2020-01-5025 Published 24 Feb 2020

Model Predictive Automatic Lane Change Control for Intelligent Vehicles

Meng Ren School of Automotive Engineering, Tongji University, China

Guangqiang Wu China

Xunjie Chen and Xuyang Liu School of Automotive Engineering, Tongji University, China

Citation: Ren, M., Wu, G., Chen, X., and Liu, X., "Model Predictive Automatic Lane Change Control for Intelligent Vehicles," SAE Technical Paper 2020-01-5025, 2020, doi:10.4271/2020-01-5025.

Abstract

As a basic link of driving behavior in urban roads, vehicle lane changing has a significant impact on traffic flow characteristics and traffic safety, and the automation of lane change is also a key issue to be solved in the field of intelligent driving. In this paper, the research on the automatic lane change control for intelligent vehicles is carried out. The main work is to build the overall structure of the vehicle's automatic lane change behavior, of which the planning and tracking are focused. The strategy of Constant Time Headway (CTH) is used in the lane change decision. The lane change trajectory adopts the model of constant velocity offset plus sine function, and the longitudinal displacement is determined by the vehicle speed when changing lanes. Model Predictive Control (MPC) theory is used to track the trajectory, which optimizes tracking accuracy and vehicle stability and constrains the range and rate of change of vehicle speed and steering angle. By using weighted quadratic cost function, linearity matrix inequality constraints and upper and lower bound constraints, the multi-objective trajectory tracking problem is eventually transformed into a constrained online convex quadratic programming problem. The results of simulation and HIL test show that the scheme of automatic lane change can make the vehicle smoothly complete the lane changing behavior, and the errors can meet the error requirements of lane change. Compared with other controller, the method shows smaller lateral acceleration, stronger robustness and higher control precision during the test. Moreover, the computational time of the proposed MPC controller, implemented using the PXI, is 47.994ms during one sampling period, which can satisfy the real-time requirement.

1. Introduction

As a basic link in urban road driving behavior, vehicle lane change has a significant impact on traffic safety and traffic flow characteristics, and its automation is a key issue in the field of intelligent driving. At the same time, the automatic lane change behavior of the vehicle is also the most necessary part to achieve the L3 level autonomous driving [1]. The automatic lane change control of the vehicle involves the joint control of the lateral and longitudinal directions of the vehicle, and its stability and efficiency of the lane change are required at the same time, which make its control mechanism more complicated than the purely longitudinal control (such as adaptive cruise control) and pure lateral control (such as lane keeping control) in the field of intelligent driving.

There are huge number of previous researches of path planning and vehicle control [2]. In literature [3], based on the four-stage lane change theory, the B-spline theory is introduced to re-plan the lane change trajectory, and then a new highway free lane change model is established. In reference [4], a cubic polynomial curve is adopted to represent the lane-changing trajectory curve. This type of curve has the second order smoothness, which makes the position and velocity of vehicle continuous in the lane-changing process. In literature [5], a new lane change trajectory model by combining the constant velocity offset trajectory function and the sinusoidal function lane change trajectory is proposed, which has the characteristics of small lateral acceleration and good smoothness. In literature [6], the sinusoidal function lane change trajectory, circular lane change trajectory, polynomial lane change trajectory and trapezoidal lateral acceleration lane change trajectory are compared. In reference [7], a trajectory tracking control algorithm based on MPC algorithm was researched by adding the dynamic constraints of centroid, acceleration and tire side angle on the basis of vehicle dynamics model, which improve the stability of unmanned vehicles. In reference [8], a time-dependent polynomial curve is used to represent the lane-changing trajectory and an optimal trajectory is obtained by minimizing the cost function weighing comfort and efficiency in the lane-changing process. The model can dynamically plan lane-changing trajectories in response to the state changes of the surrounding vehicles in real-time. However, process parameters are included in

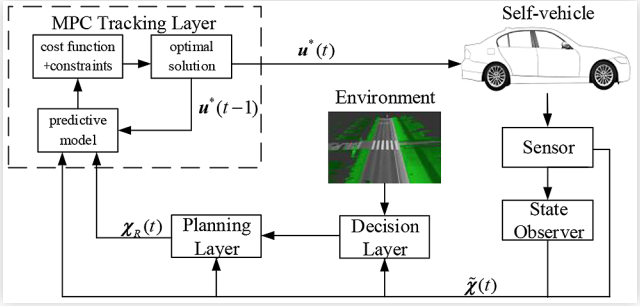
above path planning methods, which have no actual physical meaning. Therefore, these algorithms are not yet effectively applied to real-time control.

In terms of the lane change trajectory tracking, in literature [9], the information of the road ahead is used to form feedforward control, and the error of the current state variable of the vehicle is used to form feedback control. Then the feedforward and feedback control are used for lateral trajectory tracking control of unmanned vehicles. In literature [10], a sliding mode steering controller based on exponentially convergent disturbance observer is designed to track the expected yaw rate of the intelligent car's automatic lane change. In literature [11], based on the error of the current pose compared with the expected pose, the integral inversion algorithm is established to realize the tracking control of the vehicle lane change trajectory. In literature [12], the linear quadratic optimal control method is adopted, the optimized goals of which are dynamic tracking error and energy consumption. Then, the optimal control of speed adaptive trajectory tracking is achieved by adjusting the weight coefficients of two optimization goals at different speeds. In literature [13], a multi-constrained MPC is proposed for autonomous ground vehicle trajectory tracking. The active steering linear error model is applied in the controller, and softening constraint technique is used to ensure the smoothness of the trajectory. In reference [14], a NMPC method is proposed for automated lane-change behavior. In the NMPC method, switched weighting parameters are used in the cost function over the prediction horizon, which can reduce the number of reference points for control compared with the conventional framework. The NMPC method can improve tracking accuracy, but at the same time, the cost of solution time increases accordingly. Considering that MPC can easily deal with constrained problems and obtain theoretical optimal solutions to some extent, as the same time in order to improve the efficiency of solution, the trajectory tracking algorithm is based on linear MPC in the paper.

The automatic lane changing behavior of the vehicle includes four parts: environment perception, lane change decision, lane change trajectory planning and lane change trajectory tracking. The main work of this paper is to build the overall structure of automatic lane change behavior, and the planning and tracking of the lane change trajectory are mainly concerned.

The organization of this paper is as follows. At the end of Section 1, the overall structure of the automatic lane change is established. In Section 2, the vehicle model for the controller design is presented. In Section 3, the simple decision layer model is established. In Section 4, the lane change trajectory is planned. In Section 5, the lane change trajectory tracking strategy is designed. In Section 6, Simulink/CarSim co-simulations at different running conditions are carried out and simulation results are analyzed. In Section 7, the actual feasibility of algorithm in this paper is verified through a hardware-in-the-loop test. Finally, in Section 8, the conclusions are given.

FIGURE 1 The structure of automatic lane change



2. Vehicle Model

The vehicle's three degrees of freedom are considered: longitudinal, lateral and yaw, while the vehicle's vertical, pitch, roll and lateral characteristics of tire are ignored (ie, yaw angle is equal to heading angle). Then, a three-degree-of-freedom monorail kinematics model of vehicles is established.

In this paper, (X, Y) represent the longitudinal and lateral coordinates in the inertial coordinate system, and (x, y) represent the longitudinal and lateral coordinates in the vehicle coordinate system. Considering that the front wheel angle of the vehicle is small (generally less than 20°), the speed at the center of rear axle is substantially the same as the speed at the center of mass. Thus, the speed of vehicle v is replaced with the speed at the center of rear axle v_r. Then, the three-degree-of-freedom monorail kinematics model of vehicles can be obtained according to Figure 2

$$\begin{cases} \dot{X} = v\cos\psi \\ \dot{Y} = v\sin\psi \\ \dot{\psi} = \dfrac{v}{L}\tan\delta \end{cases} \tag{1}$$

FIGURE 2 Three-degree-of-freedom monorail kinematics model of vehicles

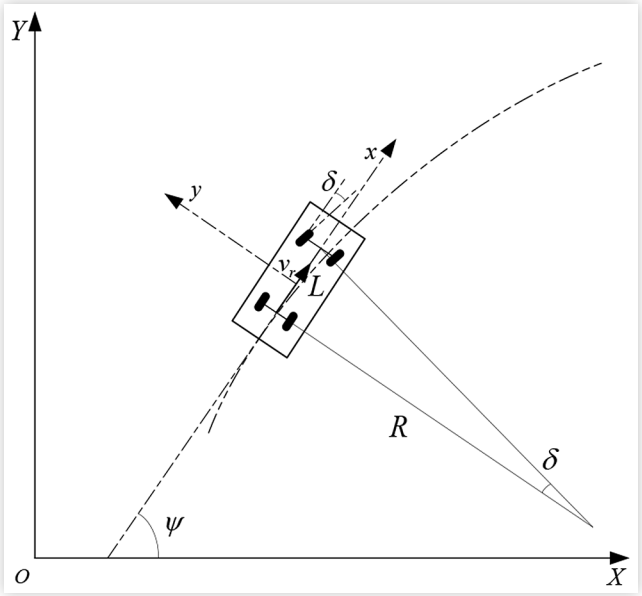
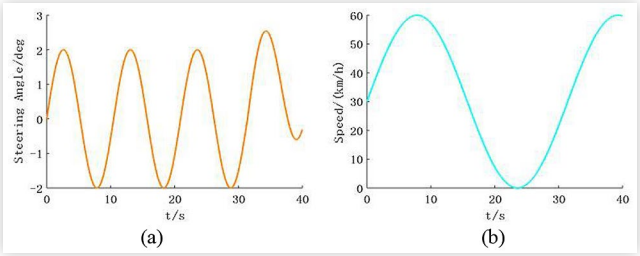


FIGURE 3 The input of vehicle model (steering angle (a) and speed (b)) with respect to t



Where $\dot{X}$ and $\dot{Y}$ are the vehicle lateral speed and longitudinal speed in the inertial coordinate system, $\dot{\psi}$ is the yaw rate of vehicle, δ is the steering angle, L is the wheelbase, and R is the turning radius of vehicle.

Equation (1) is converted to the form of state-space equation

$$\dot{\boldsymbol{\chi}} = f(\boldsymbol{\chi}, \boldsymbol{u}) \tag{2}$$

Where $\boldsymbol{u} = [v, \delta]^T$ and $\boldsymbol{\chi} = [X, Y, \psi]^T$.

To verify the established vehicle kinematics model, the kinematics model was built in Simulink, and a vehicle model was established in CarSim under the same input, which means that the steering angle and the speed of these two model are same as Figure 3. The output is the vehicle position and heading. The initial state variable of the vehicle is [0, 0, 0]. Then the state variable with respect to t are obtained as Figure 4.

It can be seen from Figure 4 that the position and heading angle of the vehicle kinematic model and the CarSim model are in agreement at the same input. That is, the vehicle model represented by the equation (2) can better reflect the kinematics characteristics of the vehicle while traveling. Therefore, the subsequent algorithm in this paper is designed based on this model.

In order to improve the efficiency of solution, this paper adopts the linear model predictive control strategy, and the nonlinear vehicle model needs to be linearized. For the model described in equation (2), the nonlinear vector function can be expanded to a Taylor series at the desired trajectory point when the vector function f has a continuous partial derivative for $\boldsymbol{\chi}$ and $\boldsymbol{u}$

$$\dot{\boldsymbol{\chi}} = f(\boldsymbol{\chi}_R, \boldsymbol{u}_R) + \left.\frac{\partial f}{\partial \boldsymbol{\chi}^T}\right|_{\substack{\boldsymbol{\chi}=\boldsymbol{\chi}_R \\ \boldsymbol{u}=\boldsymbol{u}_R}} (\boldsymbol{\chi} - \boldsymbol{\chi}_R) + \left.\frac{\partial f}{\partial \boldsymbol{u}^T}\right|_{\substack{\boldsymbol{\chi}=\boldsymbol{\chi}_R \\ \boldsymbol{u}=\boldsymbol{u}_R}} (\boldsymbol{u} - \boldsymbol{u}_R) + R(\boldsymbol{\chi}, \boldsymbol{u}) \tag{3}$$

By subtracting the vector function value $f(\boldsymbol{\chi}_R, \boldsymbol{u}_R)$ at the desired trajectory point and ignoring its high-order term $R(\boldsymbol{\chi}, \boldsymbol{u})$ from equation (3), a linear vehicle model is obtained as

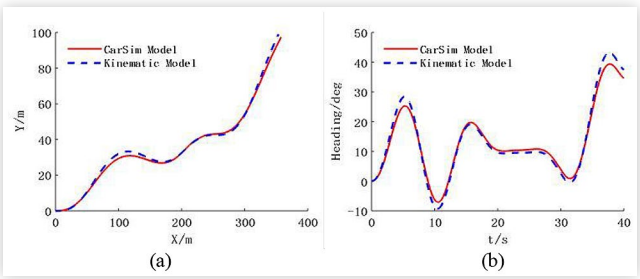
$$\dot{\tilde{\boldsymbol{\chi}}} = \begin{bmatrix} 0 & 0 & -v_R \sin\psi_R \\ 0 & 0 & v_R \cos\psi_R \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X - X_R \\ Y - Y_R \\ \psi - \psi_R \end{bmatrix} + \begin{bmatrix} \cos\psi_R & 0 \\ \sin\psi_R & 0 \\ \dfrac{\tan\psi_R}{L} & \dfrac{v_R}{L\cos^2\delta_R} \end{bmatrix} \begin{bmatrix} v - v_R \\ \delta - \delta_R \end{bmatrix} \tag{4}$$

Considering that the actual controller is a digital controller, equation (4) is discretized by using Euler discretization

$$\tilde{\boldsymbol{\chi}}\left[(k+1)T\right] = \boldsymbol{A}_k \tilde{\boldsymbol{\chi}}(kT) + \boldsymbol{B}_k \boldsymbol{u}(kT) \tag{5}$$

Where $\boldsymbol{A}_k = \begin{bmatrix} 1 & 0 & -Tv_R \sin\psi_R \\ 0 & 1 & Tv_R \cos\psi_R \\ 0 & 0 & 1 \end{bmatrix}$, $\boldsymbol{B}_k = \begin{bmatrix} T\cos\psi_R & 0 \\ T\sin\psi_R & 0 \\ T\tan\psi_R / L & Tv_R / (L\cos^2\delta_R) \end{bmatrix}$, and T = 50ms is the sampling time.

FIGURE 4 (a) is the path of vehicle models in the inertial coordinate system, and (b) is the heading angle with respect to t



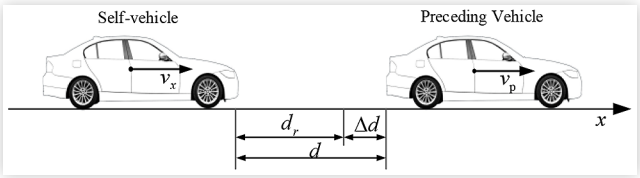
3. Lane Change Decision

The lane change decision is determined by the Constant Time Headway (CTH) strategy in literature [15]. Change lane when the distance between self-vehicle and preceding vehicle is less than safe distance

$$d_r = \tau_h v_x + d_0 \tag{6}$$

Where d_r is the safety distance, τ_h is the time headway, v_x is the longitudinal speed of self-vehicle, and d_0 = 3m is the distance between self-vehicle and preceding vehicle after self-vehicle stopping, as shown in Figure 5.

FIGURE 5 Constant time headway method



4. Lane Change Trajectory Planning

In order to ensure the smoothness of the lane change trajectory and prevent the vehicle from excessive lateral acceleration, this paper uses the constant velocity offset plus sine function lane change model to plan the lane change trajectory. The desired vehicle speed, desired heading angle, desired position, and desired steering angle with respect to t are obtained based on the lane change trajectory, then the four desired values are input to the trajectory tracking controller. According to the literature [5], the form of the constant velocity offset plus sine function lane change model is

$$y(x)=\frac{d_w}{2\pi}\left[\frac{2\pi x}{l_w}-\sin\left(\frac{2\pi x}{L}\right)\right] \tag{7}$$

Where assume that $x = 0$ when the lane change begins; d_w is the distance between the centerline of the two lanes; l_w is the longitudinal displacement change during the lane change of the vehicle. In this paper, the desired longitudinal speed $v_{x,R}$ remains unchanged during the lane change, thus $l_w = (t_f - t_0)v_{x,R}$, where t_0 and t_f are the time to start and end the lane change, generally $(t_f - t_0) = 3.6$s. Then, according to the lane change trajectory, the desired values can be calculated.

A. Desired Vehicle Speed v_R

Longitudinal displacement x_R of the vehicle during the lane change is

$$x_R = v_{x,R}(t-t_0) \tag{8}$$

Bringing equation (8) into equation (7), the function of desired lateral displacement y_R with respect to t can be obtained during lane changing process

$$y_R=\frac{d_w}{2\pi}\left[\frac{2\pi v_{x,R}(t-t_0)}{l_w}-\sin\left(\frac{2\pi v_{x,R}(t-t_0)}{l_w}\right)\right] \tag{9}$$

The desired lateral velocity $v_{y,R}$ is obtained by taking the first derivative of equation (9) with respect to t

$$v_{y,R}=\frac{d_w v_{x,R}}{l_w}\left[1-\cos\left(\frac{2\pi v_{x,R}(t-t_0)}{l_w}\right)\right] \tag{10}$$

Also the desired vehicle speed v_R can be expressed as

$$v_R=\sqrt{{v_{y,R}}^2+{v_{x,R}}^2} \tag{11}$$

Then, bringing equation (10) into equation (11), the function of desired vehicle speed with respect to t during lane change is obtained as

$$v_R=v_x\sqrt{\frac{d_w^2}{l_w^2}\left[1+\cos\left(\frac{2\pi v_{x,R}t}{l_w}-\pi\right)\right]^2+1} \tag{12}$$

B. Desired Heading Angle ψ_R

Taking the first and second derivatives of equation (7) with respect to x

$$\dot{y}(x)=\frac{d_w}{l_w}\left[1-\cos\left(\frac{2\pi x_R}{l_w}\right)\right] \tag{13}$$

$$\ddot{y}(x)=\frac{2\pi d_w}{l_w^2}\sin\left(\frac{2\pi x_R}{l_w}\right) \tag{14}$$

The desired heading angle ψ_R can be expressed as

$$\psi_R=\arctan\dot{y}(x) \tag{15}$$

Combining equations (8), (13) and (15), the function of desired heading angle with respect to t during lane change is obtained as

$$\psi_R=\arctan\left[\frac{d_w}{l_w}-\frac{d_w}{l_w}\cos\left(\frac{2\pi v_{x,R}(t-t_0)}{l_w}\right)\right] \tag{16}$$

C. Desired Position (X_R, Y_R)

The vehicle coordinate system is transformed to the inertial coordinate system

$$\begin{cases} X_R = X_0 + x_R\cos\psi_R - y_R\sin\psi_R \\ Y_R = Y_0 + x_R\sin\psi_R + y_R\cos\psi_R \end{cases} \tag{17}$$

Where (X_0, Y_0) is the inertial coordinate point at the start of the lane change, (x_R, y_R) can be obtained from equations (8) and (9), and ψ_R can be obtained from equation (15).

D. Desired Steering angle δ_R

The radius of curvature is calculated by

$$R=\frac{\left(1+\dot{y}^2(x)\right)^{\frac{3}{2}}}{\ddot{y}(x)} \tag{18}$$

According to Figure 2, the relationship between turning radius and steering angle can be obtained

$$\delta=\arctan(L/R) \tag{19}$$

Combining equations (8), (13), (14) and (19), the function of desired steering angle with respect to t during lane change can be obtained

$$\delta_R = \arctan \frac{2\pi d_w l_w L \sin\left(\frac{2\pi v_{x,R}(t-t_0)}{l_w}\right)}{\left[l_w^2 + d_w^2\left(1-\cos\left(\frac{2\pi v_{x,R}(t-t_0)}{l_w}\right)\right)^2\right]^{\frac{3}{2}}} \tag{20}$$

5. Lane Change Trajectory Tracking

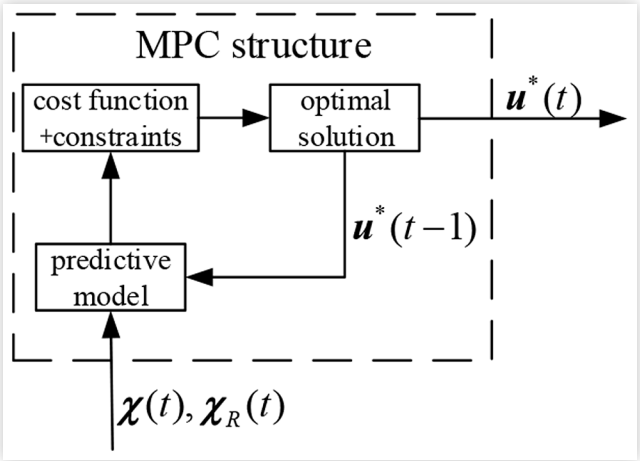
The trajectory tracking layer of Figure 1 is drawn separately as Figure 6. It can be seen that there are three elements of MPC: predictive model, cost function and constraints. Then the optimal control variable is obtained based on cost function.

A. Predictive Model Establishment

By using equation (5), the state variable at $(k + 1)T$ can be obtained according to the state variable and control variable at kT, but usually the cost function of MPC needs to find the overall optimality of the state variables at multiple time points in the future. At the same time, in order to constrain the change of the control variable, the control increment $\Delta \boldsymbol{u}$ needs to be expressed in an explicit form. Thus, equation (5) is linearly transformed to

$$\begin{aligned} \xi(t+k+1|t) &= \tilde{\boldsymbol{A}}_{k,t}\xi(t+k|t) + \tilde{\boldsymbol{B}}_{k,t}\Delta\boldsymbol{u}(t+k|t) \\ \eta(t+k|t) &= \boldsymbol{C}_{k,t}\xi(t+k|t) \end{aligned} \tag{21}$$

FIGURE 6 MPC structure of lane change trajectory tracking



Where $\xi(t+k|t) = \begin{bmatrix} \tilde{\chi}(t+k|t) \\ \tilde{u}(t+k-1|t) \end{bmatrix}$, $\tilde{\boldsymbol{A}}_{k,t} = \begin{bmatrix} \boldsymbol{A}_{k,t} & \boldsymbol{B}_{k,t} \\ \boldsymbol{0}_{m\times n} & \boldsymbol{I}_m \end{bmatrix}$, $\tilde{\boldsymbol{B}}_{k,t} = \begin{bmatrix} \boldsymbol{B}_{k,t} \\ \boldsymbol{I}_m \end{bmatrix}$, $\tilde{\boldsymbol{C}}_{k,t} = \begin{bmatrix} \boldsymbol{I}_{n\times n} & \boldsymbol{0}_{n\times m} \\ \boldsymbol{0}_{m\times n} & \boldsymbol{0}_{m\times m} \end{bmatrix}$, $n = 3$ is the state variable dimension, and $m = 2$ is the control variable dimension.

In order to further improve the efficiency of the solution, the state-space equation coefficient matrix in the predictive horizon is set to be unchanged, that is

$$\begin{aligned} \tilde{\boldsymbol{A}}_{k,t} &= \tilde{\boldsymbol{A}}_t, \quad k = t+1, t+2, \cdots, t+N_p-1 \\ \tilde{\boldsymbol{B}}_{k,t} &= \tilde{\boldsymbol{B}}_t, \quad k = t+1, t+2, \cdots, t+N_c-1 \end{aligned} \tag{22}$$

Where N_p is defined as the predictive horizon, and N_c is defined as the control horizon. Then, according to equations (21) and (22), the state variable and output at $(t + k)$ can be obtained when system is at t as

$$\begin{aligned} \xi(t+k|t) = \tilde{\boldsymbol{A}}_t^{N_p}\xi(t|t) + \tilde{\boldsymbol{A}}_t^{N_p-1}\tilde{\boldsymbol{B}}_t\Delta\boldsymbol{u}(t|t) + \\ \cdots + \tilde{\boldsymbol{A}}_t^{N_p-N_c-1}\tilde{\boldsymbol{B}}_t\Delta\boldsymbol{u}(t+k|t) \end{aligned} \tag{23}$$

$$\begin{aligned} \eta(t+k|t) = \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p}\xi(t|t) + \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p-1}\tilde{\boldsymbol{B}}_t\Delta\boldsymbol{u}(t|t) + \\ \cdots + \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p-N_c-1}\tilde{\boldsymbol{B}}_t\Delta\boldsymbol{u}(t+k|t) \end{aligned} \tag{24}$$

Finally, the output is represented as a vector form in the entire predictive horizon according to equation (24)

$$\boldsymbol{Y}_{out}(t) = \boldsymbol{\phi}_t\xi(t|t) + \boldsymbol{\Theta}_t\Delta\boldsymbol{U}(t) \tag{25}$$

Where $\boldsymbol{Y}_{out}(t) = \begin{bmatrix} \eta(t+1|t) \\ \eta(t+2|t) \\ \vdots \\ \eta(t+N_c|t) \\ \vdots \\ \eta(t+N_p|t) \end{bmatrix}$, $\Delta\boldsymbol{U}(t) = \begin{bmatrix} \Delta\boldsymbol{u}(t|t) \\ \Delta\boldsymbol{u}(t+1|t) \\ \vdots \\ \Delta\boldsymbol{u}(t+N_c-1|t) \end{bmatrix}$,

$$\boldsymbol{\phi}_t = \begin{bmatrix} \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t \\ \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^2 \\ \vdots \\ \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_c} \\ \vdots \\ \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p} \end{bmatrix}, \boldsymbol{\Theta}_t = \begin{bmatrix} \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{B}}_t & 0 & 0 & 0 \\ \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t\tilde{\boldsymbol{B}}_t & \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{B}}_t & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_c-1}\tilde{\boldsymbol{B}}_t & \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_c-1}\tilde{\boldsymbol{B}}_t & \cdots & \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{B}}_t \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p-1}\tilde{\boldsymbol{B}}_t & \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p-2}\tilde{\boldsymbol{B}}_t & \cdots & \tilde{\boldsymbol{C}}_t\tilde{\boldsymbol{A}}_t^{N_p-N_c-1}\tilde{\boldsymbol{B}}_t \end{bmatrix}$$

Equation (25) is the final predictive model. When the current state variable of the system and the control increment vector in the control horizon are given, the output of the system at any time in the predictive horizon can be obtained according to equation (25), which is the so-called "predictive" function in model predictive control [16].

B. Cost Function Design

The model predictive control problem can generally be converted into a quadratic programming problem (QP) to solve it. The method of converting MPC problem into a QP problem is described in Literature [17]. For the MPC used in vehicle trajectory tracking, the cost function is required to enable the vehicle to follow desired trajectory as accurately as possible while ensuring smoothness. The stability of vehicle is improved by optimizing control increment, and the accuracy of the trajectory tracking is improved by optimizing state variable. In addition, in order to prevent the occurrence of no feasible solution, a relaxing factor should be added to the cost function. Based on above analysis, the following cost function is established

$$J(t)=\sum_{k=t+1}^{N_p}\left\|\boldsymbol{\eta}(t+k|t)\right\|_{\Gamma_\eta}^2+\sum_{k=t+1}^{N_c-1}\left\|\Delta\boldsymbol{u}(t+k|t)\right\|_{\Gamma_u}^2+\rho\varepsilon^2 \quad (26)$$

Where ε is the relaxing factor; ρ is the penalty coefficient, which is used to punish the relaxing factor to extend the relaxing ability of constraint boundary [18]; and $\|\ \|$ means to calculate the Euclidean norm of intermediate item. Ignoring constant term in equation (26), equation (26) is converted to the standard quadratic form as

$$J(t)=\left[\Delta U(t)^T,\varepsilon\right]^T P_t\left[\Delta U(t)^T,\varepsilon\right]+Q_t\left[\Delta U(t)^T,\varepsilon\right] \quad (27)$$

Where $P_t=\begin{bmatrix}\Theta_t^T\Gamma_\eta\Theta_t+\Gamma_u & 0\\ 0 & \rho\end{bmatrix}$ and $Q_t=\begin{bmatrix}2\left(\phi_t\xi(t|t)\right)^T\Gamma_\eta\Theta_t & 0\end{bmatrix}$.

C. Constraints Description

The optimization variable in this paper is the control increment sequence in control horizon $\Delta\boldsymbol{U}(t)$, the first component of which plus control variable of last moment as the current control variable $\boldsymbol{u}(t)$. Considering that the control variable (ie, speed and steering angle) and its rate of change are limited, the control variable and control increments are constrained as

$$\begin{cases}\boldsymbol{u}_{\min}(t+k|t)\le\boldsymbol{u}(t+k|t)\le\boldsymbol{u}_{\max}(t+k|t)\\ \Delta\boldsymbol{u}_{\min}(t+k|t)\le\Delta\boldsymbol{u}(t+k|t)\le\Delta\boldsymbol{u}_{\max}(t+k|t)\end{cases} \quad (28)$$

Where $\boldsymbol{u}(t+k+1|t)=\boldsymbol{u}(t+k|t)+\Delta\boldsymbol{u}(t+k|t)$, $k=0,1,\cdots,N_c-1$.

Let $\boldsymbol{U}_0=\mathbf{1}_{N_c}\otimes\boldsymbol{u}(t-1)$ and $\boldsymbol{L}=\underbrace{\begin{bmatrix}1&0&\cdots&\cdots&0\\1&1&0&\cdots&0\\1&1&1&\cdots&0\\\vdots&\vdots&\vdots&\ddots&\vdots\\1&1&\cdots&1&1\end{bmatrix}}_{N_c\times N_c}\otimes\boldsymbol{I}_m$,

where$(t-1)$, $\mathbf{1}_{N_c}$ is a column vector with a row number of N_c, $\boldsymbol{I}_m$ is the identity matrix of dimension m, and $\otimes$ is the Kronecker product. Then

$$\boldsymbol{U}(t)=\boldsymbol{U}_0+\boldsymbol{L}\Delta\boldsymbol{U}(t) \quad (29)$$

Combining equation (28) and (29), the constraints of control increment sequence are obtained

$$\begin{cases}\boldsymbol{U}_{\min}\le\boldsymbol{U}_0+\boldsymbol{L}\Delta\boldsymbol{U}(t)\le\boldsymbol{U}_{\max}\\ \Delta\boldsymbol{U}_{\min}\le\Delta\boldsymbol{U}(t)\le\Delta\boldsymbol{U}_{\max}\end{cases} \quad (30)$$

Considering the relaxing factor, the constraints are expressed as the form of linear matrix inequality constraint and upper and lower bound constraints

$$\begin{aligned}s.t.\quad&\begin{bmatrix}\boldsymbol{L}&\mathbf{0}_{N_cm}\\-\boldsymbol{L}&\mathbf{0}_{N_cm}\end{bmatrix}\begin{bmatrix}\Delta\boldsymbol{U}(t)\\\varepsilon\end{bmatrix}\le\begin{bmatrix}\boldsymbol{U}_{\max}-\boldsymbol{U}_0\\-\boldsymbol{U}_{\min}+\boldsymbol{U}_0\end{bmatrix}\\&\begin{bmatrix}\Delta\boldsymbol{U}_{\min}\\\varepsilon_{\min}\end{bmatrix}\le\begin{bmatrix}\Delta\boldsymbol{U}(t)\\\varepsilon\end{bmatrix}\le\begin{bmatrix}\Delta\boldsymbol{U}_{\max}\\\varepsilon_{\max}\end{bmatrix}\end{aligned} \quad (31)$$

6. Simulink/CarSim Co-Simulation

By using Simulink and CarSim, a simulation platform of vehicle automatic lane change is established on the basis of Figure 1. The vehicle model is built in CarSim to replace the real vehicle in actual control system, and the forward ranging radar is set to detect the relative distance between self-vehicle and preceding vehicle as a environment sensing layer. The driving environment is a one-way three-lane straight road. The output of the control system is the vehicle speed v and steering angle δ, and the input is the coordinates of the vehicle in the inertial coordinate system (X, Y), heading angle ψ, longitudinal speed of vehicle v_x and relative distance d_r. The simulation platform is as follows

In order to express the quality of the lane change algorithm at different speeds, a simulation experiment is carried out with speeds of 30km/h, 60km/h and 90km/h on a dry concrete road (tire-road friction coefficient $\mu = 0.8$). The following results are obtained by running the simulation platform

According to the experimental results, the lane change trajectories and the tracking errors of direction X and Y at different speeds in inertial coordinate system are plotted as Figure 9, 10 and 11. At the same time, the trajectory tracking

FIGURE 7 The vehicle automatic lane change simulation platform based on Simulink/CarSim. The red module is the vehicle model of CarSim, the yellow module is the decision layer based on section III, the blue module is the planning layer based on section IV, and the cyan module is the tracking layer based on section V.

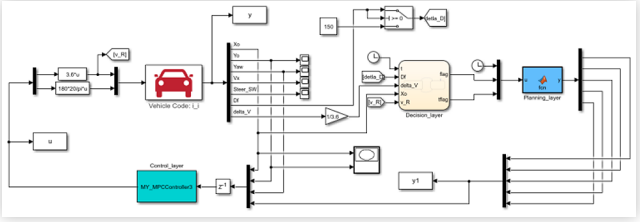


FIGURE 8 The screenshot of automatic lane change animation with a speed of 90km/h in Simulink/CarSim co-simulation

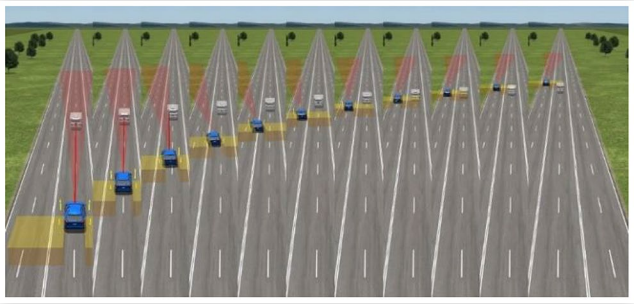


FIGURE 9 The lane change trajectory in inertial coordinate system. (a) is the lane change trajectory with a speed of 30km/h, (b) is with a speed of 60km/h; (c) is with a speed of 90km/h.

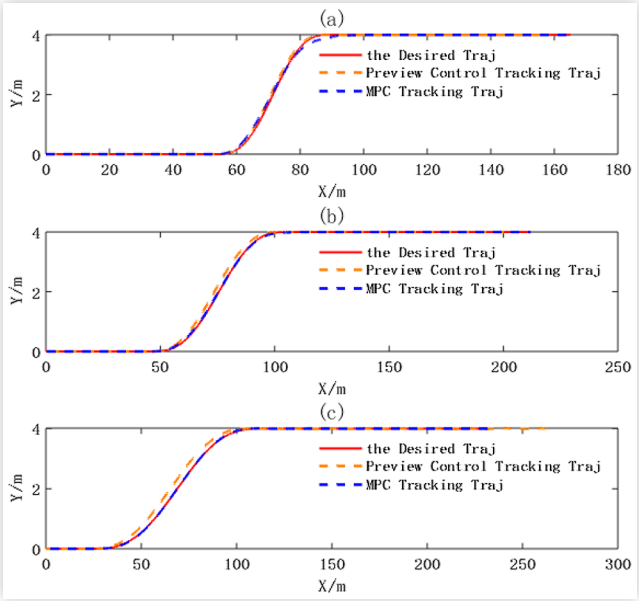


FIGURE 10 The preview control tracking errors of the direction *X* and *Y* in the inertial coordinate system. (a) is the error in the direction *X*, (b) is the error in the direction *Y*.

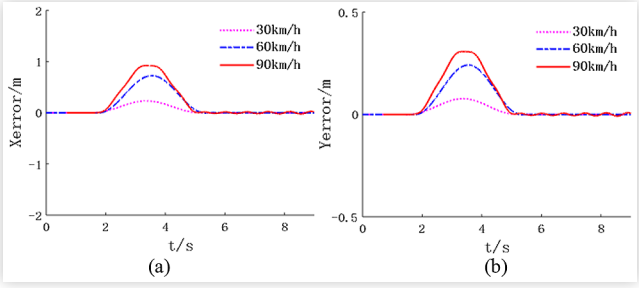
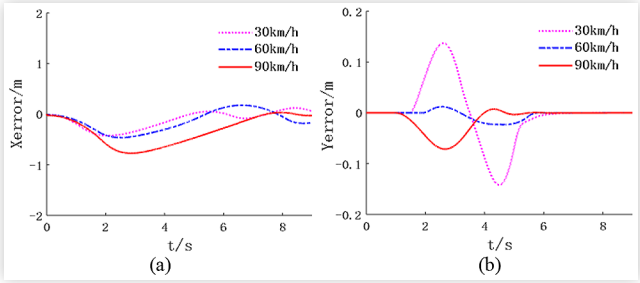


FIGURE 11 The MPC tracking errors of the direction *X* and *Y* in the inertial coordinate system. (a) is the error in the direction *X*, (b) is the error in the direction *Y*.



error of the forward preview control of single-point in the same path is plotted to compare with the algorithm designed in this paper. The method uses the distance and angle between the actual position of the vehicle and the pre-point on the desired trajectory for vehicle control on the basis of the Ackerman steering model. The steering angle is obtained by preview control as equation (32)

$$\delta = \arctan\left(\frac{2Le_p}{l_p}\right) \tag{32}$$

Where L is the wheelbase, e_p is the lateral path tracking error, and e_p is the preview distance.

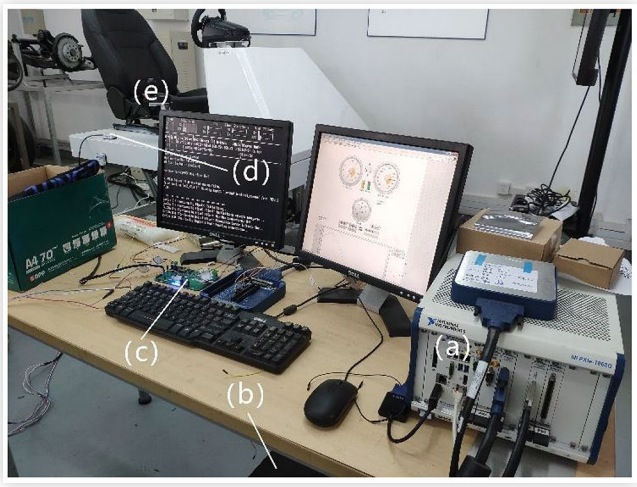
As shown in Figure 9, 10 and 11, compared with the preview control, the MPC algorithm designed in this paper can make the vehicle track the planned trajectory with smaller error at different speeds. The tracking error of the vehicle in the direction X at different speeds is within 1m, which is slightly smaller than the error of the preview control. Considering the vehicle speed, the tracking time error can be limited to within 0.1s, which has better real-time performance. The tracking error in the direction Y at different vehicle speeds is within 0.2m, which is about 1/2 of the error of the preview control. The lateral error is relatively larger at low speed, which may be due to the constraints on the rate of change of the steering angle. In short, the automatic lane change algorithm enables the vehicle to smoothly complete the lane change behavior and has the characteristics of strong robustness and high tracking accuracy.

To evaluate the performance of the proposed lane change method with respect to parameter uncertainties, the MPC controller was tested under different vehicle speeds and different tire-road friction coefficient conditions. Figure 10 and Figure 11 show that the MPC controller has good tracking accuracy at different vehicle speeds. Then variation of the tire-road friction coefficient is also included in the robustness analysis. The results show that the proposed MPC control method performs well and exhibits robustness when the tire-road friction coefficient is above 0.26.

7. HIL on the Bench

A HIL test of automatic lane change is conducted in this section. The test bench is as follows

FIGURE 12 The test bench. (a) is the NI PXI, (b) is a computer with i5 core, (c) is the signal converter, (d) is the gyroscope, (e) is the vehicle simulation platform



The lane change decision algorithm is run in the computer, and the lane change trajectory planning and tracking algorithm and the CarSim vehicle model are run in PXI. Since the PXI has a 10MHz reference clock, it can replace the actual automatic lane change controller. The trajectory tracking controller calculates the vehicle speed and the steering angle, which is input to the vehicle model of CarSim to obtain the yaw rate. PXI outputs the vehicle yaw rate as an analog signal, which is converted to CAN signal by the signal converter. Then the CAN signal is input into the vehicle simulation platform. Thus, the vehicle actual lane change trajectory is obtained according to the yaw angle of vehicle simulation platform, which is measured using a gyroscope.

The test conditions chosen for this test: the speed of the self-vehicle is 60km/h, the speed of the preceding vehicle is 30km/h, tire-road friction coefficient μ is 0.8 and the distance between the centerlines of adjacent lanes is 4m. Parameters of controller and vehicle are the same as those in the simulation.

As shown in Figure 13, the tracking effect obtained by HIL test is almost the same to that obtained by simulation, which shows that the control algorithm has strong adaptability to the actual hardware. By comparing the tracking trajectory obtained by HIL with the tracking trajectory obtained by simulation, it can be seen that the tracking trajectory of the HIL has a certain lag with respect to the desired trajectory. The reason is the tracking algorithm running in PXI requires some times to compute in real-time control. And Figure 14 shows that the trajectory error of HIL is larger than the simulated tracking error, but it is still within the error tolerance.

FIGURE 13 The lane change trajectory of desired, simulation and HIL

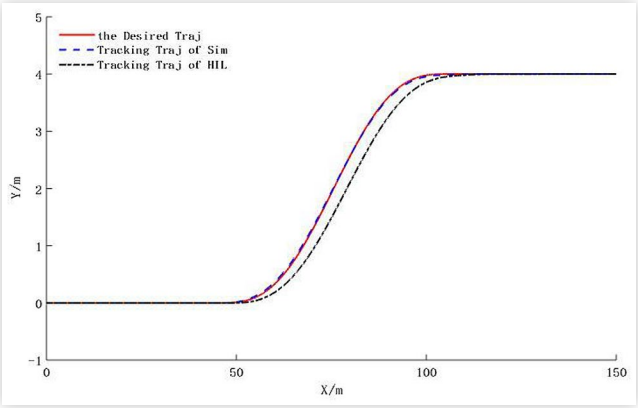
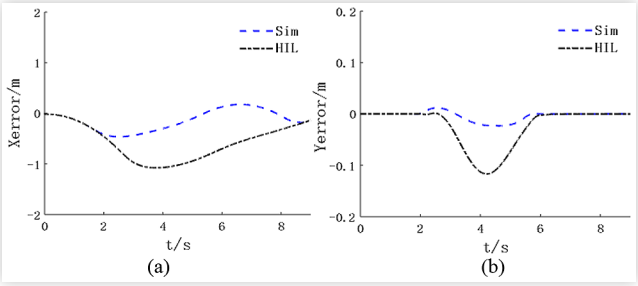


FIGURE 14 The tracking errors of the *X* and *Y* direction of simulation and HIL. (a) is the error in the direction *X*, (b) is the error in the direction *Y*.



Moreover, the computational time of the proposed MPC controller, implemented using the PXI, is 47.994ms during one sampling period, which can satisfy the real-time requirement.

8. Conclusions

In this paper, the background is the automatic lane change control system. The conditions of lane change at different speeds are selected. Conclusions through analysis are as follows

1. The lane change decision is determined by the Constant Time Headway (CTH) algorithm. Change lane when the distance between self-vehicle and preceding vehicle is less than the safe distance.
2. The constant velocity offset plus sine function lane change model is used to plan the lane change trajectory, and the lane change distance is determined according to the vehicle speed. Compared with other lane change trajectory, since the model can make vehicle maintain a small lateral

acceleration during lane change, the controller has a better robustness when the tire-road friction coefficient μ is above 0.26.

3. Model Predictive Control theory is used to track the trajectory, which optimizes tracking accuracy and vehicle stability and constrains the range and rate of change of vehicle speed and steering angle. The simulation results show that the automatic lane changing algorithm can make the vehicle complete the lane changing behavior smoothly. The tracking error in direction *Y* is about 1/2 of the error of the preview control.
4. The actual feasibility of MPC algorithm is verified through a hardware-in-the-loop test. The test results show that the error and real-time performance of the algorithm meet the requirements.

In future work, a variable speed of the lane changing vehicle would be taken into consideration. At the same time, the intelligent algorithms to solve the MPC to further improve the real-time performance of the system would be considered to be used.

References

1. Li, Y., Yang, L., and Zheng, L., "Vehicle Longitudinal and Lateral Coupling Control Based on Sliding Mode Control," *China Mechanical Engineering* 18(7):866-870, 2007.
2. Brian, P., Michal, C., Yong, S.Z. et al., "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles," *IEEE Trans. on Intelligent Vehicles* 1(1):33-55, 2016, https://doi.org/10.1109/TIV.2016.2578706.
3. Li, W., Gao, D., and Duan, J., "Research on Lane Change Model for Intelligent Vehicles," *Journal of Highway and Transportation Research and Development* 27(02):119-123, 2010.
4. Da, Y., Shiyu, Z., Cheng, W. et al., "A Dynamic Lane-Changing Trajectory Planning Model for Automated Vehicles," *Transportation Research Part C: Emerging Technologies* 228-247, 2018, https://doi.org/10.1016/j.trc.2018.06.007.
5. Yang, Z., Qi, Z., and Huang, Y., "Research on Intelligent Vehicle Free Lane Trajectory Planning," *Journal of Chongqing Jiaotong University (Natural Science Edition)* 32(03):520-524, 2013.
6. Chee, W. and Tomizuka, M., "Lane Change Maneuver of Automobiles for the Intelligent Vehicle and Highway System (IVHS)," *Proceedings of the 1994 American Control Conference* 3:3586-3587, 1994.
7. Shoutong, Y., Peng, Z., Qingyu, Z., and Xin, H., "Research on Model Predictive Control-Based Trajectory Tracking for Unmanned Vehicles," in *2019 4th International Conference on Control and Robotics Engineering (ICCRE)*, 2019, https://doi.org/10.1109/ICCRE.2019.8724158.
8. Luo, Y., Xiang, Y., Cao, K., and Li, K., "A Dynamic Automated Lane Change Maneuver Based on Vehicle-to-Vehicle Communication," *Transport. Res. Part C: Emerg. Technol.* 62(2016):87-102, 2016, https://doi.org/10.1016/j.trc.2015.11.011.
9. Ziegler, J., Bender, P., Schreiber, M. et al., "Making Bertha Drive - An Autonomous Journey on a Historic Route," *Intelligent Transportation Systems Magazine, IEEE* 6(2):8-20, 2014, https://doi.org/10.1109/MITS.2014.2306552.
10. Zhang, H. and Wang, X., "Research on Lateral Control Strategy for Automatic Lane Changing of Intelligent Vehicle," *Journal of Machine Design* 35(4):78-83, 2018, https://doi.org/10.13841/j.cnki.jxsj.2018.04.012.
11. Ronghui, Y., Feng, W., Rongben, Z., and Wenhua, X., "Lane Changing and Overtaking Control Method for Intelligent Vehicle Based on Backstepping Algorithm," *Transactions of the Chinese Society for Agricultural Machinery* 6:011, 2008.
12. Zhao, K., Guo, Q., Pei, F., and Liang, Z., "Intelligent Vehicle Trajectory Tracking Algorithm Based on Optimal Control," *Journal of Machinery and Electronics* 36(07):76-80, 2018.
13. GONG, J.-w., XU, W., JIANG, Y. et al., "Multi-Constrained Model Predictive Control for Autonomous Ground Vehicle Trajectory Tracking," *Journal of Beijing Institute of Technology* 24(04):441-448, 2015, https://doi.org/10.15918/j.jbit1004-0579.201524.0403.
14. Nobuto, S.., Hiroyuki, O.., Tatsuya, S.., et al., "Simultaneous Realization of Planning and Control for Lane-Change Behavior Using Nonlinear Model Predictive Control," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (2018), https://doi.org/10.1109/ITSC.2018.8569973.
15. Wu, G., Zhang, L., Liu, Z., and Guo, X., "Research Status and Development Trend of Vehicle Adaptive Cruise Control Systems," *Journal of Tongji University (Natural Science)* 45(04):544-553, 2017, https://doi.org/10.11908/j.issn.0253-374x.2017.04.012.
16. Gong, J., Jiang, Y., and Xu, W., *Unmanned Vehicle Model Predictive Control for Self-Driving Vehicles* (Beijing: Beijing Institute of Technology Press, 2014).
17. Kuhne, F., Lages, W.F. et al., "Model Predictive Control of a Mobile Robot Using Linearization," *Proceeding of Mechatronics and Robotics* 4:525-530, 2004.
18. Zhang, J.-H., Li, Q., and Chen, D.-P., "Drivers Imitated Multi-Objective Adaptive Cruise Control Algorithm," *Journal of Control Theory and Applications* 35(6):140-147, 2018, https://doi.org/10.7641/CTA.2017.70585.

Appendix

TABLE 1 Vehicle Parameters in CarSim

Parameters	Value
Quality	1750kg
Moment of inertia around the z axis of the vehicle coordinate system	4180 $kg \cdot m^2$
Wheelbase L	2.7m
Track	1.6m
Front tire lateral stiffness	6700 N/rad
Rear tire lateral stiffness	6300 N/rad

TABLE 2 Controller Parameters of Lane Change Trajectory Tracking

Parameters	Value
Predictive horizon N_p	60
Control horizon N_c	20
Weight matrix of state variables Γ_η	$\boldsymbol{I}_{N_pn \times N_pn}$
Weight matrix of control variables Γ_u	$5 \cdot \boldsymbol{I}_{N_cm \times N_cm}$
the penalty coefficient ρ	10
Constraint of relaxing factor ε	$0 \leq \varepsilon \leq 10$

This paper is based upon a presentation at the *SAE 2019 Intelligent & Connected Vehicles Symposium.*

ISSN 0148-7191