2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)
October 8-12, 2022, Macau, China

Modeling Human Driver Behaviors When Following Autonomous Vehicles: An Inverse Reinforcement Learning Approach

Xiao Wen, Sisi Jian* and Dengbo He*

2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC) | 978-1-6654-6880-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ITSC55140.2022.9922310

Abstract—**During the transition period, the interactions between human-driven vehicles (HVs) and autonomous vehicles (AVs), especially the car-following behaviors, need to be analyzed comprehensively to provide feedbacks to AV controllers, increase the inference ability of AVs and reflect the social acceptance of AVs. Previous studies have found that HVs behave differently when following AVs compared to when following HVs through traffic/numerical simulations or field experiments. However, these works have critical drawbacks such as simplified driving environments and limited sample sizes. The objective of this study is to realistically model and understand HV-following-AV dynamics and their microscopic interactions. An inverse reinforcement learning model (Inverse soft-Q Learning) has been implemented to retrieve HVs' reward functions in HV-following-AV events. Then a deep reinforcement learning (DRL) approach -- soft actor-critic (SAC) is adopted to estimate the optimal policy for HVs following AVs. HV-following-AV events from the high-resolution (10 Hz) Waymo Open Dataset are extracted to validate the proposed model. The results show that compared with other conventional and data-driven car-following models, the proposed model leads to significantly more accurate trajectory predictions and gains more insights into HVs' car-following behaviors.**

I. INTRODUCTION

In recent years, the technologies of autonomous vehicles (AVs) have been tested through a variety of approaches, including traffic microsimulation tools, numerical simulations, dedicated test tracks and field trials on public roads. It is generally acknowledged that before the mobility is fully automated, there will be a transition period when the traffic flow is composed of both AVs and human-driven vehicles (HVs) [1]. When sharing the road with AVs, human drivers may behave differently as compared to when sharing the road with only HVs. They may take advantage of AVs due to over-trust on them, or drive more conservatively if they have low acceptability to AVs. These differences in human drivers' behaviors can significantly affect road safety and efficiency and should be taken into account when designing AV control algorithms [2]. However, understanding of the fundamental mechanisms of such interactions, e.g., how human drivers adapt to the new driving environments when following AVs, is still yet to be investigated.

The mechanisms behind the interactions between AVs and HVs have not been studied comprehensively due to the lack of empirical data as a result of the low AV market penetration rate [2]. Previous research about the effects of AVs on the mixed traffic flow usually adopted traffic/numerical simulations or field experiments, both of which can make the reliability of their results questionable due to over-simplified traffic characteristics and vehicle interactions in their simulation settings or experiment designs [3]. Nowadays, more and more AV tech firms such as Waymo and Lyft have released high-resolution real-world datasets collected by the sensors mounted on their AV fleets at 10-Hz frequency. These datasets include abundant information about not only the AVs but also the surrounding traffic, which provides the transportation research community with new opportunities to investigate on human drivers' behavioral adaptions when interacting with AVs in real-world.

Imitation learning (IL) approaches have been frequently implemented to learn the underlying patterns from the human demonstrations and generate an optimal policy which behaves similarly to those experts [4]. Main IL approaches include behavior cloning (BC) and inverse reinforcement learning (IRL). BC directly learns a function mapping from states to actions which turns IL into a supervised learning problem. However, BC may cause the so-called "cascading errors" problem since small predictive errors will compound and lead the policy to invalid or unseen situations ultimately. Previous IRL methods such as generative adversarial imitation learning (GAIL) [5] and adversarial inverse reinforcement learning (AIRL) [6] use the adversarial training technique which learns the reward and policy functions separately and train these two jointly in a min-max game. However, the adversarial training makes these methods sensitive to hyperparameter choice or minor implementation details, thus affects their performance adversely.

To tackle the above issues, the inverse soft-Q Learning (IQ-Learn) algorithm [7] has been introduced. IQ-Learn is a novel IRL method that estimates only the Q-function, representing both reward and policy. Therefore, the min-max problem in GAIL and AIRL can be converted to a simple minimization problem over the Q-function, which enables IQ-Learn to achieve state-of-the-art results in IL settings. This study aims

X. Wen, and S. Jian, and D. He are with the Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong SAR. D. He is also with the Intelligent Transportation Thrust, The Hong Kong University of Science and Technology (Guangzhou), China. (email: xwenan@connect.ust.hk; cesjian@ust.hk; dengbohe@ust.hk). *: Corresponding authors

to imitate human drivers' trajectories when following AVs on highways and recovering their reward functions using IQ-Learn along with a deep reinforcement learning (DRL) approach -- soft actor-critic (SAC). In summary, the main contributions of this paper are listed as follows:

1) We apply an IRL approach – IQ-Learn to model human drivers' behaviors when following AVs. Instead of based on the data collected from microsimulation or field experiments, this study extracts the HV-following-AV events from the real-world dataset collected by AVs on public roads. We demonstrate that IQ-Learn can achieve higher predictive accuracy compared to conventional and data-driven car-following models and former IRL methods.
2) We use the IQ-Learn algorithm to extract reward functions of human drivers in car-following events and identify human drivers' preferred states (i.e., speed, spacing and relative speed) when they are behind AVs. The knowledge can be used for AV tech firms to infer the following human drivers' behaviors and improve the performance of their AV controllers.

The paper is organized in the following manner. The next section briefly reviews relevant studies using IL to model human driver behaviors. Sections III presents the methodologies and data sources used in this study. Sections IV discusses the recovered reward functions and the performance of the proposed model in reproducing car-following behaviors. Finally, Section V provides the conclusions and recommendations for future research.

II. RELATED WORK

Among a great deal of literature on modelling car-following behaviors, popular approaches include physics-based models and data-driven models. However, the physics-based models may show poor predictive performance especially in highly complex environments. For data-driven models, machine learning (ML)-based models may fail due to large, stochastic and continuous state space of the car-following problem and reinforcement learning (RL)-based models are difficult to craft the reward functions to perfectly encode human drivers' desired behaviors.

In recent years, applying IL methods to inferring optimal sequential policies from human demonstrations has been in the spotlight. [4] applied GAIL to the task of modeling human driver behaviors on the simulation platform where expert demonstrations were retrieved from the NGSIM dataset. They identified that compared to earlier IRL methods, GAIL has many outstanding properties, such as emergent driving behaviors and assigning high likelihood to expert actions. In another study, [8] combined GAIL and Parameter Sharing Trust Region Policy Optimization (PS-TRPO) to enable IL in the multi-agent setting. Experiment results showed that compared to the existing single-agent models, the multi-agent model generated significantly more realistic behavior, particularly over longer time horizons. Later, [9] modeled human driver heterogeneity by incorporating a social preference value (SVO) into one agent's reward functions. SVO improved the model predictive performance by quantifying the degree of an agent's selfishness or altruism. An IRL algorithm was trained for the AV to observe HVs, estimate their SVOs, and generate a control policy in real time. A reward function-based driver model that imitated human's decision-making mechanisms was proposed in [10]. They assumed that human driver behaviors consist of three processes, namely trajectory generation, trajectory evaluation, and trajectory selection. This setting converted the continuous behavior modeling problem to discrete space, thus made maximum entropy inverse reinforcement learning (IRL) tractable to learn reward functions.

In general, most of the studies focused on emulating interactions within human drivers or learning the expert policy from human demonstrations for AVs. However, studies found that human drivers may adapt to the new driving environments when following AVs [3], which highlights the necessity to evaluate drivers' behavior adaptation. IRL, which assumes that the expert follows an optimal policy with respect to an unknown reward function, can be used as a promising approach to studying HV-AV interactions.

III. METHODOLOGY

A. Problem Formulation

In HV-following-AV events, the driver state s_t at time step t is described by the features including: the speed of the following vehicle $V_n(t)$, the inter-vehicle spacing $S_{n-1,n}(t)$ and the relative speed between the following and the lead vehicles $\Delta V_{n-1,n}(t)$. The action $a_n(t)$ is defined as the longitudinal acceleration of the following vehicle. The simulation environment initializes HVs and AVs with information about their initial positions and speed. At time step t, the action taken by the HVs is sampled from the learned optimal policies. Then the state will be calculated using Newtonian equations of motion as follows:

$$V_n(t+1) = V_n(t) + \Delta T * a_n(t) \quad (1)$$

$$\Delta V_{n-1,n}(t+1) = V_n(t+1) - V_{n-1}(t+1) \quad (2)$$

$$S_{n-1,n}(t+1) = S_{n-1,n}(t) - \Delta T * \frac{\Delta V_{n-1,n}(t) + \Delta V_{n-1,n}(t+1)}{2} \quad (3)$$

where ΔT is the simulation time interval ($0.1s$), which is the data collection interval; $V_{n-1}(t+1)$ is the speed of the leading vehicle which is considered known over time.

The training process of one car-following event is set as an episode in this study. When a traffic crash happens (i.e., $S_{n-1,n}(t) \leq 0$) or the simulation reaches the maximum time step, the state will be re-initialized using the next car-following event data.

B. Baselines

The performance of IQ-Learn is compared to four baseline models, including a physics-based car-following model (i.e.,

Intelligent driver model (IDM)), a data-driven model (i.e., Long short-term memory (LSTM) neural network), and two IRL models (i.e., GAIL and AIRL). These baseline models are described as below.

Intelligent driver model (IDM): The IDM [11] was originally proposed to model highway bottleneck congestions. IDM shows collision-free dynamics and implements smooth transitions between acceleration and deceleration. The acceleration/deceleration generated by IDM considers both the desired speed and the desired distance, which is presented in the following equation:

$$a_n(t) = a_{max}^{(n)}\left(1 - \left(\frac{V_n(t)}{\widetilde{V_n}(t)}\right)^{\beta} - \left(\frac{\widetilde{S_n}(t)}{S_n(t)}\right)^2\right) \quad (4)$$

where $a_{max}^{(n)}$ is the maximum acceleration/deceleration of the following vehicle; $V_n(t)$ is the speed of the following vehicle; $\widetilde{V_n}(t)$ is the desired speed; $S_n(t)$ is the spacing between the two consecutive vehicles; β is the parameter which is usually fixed at 4.

The desired spacing $\widetilde{S_n}(t)$ is given by:

$$\widetilde{S_n}(t) = S_{jam}^{(n)} + \max\left(0, V_n(t)\widetilde{T_n}(t) + \frac{V_n(t)\Delta V_n(t)}{2\sqrt{a_{max}^{(n)}a_{comfort}^{(n)}}}\right) \quad (5)$$

where $S_{jam}^{(n)}$ is the minimum spacing at standstill; $\widetilde{T_n}(t)$ is the desired time headway; $\Delta V_n(t)$ is the relative speed; $a_{comfort}^{(n)}$ is the comfortable deceleration.

Long short-term memory (LSTM) neural network: Similarly, the inputs of the LSTM consist of the following vehicle speed, spacing and relevant speed at time step t. The output is the longitudinal acceleration of the following vehicle at the next time step. The state for the next step will be updated accordingly. The objective function for the LSTM model is described as follows:

$$C(W, B) = \frac{(S_{n-1,n}(t) - S_{n-1,n}^{obs}(t))^2}{(S_{n-1,n}^{obs}(t))^2} \quad (6)$$

where $S_{n-1,n}(t)$ is the simulated spacing at time step t, and $S_{n-1,n}^{obs}(t)$ is the observed spacing at time step t; W and b represent the weights and biases in the LSTM model. The LSTM model gradually minimizes the objective function by back-propagating a small update through time in the direction of optimizing the weights and biases [12].

Generative adversarial imitation learning (GAIL): [5] proposed GAIL which solves the IL problem by matching the state-action occupancy distribution of the expert policy. A discriminator (D_ψ) parametrized by ψ is trained to learn to distinguish whether a trajectory is from the expert demonstrations (π_E) or synthetic demonstrations generated by the policy (π_θ). The policy (π_θ) parameterized by θ is trained to generate synthetic trajectories to "fool" the discriminator (D_ψ). The objective function of GAIL is formulated as a min-max game between the discriminator (D_ψ) and the policy (π_θ):

$$\min_\theta \max_\psi E_{\pi_E}[logD_\psi(s,a)] + E_{\pi_\theta}[log(1 - D_\psi(s,a))] \quad (7)$$

In order to fit π_θ, a surrogate reward function can be calculated as:

$$\tilde{r}(s_t, a_t; \psi) = -\log(1 - D_\psi(s_t, a_t)) \quad (8)$$

As the state-actions pairs (s_t, a_t) sampled from π_θ become more and more similar to the elements sampled from π_E, the value of the reward function will increase. After performing rollouts, surrogate reward function $\tilde{r}(s_t, a_t; \psi)$ is calculated and proximal policy optimization (PPO) is used to update the policy parameters.

Adversarial inverse reinforcement learning (AIRL): Based on Guided Cost Learning (GCL) and adversarial training, [6] proposed AIRL, an efficient sampling-based approximation to Maximum entropy inverse reinforcement learning (MaxEnt IRL). The discriminator (D_ψ) parametrized by ψ is formulated as the follows:

$$D_\psi(s,a) = \frac{exp(f_\psi(s,a))}{\exp\left(f_\psi(s,a)\right) + q_\theta(a|s)} \quad (9)$$

where $f_\psi(s,a)$ is the learned function and trained to infer the reward function; $q_\theta(a|s)$ is the probability of the adaptive sampler and trained to minimize the Kullback-Leibler (KL) diverge between the trajectory distribution induced by the reward function and that generated by the policy (q_θ).

The discriminator (D_ψ) and the policy (q_θ) are trained to maximize the objective functions given by Eqs. (10) and (11):

$$\max_\psi E_{\pi_E}\left[logD_\psi(s,a)\right] + E_{q_\theta}\left[log(1 - D_\psi(s,a))\right] \quad (10)$$

$$\max_\theta E_{q_\theta}\left[logD_\psi(s,a) - log(1 - D_\psi(s,a))\right] \quad (11)$$

f_ψ is further decomposed to a reward estimator g_ψ and a potential function $h_\emptyset$:

$$f_{\psi,\emptyset}(s,a,s') = g_\psi(s,a) + \gamma h_\emptyset(s') - h_\emptyset(s) \quad (12)$$

where ψ and $\emptyset$ are parameters trained to maximize the objective functions of the discriminator (D_ψ) in Eq. (10); γ is the discount factor. Similarly, PPO is adopted as the policy optimization algorithm using the estimated reward function.

C. IQ-Learn

Both GAIL and AIRL adopt adversarial training strategy and formulate the IRL problem as a min-max game between reward and policy, which is sensitive to hyperparameter choices or minor implementation details [7]. IQ-Learn is capable of learning a single Q-function that represents both reward and policy. Hence, the complicated min-max game in traditional IRL settings is converted to a simple minimization problem over the Q-function. The pseudocode is presented in Algorithm 1. For continuous control, IQ-Learn is built upon a DRL algorithm -- soft actor-critic (SAC). The modified actor-critic update rules include:

1) For a fixed policy π_ϕ, optimize the Q-function by maximizing $\mathcal{J}(\pi_\phi, Q_\theta)$ as Eq. (13) shows:

$$\mathcal{J}(\pi_\phi, Q_\theta) = E_{\rho_E}\left[\phi\left(Q_\theta - \gamma E_{s'\sim\mathcal{P}(\cdot|s,a)} V^{\pi_\phi}(s')\right)\right] - (1-\gamma)\, E_{\rho_0}[V^{\pi_\phi}(s_0)] \quad (13)$$

2) For a fixed Q_θ, optimize π_ϕ by implementing SAC update.

The recovered policy of IQ-Learn is presented in Algorithm 2. IQ-Learn recovers the reward functions for each transition (s, a, s') using the learned Q-function as follows:

$$r(s, a, s') = Q(s, a) - \gamma V^{\pi}(s') \quad (14)$$

Algorithm 1 Inverse soft Q-Learning

1: Initialize an Q-function Q_θ and random policy π_ϕ
2: **for** step t in $\{1,2,3, \dots N\}$ **do**
3: Train Q-function using the objective $\mathcal{J}(\theta)$ from Eq. (13):
$\theta_{t+1} \leftarrow \theta_t - \alpha_Q \nabla_\theta[-\mathcal{J}(\theta)]$
4: Improve policy π_ϕ with SAC style actor critic:
$\phi_{t+1} \leftarrow \phi_t - \alpha_\pi \nabla_\phi E_{s\sim D, a\sim\pi_\phi(\cdot|s)}[Q(s,a) - log\pi_\phi(a|s)]$
5: **end for**

Algorithm 2 Recover policy and reward

1: Given a trained Q-function Q_θ and trained policy π_ϕ
2: Recover policy π:
$\pi \coloneqq \pi_\phi$
3: For state s, actions a and next state $s'\sim\mathcal{P}(\cdot\,|s,a)$
4: Recover reward $r(s, a, s') = Q_\theta(s, a) - \gamma V^{\pi}(s')$

IV. EXPERIMENT RESULTS

A. Data Sources

The car-following events that are used to train the models have been extracted from the Waymo Open Dataset released by Waymo LLC, in which a fleet of SAE Level 4 vehicles has been tested on public roads in the U.S. [13][14]. The Waymo Open Dataset is constituted of two parts: the perception and motion parts. These two parts both contain high-quality and continuous records of road agents' type, size (e.g., length, width and height), position (e.g., latitude and longitude) and movement (e.g., speed) at 10-Hz frequency.

The perception part contains 1,000 20-second video clips, each of which is composed of well-synchronized and calibrated high-resolution LiDAR and camera data recorded in urban and suburban areas [14]. The lidar data contains approximately 12 million annotated 3D ground truth bounding boxes and the camera data also contains 12 million annotated 2D fitting bounding boxes.

The motion part consists of 103,354 20-second video clips representing 574 hours of driving data collected over 1,750 km of roadways. Similar to the perception part, each clip in the motion part contains high-quality 3D ground truth bounding boxes for each road user. Compared to the perception part, the motion part additionally provides corresponding high-resolution maps.

Considering the sample size of the Waymo Open Dataset, timespan of each video clip and sensor detection range, the car-following events used in this study must satisfy the following criteria:

1) The leading and following vehicles were driving in the same lane on a straight highway segment;
2) Neither lead vehicle nor following vehicle changed lanes in the event;
3) The spacing distance between the leading and following vehicles should be less than 85m;
4) The following vehicle's speed should be greater than 10km/h to exclude traffic congestion scenarios;
5) The duration of car-following event should be at least 15 seconds long.

A total of 264 HV-following-AV events are extracted from the dataset.

B. Driver Behavior Classification

In this paper, drivers' heterogeneous longitudinal maneuvering styles are captured by clustering. Critical features including maximum speed, minimum speed, speed mean, speed standard deviation, acceleration, deceleration, spacing, and time headway are used. First, the principal component analysis (PCA) is adopted to reduce the dimension of the features. Afterwards, the agglomerative hierarchical clustering is used to classify the drivers in the HV-following-AV events. 201 drivers are identified as either non-aggressive or aggressive drivers while the remaining 63 drivers belong to smaller clusters and are discarded in the subsequent analysis. More details on the classification of car-following styles can be found in our previous study [3]. Table I summarized the statistics of non-aggressive and aggressive drivers. It should be noted that the presented values are the aggregation and average of vehicular kinematics of corresponding human driver groups. For each group of drivers, 80% of the total car-following events are randomly selected for training and the remaining 20% car-following events are used for testing.

TABLE I
CLUSTERING INFORMATION

Features	Non-aggressive	Aggressive
maximum vehicle speed (m/s)	12.407	12.153
minimum vehicle speed (m/s)	8.881	6.288
Mean vehicle speed (m/s)	10.768	9.025
Standard deviation of vehicle speed (m/s)	1.019	1.787
Vehicle acceleration (m^2/s)	0.287	0.494
Vehicle deceleration (m^2/s)	0.240	0.471
Spacing (m)	21.657	14.094
Time headway (s)	2.745	2.239
Number of drivers	89	112

C. Performance Comparison

As suggested by Punzo and Montanino [15], the root mean squared percentage error of spacing $RMSPE(S)$ and speed $RMSPE(V)$ are adopted as the model evaluation metrics:

$$RMSPE(S) = \sqrt{\frac{\sum_{i=1}^{N}(S_i^{sim} - S_i^{obs})^2}{\sum_{i=1}^{N}(S_i^{obs})^2}} \quad (15)$$

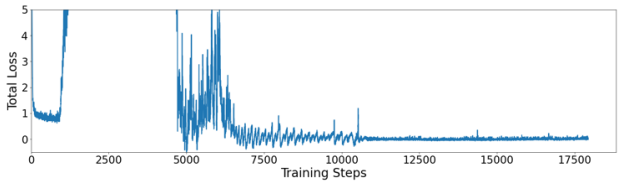
$$RMSPE(V) = \sqrt{\frac{\sum_{i=1}^{N}(V_i^{sim} - V_i^{obs})^2}{\sum_{i=1}^{N}(V_i^{obs})^2}} \quad (16)$$

where S_i^{sim} and V_i^{sim} are the i^{th} simulated spacing and speed, respectively; S_i^{obs} and V_i^{obs} are the i^{th} observed spacing and speed, respectively; N is the number of observations.

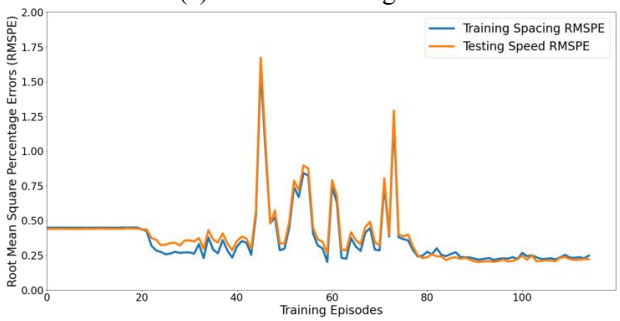
D. IQ-Learn Training and Convergence

For non-aggressive drivers, the IQ-Learn model is trained with 120 episodes with each episode representing a car-following event. During the training process, car-following events are fed into the model sequentially. To detect if there are overfitting issues, the RMSPE of spacing ($RMSPE(S)$) for the entire training and testing dataset is computed whenever a training episode ended. The same training strategy is applied to the aggressive drivers with 170 episodes.

Fig.1 and Fig. 2 show the IQ-Learn training loss and $RMSPE(S)$ of training and testing datasets for non-aggressive and aggressive drivers, respectively. In Fig.1(a), it can be observed that at round 11,000 steps, the training loss almost converged to zero. It should be noted that there are periodical jumps of the training loss which is attributed to the fact that the environment would be re-initialized and the state would be changed abruptly at the end of each training episode. But one can still observe the overall tendency where the training loss kept decreasing and then stabilized. Fig. 1(b) shows the model performance improvement in terms of $RMSPE(S)$. There is a clear decreasing trend in $RMSPE(S)$ throughout the training process and no significant improvement is observed after 85 episodes. Finally, the IRL model that generated the smallest sum of training and testing $RMSPE(S)$ is selected. The same model selection strategy is applied to the aggressive drivers.

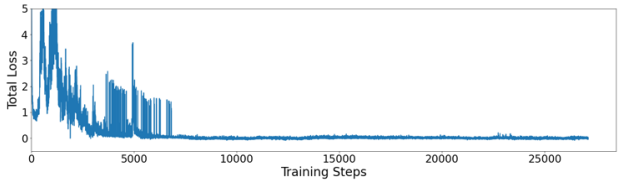


(a) Track of training loss

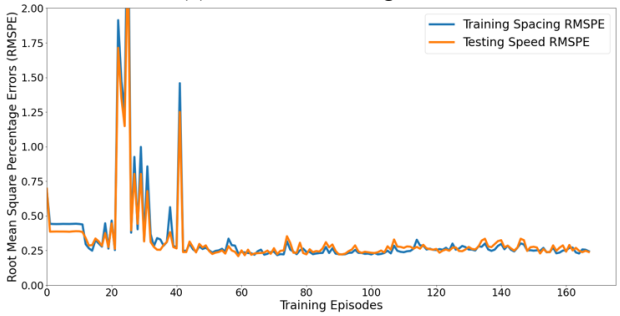


(b) Track of RMSPE of spacing

Fig. 1. Training process for non-aggressive drivers.



(a) Track of training loss



(b) Track of RMSPE of spacing

Fig. 2. Training process for aggressive drivers.

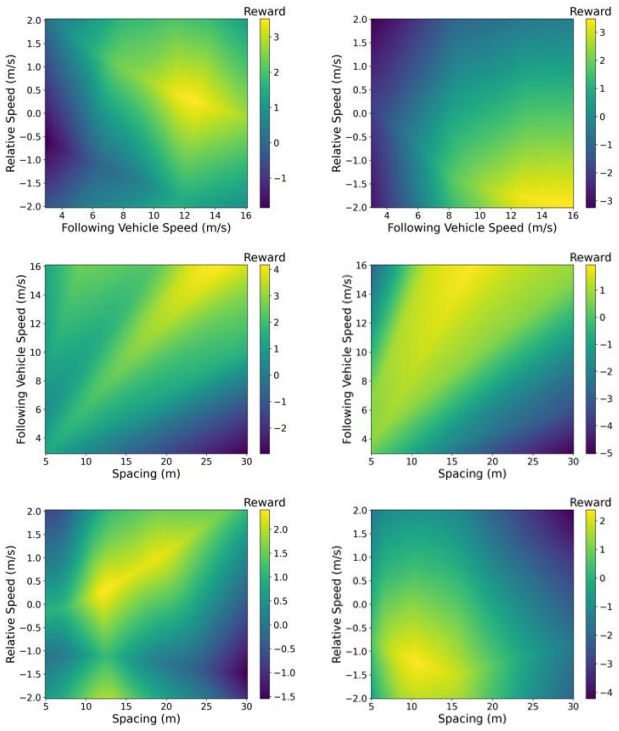
Table II compares the testing performance of IQ-Learn model to four baseline models. As can be shown from Table II, IQ-Learn outperforms the other algorithms in terms of $RMSPE\ (S)$ and $RMSPE\ (V)$ for both non-aggressive and aggressive drivers. It is noteworthy to mention that from the microscopic perspective, there are no perfect car-following models that can completely solve the discrepancy between observed and simulated data. There are always stochasticity or randomness in drivers' behaviors (i.e., some driver behaviors may reveal no perceptible patterns and cannot be predicted).

TABLE II
TESTING RMSPE OF SPACING AND SPEED

	Non-aggressive		Aggressive	
	$RMSPE\ (S)$	$RMSPE\ (V)$	$RMSPE\ (S)$	$RMSPE\ (V)$
IDM	0.277	0.070	0.260	0.075
LSTM	0.321	0.096	0.320	0.140
GAIL	0.247	0.068	0.278	0.089
AIRL	0.344	0.082	0.320	0.119
IQ-Learn	**0.202**	**0.053**	**0.208**	**0.073**

E. Recovery of Reward Functions

The reward functions recovered by applying the IQ-Learn algorithm are visualized in Fig. 3. The reward functions are presented as bivariate feature spaces where the other features are held at their mean values. The brighter the color is, the higher the reward is. Higher rewards indicate that human drivers prefer to stay at corresponding states. The human driver behavior preferences can be inferred through the recovered reward functions, which can provide insights on human drivers' behaviors when interacting with AVs. It should be noted that the inferred behavior preferences based on the reward functions are correlated with the mean values of the other features, i.e., some HV preferences may differ if the values of the other feature change.



(a) Non-aggressive drivers (b) Aggressive drivers

Fig. 3. Reward functions for HVs following AVs.

For non-aggressive drivers shown one the left side of Fig. 3 (see Fig. 3(a)), one can observe an obvious decreasing tendency of preferred relative speed is associated with the acceleration of following human drivers. Especially, if the speed of HVs is between 10 m/s to 14m/s, human drivers preferred to drive slightly faster (i.e., the relative speed is between 0 and 1m/s) than the leading AVs. Second, the preferred spacing to the lead AVs increased linearly with the increase of the HV speed. Furthermore, the increase of the spacing resulted in an increased preferred relative speed. This is reasonable since the following HVs may need to catch up with the lead AVs if the AVs drove away from them.

One the right side of Fig. 3, Fig 3(b) reveals that aggressive drivers preferred to drive at a high speed, ranging from 12 m/s to 14m/s, but are 1m/s to 2m/s slower than the lead AVs. The desired spacing also increased with the increment of the following vehicle speed among aggressive drivers. When the comparisons are made between non-aggressive and aggressive drivers, one can identify that the latter keeps a shorter spacing distance to the AVs than the former given the same speed. Moreover, when the spacing is between 7.5m to 12.5m and the relative speed ranged from $-0.5m/s$ to 1.5m/s, aggressive drivers will get the highest rewards, indicating their preferences at corresponding states.

V. CONCLUSIONS

In this study, a novel IRL algorithm -- IQ-Learn has been proposed to replicate human driver trajectories when following AVs on highway segments. The HV-following-AV events used in this study are extracted from the real-world dataset released by Waymo. Compared to other models such as IDM, LSTM, GAIL and AIRL, IQ-Learn exhibits superior performance for modeling and reproducing interactions between HVs and AVs in terms of $RMSPE(S)$ and $RMSPE(V)$. Moreover, the reward functions based on the IQ-Learn algorithm display the preferences of human drivers when following AVs.

Several research directions worth further investigation. First, it would be an opportunity to compare the reward functions of HV-following-AV and HV-following-HV events. Second, it would be interesting to investigate the effects of AVs on HVs' following trajectories in more complicated scenarios. Third, apart from car-following, other interactions between HVs and AVs need to be examined.

REFERENCES

[1] X. Hu, Z. Zheng, D. Chen, X. Zhang, and J. Sun, "Processing, assessing, and enhancing the Waymo autonomous vehicle open dataset for driving behavior research," *Transportation Research Part C: Emerging Technologies*, vol. 134, Jan. 2022, Art. no. 103490.

[2] X. Di and R. Shi, "A survey on autonomous vehicle control in the era of mixed-autonomy: From physics-based to AI-guided driving policy learning," *Transportation Research Part C: Emerging Technologies*, vol. 125, Mar. 2021, Art. no. 103008.

[3] X. Wen, Z. Cui and S. Jian, "Characterizing car-following behaviors of human drivers when following automated vehicles using the real-world dataset," Accident Analysis & Prevention, vol. 172, Jul. 2022, Art. no. 106689.

[4] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer. "Imitating driver behavior with generative adversarial networks," in *Proc. IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 204-211.

[5] J. Ho, and S. Ermon. "Generative adversarial imitation learning," in *Proc. Advances in neural information processing systems*, 2016, pp. 4565-4573.

[6] J. Fu, K. Luo, and S. Levine. "Learning Robust Rewards with Adversarial Inverse Reinforcement Learning," in *Proc. International Conference on Learning Representations*, 2018.

[7] D. Garg, S. Chakraborty, C. Cundy, J. Song, and S. Ermon. "IQ-Learn: Inverse soft-Q Learning for Imitation," in *Proc. Advances in Neural Information Processing Systems*, 2021.

[8] R. P. Bhattacharyya, D. J. Phillips, B. Wulfe, J. Morton, A. Kuefler, and M. Kochenderfer. "Multi-agent imitation learning for driving simulation" in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1534-1539.

[9] W. Schwarting, A. Pierson, J. Alonso-Mora, S. Karaman, and D. Rus. "Social behavior for autonomous vehicles," *Proceedings of the National Academy of Sciences*, vol. 116, Dec. 2019, Art. no. 50.

[10] Z. Huang, J. Wu, and C. Lv. "Driving behavior modeling using naturalistic human driving data with inverse reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, Jun. 2021.

[11] M. Treiber, A. Hennecke, and D. Helbing. "Congested traffic states in empirical observations and microscopic simulations," *Physical review E*, vol. 62, Aug. 2000, Art. no. 1805.

[12] M. Zhou, X. Qu, and X. Li. "A recurrent neural network based microscopic car following model to predict traffic oscillation," *Transportation research part C: emerging technologies*, vol. 84, Nov. 2017, pp. 245-264.

[13] S. Ettinger, C. Shuyang, C. Benjamin, L. Chenxi, Z. Hang, P. Sabeek, Y. Chai et al. "Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset," in *Proc. IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9710-9719.

[14] P. Sun, K. Henrik, D. Xerxes, C. Aurelien, P. Vijaysai, T. Paul, G. James et al. "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2446-2454.

[15] V. Punzo and M. Marcello. "Speed or spacing? Cumulative variables, and convolution of model errors and time in traffic flow models validation and calibration," *Transportation Research Part B: Methodological*, vol. 91, Sep. 2016, pp. 21-33.