2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Development of BP Neural Network PID Controller and Its Application on Autonomous Emergency Braking System

Liuhui Wang[1], Zhenfei Zhan[2], Xin Yang[3], Qingmiao Wang[4], Yufeng Zhang[5], Ling Zheng[6], Gang Guo[7]

*Chongqing University, the State Key Lab of Mechanical Transmission, China

Abstract—**Rear-end collision is one of the most common collision modes in China, which often leads to severe accident consequences. Autonomous Emergency Braking (AEB) system which can avoid or mitigate rear-end collision is one of the Advanced Driver Assistance System (ADAS) technologies. Traditional PID controller cannot effectively control the AEB system with strong nonlinear characteristics. Therefore, Back Propagation (BP) neural network PID controller is proposed in this paper. The PID parameters can be adjusted in real time based on the self-learning property and self-adapting property of BP neural network. The dynamics model is built in CarSim, and the inverse dynamics model is built in Simulink. Through the coordination control of the throttle angle and brake pressure, the host vehicle can brake automatically to avoid collisions in case of emergency. In addition, three kinds of test scenarios for the target car, stationary, slight braking, emergency braking, are setup based on complex environment in China. Finally, the simulations are conducted in these scenarios. And the simulation results indicate the feasibility and effectiveness of BP neural network PID controller in AEB system.**

I. INTRODUCTION

In China, the vehicle volume has increased rapidly in recent years. According to the National Bureau of Statistics of China, there are 58,523 people died of traffic accidents injuries in 2014. China has become one of the highest traffic fatality rate countries in the world [1, 2]. As an effective solution to alleviate serious crashes and accidents, Advanced Driver Assistive System (ADAS) devices are equipped in more and more passenger and commercial vehicles [3, 4]. Autonomous Emergency Braking (AEB) system is one of the most popular techniques in the field of ADAS. AEB is able to alert the drivers, and it can be also automatically activated to avoid or mitigate the accidental injuries in rear-end collision situations, which can compensate the drivers for their slow or insufficient braking operations [4-6]. Euro-NCAP and Australia-NCAP show that that AEB system can help to avoid about 38% of the rear-end accidents [7]. Many automobile makers, such as Volvo, Mercedes, Audi, Lexus, et, are equipping AEB systems in lots of their models. It is expected that in the near future, the AEB system will become a standardized feature in vehicles.

A number of public domain assessment program, such as Euro-NCAP, ANCAP, IIHS, J-NCAP, ECE, etc, have published their testing procedures for AEB [8]. The evaluation of AEB system performance is included in 2014 Euro-NCAP safety rating. Models without AEB system will not reach the five star rating [9]. As the driving environments vary in different countries, the testing methods of AEB should match with the actual situations. For example, the evaluation of AEB in Euro-NCAP is divided into three working conditions, including AEB City, AEB Inter Urban and AEB Pedestrian. For AEB Inter-Urban, the system is tested in three scenarios (Car to Car Rear Stationary (CCRs), Car to Car Rear Moving (CCRm) and Car to Car Rear Braking (CCRb)). Compared with Euro-NCAP, the test method of IIHS only includes CCRs. The traffic accident data shows that there is few traffic accidents caused by emergency braking of target vehicles in Japan. So there is no CCRb in J-NCAP [8]. Up to date, the standardized AEB testing procedure for China market is not established yet. Hence, it would be beneficial to conduct research aimed to evaluation the AEB system under China's real-world driving conditions.

The PID controller is a widely used technique in AEB systems because of its advantages of simple structure and strong robustness. However, due to the complicated nonlinear characteristics of control systems, and the time-varying characteristics of the real-world driving process, the conventional PID controller cannot control the AEB systems effectively [10-12]. BP neural network is a multilayer feed-forward network trained by the error back-propagation algorithm [13]. As there are obvious advantages in distributed storage, self-organizing, self-adaption and self-learning, the BP neural network can adjust the PID parameters in real time [14-16]. In this paper, the preview control idea is incorporated, and it is optimized by considering different driving scenarios.

The rest of the paper is organized as follows: The framework of BP neural network PID controller is briefly introduced, followed by the adjustment method of PID parameters. Next, three kinds of test scenarios for the target car, stationary, slight braking, emergency braking, are setup based on complex environment in China and drivers' driving habits. Then the vehicle longitudinal dynamics model is designed in MATLAB/Simulink and CarSim, and the simulations under typical working conditions are conducted to investigate the performance of the proposed controller. Finally, some conclusions are given.

II. PROPOSED BP NEURAL NETWORK PID CONTROL

In this section, a 3-layer forward BP neural network is designed, including three input neurons, five hidden neurons and three output neurons. The diagram of BP neural network PID controller is shown in figure 1. It includes the general PID controller and the BP neural network. The forward BP neural network can adjust the PID controller parameters k_p, k_i, k_d in real time.

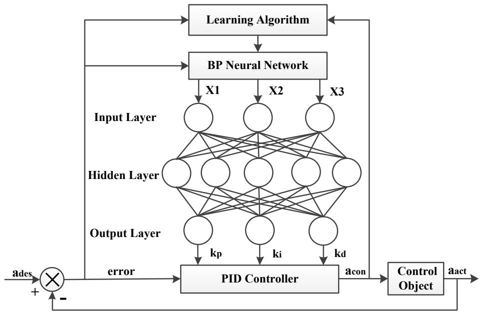


Figure 1. The diagram of BP neural network PID controller

Where a_{des} is the desired acceleration, a_{con} is the control acceleration of the inverse dynamics model, and a_{act} is the actual acceleration of the host car.

The algorithm procedure based on BP neural network PID can be described as in figure 2. It consists of several major steps: (1) Initialization. Determine the node number of input neurons, hidden neurons and output neurons, then select the learning rate η and inertial coefficient α; (2) Sample input the parameters $a_{des}(k)$ and $a_{act}(k)$, and calculate error(k)= $a_{des}(k)$ - $a_{act}(k)$; (3) Calculate the input and output of neuron at each layer; (4) Calculate the output $a_{con}(k)$ of the controller; (5) Adjust the weight and threshold of BP neural network online; (6) Let k=k+1, return to (2), until the iteration meets the requirements.

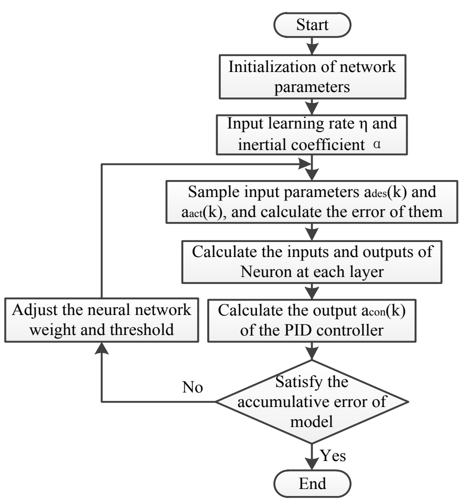


Figure 2. The flowchart of the BP neural network PID controller

The input variable of PID controller is

$$O_i^{(1)} = x(\mathrm{i}) \qquad i = 1,2,3 \tag{1}$$

$$\begin{cases} x(1) = error(k) - error(k-1) \\ x(2) = error(k) \\ x(3) = error(k) - 2error(k-1) + error(k-2) \end{cases} \tag{2}$$

The input of network hidden layer is

$$net_i^{(2)}(\mathrm{k}) = \sum_{j=1}^{3} w_{ij}^{(2)} O_j^{(3)} \quad i = 1,2,3,4,5 \tag{3}$$

The activation function of hidden layer takes hyperbolic tangent function which can continuously expand the feature effect in the loop process.

$$f(\mathrm{x}) = \tanh(\mathrm{x}) = \frac{e^x - e^{-x}}{e^x + e^x} \tag{4}$$

The output of hidden layer is

$$O_{\mathrm{i}}^{(2)}(\mathrm{k}) = f(\mathrm{net}_i^{(2)}(\mathrm{k})) \quad \mathrm{i} = 1,2,3,4,5 \tag{5}$$

The input of output layer is

$$net_l^{(3)}(\mathrm{k}) = \sum_{i=0}^{5} w_{li}^{(3)} O_i^{(2)} \quad l = 1,2,3 \tag{6}$$

When the activation function is taken the sigmoid function, the data does not easily diverged in the process of transmission. In addition, because the PID parameters cannot be negative, the non-negative sigmoid function is taken for the activation function of output neurons.

$$g(\mathrm{x}) = \frac{1}{2}(1 + \tanh) = \frac{e^x}{e^x + e^{-x}} \tag{7}$$

The output of output layer is

$$\begin{cases} O_1^{(3)}(\mathrm{k}) = \dfrac{e^{net_1^{(3)}}}{e^{net_1^{(3)}} + e^{-net_1^{(3)}}} = k_p \\ O_2^{(3)}(\mathrm{k}) = \dfrac{e^{net_2^{(3)}}}{e^{net_2^{(3)}} + e^{-net_2^{(3)}}} = k_i \\ O_3^{(3)}(\mathrm{k}) = \dfrac{e^{net_2^{(3)}}}{e^{net_2^{(3)}} + e^{-net_2^{(3)}}} = k_{\mathrm{d}} \end{cases} \tag{8}$$

The performance index function of system is the following function.

$$E(\mathrm{k}) = \frac{1}{2}(\mathrm{a_{des}}(\mathrm{k}) - \mathrm{a_{act}}(\mathrm{k}))^2 \tag{9}$$

The error of the traditional BP network is adjusted along the direction of steepest descent. The direction is the negative gradient direction [9]. The BP algorithm only reflects the local characteristics of error function at some point. When the iteration point is close to a minimal value, both the search step length and the velocity of convergence decrease. In order to avoid the local minimum of network learning, an inertia item which makes search fast convergence and global minimum is added.

$$\Delta w_{li}^{(3)}(\mathrm{k}) = -\eta \frac{\partial E(\mathrm{k})}{\partial w_{li}^{(3)}} + \alpha \Delta w_{li}^{(3)}(\mathrm{k}-1) \tag{10}$$

Where η is the learning rate, α is the inertial coefficient.

$$\frac{\partial E(\mathrm{k})}{\partial w_{li}^{(3)}} = \frac{\partial E(\mathrm{k})}{\partial a_{act}(\mathrm{k})} \bullet \frac{\partial a_{act}(\mathrm{k})}{\partial \Delta u(\mathrm{k})} \bullet \frac{\partial \Delta u(\mathrm{k})}{\partial O_l^{3}(\mathrm{k})} \bullet \frac{\partial O_l^{3}(\mathrm{k})}{\partial net_l^{(3)}} \bullet \frac{\partial net_l^{(3)}}{\partial w_{li}^{(3)}} \tag{11}$$

$$\begin{cases} \dfrac{\partial net_l^{(3)}}{\partial w_{li}^{(3)}} = O_i^{2}(\mathrm{k}) \\ \dfrac{\partial u(\mathrm{k})}{\partial O_1^{(3)}(\mathrm{k})} = error(\mathrm{k}) - error(\mathrm{k}-1) \\ \dfrac{\partial u(\mathrm{k})}{\partial O_2^{(3)}(\mathrm{k})} = error(\mathrm{k}) \\ \dfrac{\partial u(\mathrm{k})}{\partial O_3^{(3)}(\mathrm{k})} = error(\mathrm{k}) - 2\,\mathrm{error}(\mathrm{k}-1) + \mathrm{error}(\mathrm{k}-2) \end{cases} \tag{12}$$

The learning algorithm of network output layer is shown in Equation 13-14.

$$\delta_l^{(3)} = \mathrm{error}(\mathrm{k})\,\mathrm{sgn}\left(\frac{\partial a_{act}(\mathrm{k})}{\partial u(\mathrm{k})}\right) \frac{\partial u(\mathrm{k})}{\partial O_l^{(3)}(\mathrm{k})} g'(\mathrm{net}_l^{(3)}(\mathrm{k}))$$

$$l = 1,\ 2,\ 3 \tag{13}$$

$$\Delta w_{li}^{(3)}(\mathrm{k}) = \eta \delta_l^{(3)} O_l^{(2)}(\mathrm{k}) + \alpha \Delta w_{li}^{(3)}(\mathrm{k}-1) \quad i = 1,2,3,4,5 \tag{14}$$

Similarly, the learning algorithm of network hidden layer is shown in Equation 15.

$$\delta_i^{(2)} = \sum_{l=1}^{3} \delta_l^{3} w_{li}^{(3)} f'(\mathrm{net}_i^{(2)}(\mathrm{k})) \quad i = 1,2,3,4,5 \tag{15}$$

$$\Delta w_{ij}^{(2)}(\mathrm{k}) = \eta \delta_i^{(2)} O_i^{(1)}(\mathrm{k}) + \alpha \Delta w_{ij}^{(2)}(\mathrm{k}-1) \quad j = 1,2,3 \tag{16}$$

The control acceleration of the inverse dynamics model can be gained through this control algorithm.

$$\begin{aligned} a_{con}(k) &= a_{con}(k-1) + k_p[\mathrm{error}(\mathrm{k}) - error(\mathrm{k}-1)] + k_i\,\mathrm{error}(\mathrm{k}) \\ &+ k_d[error(\mathrm{k}) - 2error(\mathrm{k}-1) + error(k-2)] \end{aligned} \tag{17}$$

III. CASE STUDY

A. Test Scenario Setup

The test scenarios are setup in CarSim. CarSim which can be used to simulate and analyze vehicle dynamic responses is developed by the Mechanical Simulation Corporation [5]. As shown in the figure 3, the host car (car 2) which follows the target car (car 1) is driving on the road. The E-class sedan model is selected to verify the proposed controller in this paper. This model has front independent suspensions, rear solid axles, 14 multi-body degrees of freedom, and 54 state variables. The host car is E-class sedan with a length of 5.3 m, and the target car is the large European van with a length of 3.7 m. In order to avoid the rear-end collisions, the relative distance between two car centroids is at least 4.7 m.

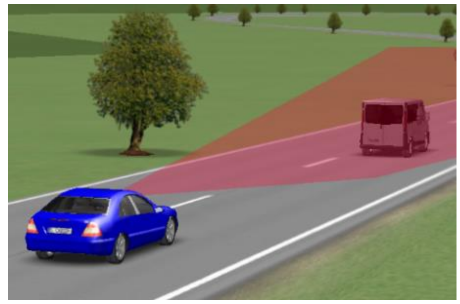


Figure 3. Simulation scenarios in CarSim

Three kinds of test scenarios for target vehicle, stationary, slight braking, emergency braking, are setup based on the AEB test scenarios of 2018 edition C-NCAP draft and several domestic automobile factories [8, 17]. These testing scenarios are most common in rear-end collision. The parameters which are set according to traffic accident statistics of China are shown in Table 1. Where v_1 is the velocity of host car, v_2 is the velocity of target car, a_2 is the acceleration of target car, and D is the relative distance between two car centroids.

TABLE I. TEST SCENARIO PARAMETERS

Test number	v_1/ km·h^{-1}	v_2 /km·h^{-1}	a_2/m·s^{-2}	D/m
B1-1	50	50	-4	40
B1-2	50	50	-4	20
B2-1	20	0	0	40
B2-2	30	0	0	40
B2-3	40	0	0	40
B3-1	35	20	-2	40
B3-2	45	20	-2	40
B3-3	65	20	-2	40

B. AEB System Model

In this section, AEB system model is designed to illustrate the flexibility of the BP neural network PID controller. The vehicle dynamics model is provided in CarSim, and the inverse dynamics model is built in Simulink. The schematic diagram of AEB system model can be described as figure 4.

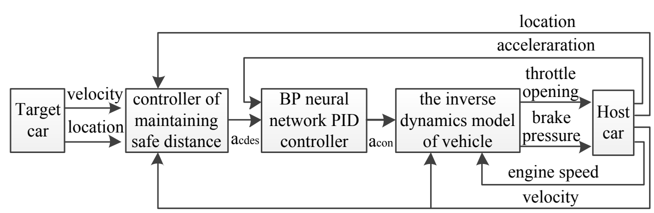


Figure 4. The schematic diagram of AEB system model

The controller of maintaining safe distance is designed based on braking distance algorithm of Mazda. The critical braking distance is shown in equation 18.

$$\mathrm{d}_{br} = \frac{1}{2}\left(\frac{v_1^2}{a_1} - \frac{v_2^2}{a_2}\right) + v_1\tau_1 + v_{rel}\tau_2 + d_0 \tag{18}$$

Where a_1 is the maximum deceleration of host car, a_2 is the maximum deceleration of target car, v_{rel} is the relative velocity of two cars, τ_1 and τ_2 are the delay time of system and driver respectively, and d_0 is the relative distance of two car after parking. d_0 is set to 10 m in this paper. When the actual distance between two cars is longer than d_{br}, a_{des} is equal to 0.5, otherwise it is equal to -8 $m \cdot s^{-2}$.

The inverse dynamics model is referred to the model established by Li and Wang [18, 19]. The desired brake pressure can be calculated through the following formula.

$$P_{des} = \frac{\left| -m a_{cdes} - 1/2 C_D A \rho v^2 - fmg \right|}{K_b} \quad (19)$$

Where CD is the air drag coefficient, A is the windward area, ρ is the air density coefficient, f is the rolling resistance coefficient, m is the mass of a whole car, and Kb is the ratio of braking force to braking pressure. The coefficient Kb can be calculated, and it is equal to 1466.7 in this paper.

The desired engine torque T_{des} can be obtained through the following formula.

$$T_{des} = \frac{m a_{des} + \sum F(\mathrm{v})}{k_d} \quad (20)$$

Where $\Sigma F(v)$ is the sum of the resistance of vehicle, and k_d is a variable that can be observed in real time.

As shown in figure 5, the torque characteristic of the engine is non-linear. And the engine torque characteristic curves can be get in CarSim. The throttle opening δ_{des} can be obtained according to T_{des} and engine speed we,

$$\delta_{des} = f(T_{des}, w_e) \quad (21)$$

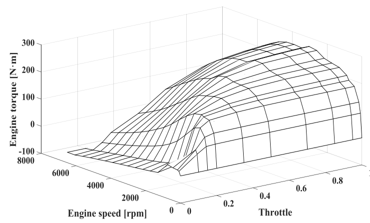


Figure 5. Engine torque characteristic curves

The co-simulation model of AEB system is shown in figure 6.

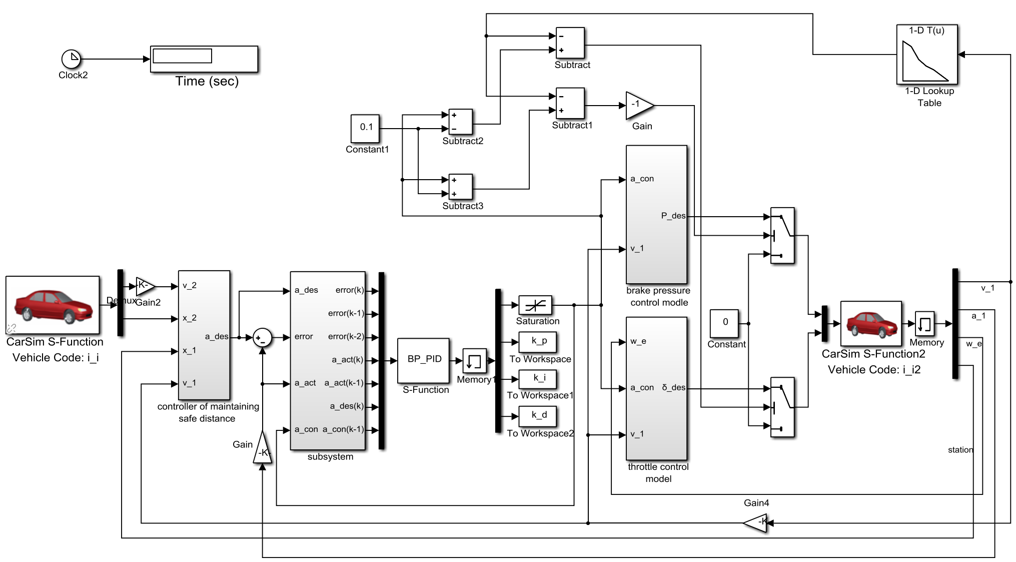
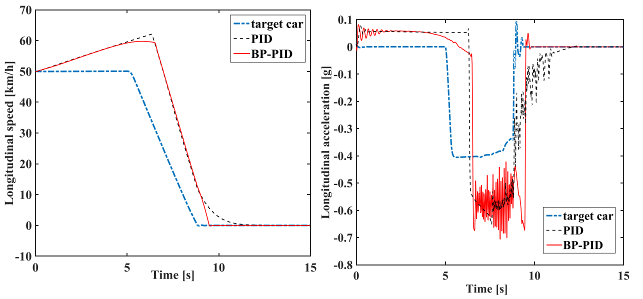


Figure 6. Co-simulation model of AEB system

IV. SIMULATION RESULTS

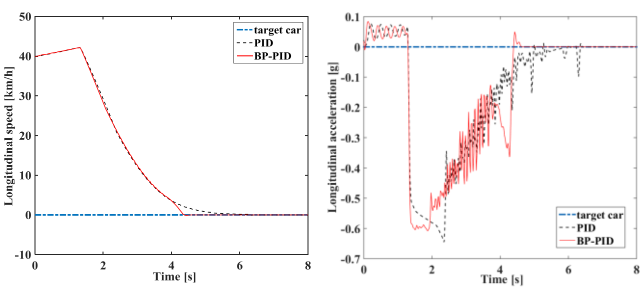
Based on the testing scenarios described above, this section compares the simulation results of BP neural network PID controller with results of conventional PID controller. Figures 7 to 9 show the speed and acceleration profiles for three different driving scenarios. The conventional PID controller parameters, k_p, k_i, k_d , are manually adjusted to the appropriate values as 1, 0.5 and 0 respectively. Figure 7 shows the speed and acceleration profiles for B1-1 driving scenario. The host car starts at a speed of 50km/h from the origin. The target car starts at a speed of 50km/h, and it is 40 meters away from the origin. When the driving time reaches 5 s, the target car starts braking with the deceleration of 2 m·s-2. The host car with the BP neural network PID controller stops at 9.8 s, while the host car with the conventional PID controller stops at 12 s. Although the a_{des} of two controllers is the same, the a_{con} is different after the controller. BP neural network PID controller can make the a_{con} achieve an ideal value by adjusting the parameters online, so the host car with the BP neural network PID controller starts to slow down earlier. When it is getting close to the target car, the velocity of host car can be adjusted automatically. The advanced AEB system with BP neural network PID controller exhibits a better performance. The required deceleration to stop the vehicle is about 5.8 $m \cdot s^{-2}$, which belongs to moderate braking range and makes the driver feel temporarily uncomfortable. But the deceleration will decrease rapidly, and the comfort level will improve.



(a) Longitudinal speed profile (b) Longitudinal acceleration profile

Figure 7. Longitudinal speed and acceleration profiles for the target car and the host car (speed ~50 km/h)

Figure 8 shows the speed and acceleration profiles for B2-3 driving scenario. The target car maintains stationary at a distance of 40 meters from the origin. The host car starts at a speed of 40km/h from the origin. At the beginning, the performances of the two controllers are similar. However, when the car is about to stop, the advanced AEB system also exhibits a better performance. The host car with the BP neural network PID controller stops at 4.6 s, while the host car with the PID controller stops at 6.4 s. Meanwhile, the peak acceleration of the host car with BP neural network PID controller is about 5.8 $m{\cdot}s^{-2}$, and the deceleration will decrease rapidly.

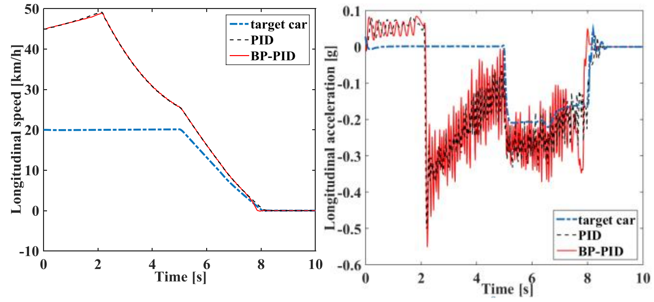


(a) Longitudinal speed profile (b) Longitudinal acceleration profile

Figure 8. Longitudinal speed and acceleration profiles for the target car and the host car (speed ~40 km/h)

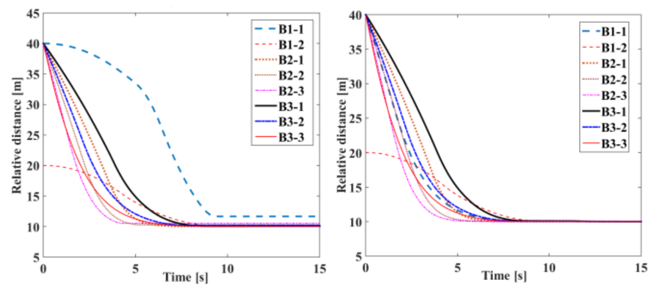
Figure 9 shows the speed and acceleration profiles for B3-2 driving scenario. The target car starts at a speed of 50 km/h, and it is 40 meters away from the origin. When the driving time reaches 5 s, the target brakes slightly, and deceleration is 2 $m{\cdot}s^{-2}$. The host car starts at a speed of 45 km/h from the origin. The host car with the BP neural network PID controller stops at 8.5 s, and the host car with the PID controller stops at 8.75 s. The fluctuation range of the acceleration of the host car with conventional PID controller is less than the host car with BP neural network PID controller. Thus, as long as the parameters of PID controller are adjusted to appropriate level, an ideal brake performance of the host car will be achieved.

Figure 10 shows the relative distance profiles for the host car with different controllers in all test scenarios described above. As shown in the figure 10 (a), all relative distances between two car centroid are about 10 m after parking, which are greater than the minimum safe distance. BP neural network PID controller shows a great performance for the rear-end collision avoidance. As shown in the figure 10 (b), all relative distances are longer than the minimum safe distance when the conventional PID controller parameters are manually adjusted to the appropriate values. Therefore, it also shows a great braking performance.



(a) Longitudinal speed profile (b) Longitudinal acceleration profile

Figure 9. Longitudinal speed and acceleration profiles for the host car and target car (speed ~45 km/h)



(a) (b)

Figure 10. The relative distance profiles for the host car with different controllers (a) BP neural network PID controller (b) PID controller

However, if conventional PID controller parameters were not selected properly, the rear-end collision would not be avoided. For example, when they are set to 0.5, 3 and 0.5 respectively, the relative distance profiles for the host car equipped with conventional PID controllers are shown in figure 11. The value of baseline is 4.7 m, which is the minimum safe distance. If the relative distance between two cars is shorter than 4.7 m, the rear-end collision will not be avoided.

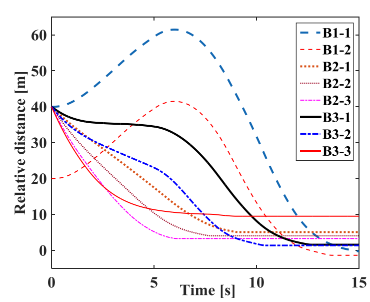


Figure 11. The relative distance profiles for the host car equipped with conventional PID controller

The AEB system shows a great performance for rear-end collision avoidance in the B2-1 and B3-3 test scenarios, but the rear-end collisions occur in all other test scenarios. Therefore, it is necessary to manually set appropriate parameter values to achieve great performance in different test scenarios. The parameters of the PID controller are selected from a large number of values. In addition, when the parameter are set, they will not be automatically changed to adapt to other working conditions. However, BP neural network PID controller can automatically adjust the controller parameters in real time to adapt to various working conditions. Therefore, BP neural network controller is more flexible than PID controller.

Three parameters have different effects on the control system with the PID controller. With the increase of k_p, the response speed of the system will increase. When k_p is too large, the overshoot will raise. The system will be oscillating, which will influence the stability. When ki increases, the overshoot will decrease, and the stability increases. The time of eliminating the static difference is longer. When k_d increases, the response speed of the system will increases. The overshoot will decrease, and the stability will increase. But the ability to suppress the disturbance is weakened. In this paper, we choose a scene to analyze the effect of controller parameters variation. For example, k_p and k_i have a little influence on the relative distance between two cars, but kd has a significant effect in B1-1 driving scenario. Figure 12 shows the relative distance profiles for the different k_d. The relative distance between two cars is reasonable when k_d ranges from 0 to 0.2. When k_d increases, the relative distance is shorter than 4.7 m. The rear-end collision will not be avoided. The baking performance of the host car is poor.

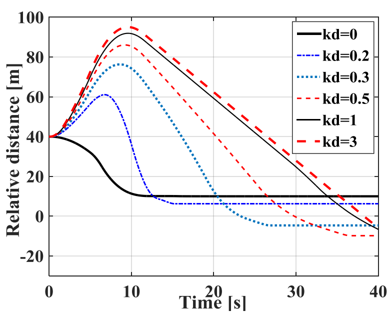


Figure 12. The relative distance profiles for the different k_d.

V. CONCLUSION

This paper proposed BP neural network PID controller based on AEB development method and process for traffic conditions in China. The proposed controller is robust to external disturbance and model uncertainty. For the controller, the BP neural network has the approximation ability to any nonlinear function, which can be used to achieve the path tracking for emergency braking. Three kinds of test scenarios for target vehicle, stationary, slight braking, emergency braking, are setup based on the AEB test scenarios of 2018 edition C-NCAP draft and several domestic automobile factories. The simulation results validated the accuracy and flexibility of the BP neural network PID controller. Due to self-tuning and optimization of the parameters, the performance of the proposed controller is more prominent than traditional PID controller in the braking time. Future work will further improve and perfect this method. Meanwhile, more complex scenarios will be setup to test the effectiveness of the controller.

ACKNOWLEDGMENT

This work was supported by Chongqing foundation and advanced research project (cstc2015jcyjBX0097), Chongqing key industry common key technology innovation project (cstc2015zdcyztzx30001), and the National Key Research and Development Program of China (Grant No. 2016YFB0100904).

REFERENCES

[1] State Statistics Bureau. http://www.stats.gov.cn.

[2] Zhao, M., Wang, H., Chen, J., Xu, X. et al., "Method to Optimize Key Parameters and Effectiveness Evaluation of the AEB System Based on Rear-End Collision Accidents," SAE Int. J. Passeng. Cars – Electron. Electr. Syst. 10(2):310-317, 2017.

[3] Senapati, P., Das, S., and Vora, P., "Intelligent Braking and Maneuvering System for an Automobile Application," SAE Technical Paper 2017-26-0080, 2017.

[4] Rawashdeh, Z., Nguyen, T., Pottammal, A., and Malhan, R., "Comfortable Automated Emergency Brake for Urban Traffic Light Based on DSRC and On-Board Sensors," SAE Technical Paper 2017-01-0108, 2017.

[5] Lee, I. and Luan, B., "Design of Autonomous Emergency Braking System Based on Impedance Control for 3-Car Driving Scenario," SAE Technical Paper 2016-01-1453, 2016.

[6] Coelingh E., Eidehall Z. and Bengtsson M., "Collision Warning with Full Auto Brake and Pedestrian Detection-a practical example of Automatic Emergency Braking, " 2010 IEEE International Conference on Intelligent Transportation Systems, Portugal, pp. 155-160, Sept. 2010.

[7] Fildes B, Keall M, Bos N, et al., "Effectiveness of low speed autonomous emergency braking in real-world rear-end crashes", Accident Analysis & Prevention, 81:24, 2015.

[8] Jidong Gao, Hui Z. and Bolin G., "Test Protocols of Autonomous Emergency Braking Systems", Auto Engineer, 0(1):11-15, 2017.

[9] EURO NCAP. https://www.euroncap.com/en.

[10] Liu, Y. J., et al. "Neural Network Control-Based Adaptive Learning Design for Nonlinear Systems with Full State Constraints, " IEEE Transactions on Neural Networks & Learning Systems, 27 (7) :1562-1571, 2016.

[11] Fan J, Zhong J, Zhao J, et al., "BP neural network tuned PID controller for position tracking of a pneumatic artificial muscle, " Technology & Health Care Official Journal of the European Society for Engineering & Medicine, 2(s2):231-8, 2015.

[12] Ma L, Yao Y, Wang M., "The Optimizing Design of Wheeled Robot Tracking System by PID Control Algorithm Based on BP Neural Network," 2016 IEEE International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration, 34-39, 2016.

[13] Chen S., "Studies on Learning Algorithms for BP Net," Journal of Basic Science & Engineering, 1995.

[14] Xuewu J., Jian W., Youqun Z., et al., "Path Planning and Tracking for Vehicle Parallel Parking Based on Preview BP Neural Network PID Controller," Trans. Tianjin Univ., 21(3): 199-208, 2015.

[15] Jing S, Guo S, Zhao X, et al., "BP neural network PID controller of pocket dropout," 2016 IEEE International Conference on Computer and Communications, 67-71, 2016.

[16] Zhenhai G, Bo Z., "Vehicle lane keeping of adaptive PID control with BP neural network self-tuning," 2005 IEEE Intelligent Vehicles Symposium, Proceedings. 84-87, 2005.

[17] Lijun J., "Research on Test Scenarios of Automatic Emergency Braking System," Auto Tech, 0(1):39-43, 2014.

[18] Shifu Li, "Modeling and Simulation of Vehicle Collision Avoidance Control System" Hunan Univ., 2009.

[19] Rongrong Wang., "Modeling and research of vehicle collision avoidance system based on fuzzy control," Highways & Automotive Applications, 0 (2): 9-15, 2012.