2022-01-0807 Published 29 Mar 2022

Comfort Improvement for Autonomous Vehicles Using Reinforcement Learning with In-Situ Human Feedback

Jun Xiang and Longxiang Guo Clemson University

Citation: Xiang, J. and Guo, L., "Comfort Improvement for Autonomous Vehicles Using Reinforcement Learning with In-Situ Human Feedback," SAE Technical Paper 2022-01-0807, 2022, doi:10.4271/2022-01-0807.

Received: 29 Jan 2022 Revised: 29 Jan 2022 Accepted: 25 Jan 2022

Abstract

In this paper, a reinforcement learning-based method is proposed to adapt autonomous vehicle passengers' expectation of comfort through in-situ human-vehicle interaction. Ride comfort has a significant influence on the user's experience and thus acceptance of autonomous vehicles. There is plenty of research about the motion planning and control of autonomous vehicles. However, limited studies have explicitly considered the comfort of passengers in autonomous vehicles. This paper studies the comfort of humans in autonomous vehicles longitudinal autonomous driving. The paper models and then improves passengers' feelings about autonomous driving behaviors. This proposed approach builds a control and adaptation strategy based on reinforcement learning using human's in-situ feedback on autonomous driving. It also proposes an adaptation of humans to autonomous vehicles to account for improper human driving expectations. The proposed approaches are implemented and tested with human-in-the-loop experiments and the results demonstrate that the proposed approaches can successfully adapt the vehicle behaviors, improve the ride comfort of humans in autonomous vehicles, and also correct improper human driving expectations.

Introduction

Autonomous vehicles have become a driving force to transform our existing transportation system in the near future and will help improve a lot of driving performance such as driving safety, congestion, emissions, etc. [1, 2]. The gradual maturity of autonomous driving technology and increasing consumer expectations spark interest in passenger comfort research [3, 4] . Although passengers' comfort has always been an essential topic for automotive-related research, ride comfort in AVs is dramatically different from ride comfort in a traditional vehicle. Traditionally, researchers mostly investigated ergonomic factors such as seat vibrations and noise. The introduction of automated driving functions would lead researchers to shift toward vehicle control factors, but limited methods have explicitly considered passengers in AVs [3].

Headway distance (gap) and speed are probably the first two to be noticed and the most significant control factors that affect the ride comfort [5]. Many traditional studies have focused on the gap acceptance of human drivers [6-8]. However, these works mainly focused on studying the gap acceptance and comfort of the population or a specific population is most likely has a neutral driving style, however, his/her headway selection may make a conservative driver feel nervous and unsafe, and at the same time, make an aggressive driver feel impatient and uncomfortable.

To make AVs comfortable for individuals, the concept of personalized autonomous driving has arisen in recent years. General approaches to achieve personalized autonomous driving can be categorized into analytical model-based approaches and heuristic approaches. Analytical approaches use explicit mathematical models to model the human expected driving behaviors [9-13]. Heuristic approaches do not assume a predefined mathematical model. Instead, they employ machine learning techniques such as artificial neural network [14, 15], Gaussian mixture models [16] and deep learning [17, 18] to approximate the human expected driving behaviors. However, existing approaches have two drawbacks. Firstly, they require humans to drive the vehicle to generate complete demonstrations, which does not match the application scenario of autonomous vehicles. Secondly, a person's expectations of driving behavior may vary as one's identity changes from driver to passenger. A control model trained on a person's driving data may still make him/her uncomfortable as a passenger.

Therefore, we propose to learn human expectations on autonomous vehicles without a need of human demonstrations but just intuitive force feedback from humans. More specifically, Reinforcement learning (RL) is used as the tool to find the most comfortable gap and speed for specific passengers with only their pressing force input data. The goal of reinforcement learning is to learn policies for

decision-making by optimizing a cumulative future reward function. Different from existing reinforcement learning approaches that use some calculated criteria like safety or efficiency as the reward, we directly integrate the real-time in-situ human pressing force into the reward as a representation of human comfort. In addition, it is known that reinforcement learning may sometimes learn unrealistically high action values because it includes a maximization step over estimated action values, which tends to prefer overestimated to underestimated values [19]. To address this, we have integrated some extra constraints and penalties in reinforcement learning to avoid such unrealistic actions. It is also known that human expected driving behaviors may not always be correct. For instance, one passenger may expect very aggressive driving behavior which may lead to safety issues. In contrast, another passenger may expect a very conservative driving behavior which may lead to efficiency issues. To address this, we propose an adaptation of humans to autonomy through human-vehicle interfaces to provide real-time feedback to humans in order to correct their improper expectations. Simulator-based research on the comfort of occupants in autonomous vehicles has been very popular in recent years and has produced many valuable results [20-22]. In this paper, we also propose to use a driving simulator to train and test the proposed reinforcement learning-based approach.

The contributions of the paper can be summarized as follows:

- Propose a reinforcement learning-based approach to learn human comfort expectations on autonomous vehicles using in-situ human pressing force feedback without a need for human driving demonstrations.
- Build a reinforcement learning-based comfortable autonomous driving control with an adaptation design for humans to correct their improper driving expectations.
- Conduct human-in-the-loop experiments to validate and evaluate the proposed approaches.

The rest of the paper is organized as follows. The third section gives details of the reinforcement learning-based learning of human comfort expectation and the adaptation of human passengers. The fourth section presents the human-in-the-loop experimental results and analysis.

Human Comfort Improvement by Reinforcement Learning

In the first part of this section, the method of learning human comfortable expectations and adapting a controller to human expectations using reinforcement learning is described in detail. Then, the adaption of humans to autonomy is introduced.

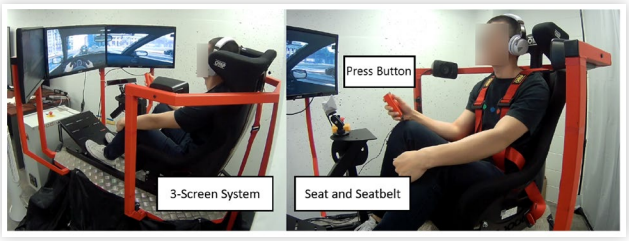
Reinforcement Learning of Human Comfortable Expectations with In-Situ Human Feedback

Comfort Data Collection through In-Situ Human-Vehicle Interaction In this paper, comfort data are collected through in-situ human-vehicle interaction. The participants will experience a virtual automated car-following ride, which is simulated by Simulink, as shown in Figure 1 left. The lead vehicle in the virtual ride will track a preset test drive cycle. The host vehicle, in which the participants are sitting virtually, is controlled by an existing car-following controller. The velocity and acceleration of both vehicles and the headway distance (gap) between them are recorded. To discretize the data for reinforcement learning, those recorded data are rounded to integers. During the virtual ride, the participants can feel and react to the states of the two-vehicle system. Although many features can affect ride comfort, this paper will focus on the influence of relative velocity (RV) and gap. In the real world, there are generally three different types of drivers: aggressive, neutral, and conservative. These three types of participants will be involved in this paper since they should have different feelings over the gap and relative speed conditions. An aggressive participant may feel more comfortable with a small gap and high relative velocity. On the contrary, a conservative participant may feel more comfortable with a large gap and low relative velocity. A neutral participant would feel uncomfortable with extreme gap and relative velocity.

Before the data collection process starts, the participants need to watch a three-minute sample video to get familiar with the in-situ interface. During the data collection, the participants will experience five three-minute-long rides. Three minutes is long enough to make the participants experience the car-following process comprehensively and not too long to keep them focused. If the participants feel uncomfortable with the relative velocity and/or gap, they can press a button, as shown in Figure 1 right, to express their discomfort. The button will measure the pressing force, and a larger pressing force stands for a greater sense of discomfort.

After the participants finish the visual rides, all the data will be fit to a reward policy. The reward policy can show the relationship between comfort level and gap and relative velocity. The reward policy table will play an essential role in this paper. During the Q-Learning training process, the

FIGURE 1 In-situ human-vehicle interaction.



reward policy table will be used as the RL reward policy, which generates the reward for each step in RL. During the evaluation process, the reward policy will be used to evaluate the trained agent. The detailed data fitting process will be introduced in section 3.1.2 below.

Comfort Data Regression and Normalization After the data collection process is finished, the raw data, which are pressing force versus time, need to be converted to the form that can be used to train the agent. Firstly, the data should be converted from pressing force versus time to pressing force versus relative velocity and pressing force versus gap.

The pressing force eventually is going to be used as the comfort reward for each state. Later in the next subsection, another type of reward, the safety reward, will be introduced. These two types of rewards will be combined to form the reward function of the RL. Thus, to balance the weights of these two rewards, the pressing force data should be normalized so that the comfort reward shares the same value range as the safety reward. Also, different participants may press the button with different forces to express the same level of discomfort. Normalization is critical for analyzing data from different participants fairly. Furthermore, the training with a normalized reward policy is easier to track during the training process. Lastly, the normalized data can be used directly for future training such as deep-q learning.

The next step is regression. Despite the preventive design in the virtual ride experience, the participant may still be distracted and press the button wrongly during the ride, which will bring noise and inaccuracy to the data. This paper presumes that the relationship between people's comfort and gap and RV is continuous and differentiable. It is unlikely that a person feels uncomfortable when the gap is 100 meters but feels extremely comfortable when the gap is 101 meters and feels extremely uncomfortable again when the gap is 102 meters. Since the general comfort expectations of different types of participants can be roughly anticipated, it is relatively easy to get a reasonable initial guess of the form of the fitted curve.

Learning of Human Comfortable Expectations In this paper, the RL method called Q-Learning is used. Q-Learning is a model-free reinforcement learning method firstly proposed by Watkins and developed further in 1992. Agents in RL algorithms are incentivized with punishments for bad actions and rewards for good ones. The goal of the agent is to learn a behavior rule that maximizes the reward it receives [23]. There is a table called Q-table used to record those learning results, which are also called Q-value. Each Q-value represents thereby the reward of a unique state-action pair, and a higher value relative to other values promises a higher return according to the definition [23].

Q-learning specifically allows an agent to learn to act optimally in an environment that can be represented by a Markov decision process (MDP). Consider a finite MDP $\langle S, A, P, R, \eta \rangle$ with state-space S, action space A, state transition probability P, reward function R and discount factor η. In this paper, the state space will be formed by the host vehicle's velocity, the relative velocity, the headway distance, and the front vehicle's trend. The trend of the front vehicle is based on whether it is accelerating or decelerating. Action space will contain the acceleration choices for the host vehicle. The transition probability will be determined by the state trajectories of both vehicles and the physics of the vehicle motion.

The reward function will be formed by the comfort reward defined in the previous subsection and an extra safety reward. The detailed reward function is given by the equation below:

$$R\left(s_t, a_t\right) = w_1 CR\left(s_t + 1\right) + w_2 SR\left(s_t + 1\right) \tag{1}$$

where $R(s_t, a_t)$ is the reward for taking action a_t in state s_t. w_1 and w_2 denote the weights that balance the comfort reward and safety reward. $CR(s_{t+1})$ denotes the comfort reward produced by reward policy based on the new state. $SR(s_{t+1})$ denotes the safety penalty that prevents the agent from driving too aggressively. After normalization, the value of CR and SR will be between 0 and 1. In this paper, a higher reward value means greater discomfort or safety hazard, and the rewards are actually penalties, thus both w_1 and w_2 are negative. The SR reward function is given below

$$SR\left(s\right) = \begin{cases} 1 & gap \left\langle gap_{\min}, gap \right\rangle gap_{\max}, \\ & RV \left\langle RV_{\min}, RV \right\rangle RV_{\max}, \\ & V < 0 m/s, TTC < TTC_{\min} \\ 0 & otherwise \end{cases} \tag{2}$$

A safety penalty will be imposed if one of the following three situations happens. The first situation is the car-following is out of state. Because of the property of the Q-learning, the training must be reset if the agent is out of state. The valid state range is defined by the minimum gap gap_{min}, maximum gap gap_{max}, minimum relative speed RV_{min} and maximum relative speed gap_{max}. The second situation is the car is reversing. In the real world, no human driver will reverse the car during the car following. The third situation is violating the safety constraints, which are defined by minimum time to collision TTC_{min}.

Q-learning is based on iteratively improving the state-action value function (or Q-function), which represents an expectation of the future reward when taking action a in state s and following policy π from thereon after. The Q-function is:

$$Q^{\pi}\left(s, a\right) = E_{\pi}\left\{R_t \middle| s_t = s \middle|, a_t = a\right\} \tag{3}$$

where $Q^{\pi}(s, a)$ denotes Q-Value, and it represents the reward value for each action's behaviors in each state of AVs. R_t denotes the future reward.

As described before, the state space will be formed by the host vehicle's velocity, RV, gap, and acceleration trend of the lead vehicle. Action space will be the acceleration choice. The behavior rule tells the host vehicle how to select actions given a certain state. At the same time, only one future step will be considered to calculate the reward.

Therefore, the Q-Table update equation in this paper is presented as follows:

$$R_{new} = R + \eta * Q\left(s_{t+1}, a\right)_{\max} \tag{4}$$

$$Q\left(s_t, a_t\right)_{new} = Q\left(s_t, a_t\right) + \theta * \left(R_{new} - Q\left(s_t, a_t\right)\right) \tag{5}$$

where η denotes the discount factor, θ denotes the learning rate, R_{new} denotes the updated estimated reward. R denotes the reward for reaching the current state, which is decided by the predefined reward policy. $Q(s_{t+1}, a)_{max}$ denotes the maximum potential future reward for the current state. $Q(s_t, a_t)_{new}$ denotes the new Q-Value for the updated Q-Table. $Q(s_t, a_t)$ denotes the old Q-Value.

Algorithm 1 Algorithm for human comfortable expectation learning.

Input: Lead vehicle velocity trajectory (LV), comfort reward policy (CR), training parameters

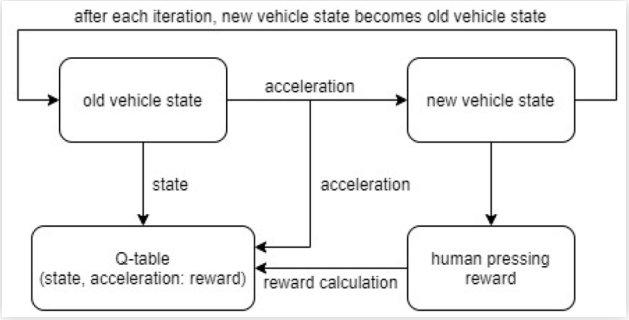
Initialization: $Q - table(Q) = zeros(s, a)$, s, leadv

Output: Updated Q-table

for $GP = GP_{min}$ to GP_{max} **do**
  for $x = 0 : B_{length} : T_{length}$ **do**
    j ← **x**;
    for i = 0 to iteration **do**
      j ←increment.
      trending ← LV(**j**) compare to leadv
      leadv ←LV(**j**).
      a = **eps**(**Q**(**s**)).
      v ← **v**(**s**) + **a**.
      rv ← **v** − **leadv**.
      gap ← **gap**(**s**) + **rv**.
      $\mathbf{s}_{t+1}$ ← **v**, **gap**, **rv**, **trending**
      if safety penalty **then**
        reward = $SR(s_{t+1}) * w_2$
        j ← **x**.
        leadv ←LV(**j**).
        v ← **leadv** ± Δv
        gap ← $w_{GP} *$ **GP** + $w_v *$ **leadv** ± Δgap
      else
        reward ←**CR**$(s_{t+1}) * w_1 + \eta *$ **Q**$(s_{t+1})_{max}$
      end if
      Q(s, a) ← **Q**$(s, a) + \theta *$ (**reward** − **Q**(s, a))
      s ← $\mathbf{s}_{t+1}$.
    end for
  end for
end for
return Q

In this paper, the Q-Learning is applied to a single-lane car-following scenario where the host vehicle follows the lead vehicle. During the training, a Q-table-based controller is used to control the host vehicle moving from initial states to new states. As shown in Figure 2, at the start of each Q-Learning iteration, the host vehicle starts at certain initial states, an acceleration from available options will be assigned using the Epsilon-Greedy Action Selection method, and the vehicle will run with this acceleration for the next 1 second. The traveled distances, velocities, and accelerations of both vehicles, together with the relative velocity and the gap are also updated. Then, the vehicle will reach new states and the reward will be decided based on the new states. The Q-table is then updated based on the reward and state transition. Such training will be repeated until the iteration limit is reached. After the training, the trained Q-table will be able to control the host vehicle effectively. The details of the learning process are given in Algorithm 1.

FIGURE 2 Reinforcement learning flow chart.

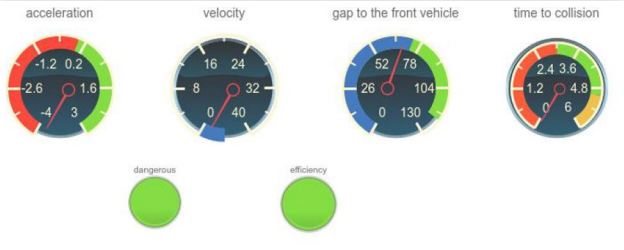


The lead vehicle's velocity trajectory is prerecorded and is divided into multiple sub-trajectory batches. T_{length} and B_{length} are the lengths of the complete training trajectory and sub-trajectory respectively in seconds. The comfort reward policy has been defined in previous sections. GP denotes the gap parameter. The larger the gap parameter is, the larger the initial gap will be after each reset. All the batches will run $N_{gap} * iteration$, where N_{gap} is the number of optional values of the gap parameter. Other training parameters include learning rate θ and discount factor η. The Q-table is initialized as all the Q-value are zeros. For each iteration, there will be one action performed at one state, and one Q-value is updated. The *eps* denotes the Epsilon-Greedy Action Selection method used to pick the action. When the new state triggers the safety penalty, the states will be reset to a random initial gap and velocity. Although the front vehicle's state trajectory is fixed, the initial gap and the host vehicle's velocity are random after each reset.

Adaptation of Human to Autonomy

The human's in-situ feedback may not always be reasonable. The human participants could have improper expectations of the automated driving style. For example, an aggressive participant could prefer a too short headway distance that may cause accidents under emergent scenarios. A conservative participant could want the vehicle to drive too slowly, which reduces the traffic efficiency significantly. Thus, apart from learning a driving model from human expectations, we also propose to adapt to the human's expectations to autonomy by providing necessary feedback to the human.

FIGURE 3 Feedback interface for human participants.



After a trained Q-table is obtained, the participants will be asked to experience an automated car-following ride again. This time, the host vehicle is controlled by the trained Q-table while the front vehicle's motion trajectory comes from test drive cycles, some of which are the same as those during training, and some are not. At the same time, safety and efficiency warnings will be provided, as shown in Figure 3.

The safety warning light will be turned on if the time to collision is smaller than 3*s*. The efficiency warning light will be turned on if the time to collision is larger than 5*s*. During the process, the participants can see the warning light if the driving is too dangerous or too inefficient. Meanwhile, the participants have the freedom to press the button whenever they want. For example, the participants can still decide to press the button if they think the vehicle drive too slowly when the safety warning is on. The new data collected will be used to evaluate the result of Q-learning. The comparison of the data before and after the participants see the warning can reveal if the adaptation works.

Experimental Results and Analysis

The experiments in this paper were conducted using a real-time driving simulator build with Matlab. The interface of the 3D simulation environment is shown in Figure1.

Results of Reinforcement Learning with In-Situ Human Feedback

During the in-situ human feedback data collection, the lead vehicle was following the EPA Urban Dynamometer Driving Schedule (UDDS) cycle, which is shown in the top figure in Figure 4. Figure 5 shows the converted, normalized, and regressed pressing force data.

The state space of the MDP, like introduced in previous sections, contains the host vehicle's speed, gap, RV, and the front vehicle's trend. In the experiment, there are 41 different speed states that are evenly distributed from $0m/s$ to $40m/s$, 111 gap states from $10m$ to $120m$, 13 relative velocity states from $-6m/s$ to $6m/s$, and 3 front vehicle trend states. The action space contains only the host vehicle's acceleration, which has 8 options: $-4m/s^2$, $-3m/s^2$, $-2m/s^2$, $-1m/s^2$, $0m/s^2$, $1m/s^2$, $2m/s^2$, and $3m/s^2$. Thus, the dimensions of the Q-Table is 41(*velocity*) $*$ 111(*gap*) $*$ 13(*relativevelocity*) $*$ 3(*trending*) $*$ 8(*acceleration*). The EPS values and other training related parameters used are shown in Table 1.

FIGURE 4 Trajectories of the front vehicle.

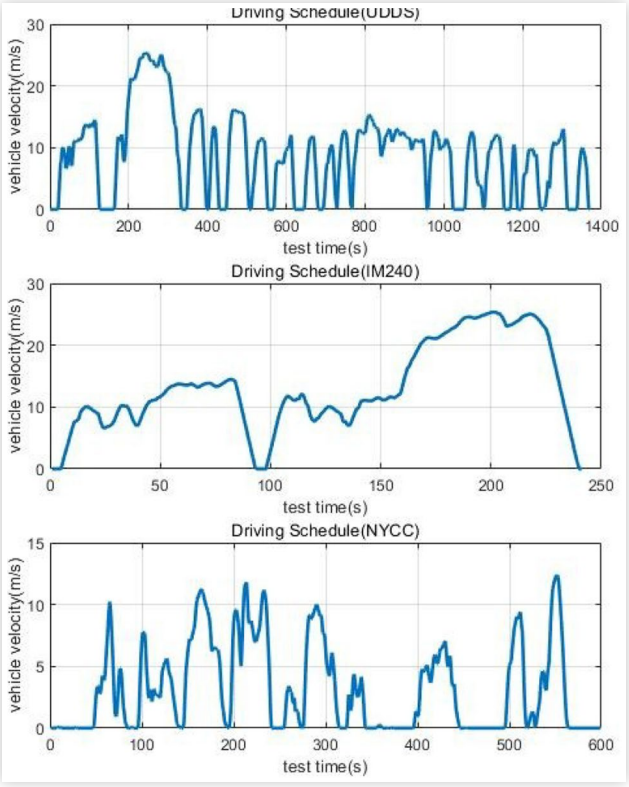
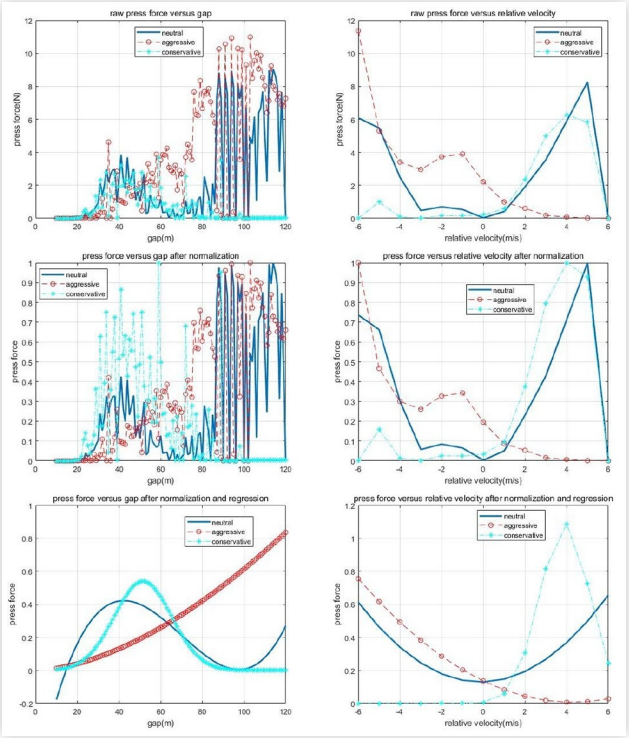


FIGURE 5 Raw press force versus gap and relative velocity.



Multiple virtual automated driving rides were designed to evaluate how well do the trained Q-learning controllers work. The rides could be classified into two main categories: objective quantitative evaluation and subjective evaluation. The objective quantitative evaluation focuses on analyzing quantitative data such as total reward value, state statistics (e.g., Average gap and change of speed), and the number of "accidents" that happen. In this paper, "accidents" are represented by the safety penalty in training. In subjective evaluation, the participants will experience the virtual ride in the host vehicle under the control of the Q-controller and evaluate the performance of the Q-controller directly.

The weights of the safety and comfort rewards need to be figured out before further experiments. Five sets of weights were tried to train the agent. The trained agent then drove the vehicle to follow the front vehicle that tracked the UDDS cycle. All training parameters remained the same for all training. All the accidents caused by the agent were counted. The results are shown in Table 2. The results reveal that it is impossible to train a reliable agent with a small w_2. Therefore, in this paper, all the training used 50 for w_2 and 1 for w_1.

TABLE 1 training parameters.

Training Parameters	Value	Iteration Percentage	EPS Value
learning rate(η)	0.5	10%	0.9
iteration	20000	20%	0.8
batch size(B_{length})	50	40%	0.6
batch number	27	60%	0.4
discount factor(θ)	1	80%	0.2
Δv	6	90%	0.1
Δgap	5	99%	0
w_{GP}	10		
w_v	2		

TABLE 2 Number of accidents with different reward weights.

w_1	w_2	Accidents
–1	0	648
–1	–1	524
–1	–10	124
–1	–25	27
–1	–50	0

TABLE 3 reward collection.

LT	IDM	Agg	IDM	Neu	IDM	Cons
UDDS	–613.6	–335.65	–628.9	–557.38	–597	–384.48
IM240	–145.9	–74.39	–113.28	–99.68	–108.6	–76.74
NYCC	–172.1	–101.92	–280.32	–138.96	–232.04	–128.5

TABLE 4 Accidents record.

LT	Agg	Neu	Cons
UDDS	0	1AC5	2AC2, 2AC3
IM240	0	4AC3, 1AC4	0
NYCC	3AC3	3AC3, 1AC4	1AC3, 1AC4, 3AC5

Evaluation of Learning Human Comfortable Expectations

In this section, three Q-controller agents trained with different comfort reward policies learned from 3 different types of participants (aggressive (Agg), neutral (Neu), and conservative (Cons)) are evaluated. Aggressive participants expect smaller headway and higher speed. Conservative participants expect larger headway and lower speed. All the Q-controllers were trained with the same training parameters and lead trajectory (UDDS). Parameters setting is shown in Table 1.

Each controller ran three tests following three different driving cycles. One is the same cycle as the training (UDDS), and the others are new cycles. The two new cycles were the Inspection and Maintenance (IM240) and the New York City Cycle (NYCC), which are also shown in Figure 4. An IDM car-following controller ran the same tests as a baseline.

The first quantitative data are the reward collection. In this paper, reward collection denotes the cumulative reward a trained agent can gather in one car-following test. Reward gathering ability is an essential index that reflects the performance of Q-learning. The results are shown as table 3. The results show that all the controllers could collect higher rewards than the base IDM controllers in all three lead cycles, proving that the Q-training successfully adapted the vehicle behavior to the human's expectations.

The second quantitative data are the number of "accidents". A successfully trained Q-controller should minimize the number of "accidents". In this paper, five classes of "accidents" are defined. Accident Class1(AC1) is defined as the gap being too small. Whenever the gap is smaller than $10m$, an AC1 event will be triggered and recorded. Accident Class2(AC2) is defined as the gap being too large. Whenever the gap is larger than $120m$, an AC2 event will be triggered and recorded. Accident Class3(AC3) is defined as relative velocity being too small. Whenever the relative velocity is smaller than $-6m/s^2$, an AC3 event will be triggered and recorded. Accident Class4(AC4) is defined as relative velocity being too large. Whenever the relative velocity is large than $6m/s^2$, an AC4 event will be triggered and recorded. Accident Class5(AC5) is defined as vehicle reversing. Whenever the velocity is smaller than $0m/s$, an AC5 event will be triggered and recorded. The numbers of occurred "accidents" are shown as table 4.

The length of UDDS is $1370s$, IM240 is $240s$, NYCC is $599s$. The frequency of "accidents" is relatively low but not low enough. A well-trained Q-controller should avoid "accidents" completely. However, the cause of most "accidents" is not directly related to the Q-learning algorithm. AC3 and AC4 happen when relative velocity is too large or too small. In this paper, the available acceleration options are limited. If the lead vehicle accelerates or decelerates dramatically, AC3/4 cannot be avoided even if the most optimized acceleration option is chosen. Similarly, AC2 happened when the lead vehicle accelerates too fast with a big gap. For instance, if the

current gap is 119m, the current relative velocity is 0m/s, and the front vehicle has an acceleration of 5m/s^2; there is no way the agent can avoid AC4 to happening with the currently available actions. Under-coverage causes AC5. With full coverage, the agent will not be able to decelerate to negative velocity. Therefore, all the AC can be avoided with a larger iteration number and range of states and actions.

The third quantitative data are the gap and Time to Collision (TTC). Gap and TTC are two critical values that determine the driving style. The average gap and TTC for each Q-controller in each driving cycle are shown in table 5.

The results show that the aggressive controller has the smallest gap and TTC, the conservative controller has the largest gap and TTC, and the neutral controller has the medium gap and TTC. These results show that the trained Q-controller matched the different styles of participants. This is another piece of evidence that proves the Q-learning adaption is working.

Evaluations of Human Comfort Improvement

The gap and relative velocity trajectories of all Q-controllers in all cycles are shown in Figure 6, 7, and 8. It can be seen from the figures that the aggressive Q-controller kept a smaller gap than the other Q-controllers while the conservative Q-controller kept a larger gap. The relative velocity trajectories are not showing as big differences between each other as the gap trajectories. However, it is safe to say the aggressive Q-controller is trying to avoid negative RV when the conservative Q-controller is trying to avoid positive RV. The neutral controller performed in between the aggressive and the conservative controller. These driving styles corroborate how each type is defined.

We asked all the participants to experience the autonomous driving journey again. During this time, the host vehicle was controlled by the Q-controller trained with the data generated by the corresponding participant. They had access to the safety and efficiency warning signal, which would warn them if the acceleration option they did not want to be had been the most aggressive or conservative option. Figure 9 - 11 show the raw pressing force data collected from different participants during the IDM ride and the Q-controller ride. In the figures, the legend *IDM* refers to the original IDM controller, legend $Q-$ *controller*1*th* refers to the learned RL-based controller, and legend $Q-$ *controller*2*nd* refers to the RL-based controller with warnings for the participants. The results show that the aggressive and the neutral participants felt much more comfortable and pressed less during the Q-controller ride than during the IDM ride. However, the conservative participant felt a little more comfortable with the IDM. The reason is the initial IDM setting is already extremely conservative. There is no significant room for improvement. For the second virtual ride, since the participants had access to the warning notification, the participants knew that what Q-controller was doing was the best under safety constraints. The results show that the aggressive and the conservative participants pressed much less in the second virtual ride. It proves that showing participants the safety warning can increase ride comfort if the participants have improper and extreme driving expectations. The human-vehicle interface could correct Their improper expectations. The result shows that the neutral participants' average pressing force decreased the most dramatically (decrease from 1.042 to 0.0503, 95.2%), which indicates that

TABLE 5 Average Gap(m)/TTC(s).

LT	Agg	Neu	Cons
UDDS	30.69/2.75	31.91/2.83	43.31/4.11
IM240	41.20/3.23	44.83/3.36	52.70/4.09
NYCC	18.73/2.13	18.98/2.35	20.81/2.78

FIGURE 6 Gap and relative velocity trajectory on UDDS.

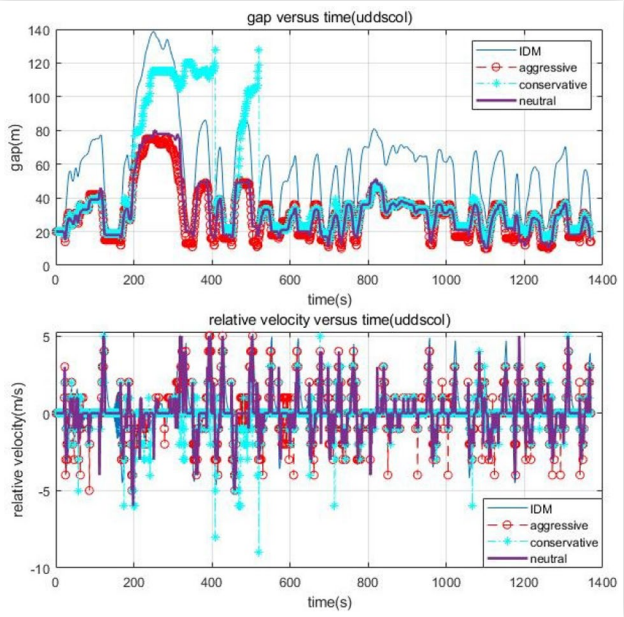


FIGURE 7 Gap and relative velocity trajectory on IM240.

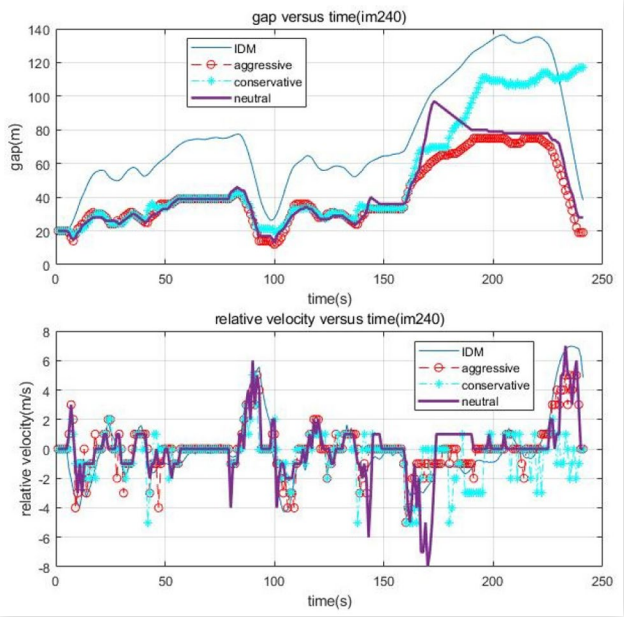


FIGURE 8 Gap and relative velocity trajectory on NYCC.

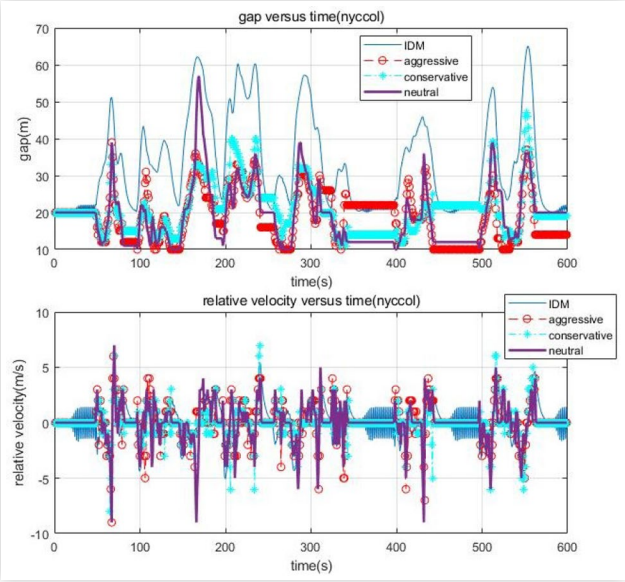


FIGURE 9 Comparison of Evaluation of controller (aggressive).

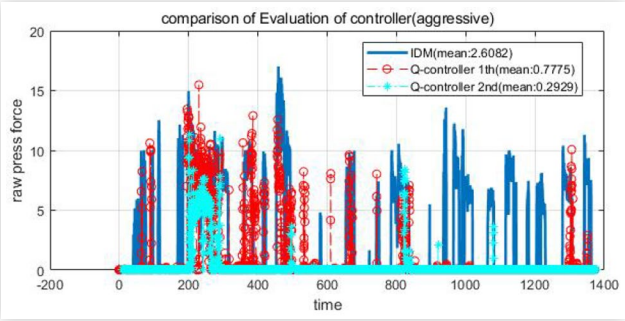
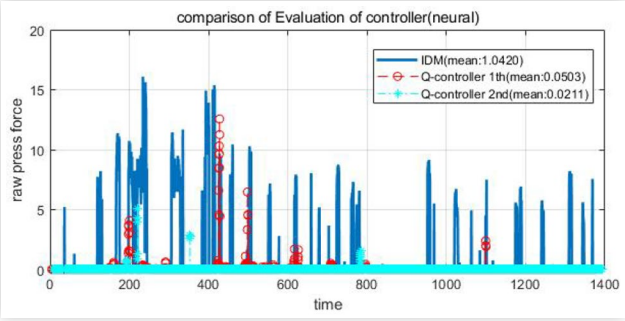


FIGURE 10 Comparison of Evaluation of controller (neutral).



this method works exceptionally effectively in driving behaviors for neutral participants. Because of the safety penalty, the trained agent tended to drive neutrally to avoid triggering the safety penalty. It explains why the trained agent satisfied the neutral participant the most even though the reward collection score is not impressive.

FIGURE 11 Comparison of Evaluation of controller (conservative).

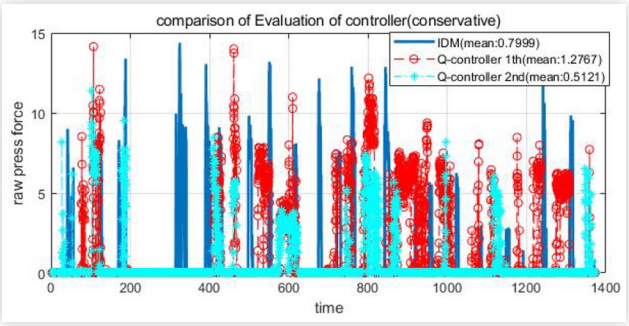
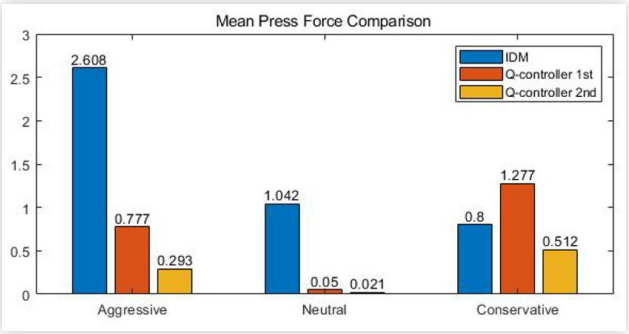


FIGURE 12 Comparison of Evaluation of controller.



Conclusion

In this paper, a reinforcement learning-based method is proposed to adapt automated driving behaviors to human expectations for better ride comfort. The method utilizes human in-situ feedback to generate the Q-table for the learning process. We also propose to use the trained Q-learning controller to correct bad expectations from humans by providing necessary warning feedback. Experiments have been conducted on three different types of human participants. The results showed that the trained automated driving agent could increase the ride comfort for the corresponding participant. Simultaneously, the human participants could acquire effective warning information when they are having improper feedback. Such information successfully helped the participants to develop better expectations during automated driving rides. In future work, we will apply this proposed method to a wider range of autonomous driving scenarios such as lane switching, cornering, and overtaking.

References

1. Duarte, F. and Ratti, C., "The Impact of Autonomous Vehicles on Cities: A Review," *Journal of Urban Technology* 25, no. 4 (2018): 3-18.

2. Ard, T., Guo, L., Dollar, R.A., Fayazi, A. et al., "Energy and Flow Effects of Optimal Automated Driving in Mixed Traffic: Vehicle-in-the-Loop Experimental Results," *Transportation Research Part C: Emerging Technologies* 130 (2021): 103168.
3. Elbanhawi, M., Simic, M., and Jazar, R., "In the Passenger Seat: Investigating Ride Comfort Measures in Autonomous Cars," *IEEE Intelligent Transportation Systems Magazine* 7, no. 3 (2015): 4-17.
4. Su, H. and Jia, Y., "Study of Human Comfort in Autonomous Vehicles Using Wearable Sensors," *IEEE Transactions on Intelligent Transportation Systems* (2021).
5. Corbridge, C., "Vibration in Vehicles: Its Effect on Comfort," PhD thesis, University of Southampton, 1987.
6. Xu, F. and Tian, Z.Z., "Driver Behavior and Gap-Acceptance Characteristics at Roundabouts in California," *Transportation Research Record* 2071, no. 1 (2008): 117-124.
7. De Vos, A.P., Theeuwes, J., Hoekstra, W., and Coëmet, M.J., "Behavioral Aspects of Automatic Vehicle Guidance: Relationship Between Headway and Driver Comfort," *Transportation Research Record* 1573, no. 1 (1997): 17-22.
8. Trnros, J., Nilsson, L., Ostlund, J., and Kircher, A., "Effects of Acc on Driver Behaviour, Workload and Acceptance in Relation to Minimum Time Headway," in *9th World Congress on Intelligent Transport Systems ITS America,* ITS Japan, ERTICO (Intelligent Transport Systems and Services-Europe), 2002.
9. Kesting, A. and Treiber, M., "Calibrating Car-Following Models by Using Trajectory Data: Methodological Study," *Transportation Research Record* 2088, no. 1 (2008): 148-156.
10. Hidas, P., "Modelling Vehicle Interactions in Microscopic Simulation of Merging and Weaving," *Transportation Research Part C: Emerging Technologies* 13, no. 1 (2005): 37-62.
11. Ungoren, A.Y. and Peng, H., "An Adaptive Lateral Preview Driver Model," *Vehicle System Dynamics* 43, no. 4 (2005): 245-259.
12. Bolduc, A.P., Guo, L., and Jia, Y., "Multimodel Approach to Personalized Autonomous Adaptive Cruise Control," *IEEE Transactions on Intelligent Vehicles* 4, no. 2 (2019): 321-330.
13. Guo, L., Manglani, S., Liu, Y., and Jia, Y., "Determining Headway for Personalized Autonomous Vehicles by Learning from Human Driving Demonstration," in *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, 1461-1466, IEEE, 2017.
14. Xu, L., Hu, J., Jiang, H., and Meng, W., "Establishing Style-Oriented Driver Models by Imitating Human Driving Behaviors," *IEEE Transactions on Intelligent Transportation Systems* 16, no. 5 (2015): 2522-2530.
15. Dougherty, M., "A Review of Neural Networks Applied to Transport," *Transportation Research Part C: Emerging Technologies* 3, no. 4 (1995): 247-260.
16. Wiest, J., Höffken, M., Kreßel, U., and Dietmayer, K., "Probabilistic Trajectory Prediction with Gaussian Mixture Models," in *2012 IEEE Intelligent Vehicles Symposium*, 141-146, 2012.
17. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B. et al., "End to End Learning for Self-Driving Cars," *arXiv preprint arXiv:1604.07316* (2016).
18. Loiacono, D., Lanzi, P.L., Togelius, J., Onieva, E. et al., "The 2009 Simulated Car Racing Championship," *IEEE Transactions on Computational Intelligence and AI in Games* 2, no. 2 (2010): 131-147.
19. Hasselt, H.V., Guez, A., and Silver, D., "*Deep Reinforcement Learning with Double Q-Learning*," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16*, p. 2094-2100, AAAI Press, 2016.
20. Hartwich, F., Beggiato, M., and Krems, J.F., "Driving Comfort, Enjoyment and Acceptance of Automated Driving-Effects of Drivers' Age and Driving Style Familiarity," *Ergonomics* 61, no. 8 (2018): 1017-1032.
21. Lewis-Evans, B. and Rothengatter, T., "Task Difficulty, Risk, Effort and Comfort in a Simulated Driving Task-Implications for Risk Allostasis Theory," *Accident Analysis & Prevention* 41, no. 5 (2009): 1053-1063.
22. Tatsuno, J. and Maeda, S., "Driving Simulator Experiment on Ride Comfort Improvement and Low Back Pain Prevention of Autonomous Car Occupants," in: , *Advances in Human Aspects of Transportation*, (Springer, 2017), 511-523.
23. Bayerlein, H., De Kerret, P., and Gesbert, D., "Trajectory Optimization for Autonomous Flying Base Station via Reinforcement Learning," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 1-5, 2018.

Acknowledgement

This work was partially supported by the National Science Foundation under Grants CNS-1755771 and IIS-1845779.

Definitions, Acronyms, Abbreviations

AV - Automated Vehicle

RL - Reinforcement Learning

MDP - Markov Decision Process.

ISSN 0148-7191