Reinforcement Learning based Throttle and Brake Control for Autonomous Vehicle Following

Qi Zhu, Zhenhua Huang, Zhenping Sun, Daxue Liu and Bin Dai
College of Mechatronic Engineering and Automation
National University of Defense Technology
Changsha, China
zhuqics@hotmail.com

***Abstract*—In this paper, we focus on the basic form of autonomous follow driving problem with one leader and one follower. A reinforcement learning based throttle and brake control approach is developed for the follower vehicle. Near optimal control law is directly learned by "trial and error" with the neural dynamic programming algorithm. According to the timely updated following state, the learned control policy can deliver appropriate throttle and brake control commands for the follower. Simulation tests to illustrate the effectiveness of the presented method are carried out with the highly recognized vehicle dynamic simulator CarSim.**

***Index Terms*—autonomous driving, autonomous following, throttle and brake control, neural dynamic programming, reinforcement learning.**

I. Introduction

Autonomous driving technologies has become hot topics among the academia and industry [1], [2], [3]. In this paper, we aim at the autonomous follow driving problem which is one of the autonomous vehicles application forms with great potential to improve the transportation efficiency and save energy. Specifically, the follower's throttle and brake control of the basic autonomous following problem with one follower and one leader is well investigated.

The goal of controlling the follower's throttle and brake is to regulate the follower's speed and position to drive after the leader with the given target space. This longitudinal control problem for autonomous following has taken wide attentions from many researchers. As an intuitive and widely applied control law, proportional integral differential (PID) controllers are designed to regulate the throttle and brake for the autonomous following problem [4], [5]. But experience and skills are needed to tune the control parameters. Generally, to fulfill the autonomous follow driving with a given target space, the whole longitudinal control problem is always divided into two level. The upper level is to determine the optimal follower's acceleration or speed. The desired acceleration or speed are then tracked with adaptive throttle and brake controllers in the lower level. With this scheme, model predict controllers are developed for the formation control of off-road autonomous vehicles [6], [7]. Linear optimal upper-level controllers are proposed for the full speed range cruise control in [8] and [9]. Then the throttle and brake are controlled with a model free control [10]. Sliding-model approaches [11], [12] and H_∞ theories [13] based controllers are also presented to realize the autonomous follow driving.

Reinforcement learning (RL) algorithms are considered as effective approaches to solve complex sequential decision and control problems [14], [15]. The goal of RL methods is to derive the optimal action policy of different states for the agent through iteratively interacting with the environment. By repeatedly "trail-and-error", the agent try to find the optimal action policy at every state to maximize the total rewards from the environment. Neural dynamic programming (NDP) [16], as a kind of model-free reinforcement learning method, possesses encouraging performance on learning speed, success rate of learning.

In this paper, a reinforcement learning mechanism based on NDP is provided to directly learn the near optimal throttle and brake control policy to regulate the following distance to the target value. In the remaining of this paper, the MDP modeling of the autonomous following problem is introduced in Section II. The learning control method based on neural dynamic programming is illustrated in detail in Section III. Simulation results for verifying the effectiveness of the proposed approach are shown in Section IV. Finally, conclusions and future work are discussed.

II. Autonomous following problem modelling

The objective of this paper is to develop a throttle and brake control approach for autonomous following problems. It is noted that the transmission of the follower is assumed to be unchanged all along. Generally, there are three basic kinds of following policies: constant separation, constant time-headway, and constant safety-factor [17]. In this paper, constant separation policy is adopted. Thus, proper throttle and brake control commands are expected to drive the follower vehicle behind the leader with a fix distance. When the leader speeds up or slows down, the controller should respond quickly and accurately to converge the following distance to the target value.

As illustrated in Fig. 1, we assumed that the speed of itself v_f, leader's speed v_l and follow space s can be obtained by the follower's controller. Actually, several kinds

978-1-5386-3524-7/17/$31.00 ©2017 IEEE

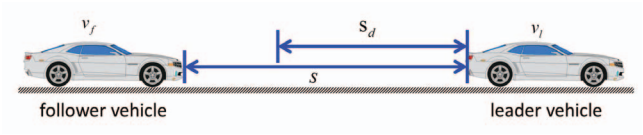

Fig. 1. The autonomous following problem.

of sensor like odometers, IMUs, radars, and LIDARs are available to acquire these information [18], [19]. Moreover, with vehicle-to-vehicle devices, vehicles' running state can be spread between each other [20]. In autonomous follow applications, the leader can be operated by the human driver or some other autonomous driving systems. In this paper, lateral controls of the leader and the follower both are fulfilled with the CarSim built-in preview based path tracker. While the leader's speed is regulated with the approach in [21], the longitudinal control of the follower is implemented with the presented reinforcement learning (RL) based method. To apply RL algorithms, field problems should be formatted as the Markov Decision Processes (MDPs). The principle of MDPs and MDP definitions for the autonomous following problem are introduced in detail in the rest of this section.

A. Markov Decision Process

Reinforcement learning algorithms are developed on the basis of Markov Decision Processes (MPDs). By modeling the interaction process between the agent and the environment, MDPs provide a framework to describe and analyze sequential decision-making problem [22].

A Markov decision process can be depicted as a 5-tuple $(S, \mathcal{A}, P, R, \gamma)$ where S is the states set, $\mathcal{A}$ is the action set, P denotes the state transition probability model, R is the immediate reward, and $\gamma \in [0, 1)$ is the discount factor for future rewards. Policy $\pi : S \mapsto \Omega(\mathcal{A})$ of an MDP is a mapping from state space to action space, while $\Omega(\mathcal{A})$ represents a probability distribution in the action space.

The action-value function of policy π is defined as follow:

$$Q^{\pi}(s,a) = E^{\pi}\left\{\sum_{t=0}^{\infty}\gamma^{t} r_t \,|s_0 = s, a_0 = a\right\} \tag{1}$$

For any state-action pair (s, a), the action-value function $Q^{\pi}(s, a)$ satisfies the Bellman equations:

$$Q^{\pi}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s,a,s') \sum_{a' \in A} \pi(s',a') Q^{\pi}(s',a') \tag{2}$$

where $\pi(s', a')$ is the probability of performing action a' in next state s'; $P(s, a, s')$ is the probability of making a transition to next state s' when taking action a in current state s. Written in matrix form, it can be expressed as follows:

$$Q^{\pi} = R + \gamma \mathrm{P}\Pi_{\pi} Q^{\pi} \tag{3}$$

$$(I - \gamma \mathrm{P}\Pi_{\pi}) Q^{\pi} = R \tag{4}$$

For a deterministic policy space, there exists an optimal policy π^* which maximizes the action-value function Q^{π} for all the state-action pairs:

$$Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a) \tag{5}$$

when $Q^*(s, a)$ is computed, the optimal policy can be obtained easily by

$$\pi^* = \arg\max_{a} Q^{\pi}(s,a) \tag{6}$$

B. The definition of MDP for the autonomous following problem

According to the theory of MDPs above, the proper definition of MDP for autonomous follow driving problem should be fulfilled first before learning online.

The objective of our method is to derive proper throttle or brake control commands to drive the follower vehicle to maintain the target distance from the leader. Thus the following distance deviation $e_d = d - d_t$ is one of the MDP's states for the automatic follow problem. Additionally, follower's relative speed to the leader $e_v = v_f - v_l$ is an important factor affecting the following space. Thus it is treated as the other element of the MDP's state. The complete state of the MDP for the autonomous follow problem is defined as Eq. (7).

$$s = (e_d, e_v) \tag{7}$$

The action of the MDP for the autonomous follow problem is the control value to the vehicle. Since we only focus on the longitudinal control and the transmission is supposed to be unchanged, the control value to the follower only includes the throttle and brake. Due to the throttle and brake cannot be applied at the same time, the longitudinal system is regarded as a one input plant. For the convenience of computation, the throttle and brake are normalized in the range of $[-1, 1]$, and noted as u. u goes from 0 to 1 means minimum throttle to the maximum, and from 0 to -1 represent minimum brake pressure to the maximum.

Immediate reward of MDPs is the measurement for action taken in certain states. It is important to guide the learning process. In this paper, the reward is designed as Eq. (8).

$$r(s,a) = -(k_1 \cdot e_d + k_2 \cdot e_v) \tag{8}$$

where k_1 and k_2 are positive weights parameters. From Eq. (8), it can be found that any follow distance deviation or relative speed will be punished.

III. Online Learning Throttle and Brake Control

State transition probability model P in the MDP definition reflects the effects when actions are applied to the system. It is crucial to observe the system's operation process and to solve the MDP problems. However, since the vehicle's

longitudinal system consists of several components like engine, transmission and tyres, large delay and strong nonlinear characteristic exist in the power and braking delivery process. It is difficult to find an explicit relationship from throttle or brake inputs to the vehicle's speed or position. Therefore, a reinforcement learning algorithm named Neural Dynamic Programming (NDP) [16] independent of priori models is employed in this paper. As an online and model-free method, NDP can update the action policy and estimation of the value functions simultaneously by iteratively interact with the environment. It is a typical actor-critic algorithm, neural networks with three layers are designed for the actor and critic respectively.

Fig. 2 illustrates the critic component of NDP, MDP state s_t and the performed action a_t at time t are inputted into the critic network. As defined above, for the autonomous following problem, s_t is the couple of the follower's relative speed e_v and the follow distance deviation e_d. Action a_t is the normalized control value of throttle and brake u. Output of this critic network is the approximate of the state-action value function $Q(s_t, u_t)$ of the MDP at time t, denoted as $\tilde{Q}(s_t, a_t)$.

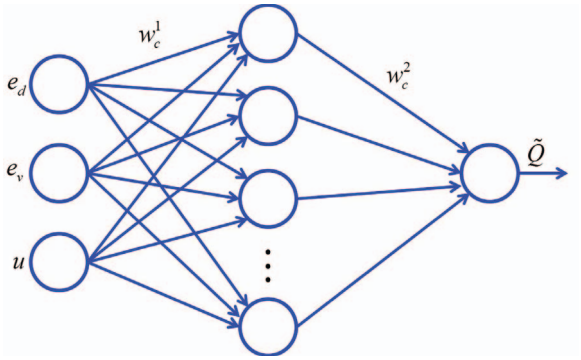

Fig. 2. Schematic diagram of the critic network.

For the critic, the prediction error is defined as Eq. (9). Where γ is the discount factor for future rewards, $r(s_t, a_t)$ denotes the reward as (8) received by execute action a_t at state s_t. Subscripts t and $t+1$ refer to the successive time steps.

$$e_c(t) = r(s_t, a_t) + \gamma \tilde{Q}(s_{t+1}, a_{t+1}) - \tilde{Q}(s_t, a_t) \tag{9}$$

Then the loss function to be minimized of the critic neural network is defined as Eq. (10).

$$E_c(t) = \frac{1}{2} e_c^2(t) \tag{10}$$

Gradients to the network parameters are crucial to solve the optimal loss value. Hierarchical gradient rules are derived as Eq. (11) and Eq. (12).

$$\begin{aligned} W_{t+1}^c &= W_t^c - \alpha_t \Delta W_t^c \\ &= W_t^c - \alpha_t e_c(t) \frac{\partial e_c(t)}{\partial W_t^c} \end{aligned} \tag{11}$$

$$\frac{\partial e_c(t)}{\partial W_t^c} = \gamma \frac{\partial \tilde{Q}(s_{t+1}, a_{t+1})}{\partial W_t^c} - \frac{\partial \tilde{Q}(s_t, a_t)}{\partial W_t^c} \tag{12}$$

Where α_t is the learning rate in the critic network and the $\frac{\partial \tilde{Q}(s_{t+1}, a_{t+1})}{\partial W_t^c}$ and $\frac{\partial \tilde{Q}(s_t, a_t)}{\partial W_t^c}$ are determined via the predefined structure of the critic network.

Fig. 3 indicates the actor component of NDP, only MDP state s_t is inputted into the actor network. Similar to the critic, for the autonomous following problem, s_t is consist of thee follower's relative speed e_v and follow space deviation e_d as Eq. (7). Output of the actor network is the desire action a_t that should be executed at state s_t. In this paper, it is exactly the normalized throttle and brake control value u.

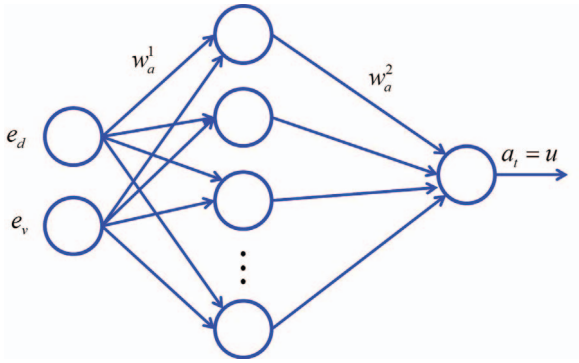

Fig. 3. Schematic diagram of the actor network.

For the actor network, the aim is to minimize the accumulative total reward $Q(s_t, a_t)$ derived by performing action a_t at state s_t. The gradient of q_t can be utilized to obtain the optimal action as Eq. (13). But the explicit form of $Q(s_t, a_t)$ is unknown. Thus the approximated value $\tilde{Q}(s_t, a_t)$ derived from the critic network is employed instead to solve the optimal action.

$$\frac{\partial Q(s_t, a_t)}{\partial a_t} = 0 \tag{13}$$

Then the objective function of the actor network is defined as Eq. (14).

$$E_a(t) = \frac{1}{2} \tilde{Q}^2(s_t, a_t) \tag{14}$$

Similar gradient rules can be found for parameters in the actor networks as Eq. (15) and Eq. (16).

$$\begin{aligned} W_{t+1}^a &= W_t^a - \beta_t \Delta W_t^a \\ &= W_t^a - \beta_t \frac{\partial E_a(t)}{\partial W_t^a} \end{aligned} \tag{15}$$

$$\frac{\partial E_a(t)}{\partial W_t^a} = \tilde{Q}(s_t, a_t)\frac{\partial \tilde{Q}(s_t, a_t)}{\partial a_t}\frac{\partial a_t}{\partial W_t^a} \tag{16}$$

Where β_t is the learning rate in the actor network. $\tilde{Q}(s_t, a_t)$ is the approximated of $Q(s_t, a_t)$ outputted from the critic network, $\frac{\partial \tilde{Q}(s_t, a_t)}{\partial a_t}$ and $\frac{\partial a_t}{\partial W_t^a}$ are determined according to the structure of the critic and actor network respectively.

With the above defined MDP models and gradient rules for the parameters of the critic and actor networks, the state action value function Q and the action policy π can be updated iteratively and converge to an optimal or a near optimal stage. Then appropriately normalized throttle and brake control actions for the autonomous follow driving problem u can be derived.

IV. Simulation Results

In this section, simulation results that demonstrate the performance of the presented learning based throttle and brake control method are shown. The simulation is completed with Matlab and CarSim. CarSim is a well-known high fidelity vehicle dynamic simulator and is highly recognized by the automotive industry. Our learning based controller is implemented with Matlab and communicates with CarSim with the provided API.

Before the learning process, actor and critic networks structure should be defined clearly. In this paper, both actor and critic utilize feedforward neural networks with three layers. As Fig. 2 and Fig. 3 show, there are 3 and 2 input nodes for the critic and actor network individually, 1 output node for both. Both two networks have 10 hidden nodes. The input signals are scaled into the range of $[-1, 1]$ during the iteration. The hidden and output nodes of the actor network have sigmoid-type activation function. The hidden nodes of the critic network have sigmoid-type activation function and the output node of the critic network has linear-type activation function. Weights parameters in the actor and critic networks are randomly initialized within the interval $[-1, 1]$. Updating rates α_t and β_t are both set as 0.01. The discount factor γ is set as 0.9. Learning procedure is terminated until the maximize iteration is achieved or the following distance converges to the target value.

A representative procedure is designed to conduct the test experiment. In the test, initial speeds of the follower and the leader are set as 0 and $30km/h$ respectively. Initial follow distance is $10m$, the target is set as $25m$. The leader's speed is changed with time. During the first $30s$, it maintains at $30km/h$. In the second $30s$, it transits to $20km/h$ smoothly. Finally, it speeds up to $40km/h$.

The following results are illustrated in Fig. 4. In the upper left sub-figure, the leader's follower's speeds are recorded as the blue solid curve and the red dashed curve respectively. In the upper right sub-figure, the target following distance and actual follow distance are indicated as the blue solid curve and the red dashed curve individually. From these two sub-figures, it can be found that the follower's speed can be regulated rapidly when large relative speed or large follow distance deviation exist. And the actual following distance can converge to the target value after short regulations at the very beginning. The lower two sub-figures in Fig. 4 described the follower's relative speed and the follow distance error. There are acceptable small overshoots during the regulation process.

Control value derived from the actor network is recorded as the red dashed curve in Fig. 5. Actual normalized throttle and brake value exported from CarSim are indicated as the blue solid curve and the green dashed curve. Note that for the convenience of comparing, the brake values are multiply by -1 since the exported raw values are always positive. Appropriate control commands are generated from the proposed learning based throttle and brake control approach from Fig. 5.

V. Conclusions and Future Work

The throttle and brake control methods for autonomous follow driving are investigated in this paper. A neural dynamic programming based throttle/brake control approach is proposed. The control policy is directly learned by "trial and error". According to the following state including follower's relative speed and follow distance deviation, the learned control law can derive near optimal control commands for the autonomous following problem. The effectiveness of the proposed method is illustrated by test experiments with highly recognized vehicle dynamic simulator.

However, slight vibrating exists in the control commands during the follow distance regulation process according to the simulation results. This is harmful to the actuators like throttle and brake. Some proper mechanisms should be introduced in the future to eliminate the vibrating of the control value. Besides, experiments on real vehicles should be conducted to further verify the performance of the presented method.

References

[1] M. Iida, M. Kudou, K. Ono, and M. Umeda, "Automatic following control for agricultural vehicle," in *International Workshop on Advanced Motion Control, 2000. Proceedings*, 2000, pp. 158–162.

[2] R. Kianfar, B. Augusto, A. Ebadighajari, U. Hakeem, J. Nilsson, A. Raza, R. S. Tabar, V. K. Irukulapati, C. Englund, and P. Falcone, "Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 994–1007, 2012.

[3] M. Faieghi, A. Jalali, and S. K. Mashhadi, "Robust adaptive cruise control of high speed trains." *Isa Transactions*, vol. 53, no. 2, p. 533, 2014.

[4] P. A. Ioannou and Z. Xu, "Throttle and brake control systems for automatic vehicle following," *Journal of Intelligent Transportation Systems*, vol. 1, no. 4, pp. 345–377, 1994.

[5] J. Lygeros, D. N. Godbole, and S. Sastry, "Verified hybrid controllers for automated vehicles," *Automatic Control IEEE Transactions on*, vol. 43, no. 4, pp. 522–539, 1998.

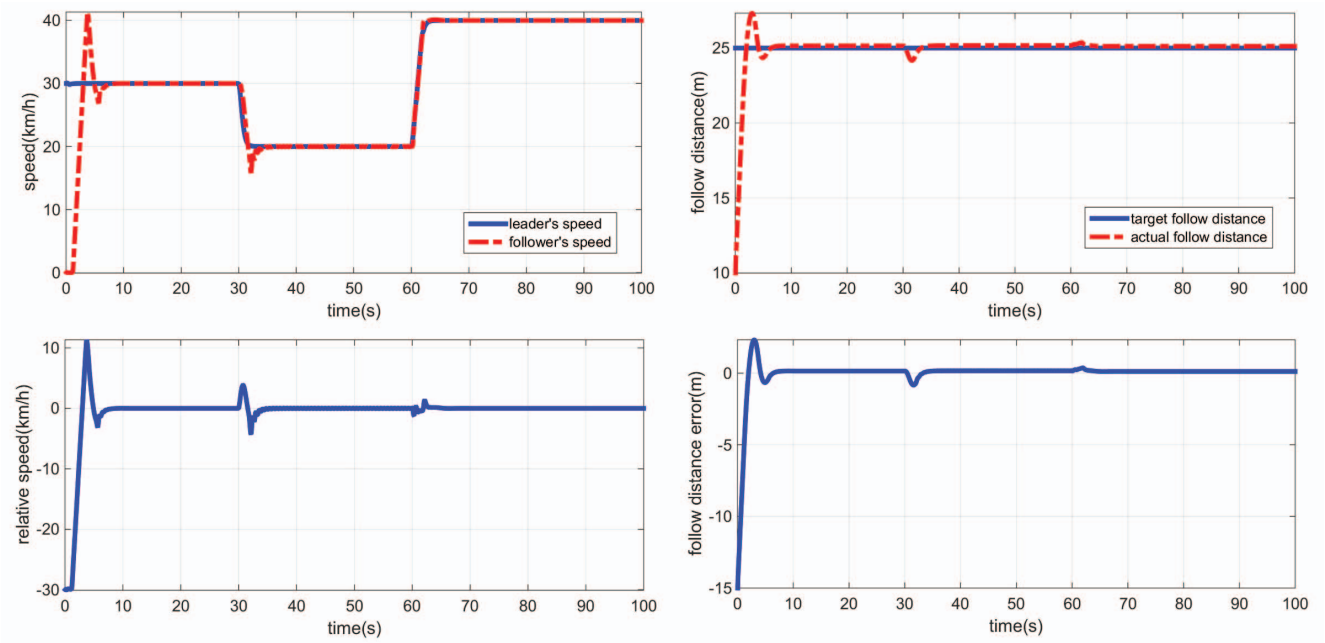

Fig. 4. Simulation results. Upper left: leader's and follower's speed profile. Upper right: actual and target follow distance. Low left: follower's relative speed to leader. Low right: follow distance error.

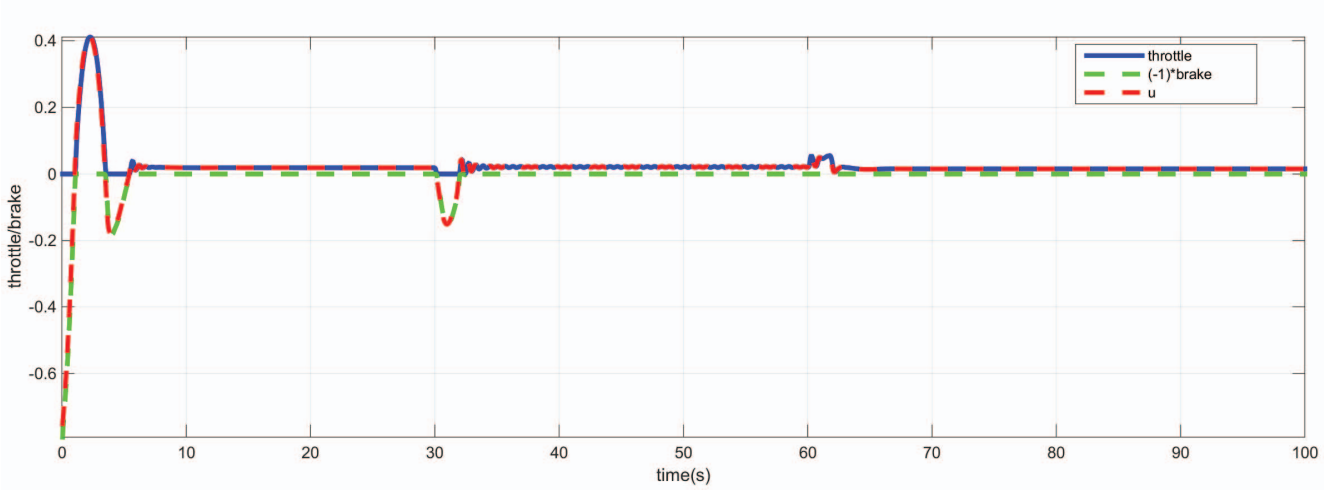

Fig. 5. Desired and actual throttle/brake control value of the follower vehicle.

[6] A. Guillet, R. Lenain, B. Thuilot, and P. Martinet, "Adaptable robot formation control: Adaptive and predictive formation control of autonomous vehicles," *IEEE Robotics and Automation Magazine*, vol. 21, no. 1, pp. 28–39, 2014.

[7] J. Bom, B. Thuilot, F. Marmoiton, and P. Martinet, "A global control strategy for urban vehicles platooning relying on nonlinear decoupling laws," in *Intelligent Robots and Systems*, 2005, pp. 2875–2880.

[8] S. Moon, I. Moon, and K. Yi, "Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance," *Control Engineering Practice*, vol. 17, no. 4, pp. 442–455, 2009.

[9] K. Dongwook, L. Junyoung, K. Boemjun, L. Kangwon, and Y. Kyoungsu, "Integrated risk management based automated vehicle following system on inner-city streets," in *17th International Conference on Intelligent Transportation Systems (ITSC)*, 2014, Conference Proceedings, pp. 418–423.

[10] K. H and Y. K, "Design of a model reference cruise control algorithm," *SAE International Journal of Passenger Cars*, vol. 5, no. 2, pp. 440–449, 2012.

[11] A. Fritz and W. Schiehlen, "Nonlinear acc in simulation and measurement," *Vehicle System Dynamics*, vol. 36, pp. 159–177, 2001.

[12] B. Ganji, A. Z. Kouzani, S. Y. Khoo, and M. Nasir, "A sliding-mode-control-based adaptive cruise controller," in *International Conference on Control and Automation*, 2014, pp. 394–397.

[13] E. Kayacan, "Multiobjective h_∞ control for string stability of coopera-

tive adaptive cruise control systems," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 1, pp. 52–61, 2017.
[14] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *Neural Networks IEEE Transactions on*, vol. 9, no. 5, p. 1054, 1998.
[15] X. Xu, C. Liu, and D. Hu, "Continuous-action reinforcement learning with fast policy search and adaptive basis function selection," *Soft Computing*, vol. 15, no. 6, pp. 1055–1070, 2010.
[16] J. Si and Y. Wang, "Online learning control by association and reinforcement," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 264–276, 2001.
[17] R. J. Caudill and W. L. Garrard, "Vehicle-follower, longitudinal control for automated transit vehicles *," *IEEE Transactions on Vehicular Technology*, vol. 28, no. 1, pp. 36–45, 1977.
[18] C. Urmson, J. Anhalt, D. Bagnell, C. R. Baker, R. Bittner, M. N. Clark, J. M. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.
[19] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. A. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. W. Hoffmann, B. Huhnke *et al.*, "Junior: The stanford entry in the urban challenge," *Journal of Field Robotics*, vol. 25, no. 9, pp. 569–597, 2008.
[20] D. Jia and N. Dong, "Platoon based cooperative driving model with consideration of realistic inter-vehicle communication ," *Transportation Research Part C Emerging Technologies*, vol. 68, pp. 245–264, 2016.
[21] J. Wang, Z. Sun, X. Xu, D. Liu, J. Song, and Y. Fang, "Adaptive speed tracking control for autonomous land vehicles in all-terrain navigation: An experimental study," *Journal of Field Robotics*, vol. 30, no. 1, pp. 102–128, 2013.
[22] R. Sutton, *Reinforcement Learning: An Introduction*. MIT Press, 1998.