2014 International Conference on Informative and Cybernetics
for Computational Social Systems (ICCSS)

Intelligent Vehicle Trajectory Tracking Based on Neural Networks Sliding Mode Control*

GUO Lie, GE Ping-shu, YANG Xiao-li, LI Bing

***Abstract*—The problem of lateral control in intelligent vehicle trajectory tracking for automated highway system is studied. The article deduced the vehicle's desired yaw rate through real time planning virtual path between the vehicle mass center and prediction aiming point which is planned according to the vehicle's kinematic model and pose error model. Based on the lateral dynamic model of vehicle, radical basis function (RBF) neural networks based sliding mode variable structure trajectory tracking controller is designed. A multi-body dynamics model of vehicle is built in ADAMS/Car. The interactive combination control dynamic simulation between Matlab/Simulink and ADAMS is realized through designing the data interface between Matlab and ADAMS. Simulations were conducted and the results show that the proposed algorithm improves the control precision of the system and improves the tracking performance of the system.**

I. INTRODUCTION

To alleviate the serious traffic problem and enhance the efficiency of the transport, intelligent vehicle highway system has been proved to be one of the most important and promising solution[1]. The main research purpose on intelligent vehicle technologies is to generate warnings within a specified area of interest, which can be applied to automobile safety systems utilizing autonomous electromechanical sensors. There has been recent interest in intelligent vehicle that offer a significant enhancement of safety and convenience to drivers and passengers. To realize the highly intelligence, the vehicle should percept its status and environment surrounding and then make an appreciate decision and execute certain control command to a safety state[2].

Trajectory tracking is one of the basic ability of intelligent vehicle, aiming to complete the scheduled tasks safely and effectively. Trajectory tracking control means tracking reference trajectories scheduled or given by path planners[3]. Recently, the trajectory tracking has gained much concerns and different kind of methods and controllers have been developed to fulfill such missions. For example, Kim et al.[4] designed a torque controller for wheeled mobile robot to track the target reference trajectory by considering the kinematic model of wheeled mobile robot. Kumar et al.[5] designed a tracking controller for a four wheeled mobile robot based on Backstepping control algorithm. Guo et al.[6] realized the lane changing trajectory tracking control on curved road using the backstepping algorithm based on the kinematics model of the vehicle.

The above studied the lateral control based on the kinematics model of intelligent vehicle. However, the vehicle's lateral error which based on the position error feedback control can't tend to be zero. Recently, the researchers gradually introduced the intelligent vehicle dynamics. Wang et al.[7] designed a sliding mode variable structure trajectory tracking lateral controller based on vision guided and applied it to real vehicle to verified the stability of the controller. Zarabadipour et al.[8] .presented a mobile robot system controlled by a feedback error learning (FEL) neural network and proportional-derivative (PD) controllers. Lin and Cook[9] designed the vehicle trajectory tracking controller based on the optimal control theory, and analysis the tracking behavior in different vertical velocity. Raksincharoensak et al.[10] adopted the yawing moment control strategy to realize the lane keeping and obtained a good result.

Vehicle trajectory tracking is strongly influenced by the selection of the look-ahead point. In different road curvature and longitudinal velocity, the control effect has a big difference even in the same look-ahead distance. In order to reduce the effect of the road curvature and look-ahead distance to the lateral control and improve the effect of lateral control precision, this paper presents the vehicle's desired yaw rate through real time planning virtual path between the vehicle CG and desired position and then a RBF neural networks sliding mode variable structure trajectory tracking controller is designed based on the lateral dynamic model of vehicle. This paper is organized as follows. Section II presents the vehicle model and section III designs the controller. Simulations are conducted to verify the proposed controller in section IV. Section V concludes this paper.

II. VEHICLE MODEL

A. Vehicle Dynamics Model

To facilitate the study of the lateral motion and yawing motion in the process of trajectory tracking, the vehicle dynamic model is simplified to be a linear 2 DOF bicycle mode when travels on a small curvature and flat road, as shown in Fig. 1.

*Research supported by the National Natural Science Foundation of China (51305065) and the Fundamental Research Funds for the Central Universities (DUT13JS02)

GUO Lie is with the School of Automotive Engineering, Dalian University of Technology, Dalian 116024, China (phone: 0086-411-84706475; fax: 0086-411-84706475; e-mail: guo_lie@ dlut.edu.cn).

GE Ping-shu is with the College of Electromechanical & Information Engineering, Dalian Nationalities University, Dalian 116600, China.

YANG Xiao-li, LI Bing are with the School of Automotive Engineering, Dalian University of Technology, Dalian 116024, China.

978-1-4799-4752-2/14/$31.00 ©2014 IEEE

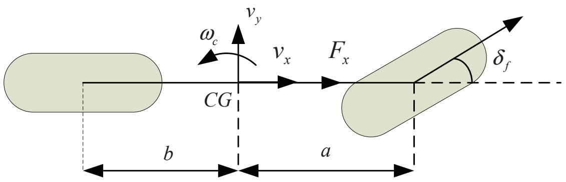


Figure 1. Vehicle linear 2DOF bicycle model

In the above model, some assumptions should be taking into account[11]. The roll, pitch and vertical motion are neglected; the normal forces acting on tires are approximated as static values; the actuator dynamics for brake, throttle and steering are discounted; the front wheel steering angle is chosen to be the input variable. Then the dynamic model can be expressed as:

$$\begin{bmatrix} \dot{v}_y \\ \dot{\omega}_c \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} \begin{bmatrix} v_y \\ \omega_c \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \delta_f \tag{1}$$

where v_y denotes the lateral velocity, ω_c denotes the yaw rate, δ_f denotes the front wheel steering angle and

$$\begin{cases} f_{11} = -\dfrac{C_f + C_r}{m v_x} \\ f_{12} = \dfrac{C_f a - C_r b}{m v_x} - v_x \\ f_{21} = \dfrac{-C_f a + C_r b}{I v_x} \\ f_{22} = -\dfrac{C_f a^2 + C_r b^2}{I v_x} \\ g_1 = \dfrac{C_f}{m} \\ g_2 = \dfrac{C_f a}{I} \end{cases} \tag{2}$$

with C_f/C_r denotes the cornering stiffness of the front/rear tire, v_x denotes the longitudinal velocity, m is the total vehicle mass, I is the yaw moment of inertia about vertical axis at centre of gravity (CG), a/b is the distance from CG to the front/rear axle. Parts of the vehicle parameters are shown in Table 1.

TABLE I. THE VEHICLE MODEL PARAMETERS USED FOR SIMULATION

Parameter	**definition**	**value**
m	total mass	2010(kg)
I	rotational inertia	2280(kg.m^2)
a/b	distance from CG to front/rear axle	1.335/1.265(m)
C_f/C_r	cornering stiffness of front/rear tire	40/40(kN/rad)

B. Vehicle Kinematic Model

The problem of trajectory tracking is to design a controller so that the vehicle can quickly follow the desired trajectory with no error. Fig.2 shows the definition of posture error to track a predefined trajectory. The position and orientation of the vehicle can be explained under the world coordinate system X-Y and the body-fixed coordinate system x_c-y_c. The CG of the vehicle in the world coordinate system is $P(X_c, Y_c)$ and the angle between the vehicle's longitudinal axis and X axis is φ_c. The kinematic model of the vehicle is defined as[12]:

$$\begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ \dot{\varphi}_c \end{bmatrix} = \begin{bmatrix} \cos\varphi_c & 0 \\ \sin\varphi_c & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} \tag{3}$$

where v_c and ω_c denotes the velocity and yaw velocity of CG, respectively.

The vehicle's posture error is not the lateral deviation or the orientation deviation of CG, but the deviation of a certain point $P(X_p, Y_p)$ in front of the vehicle, which is called the look-ahead point. The distance between the CG and the look-ahead point is defined as look-ahead distance[3]. The trajectory tracking error can be described as (x_e, y_e, φ_e), as shown in Fig. 2. The aim of tracking is to design a stable controller with certain commands. In the body-fixed coordinate system x_c-y_c, the look-ahead distance of the vehicle is x_e, the distance between the look-ahead point and the tangent of road is called lateral deviation y_e, and the angle between the center line of the vehicle and the tangent of the road is called orientation deviation φ_e. According to the geometrical relationship of the world and the body-fixed coordinate system, the error vector for tracking can be defined as:

$$p_e = \begin{bmatrix} x_e \\ y_e \\ \varphi_e \end{bmatrix} = \begin{bmatrix} \cos\varphi_c & \sin\varphi_c & 0 \\ -\sin\varphi_c & \cos\varphi_c & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_p - X_c \\ Y_p - Y_c \\ \varphi_p - \varphi_c \end{bmatrix} \tag{4}$$

where (x_p, y_p, φ_p) denotes the virtual vehicle pose.

The corresponding error derivatives are

$$\dot{p}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} \omega_c y_e - v_c + v_r \cos\theta_e \\ -\omega_c x_e + v_r \sin\theta_e \\ \omega_r - \omega_c \end{bmatrix} \tag{5}$$

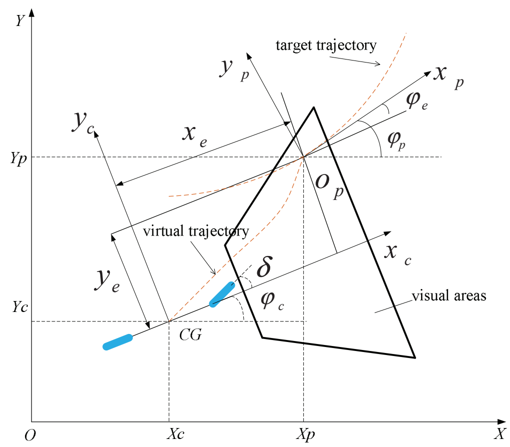


Figure 2. Posture error definition for trajectory tracking under the world coordinate system X-Y and the body-fixed coordinate system x_c-y_c

III. Controller Design

A. Trajectory Planning

In order to realize the trajectory tracking of the intelligent vehicle, this paper plan virtual path between the CG and the desired position of the vehicle in real time. Assuming the virtual path $y(x)$ is a cubic polynomial curve, which can be confined as follows[13]:

$$y = n_1 + n_2 x + n_3 x^2 + n_4 x^3 \tag{6}$$

with constraints

$$\begin{cases} y(0) = 0 \\ \dot{y}(0) = 0 \\ y(x_e) = y_e \\ \left. \dfrac{\ddot{y}}{(1+\dot{y}^2)^{3/2}} \right|_{x=0} = \psi \end{cases} \tag{7}$$

where ψ is the curvature of the vehicle traveling trajectory, $\psi = \omega_c / v_c$.

The equation of the virtual path can be expressed as follows:

$$y(x) = \frac{\omega_c}{2v_c} x^2 + \frac{y_e - \dfrac{\omega_c}{2v_c} x_e^2}{x_e^3} x^3 \tag{8}$$

The position of the vehicle is determined by the CG's velocity and yaw velocity. Therefore if the change of the vehicle CG's velocity is known, the vehicle trajectory can be determined by controlling the yaw velocity. Supposing the intelligent vehicle's CG can stably follow the trajectory curve equation $y(x)$ with no error. At one moment, when the vehicle CG's position on equation curve is (x, y), the vehicle velocity is v_c, and the direction of the velocity and the tangent of the curve are the same, that is to say, the vehicle's driving curvature equals the curvature of the curve at the position (x, y), we can get the ideal yaw rate ω as follows:

$$\omega = v_c \gamma \tag{9}$$

where γ is the road curvature.

Differentiating the curvature, yields

$$\dot{\gamma} = \frac{d\left(\dfrac{\ddot{y}}{(1+\dot{y}^2)^{3/2}}\right)}{dx} \frac{dx}{dt} = v_c \frac{\dddot{y}(1+\dot{y}^2) - 3\ddot{y}^2 \dot{y}}{(1+\dot{y}^2)^{5/2}} \tag{10}$$

Differentiating the ideal yaw rate, yields

$$\dot{\omega} = \dot{v}_c \gamma + v_c \dot{\gamma} \tag{11}$$

Then we can get the change rate of the ideal yaw rate of the vehicle which travels along virtual path.

$$\dot{\omega}\big|_{x=0} = \dot{v}_c \psi + v_c \dot{\psi} = \frac{\dot{v}_c \omega_c}{v_c} + 6v_c^2 \frac{y_e - \dfrac{\omega_c}{2v_c} x_e^2}{x_e^3} \tag{12}$$

When the vehicle tracks the target path, the change rate of the ideal yaw rate represents the change tendency of yaw rate at this moment. Then yield the desired yaw rate is

$$\omega_r = \omega_c + \alpha \dot{\omega}\big|_{x=0} = \omega_c + \alpha\left(\frac{\dot{v}_c \omega_c}{v_c} + 6v_c^2 \frac{y_e - \dfrac{\omega_c}{2v_c} x_e^2}{x_e^3}\right) \tag{13}$$

where α is the proportion coefficient.

B. Neural Networks Sliding Mode Control

In order to enhance the robustness and reduce the randomness of the control system, this paper designs a sliding mode variable structure controller based on RBF neural network. The desired yaw velocity is about to be followed stability by the designed controller through controlling the vehicle's yaw velocity, which can achieve the desired trajectory tracking.

The sliding mode switching function is defined as:

$$s_1 = \omega_c - \omega_r \tag{14}$$

where ω_c denotes the current yaw velocity of the vehicle and ω_r denotes the desired yaw velocity of the vehicle.

Differentiating s_1 in (14) and substituting (13) into it, yields the following expression:

$$\dot{s}_1 = \dot{\omega}_c - \dot{\omega}_r = f_{21} v_y + f_{22} \omega_c + g_2 \delta_f - \dot{\omega}_r \tag{15}$$

In the sliding mode controller, the control law is usually composed of the equivalent control u_{eq} and the switching control u_{sw}. The equivalent control is able to control the system state on a sliding surface. While the switching control forces the system state to slide along the sliding surface. That is to say $s\dot{s} \le 0$.

Then the equivalent control of tire steering angle can be achieved by setting $\dot{s}_1 = 0$

$$\delta_{eq} = (f_2 + \dot{\omega}_r - f_{21} v_y - f_{22} \omega_c) / g_2 \tag{16}$$

Due to the switching control is the decisive factor to the controller's chattering and the stand or fall of the switching control has vital role to the whole system, therefore we design the switch control based on RBF neural network in this paper. This paper defines the switching function s_1 and its derivative $\dot{s}_1$ as the inputs of the RBF neural network, that is to say $X = [s_1 \quad \dot{s}_1]$ is the neural network input and defines the output of the RBF neural network as the switching control:

$$\delta_{sw} = W^T H \tag{17}$$

where $H = [h_1 h_2 \ldots h_j \ldots h_n]^T$ is the network hidden layer outputs, $h_j = \exp(-\|X - C_j\|^2 / 2b_j^2)$ is the Gaussian function, $j = 1, 2 \cdots n$, n is the number of hidden layers. $W = [w_1 w_2 \cdots w_i \cdots w_n]^T$ is the matrix of neural network weights. $C_j = [c_{1j} \quad c_{2j}]^T$ is the center vector of the network node. $B = [b_1 b_2 \ldots b_n]^T$ is the base width vector of neural network.

Define the RBF neural network's learning indicators as:

$$E = s_1(t)\dot{s}_1(t) \tag{18}$$

The gradient descent method is utilized and according to principle of it, the learning algorithm of neural network weights (the weight value, the base width vector, the center vector) can be obtained as follows:

$$\begin{cases} w_j(t+1) = w_j(t) + dw_j + \beta(w_j(t) - w_j(t-1)) \\ b_j(t+1) = b_j(t) + db_j + \beta(b_j(t) - b_j(t-1)) \\ c_{ij}(t+1) = c_{ij}(t) + dc_{ij} + \beta(c_{ij}(t) - c_{ij}(t-1)) \end{cases} \tag{19}$$

with

$$db_j = -\eta \frac{\partial E}{\partial b_j(t)} = -\eta \frac{\partial s_1(t)\dot{s}_1(t)}{\partial b_j(t)} = -\eta \frac{\partial s_1(t)\dot{s}_1(t)}{\partial \delta_f(t)} \frac{\partial \delta_f(t)}{\partial b_j(t)}$$
$$= -\eta s_1(t) g_3 w_j h_j \left\| X - C_j \right\|^2 \Big/ b_j^3$$

$$dw_j = -\eta \frac{\partial E}{\partial w_j(t)} = -\eta \frac{\partial s_1(t)\dot{s}_1(t)}{\partial w_j(t)} = -\eta \frac{\partial s_1(t)\dot{s}_1(t)}{\partial \delta_f(t)} \frac{\partial \delta_f(t)}{\partial w_j(t)}$$
$$= -\eta s_1(t) \frac{\partial \dot{s}_1(t)}{\partial \delta_f(t)} \frac{\partial \delta_f(t)}{\partial w_j(t)} = -\eta s_1(t) g_3 \frac{\partial \delta_f(t)}{\partial w_j(t)}$$
$$= -\eta s_1(t) g_3 h_j$$

$$dc_{ij} = -\eta \frac{\partial E}{\partial c_{ij}(t)} = -\eta \frac{\partial s_1(t)\dot{s}_1(t)}{\partial c_{ij}(t)} = -\eta \frac{\partial s_1(t)\dot{s}_1(t)}{\partial \delta_f(t)} \frac{\partial \delta_f(t)}{\partial c_{ij}(t)}$$
$$= -\eta s_1(t) g_3 w_j h_j (x_i - c_{ij}) \Big/ b_j^3$$

where η is learning rate and $\eta > 0$. β is factor of momentum, i=1, 2.

We can get the front wheel angle control law δ_f by the equivalent control δ_{eq} and the switching control δ_{sw} which is outputted by the RBF neural network

$$\delta_f = \delta_{eq} + \delta_{sw} = (-f_2 + \dot{\omega}_r)\big/ g_2 + W^T H \tag{20}$$

IV. Simulation Results

The diagram of the co-simulation control system is shown in Fig. 3. It explains the co-simulation process between Matlab/Simulink and Adams/car. A desired trajectory module and a sliding mode controller module based on RBF neural network were built in Simulink. Adams/Car was used to establish the vehicle dynamics model which is able to control the steering system and output the state variables. The vehicle dynamics model in Adams/Car transmits the position and the speed of the vehicle into Simulink and the latter decides the best steering wheel angle and transmits it into the Adams/Car.

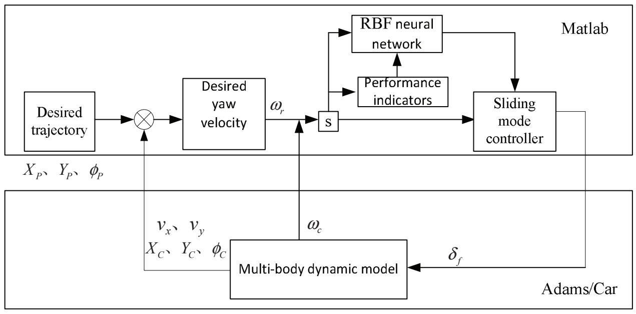


Figure 3. The co-simulation control system

The multi-dynamic model of intelligent vehicle was established in Adams/Car. The coordinates of join points and the mass and inertia of each component of the model are accord with the design parameters of the real vehicle. The vehicle model consists of the suspension system (front and rear), the steering system, the body system, the engine system, the tire and the road surface spectrum. In order to simplify the difficulty of simulation, the suspension system and the engine system were invoked from the subs

The steering system established in the Adams/Car need to make the following assumptions: all the components are rigid bodies; each kinematic pair is a rigid connection; ignore the internal clearance and friction. Then add an angle actuator on the steering wheel and set it to be the input variable of the multi-dynamic model of intelligent vehicle. In order to convert the front tire steering angle exported from the Simulink to the steering wheel angle which is the input of the Adams, This article define the steering system's angular velocity ratio (the increment of the steering wheel angle/ the increment of the front tire steering angle) as twelve.

Due to the fact that this paper doesn't consider the air resistance, the car body system is simplified to the mass and the rotational inertia of the vehicle CG. Establish the solver of the vehicle position and speed and set them to be the output variables of the vehicle model in the Adams/Car. Tire model is one of the most important parts for the multi-body dynamic model. Its structure parameters and mechanical characteristics determine the vehicle's driving performance. Because the UA tire model considers the effect of unsteady and has a good accuracy under the condition of limited several parameters, this paper chooses the UA tire model as the tire system and changes part of its characteristic parameters.

An auto runway that consists of two straight lines and a five times polynomial curve is defined as the simulation path and establishes the road surface spectrum model based on the road builder of Adams/Car.

The look-ahead distance x_e was set to be 5m and the proportion coefficient α was set to be 0.05. The number of hidden layers n was 4 and the networks learning parameters η and β were set to be 0.6 and 0.05 respectively. The initial values of neural network weights (the weight vector, the base width vector, the center vector) were set as follows: w=[0.25 0.25 0.25 0.25]^T, C_j=rands(2,4) and B=[0.05 0.05 0.05 0.05]^T. The longitudinal velocity of the vehicle in the simulation process is 8m/s. The simulation results are shown in Fig.4-7.

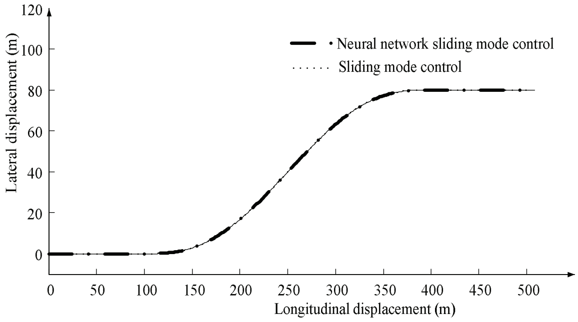
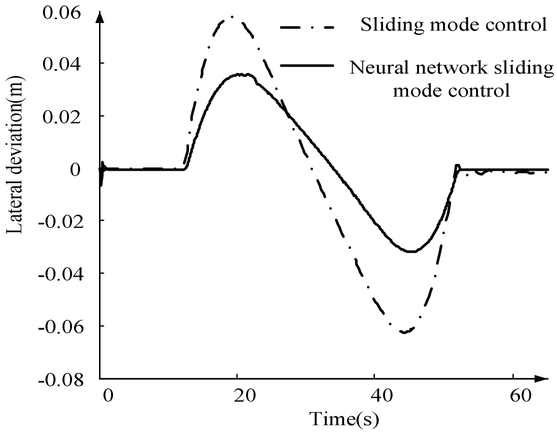
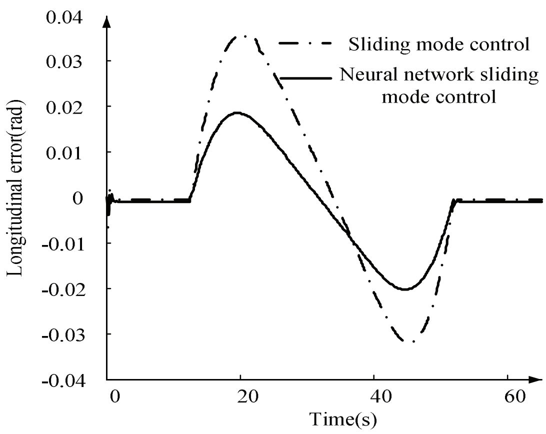


Figure 4. Trajectory tracking comparison



(a) Lateral error



(b) Longitudinal error

Figure 5. Vehicle lateral and longitudinal states tracking error

Fig.4 compares the trajectory tracking results using the designed lateral controller and Fig.5 shows the tracking error states. The lateral deviation curves are shown in Fig.5 (a). The lateral deviations based on the two methods all appear certain fluctuations, but the lateral deviation based on this paper's method always stay within the error of $\pm 0.04m$ during the whole simulation process. Compared with the result based on the sliding mode control method the lateral deviation reduces about one times; the orientation deviation curves are shown in Fig.5 (b). The orientation deviation based on this paper's method is significantly smaller than the orientation deviation based on the sliding mode control method.

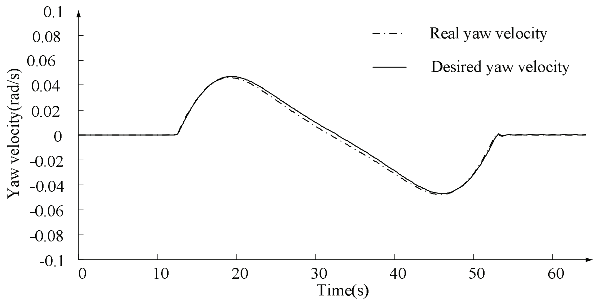


Figure 6. Vehicle's yaw rate and desired value

Fig.6 shows the duration curve of the real yaw velocity and the desired yaw velocity of the vehicle. We can find that when the desired curve's curvature is 0, the desired yaw rate converges to zero fast, when the desired curve's curvature changes, the yaw rate changes quickly too, and the vehicle's real yaw rate can better approaching the desired yaw rate, which shows the vehicle has better handling stability and safety.

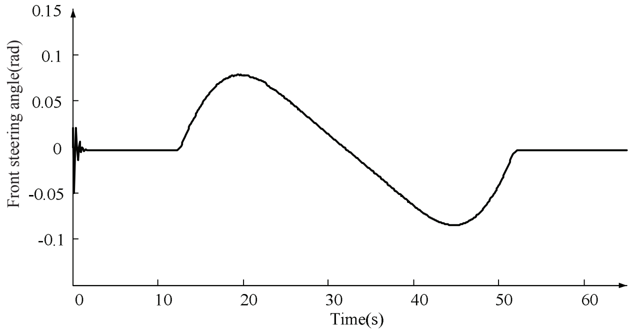


Figure 7. The input variable of the controller

Fig.7 shows the control input (the front steering angle) of the vehicle during the process of trajectory tracking. Due to the transformation of the curvature during the time period 12-52s, the front tire steering angle changes from positive to negative. And during the whole simulation process, the changing curve of the front tire steering angle is smooth and does not appear larger dithering.

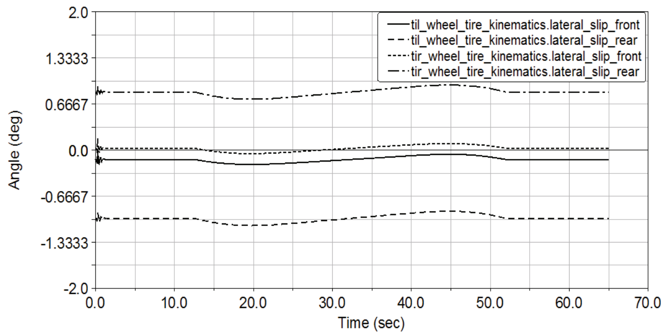


Figure 8. Lateral slip angle of tires along the desired path

Fig.8 shows the lateral slip angle changing curve of the vehicle during the process of trajectory tracking. The tire does not appear too big lateral slip angle during the whole simulation process, which meets the assumption that the tire lateral force is in direct proportion to the lateral slip angle under the small lateral slip angle. As can be seen from the Fig.5-8, the proposed algorithm can correct the deviation quickly and track the trajectory with different curvature accurately.

V. CONCLUSION

This paper has presented a trajectory tracking control algorithm for an intelligent vehicle. The vehicle's desired yaw rate was deduced by planning virtual path between the vehicle CG and the desired position. A RBF neural networks based sliding mode variable structure controller was designed considering the lateral dynamic model of the vehicle. This algorithm makes the longitudinal axis of the intelligent vehicle tend to the tangential direction of the desired trajectory by controlling the vehicle yaw rate. A multi-body dynamics model of the vehicle was built in Adams/Car. A co-simulation platform was built combining Adams/Car and Matlab/Simulink to adjust the control parameters. Simulation results show that the proposed controller is able to the precision and performance of the system.

REFERENCES

[1] L.D. Baskar, B. De Schutter, J. Hellendoorn and Z. Papp, "Traffic control and intelligent vehicle highway systems: A survey," *IET Intelligent Transport Systems*, 2011, 5(1):38-52.

[2] S. Hima, B. Lusseti, B. Vanholme, S. Glaser and S. Mammar, "Trajectory Tracking for Highly Automated Passenger Vehicles," in *the 18th IFAC World Congress*, Milano Italy, Aug.28 – Sep.2, 2011, pp.12958-12963.

[3] R. Solea and U. Nunes, "Trajectory planning and sliding-mode control based trajectory tracking for cybercars," *Integrated Computer-aided Engineerin*g, 2007, 14(1):33-47.

[4] S. Kim, K. Seo and Y. Cho, "A trajectory tracking control of wheeled mobile robot using a model reference adaptive fuzzy controller," *Journal of Institute of Control, Robotics and Systems*, 2009.

[5] U. Kumar and N. Sukavanam, "Backstepping based trajectory tracking control of a four wheeled mobile robot," *International Journal of Advanced Robotic Systems*, 2008, 5(4):403-410

[6] L. Guo, P.S. Ge, M. Yue and Y.B. Zhao, "Lane Changing Trajectory Planning and Tracking Controller Design for Intelligent Vehicle Running on Curved Road," *Mathematical Problems in Engineering*, vol. 2014, Article ID 478573, 9 pages, 2014.

[7] J. E. Wang, W. W. Chen and T. B. Wang, "Vision guided intelligent vehicle lateral control based on desired yaw rate," *Journal of Mechanical Engineering*, 2012, 48(4):108-115.

[8] H. Zarabadipour, Z. Yaghoubi and M. Aliyari Shoorehdeli, "Mobile robot control based on neural network and feedback error learning approach," *International Journal on Technical and Physical Problems of Engineering*, 2013, 5(3):91-95.

[9] Y. S. Lin and G. Cook, "Automobile steering autopilot with lane changing capability," In *Proc 23rd International Conference on Industrial Electronics, Control and Instrumentation*, New Orleans, LA, USA, 1997, 1:143-148.

[10] P. Raksincharoensak, M. Nagai and M. Shino, "Lane keeping control strategy with direct yaw moment control input by considering dynamics of electric vehicle," *Vehicle System Dynamics*,2006(44):192-201.

[11] R. Solea, A. Filipescu and U. Nunes, "Sliding-mode control for trajectory-tracking of a wheeled mobile robot in presence of uncertainties," in *Proc. of the 7th Asian Control Conference*, Hong Kong, 2009, pp. 1701-1706.

[12] Z.H. Zhang and Y.S. Xiong, "Real-Time Trajectory Tracking of a Wheeled Mobile Robot with Kinematic and Dynamic Constraints," D. Jin and S. Lin (Eds.): *Advances in Mechanical and Electronic Engineering*, LNEE 176, pp. 165–170. Springer-Verlag Berlin Heidelberg 2012.

[13] L.Guo, B.Li, P.S.Ge, R.R.Ding and Y.B.Zhao. Longitudinal and lateral coupling control trajectory tracking algorithm for intelligent vehicle. *Journal of Applied Sciences*, 2013, 13(18):3782-3789.