Research on vehicle cruise control based on online asynchronous supervised reinforcement learning

Zhonghai Han\textsuperscript{1,2}, Liangyi Yang\textsuperscript{1,2}, Qiang Zhang\textsuperscript{1,2,3}, Jie Fan\textsuperscript{2}.

\textsuperscript{1}State Key Laboratory of Vehicle NVH and Safety Technology, Chongqing 401122, China
\textsuperscript{2}China Automotive Engineering Research Institute Co., Ltd., Chongqing 401122, China
\textsuperscript{3}Chongqing University, Chongqing 400044, China

*Corresponding author: Liangyi Yang. E-mail address: yangliangyi@caeri.com.cn.

Abstract. This paper proposes a vehicle cruise control strategy based on asynchronous supervised reinforcement learning, which improves the driver's reception in the process of vehicle deceleration and following. The control strategy takes actor-critic network as the basic control unit, which guides the control strategy to achieve driver-like cruising effect by adding real driver cruising data into the supervision network. Simultaneously, combined with the driver online feedback mechanism for real-time training, it realizes offline training and online updating of network parameters. The simulation results show that the asynchronous supervised reinforcement learning algorithm can quickly update the parameters of the control network, and constantly update the control strategy by online learning combined with the actual driving data to better simulate the driver's driving characteristics.

1. Introduction

Adaptive cruise control (ACC) is an important part of advanced driver assistance systems (ADAS). On the one hand, ACC system can reduce manual driving errors and enhance the safety of active driving. On the other hand, it can improve the driving comfort and reduce the driver's driving burden.

In the early stage, linear feedforward and feedback controllers are widely used in cruise control system [1]. However, the adaptability of linear controllers is poor against unknown disturbances. However, reinforcement learning based control strategy is considered to be an effective method to solve the above problems [2]. By learning available and even optimal control strategies based on the collected manual driving data, it has strong adaptability in solving the control problems of unknown model and uncertain environment.

In order to make ACC system realize the braking cruise effect of personalized drivers in the process of deceleration and parking, this paper proposes an asynchronous supervised reinforcement learning based control strategy, which adopts asynchronous training mechanism to quickly iterate and shorten the network learning time. Simulation results show that the control strategy can make ACC system achieve similar deceleration braking behavior to real drivers.
2. System architecture and network composition

2.1. Control framework of the system

Fig. 1 shows the overall structure of the control framework. The basic function of ACC system is to achieve the desired balance state between the main vehicle and the front vehicle by controlling the relative distance $\Delta d(t)$, relative speed $\Delta v(t)$ and relative acceleration $\Delta a(t)$ between the main vehicle and the front vehicle.

The upper control module takes actor-critic network as the main body and adds pre-trained supervision network. In the process of vehicle deceleration, the offline trained upper control module inputs the lower control network proper control signal $u(t)$ according to the relative error $[\Delta d(t), \Delta v(t), \Delta a(t)]$ between the main vehicle and the front vehicle. If the driver is not satisfied with the current network control effect, he/she can intervene in the vehicle automatic cruise deceleration process online, and impose additional feedback control $u'(t)$ to adjust the deceleration effect. At this time, the control system enters the online learning mode. The control parameters can be quickly updated online through the asynchronous training mechanism, so as to adjust the control parameters to fit the new driver deceleration style. The lower control module calculates the desired throttle opening and braking pressure under the current vehicle speed and demand acceleration according to the vehicle dynamics model, and controls the acceleration and braking pedals combined with the actuator.

2.2. Supervised reinforcement learning neural network

The supervised reinforcement learning (SRL) constructed in this paper mainly includes four parts: controlled system, actor, critic and supervisor. The controlled system is a longitudinal following dynamic model, which transfers from the current state to the next state under the action of control variables. Actor is responsible for outputting control actions according to the current system state. Critic evaluates the control actions and updates the parameters of actor to improve the control strategy. The supervisor is a driver model based on real-world data, which gives a reasonable "anthropomorphic" control output according to the current vehicle state, compares it with the "mechanized" control output given by the action unit, and provides the actor unit with update tips about which operation may or may not be suitable. At the same time, the composite action of mixing two actions is sent by the weight regulator to the vehicle system. The system changes from the current vehicle state to the next state by responding to the compound input action, and gives out the reward.

2.3. Dynamic model of longitudinal following system

The main goal of ACC system is to make the main vehicle keep the desired distance and follow the vehicle ahead. The safety distance model is adopted as follows[3]:

---

Fig. 1 Control frame structure of the system

Fig. 2 Overall control framework
\[ d_s(t) = d_s^0 + hv_h(t) \]  
\text{Eq.(1)}

where \( d_s \) is the expected relative distance, \( d_s^0 \) is the safe vehicle distance when parking, \( h \) is the workshop time distance, \( v_h \) is the main vehicle speed.

2.4. Actor

Actor uses three-layer feedforward neural network to establish the mapping from system state and control signal, as shown in Fig.3. The input of the network is the system error and the output is the control signal[4].

\[ u_a(t) = \frac{1 - e^{-v(t)}}{1 + e^{-v(t)}} \]  
\text{Eq.(2)}

where \( v(t) \) is the input of the actor neural network.

\[ v(t) = \sum_{i=1}^{N_{ha}} w_a^{(2)}(i) g_i(t) \]  
\text{Eq.(3)}

where \( N_{ha} \) is the number of neurons in the hidden layer. \( w_a^{(2)} \) is the weight coefficient vector from the hidden layer node to the output node, which is trained and updated by error backward propagation. \( g_i(t) \) is the output vector of the hidden layer.

![Fig.3] Three-layer neural network

2.5. Critic

Critic also uses a three-layer feedforward neural network, and the output \( J(t) \) is expressed as follows:

\[ J(t) = \sum_{i=1}^{N_{hc}} w_c^{(2)}(i) p_i(t) \]  
\text{Eq.(4)}

where \( N_{hc} \) is the number of neurons in the hidden layer. \( w_c^{(2)} \) is the weight coefficient vector from the hidden layer to the output layer. \( p_i(t) \) is the output vector of the hidden layer node.

\[ p_i(t) = \frac{1 - e^{-q_i(t)}}{1 + e^{-q_i(t)}}, \quad i = 1, \ldots, N_{hc} \]  
\text{Eq.(5)}

where \( q_i(t) \) is the input vector of the hidden layer node.

In addition to the system state \( x(t) \), the input of critic also includes the composite control signals \( u(t) \) of the supervisor and the actor. The output of critic is the evaluation signal \( J(t) \), which is the estimation of the future cumulative discount return \( R(t) \) of the system.

\[ R(t) = \sum_{k=0}^{N_T} \gamma^k r(t+k+1) \]  
\text{Eq.(6)}

where \( N_T \) is the number of termination time steps, \( \gamma \) is the discount factor, and \( r \) is the return function. Cumulative discount return \( R(t) \) shows which action is good in the long term. Under the steady-state following driving condition, the expected distance error and relative speed tend to zero at
the same time, and the excessive acceleration jump is avoided to affect the ride comfort. Thus, reward function is designed as follows:

$$r(t) = -\left(k_1 e_d^2(t) + k_2 \Delta v^2(t) + k_3 \Delta a^2(t)\right)$$

Eq.(7)

where $k_1$, $k_2$ and $k_3$ are positive weight coefficients. $\Delta e_d$ indicate the error between the real relative distance and the expected relative distance.

2.6. Supervisor network

The supervisor is actually a driver model, which also adopts three-layer back propagation neural network. The hidden layer and output layer adopt hyperbolic tangent function.

$$a^{(i)} = \tanh(n) = \frac{e^n - e^{-n}}{e^n + e^{-n}}, \quad n = Wa^{(0)} + b$$

Eq.(8)

where $a^{(i)}$ and $a^{(0)}$ are the outputs of the current layer and the upper layer respectively. $n$ is the cumulative output. $W$ is the weight coefficient. $b$ is the bias value.

The actor network, the supervisory network and the weight regulator generate a composite action for the main vehicle, and the formula is as follows:

$$u(t) = k_n u_e(t) + (1 - k_n) u_d(t), \quad 0 \leq k_n \leq 1$$

Eq.(9)

where for $u_e(t)$, normalization is carried out in the range of $[-1,1]$. $u_e(t)$ represents the exploration behavior of action unit and $u_d(t) = u_e(t) + N(0,\sigma)$. $N(0,\sigma)$ represents the random noise with mean 0 and variance $\sigma$. $k_n$ is the parameter weight, which determines the participation degree of the driver model and the guidance strength of the supervision network.

3. Control framework based on asynchronous supervised reinforcement learning

The learning process proposed in this paper mainly includes two parts: one is the offline learning of the driving characteristics of the supervisor to make it close to the driver's behaviour, which is a general control module; the other is the real-time learning in the driving process, because the driver is not satisfied with the deceleration and following behaviour of the control system, he intervenes in the automatic control system and applies additional acceleration control. According to the feedback information, the control system can learn the driver's new style online quickly, and then update the network parameters of the supervisor.

In order to make full use of computer resources, and to ensure that the neural network converges quickly and avoid converging to the local optimal solution, this paper proposes to adopt the asynchronous learning method. The whole process includes a global network learning module and $N$ sub-worker modules, as shown in Fig.4. It should be pointed out that the structure of each sub worker module is the same as that of the global module, which is the basic actor, critic and supervisor network unit. Each worker thread will interact with the environment independently to get experience data. These threads do not interfere with each other and run independently.

![Fig.4 Schematic diagram of asynchronous network training](image-url)
In the training process, after each thread interacts with the environment to a certain amount of data, it calculates the gradient of the neural network loss function in its own thread. However, these gradients do not update the neural network in its own thread, but update the common neural network. In other words, $n$ worker threads independently update the parameters of the neural network model in the common part with the accumulated gradient. Since the global module directly obtains more "tendentious" network parameters, it is conducive to the convergence training of the network. After the global module network training, more excellent network parameters are distributed to each sub unit. After that, cycle training in turn. Compared with single network training, the asynchronous training method has $n$ times of training times in the same time step, so it takes $1/N$ time to achieve the same training effect. At the same time, due to the random network exploration, more network exploration behaviours can be carried out in one training, and the system is not easy to converge to the local optimum, so it can achieve better training effect.

4. Data simulation and test analysis

4.1. Personalized driving behaviour data collection

Firstly, two drivers with different driving styles are selected, and the driving data of the front car and the main car are collected by using the positioning inertial navigation system when they follow the driving deceleration to stop at the intersection, which is used as the training set of the neural network driver model. Fig.5 statistics the distance and acceleration of two completely different driving styles. In Fig.5 (a), when the driver tracks the car in front of him, he often starts to brake when still having a long distance to the intersection. In Fig.5 (b), the driver is more inclined to drive steadily to the intersection and suddenly brakes to make sure the vehicle stops behind the target vehicle. The fluctuation and zero drift of the collected data are caused by the accuracy of the acquisition equipment. The data after smoothing filtering is used for simulation training and analysis.

4.2. Discussion on network training and numerical simulation

The neural network toolbox of MATLAB is used to train the model. In the early stage of training, the weights of actor network are mainly updated under the supervisor. With the progress of training, reinforcement learning gradually dominates in order to obtain the optimal control strategy. During asynchronous supervised reinforcement learning (ASRL) training, one global module and four sub modules are configured according to computer resources.

A simulation experiment is designed to simulate the actual expectations of online drivers by using the driving data of the fixed workshop time distance model. The training effect of ASRL network and SRL network was compared after online driver behaviour intervention.

Each experiment has 1000 training times at most, and each algorithm has 100 repeated experiments. The success rate of training is shown in Tab.1. The experimental results show that the two algorithms
have 100% success rate within the set number range, but the average training numbers of ASRL algorithm is significantly lower than that of SRL algorithm, and it takes less time to meet the use needs. Fig.6 shows a large number of original following deceleration data and the probability distribution histogram of acceleration based on ASRL network simulation test results. Fig.6 (a) shows the acceleration distribution of the original following data collected by two drivers, and Fig.6 (b) shows the acceleration distribution of the two control strategies in the condition simulation. It can be seen that the simulation results and the original data have similar acceleration distribution, which indicates that the network successfully imitates the driver's behaviour.

### Table 1 Comparison of SRL and ASRL training

| Algorithm | Times | Success rate | Average times | Average time |
|-----------|-------|--------------|---------------|--------------|
| SRL       | 100   | 100%         | 129           | 19.7s        |
| ASRL      | 100   | 100%         | 33            | 102.4s       |

In the car-following simulation experiment, the front car is set to do continuous acceleration and deceleration behaviour. Fig.7 shows the following speed effect of ASRL training network controller under different driver data. It can be seen from the car following effect that the controller based on ASRL network can realize the car following cruise control, especially in the process of vehicle deceleration tracking. The behaviour difference of different driver data makes the network controller get different car following effect, and the output control effect is close to that of human driver. As a general human driver, the online real-time learning function of ASRL network can be understood as the system adaptive to different characteristics of the driver.

![Fig.7 ASRL network following control effect of different drivers](image)

5. Conclusion

In this study, we propose an ASRL based framework for the dynamic control of ACC system in the process of longitudinal vehicle deceleration and following braking. In particular, it introduces real-world driving data to build a supervision unit to guide the reinforcement learning process into the driver's characteristics. Secondly, combined with the real-time driver feedback information, the asynchronous reinforcement learning mechanism can make full use of the computing resources to realize the online and offline fast iteration of the neural network, greatly reducing the network training time. Through the numerical simulation, it is found that the control strategy can successfully simulate the driver's parking characteristics, so as to improve the driver's comfort.

### Acknowledgements

The authors acknowledge the Technology R & D and engineering service innovation and Entrepreneurship of intelligent connected vehicle (CQYC201903218) for financial support.

### References

[1] Milanes V, Shladover S E, Spring J, et al. Cooperative Adaptive Cruise Control in Real Traffic Situations[J]. IEEE Transactions on Intelligent Transportation Systems, 2014, 15(1):296-305.

[2] Liu T, Zou Y, Liu D, et al. Reinforcement Learning of Adaptive Energy Management with Transition Probability for a Hybrid Electric Tracked Vehicle[J]. IEEE Transactions on Industrial Electronics, 2015, 62(12): 7837-7846.
[3] Li Y, Sun D, Zhao M, et al. *MPC-based switched driving model for human vehicle co-piloting considering human factors*[J]. Transportation Research Part C Emerging Technologies, 2020, 115:102612.

[4] Makableh Y, Rios-Guteerrez F. *Intelligent cruise control using an efficient PID model neural network*[J]. Neural Parallel & Scientific Computations, 2012, 20(1):1609-18.