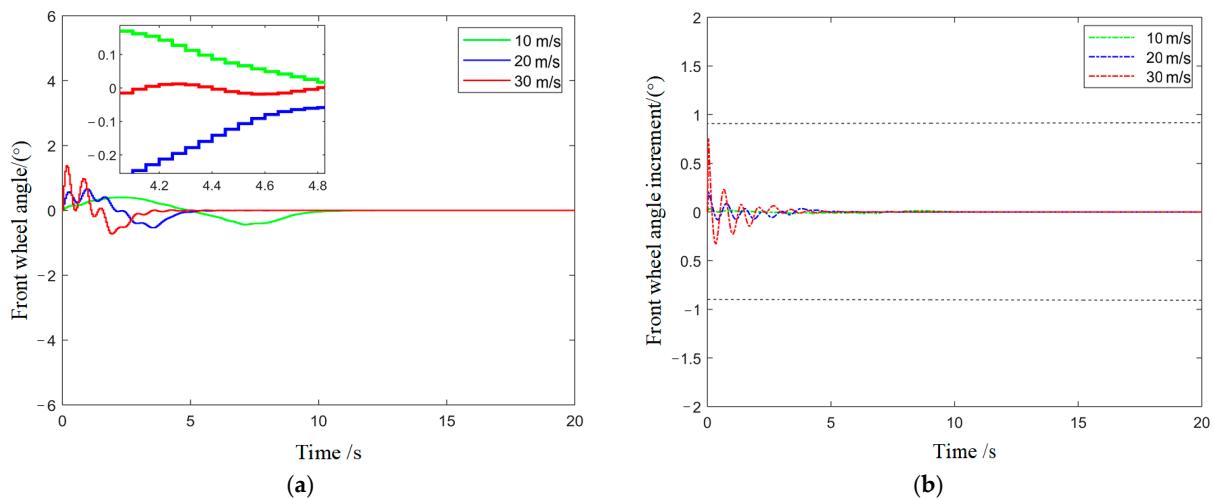


Figure 6. Front-wheel angle control amount and increment changing with time; (a) front-wheel angle changing with time, and (b) front-wheel angle increment changing with time.

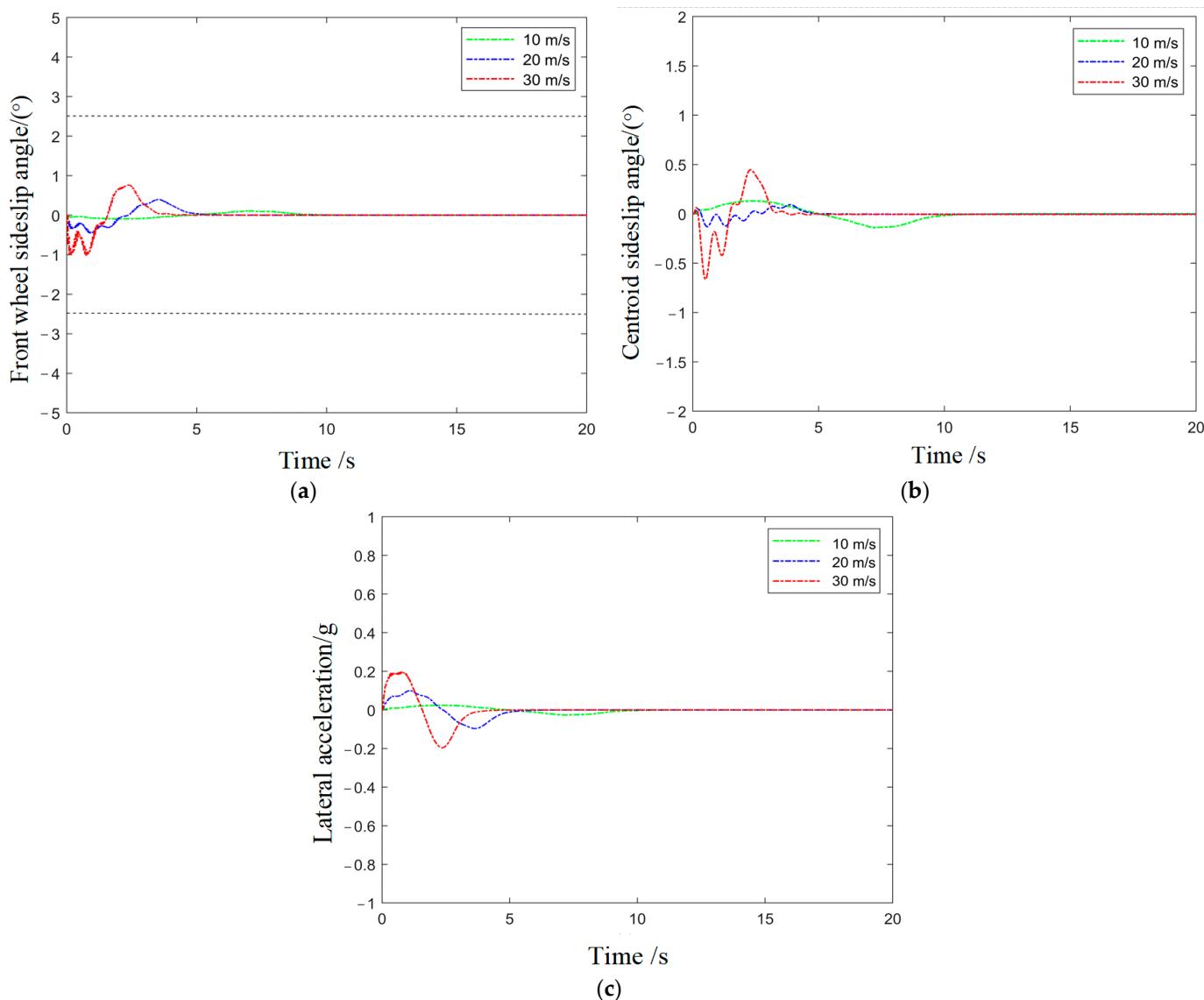


Figure 7. Dynamic constraints changing with time: (a) front-wheel side deflection angle changing with time, (b) centroid side deflection angle changing with time, and (c) lateral acceleration changing with time.

It can be seen from Figure 4a that the proposed MPC controller is capable of tracking the reference trajectory at different speeds, and the lane-changing vehicle can track the reference trajectory well at different speeds. Based on the selection of control points of the quintic Bessel curve and the characteristics of easy tracking of the curve, MPC has better robustness under the determined parameters. It can be seen from Figure 4b that the lane-changing lateral tracking deviation is $-0.1\text{--}0.3$ m, and the tracking deviation is small under different speed conditions, indicating that the control algorithm has the ability to track and control low-, medium-, and high-speed vehicles. The lane-changing tracking yaw angle change in Figure 5 further proves the point of view in Figure 4. Figure 6a shows that the front wheel angle is much smaller than the constraint range of $-10\text{--}10^\circ$ during the lane-change tracking process. From the partially enlarged view, it can be seen that the front wheel angle has no sudden change during the lane-changing process, and the control stability is relatively good. Figure 6b shows that the front wheel angle increment is also within the specified constraint range of $-0.85\text{--}0.85^\circ$, and the greater the longitudinal speed during the lane change, the greater the increment of the front wheel angle, which is consistent with the actual situation. It also shows that the control algorithm has better

stability for lane-changing tracking under high-speed conditions. Figure 7 shows the simulation results of dynamic constraint parameters changing with time during the lane-changing tracking process. From Figure 7a, we can see that the change of the front wheel side deflection angle is within the specified constraint range, and as the longitudinal speed increases, the front wheel side deflection angle increases, which is consistent with the actual situation. It can be seen from Figure 7b that the centroid side deflection angle changes much smaller than the specified constraint range of $-12\text{--}12^\circ$ at different tracking speeds, which shows that the control algorithm has better stability for lane-changing tracking under high-speed conditions.

Based on the above analysis, the model predictive control algorithm does not need to re-modify the control parameters each time for the changing speed during the lane-change tracking, which reflects that the algorithm has strong robustness for the longitudinal speed change in the lane-change tracking, and the tracking stability is also better.

5.2. The Simulation of Different Road Adhesion Conditions

The simulation scene is set to the same vehicle changing lanes on a dry asphalt pavement ($\mu = 1$) and an icy and snowy road ($\mu = 0.2$) at a speed of 30 m/s. The control parameters remain unchanged. The simulation results of trajectory-tracking control are shown in Figures 8–11.

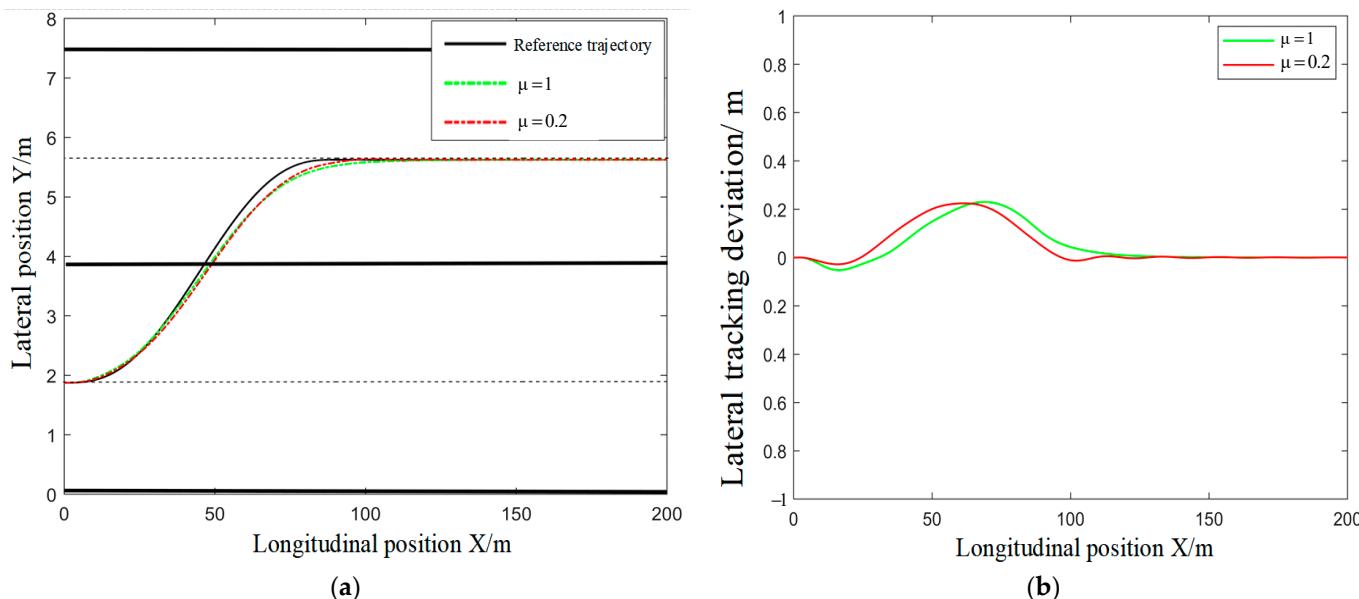


Figure 8. Tracking the lane-changing trajectory: (a) reference trajectory and actual trajectory, and (b) tracking lateral deviation.

The screenshot of the simulation 3D animation is shown in Figure 12.

It can be seen from Figure 8a that lane-changing vehicles can track the reference trajectory on roads with different adhesion conditions. Figure 8b shows that the lateral tracking deviations are all between -0.1 and 0.3 m. The lateral deviations under different road conditions are not obvious, indicating that the control algorithm is more robust for roads with different friction coefficients. Figure 9a shows that the yaw angle of the vehicle on the icy and snowy road is larger than that on the dry asphalt road, and the fluctuation is strong. This is because the adhesion coefficient of the icy and snowy road is reduced, which cannot provide sufficient vehicle steering lateral force. Finally, it causes the vehicle to skid during the steering process, and the vehicle's yaw angle increases. This is consistent with the actual situation. The control algorithm can continuously correct the tracking deviation. It can be seen from Figure 10a that when the vehicle is tracking the trajectory on an icy and snowy road, the front wheel angle is larger than that on a dry asphalt road, which verifies the analysis conclusion of Figure 8. It can be seen from the partially enlarged view that

the control algorithm does not cause a sudden change in the control amount of the front wheel angle, which shows that the control algorithm has good stability. During the tracking process, the front wheel angle and the increment of the front wheel angle are all within the limited constraint range, which further shows that the control algorithm is stable. It can be seen from Figure 11a,b that the front wheel side deflection angle and the centroid side deflection angle on ice and snow roads are larger than those on dry asphalt pavements during the lane changing, indicating that the stability of the vehicle when changing lanes on ice and snow roads is reduced, and the problem of sideslip is more likely to occur. This is consistent with the actual situation, but the control algorithm can continuously correct the deviation between the driving trajectory and the reference trajectory and finally converges to zero.

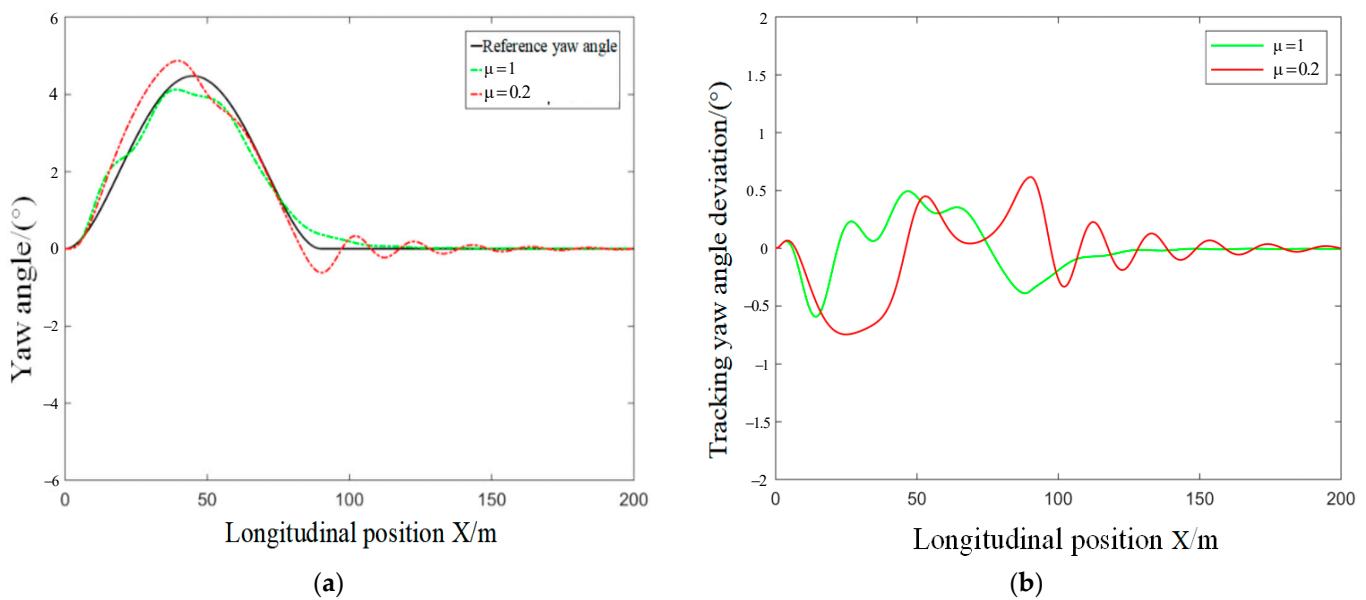


Figure 9. Tracking yaw angle changing; (a) reference yaw angle and actual yaw angle, and (b) tracking yaw angle deviation.

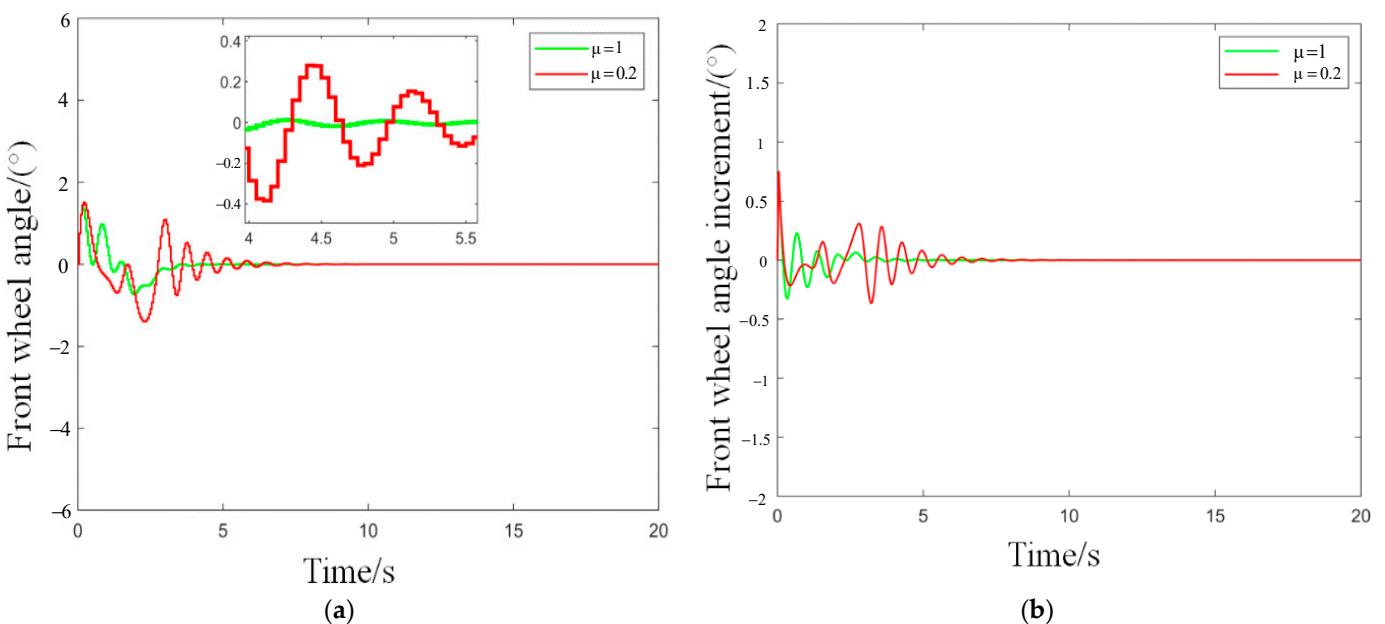


Figure 10. Front-wheel angle control amount and increment changing with time: (a) front-wheel angle changing with time, and (b) front-wheel angle increment changing with time.

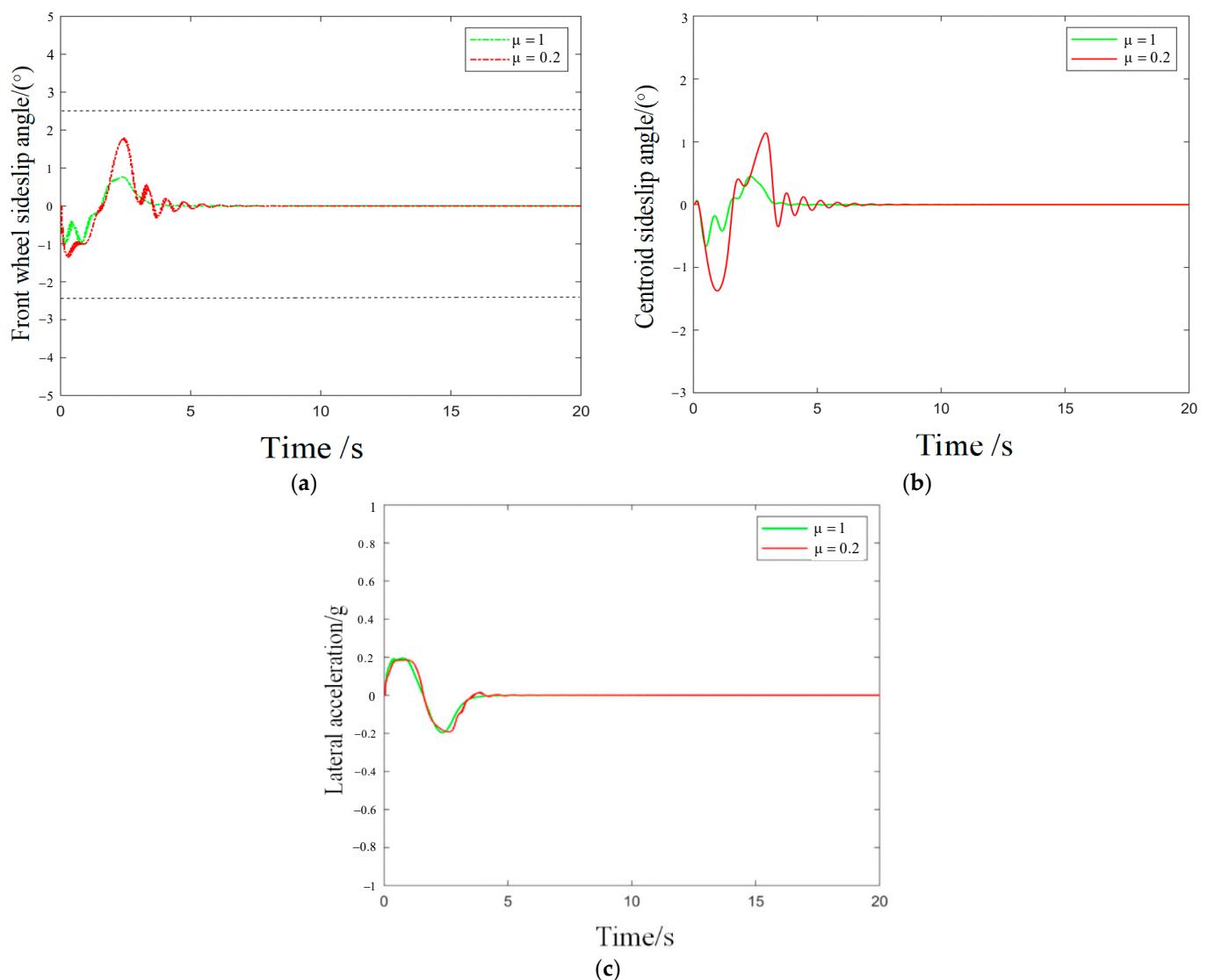


Figure 11. Dynamic constraints changing with time: (a) Front-wheel side deflection angle changing with time, (b) centroid side deflection angle changing with time, and (c) lateral acceleration changing with time.

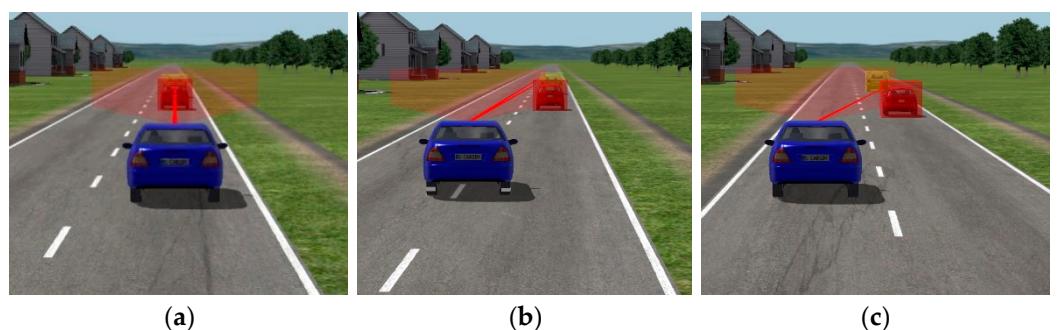


Figure 12. Free lane changing joint simulation animation scene: (a) start changing lanes, (b) changing lanes, and (c) end the lane change.

Based on the above analysis, it can be seen that the established model predictive control algorithm can continuously correct the deviation between the driving trajectory and the planned trajectory. For dry asphalt pavements and icy and snowy roads, it can complete lane-changing trajectory tracking without modifying control parameters, indicating that

the control algorithm has good robustness for roads with different adhesion conditions and can ensure driving stability.

6. Conclusions

- (1) According to the assumption that the active steering of the autonomous driving vehicle has a small angle, the lateral force of the tire and the side deflection angle, the longitudinal force, and the slip rate are approximately linear. The three degrees of freedom dynamic model of autonomous driving vehicle is established by constraining the front wheel angle of vehicles with different road adhesion and using the model predictive control algorithm to design a lane-changing lateral controller. It analyzes the control variables, incremental constraints, and dynamics constraints and establishes an objective function that reflects the comprehensive performance of vehicle lane changing.
- (2) We used MATLAB/Carsim software to carry out joint simulation experiments and used the quintic Bézier curve as the lane-changing planned reference track. Benefiting from the continuous curvature and easy tracking characteristics of the quintic Bezier curve, we continuously adjusted the control points and model parameters of the Bezier curve to obtain the optimal curve and mpc control model suitable for different speeds and road adhesion coefficients. The experiment shows that when the lane-changing trajectory is tracked under different road adhesion coefficients or different speed conditions, the control algorithm does not need to adjust the control parameters, and the tracking deviation can be continuously corrected until it converges to zero. The lateral tracking deviation is between -0.1 and 0.3 m. The constructed lateral controller also has good trajectory-tracking robustness and driving stability for low adhesion coefficient roads and high-speed driving conditions.
- (3) This research mainly solves the lateral control problem and joint simulation of lane changing of autonomous vehicles under different road adhesion and speed. We achieved the best effect of controller tracking by adjusting the position of six control points of the Bezier curve and controller parameters. When facing a complex road traffic environment, vehicle lane changing is a complex driving behavior with constantly changing lateral and longitudinal speeds. For future work, the vehicle control problem with more complex road conditions and surrounding vehicle interference will be considered, the comprehensive change of lateral and longitudinal speed will be considered, and coupling control of lane-changing maneuver between both directions based on MPC will be developed to validated and verified in the field experiments.

Author Contributions: Conceptualization, H.S. and D.Q.; methodology, H.S.; software, H.S. and H.G.; validation, T.W.; formal analysis, K.Z.; investigation, K.Z.; resources, H.S.; data curation, H.G.; writing—original draft preparation, H.S.; writing—review and editing, H.S.; visualization, T.W.; supervision, D.Q.; project administration, H.S. and T.W.; funding acquisition, D.Q. All authors have read and agreed to the published version of the manuscript.

Funding: The project was supported by the National Natural Science Foundation of China (Grant Nos. 51678320) and the Key R & D plan of Shandong Province (Grant Nos. 2019GGX101038).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Petrov, P.; Nashashibi, F. Adaptive Steering Control for Autonomous Lane Change Maneuver. In Proceedings of the 2013 IEEE Intelligent Vehicles Symposium, Gold Coast, QLD, Australia, 23–26 June 2013; pp. 835–840.
2. Bayar, G. Long distance autonomous trajectory tracking for an orchard vehicle. *Ind. Robot.* **2013**, *40*, 27–40. [CrossRef]

3. Cao, W.; Mukai, M.; Kawabe, T.; Nishira, H.; Fujiki, N. Cooperative vehicle path generation during merging using model predictive control with real-time optimization. *Control. Eng. Pract.* **2015**, *34*, 98–105. [[CrossRef](#)]
4. Yang, G.; Zhang, D.; Li, K.; Luo, Y. Cooperative Same-direction Automated Lane-changing Based on Vehicle-to-vehicle Communication. *J. Highw. Transp. Res. Dev.* **2017**, *34*, 120–129, 136.
5. Cai, B. *Multi-Vehicle Coordinated Lane Changing and Combined Control*; Dalian University of Technology: Dalian, China, 2017; pp. 40–46.
6. You, F.; Gu, G. Collaborative Lane Changing Trajectory Planning of Autonomous Vehicle. *Sci. Technol. Eng.* **2018**, *18*, 155–161.
7. Ji, J.; Tang, Z.; Wu, M.; Fang, J. Path Planning and Tracking for Lane Changing Based on Model Predictive Control. *China J. Highw. Transp.* **2018**, *31*, 172–179.
8. Huang, J.; Ji, Z.; Peng, X.; Hu, L. Driving Style Adaptive Lane-changing Trajectory Planning and Control. *China J. Highw. Transp.* **2019**, *32*, 226–239, 247.
9. Wang, Z. *Research on Autonomous Lane Change Control of Intelligent Networked Vehicles on Expressway*; Harbin Institute of Technology: Harbin, China, 2019; pp. 44–51.
10. Shao, Y.; Chen, Y. Design of Lateral Fuzzy Controller for Self-driving Cars. *J. Chongqing Jiaotong Univ. Nat. Sci.* **2019**, *38*, 7–13.
11. Leng, Y.; Zhao, S. Explicit Model Predictive Control for Intelligent Vehicle Lateral Trajectory Tracking. *J. Syst. Simul.* **2021**, *33*, 1177–1187.
12. Wang, J.; Teng, F.; Li, J.; Zang, L.; Fan, T.; Zhang, J.; Wang, X. Intelligent vehicle lane change trajectory control algorithm based on weight coefficient adaptive adjustment. *Adv. Mech. Eng.* **2021**, *13*, 1–16. [[CrossRef](#)]
13. Bae, I.; Jin, H.K.; Moon, J.; Kim, S. Lane Change Maneuver based on Bezier Curve providing Comfort Experience for Autonomous Vehicle Users. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019.
14. Gong, J.; Liu, K.; Qi, J. *Autonomous Vehicle Model Predictive Control*, 2nd ed.; Beijing Institute of Technology Press: Beijing, China, 2019; pp. 88–89.
15. Zheng, H.; Rong, J.; Ren, F. A lane changing model based on random utility theory. *J. Highw. Transp. Res. Dev.* **2004**, *21*, 88–91.
16. Liu, W.; Deng, R.; Zhang, Y.; Mo, C. Vehicle lane-changing model for overtaking lane in freeway off-ramp diverging area. *J. Highw. Transp. Res. Dev.* **2012**, *29*, 106–111.
17. Wang, C.; Li, Z.; Chen, Y.; Dai, G. Research on lane-changing models considering restricted space. *J. Highw. Transp. Res. Dev.* **2012**, *29*, 121–127.
18. Shi, B.; Yang, X.; Yu, X. Modeling and simulation of discretionary lane-change considering the combination with the carfollowing model. *J. Transp. Sci. Eng.* **2009**, *25*, 91–96.
19. Van Zanten, A.T.; Erhardt, R.; Landesfeind, K.; Pfaff, G. VDC Systems Development and Perspective. *SAE Trans.* **1998**, *107*, 424–444.
20. Van Winsum, W.; De Waard, D.; Brookhuis, K.A. Lane Change Manoeuvres and Safety Margins. *Transp. Res. Part F Traffic Psychol. Behav.* **1999**, *2*, 139–149. [[CrossRef](#)]
21. Li, X.; Qu, S.; Xia, Y. Cooperative Lane-changing Rules on Multilane under Condition of Cooperative Vehicle and Infrastructure System. *China J. Highw. Transp.* **2014**, *27*, 97–104.
22. Yang, Z.; Qi, Z.; Huang, Y. Trajectory Planning of Lane Changing for Intelligent Vehicles. *J. Chongqing Univ. Nat. Sci. Ed.* **2013**, *32*, 520–524.
23. Kesting, A.; Retiber, M.; Helbing, D. General Lane-changing Model MOBIL for Car-following Models. *Transp. Res. Rec.* **2007**, *1999*, 86–94. [[CrossRef](#)]