Modeling the Effects of Autonomous Vehicles on Human Driver Car-Following Behaviors using Inverse Reinforcement Learning

Xiao Wen, Sisi Jian* and Dengbo He*

***Abstract*—The development of autonomous driving technology will lead to a transition period where human-driven vehicles (HVs) share the road with autonomous vehicles (AVs). Understanding the interactions between AVs and HVs are critical to traffic safety and efficiency. Previous studies used traffic/numerical simulations and field experiments to investigate HVs' behavioral changes when following AVs. However, such approaches simplify the actual scenarios and may result in biased results. To this end, the objective of this study is to realistically model HV-following-AV dynamics and their microscopic interactions which are important for intelligent transportation applications. HV-following-AV and HV-following-HV events are extracted from the high-resolution (10Hz) Waymo Open Dataset. Statistical test results reveal that there are significant differences in calibrated intelligent driver model (IDM) parameters between HV-following-AV and HV-following-HV. An inverse reinforcement learning model (Inverse soft-Q Learning) is proposed to retrieve HVs' reward functions in HV-following-AV events. A deep reinforcement learning (DRL) approach -- soft actor-critic (SAC) is adopted to estimate the optimal policy for HVs when following AVs. The results show that compared with other conventional and data-driven car-following models, the proposed model leads to significantly more accurate trajectory predictions. Additionally, the recovered reward functions indicate that drivers' preferences when following AVs are different from those when following HVs.**

***Index Terms*—Autonomous vehicles, car-following, vehicle trajectory, driver behavior, inverse reinforcement learning, deep reinforcement learning**

I. Introduction

In recent years, the technologies of autonomous vehicles (AVs) have been tested and deployed through a variety of approaches, including traffic microsimulation, numerical simulations, dedicated test tracks and public road experiments. It is widely 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 roads with AVs, human drivers behave differently compared to when sharing the roads with only HVs [2][3][4]. This difference in human drivers' behaviors can significantly affect traffic safety and efficiency, thus should be investigated comprehensively [5][6]. Among driving behavioral models, car-following models intend to describe the longitudinal interactions between vehicles on the road, which forms the core component in microscopic simulation as well as in traffic flow theory. Therefore, understanding the fundamental mechanisms of such interactions, e.g., how human drivers adapt to the new driving environments when following AVs, remains among the key research questions.

Previous research on car-following interactions between HVs and AVs mainly adopted traffic/numerical simulations or field experiments, due to the lack of empirical data as a result of the low AV market penetration rate [2]. However, traffic microsimulation may simplify and ignore important aspects of traffic characteristics and vehicle interactions. Field experiments cannot reproduce the complicated driving conditions where vehicle speed has higher fluctuations and surrounding traffic interacts with the subject vehicles, both of which can result in biased estimation of AV effects. Recently, more and more AV tech firms such as Waymo and Lyft have released real-world datasets collected by the sensors mounted on a fleet of AVs at 10-Hz frequency. These datasets include abundant information about not only the AVs but also the surrounding environments, as such provides the transportation research community with new opportunities to investigate human drivers' behavioral adaptations when interacting with AVs in reality.

Human drivers' car-following behaviors involve making sequential decisions on longitudinal acceleration based on the motion of the surrounding vehicles. Conventional methods to study car-following behaviors are based on physics-based and data-driven models. However, physics-based models rely on strict assumptions about car-following behaviors and a small set of parameters which may not be generalizable to a variety of driving scenarios; data-driven models based on machine learning (ML) methods are challenging due to the large, stochastic and continuous state space in the highly interactive environment [7]. Compared to physics-based and ML models, the imitation learning (IL) based models are promising for this problem where the agent learns an optimal policy to imitate human demonstration [6][8].

Popular IL approaches include behavior cloning (BC) and inverse reinforcement learning (IRL). However, BC may cause the so-called "cascading errors" problem because small predictive errors will compound and ultimately lead the policy

X. Wen is with Intelligent Transportation, Division of Emerging Interdisciplinary Areas (EMIA) under Interdisciplinary Programs Office (IPO), The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR (e-mail: xwenan@connect.ust.hk).

S. Jian is with the Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR (e-mail: cesjian@ust.hk).

D. He is with Intelligent Transportation Thrust, Systems Hub, The Hong Kong University of Science and Technology (Guangzhou), Guangdong China 511400 (e-mail: dengbohe@ust.hk). *: Corresponding authors

to unseen situations that are underrepresented in the training data. On the other hand, previous IRL methods such as generative adversarial imitation learning (GAIL) [9] and adversarial inverse reinforcement learning (AIRL) [10] 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 and thus affects their performance negatively. To tackle the above issues, the inverse soft-Q Learning (IQ-Learn) algorithm [11] has been introduced. IQ-Learn is an IRL method that approximates only the Q-function, representing both reward and policy. As such, the min-max problem in GAIL and AIRL can be turned into a simple minimization problem over the Q-function, which enables IQ-Learn to achieve state-of-the-art results.

This study aims to imitate human drivers' trajectories when following AVs on highways and learn their reward functions using IQ-Learn with a deep reinforcement learning (DRL) approach – soft actor-critic (SAC) [12]. In more detail, the car-following data is extracted from a naturalistic driving dataset published by Waymo. Typical car-following patterns are determined by the hierarchical clustering and compared between HV-following-AV and HV-following-HV. Then the effectiveness of the proposed method is validated in terms of trajectory-reproducing accuracy and reward functions of human drivers are recovered. In summary, the main contributions of this study are listed as follows:

1) Instead of traffic/numerical simulation or field tests, we compare HV-following-AV and HV-following-HV data by calibrating the intelligent driver model (IDM) [13] for each car-following event using the real-world Waymo Open Dataset. The results of Mann-Whitney U test reveal that human drivers perform different car-following behaviors when they are following AVs compared to following HVs.
2) By using IQ-Learn with SAC to mimic HVs' car-following trajectories when following AVs, the reward functions of human drivers which indicate their preferred states are recovered. Drivers are found to show different preferences when following AVs compared to followings HVs, suggesting the significant impacts of AVs on the following human drivers. These reward functions can be used to infer the following HVs' behaviors and improve the performance of AV controllers.

The rest of the paper is organized in the following manner. The next section briefly reviews relevant studies in modeling human drivers' car-following behaviors. Sections III presents the methodologies of both the proposed model and the benchmark models. Sections IV compares calibrated IDM parameter values in HV-following-AV and HV-following-HV, discusses the performance of the proposed model and the recovered reward functions. Finally, Section V provides the conclusions and recommendations for future research.

II. Related Work

Learning the car-following policy of human drivers can be classified into two categories including: (1) physics-based models which refer to the predefined mathematical functions with a small set of parameters and (2) data-driven models which refer to leveraging artificial intelligence methods, such as deep neural networks (DNN), DRL, and IL to mimic human behaviors.

There is a great deal of literature on modeling car-following behaviors using physics-based models, e.g., the Gipps [14], IDM, full velocity difference model (FVDM) [15] and more [16][17]. Physics-based models assume that each driver acts as an automated particle, within which human cognitive process and the machine's mechanical dynamics are highly simplified [6]. The inputs of the physics-based models usually include speed, inter-vehicle spacing and relative speed while the output is the acceleration at the current time step. The main task of physics-based models is to find an optimal set of parameters that best fit the empirical data. However, the physics-based models show poor predictive performance especially in highly complex environments. This is because they usually impose strict assumptions on human driving behaviors with a limited number of parameters, which can hardly capture human's strategic planning behaviors [18].

In recent years, data-driven models have been in the spotlight since they have high flexibility and require few prior assumptions regarding the dataset [19][20][21]. Generally, three kinds of data-driven methods have been applied to modeling human drivers' car-following behaviors, including ML, DRL and IL. For example, [22] revealed that DNN based car-following models were more accurate and robust than the Gipps and psycho-physical models. In another study, [23] modeled the car-following behaviors using a recurrent neural network (RNN). It was identified that the RNN based car-following model showed superior performance in predicting traffic oscillations with different driver characteristics compared to the IDM and DNN. Using the NGSIM dataset, [24] combined Markov theory and a gated recurrent unit (GRU) to propose a new car-following model. The proposed model showed higher accuracy and could enhance the stability of trajectory prediction relative to FVDM and DNN. Later, [25] used the deep deterministic policy gradient (DDPG) algorithm to achieve human-like car-following behaviors and the DDPG outperformed the IDM and RNN in terms of predictive accuracy. In another empirical research, [26] implemented a variety of DRL frameworks in developing a car-following model with time margin, time gap, and jerk included in the reward function. The results indicated that automating entropy adjustment on Tsallis actor-critic (ATAC) achieved the highest rewards and the ATAC based model enabled vehicles to drive safely, efficiently, and comfortably, had good stability, and were more acceptable for drivers. However, ML based models may fail due to large, stochastic and continuous state space of the car-following problem. On the other hand, it is often difficult to craft the reward functions in DRL models to perfectly encode human drivers' desired behaviors in every setting.

More recently, applying IL methods to inferring optimal sequential policies by observing how experts perform that task has attracted more and more attention. [27] assumed that human drivers follow hierarchical reasoning decision-making strategy and employed Dataset aggregation (Dagger) to build a function mapping from the ego vehicle's state, all others' state, and the ego vehicle's reasoning level-k to the ego vehicle's level-k action. While Dagger learns actions from observed states, the aim of IRL is to estimate the underlying reward function prior to finding the optimal policy. For example, [28] modeled the reactions of human drivers to AVs in three different driving scenarios by approximating the human as an optimal planner with a reward function that was acquired through IRL. [7] applied GAIL to the task of modeling human driver behaviors on the simulation platform where expert demonstrations were retrieved from the NGSIM dataset. In another study, [29] combined GAIL with Parameter Sharing Trust Region Policy Optimization (PS-TRPO) to enable IL in the multi-agent setting. Experiment results showed that compared to the single-agent models, the multi-agent model generated significantly more realistic behaviors, particularly over longer time horizons. Later, [30] modeled human driver heterogeneous behaviors by incorporating a social preference value (SVO) into one's reward function. SVO improves the model predictive performance by quantifying the degree of one'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. In [31], a reward function-based driver model that imitates human's decision-making mechanisms was proposed. They assumed that human driver behaviors consisted of three processes, which were trajectory generation, trajectory evaluation, and trajectory selection. This converted the continuous behavior modeling problem to a discrete setting, thus adopted the maximum entropy inverse reinforcement learning (MaxEnt IRL) approach to learn reward functions.

Most of the studies focused on emulating interactions within human drivers or learning the optimal policy from human demonstrations for AVs to perform human-like driving behaviors. However, existing studies have found that human drivers may adapt their car-following behaviors when they are behind AVs [2], which highlights the necessity to study drivers' behavioral changes when interacting with AVs. IRL is a promising approach to solve this research question since the recovered reward function can extend to unobservable states and then the corresponding policy can be more generalizable.

Another problem is the traffic heterogeneity, which can be defined as the differences between car-following behaviors of driver/vehicle combination under comparable conditions [32]. For example, [33] proposed a two-level probabilistic approach to run stochastic simulations of three NGSIM I-80 traffic scenarios and quantitatively study the effects of heterogeneity. It was found that heterogeneity of driver/vehicle parameters significantly affected the car-following model accuracy. Specifically, simulations with homogeneous parameters demonstrated the highest errors, by one order of magnitude. [34] developed a long- and short-term driving (LSTD) model to incorporate driver's heterogeneity in modeling car-following behaviors. The long-term driving characteristics were extracted through clustering, and the changes after an external stimulus were identified and measured as the indicator of the short-term driving characteristics. The results showed a promising performance as the errors significantly decreased after applying the LSTD model to the NGSIM dataset.

III. Methodology

This section first introduces the formulation of the car-following problem, then describes the detailed implementation of IQ-Learn built upon SAC, next introduces four benchmark models: intelligent driver model (IDM), long short-term memory (LSTM) model, generative adversarial imitation learning (GAIL) and adversarial inverse reinforcement learning (AIRL), and finally provides the metrics to assess the model performance.

A. Problem Formulation

Car-following models depict the process by which drivers follow each other in the traffic stream. During the process, drivers interact with the environment by adjusting their accelerations to maximize the long-term rewards. The problem of car-following can be formulated as a Markov Decision Process (MDP) which is defined with a five-element tuple: (S, A, R, P, γ), where S is the state space, A is the action space, $R(s_t, a_t, s_{t+1})$ is the reward function which provides the reward received while interacting with the environment, $P(s_{t+1}|s_t, a_t)$ is the transition function determining the next state given the current state and action and γ is the discount factor of future cumulative rewards. The process of interacting with the environment can be defined below: At each time step t, the vehicle will take an action a_t by observing the current state s_t, according to a stochastic policy $\pi(a_t|s_t)$. Then, the environment will be updated to the next state s_{t+1} based on the transition function $P(s_{t+1}|s_t, a_t)$, and return a scalar reward $R(s_t, a_t, s_{t+1})$ to the vehicle. The objective in an MDP is to find the optimal policy that maximizes the expected return: $\pi^* = arg\max_{\pi} \sum_{t=1}^{\infty} E_{(s_t,a_t)\sim\rho_\pi}[R(s_t, a_t)]$, where t is the time step.

In our case, let s_t denote driver's state at time step t. It is described by the following features: 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 leading vehicles $\Delta V_{n-1,n}(t)$. The action that the following vehicle needs to take at time step t is the longitudinal acceleration, denoted as $a_n(t)$. The simulation environment initializes HVs and AVs with information about their initial positions and speed according to the observed data. At time step t, the action taken by HVs is sampled from the learned optimal policies. Then, the state will be updated using Newtonian equations of motion [25]:

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

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

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

where ΔT is the simulation time interval which is the data collection interval (which is set to be $0.1s$ in this study);

$V_{n-1}(t+1)$ is the speed of the leading vehicle which is set to be known over time.

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

B. IQ-Learn

In GAIL and AIRL models, the adversarial training strategy formulates the IRL problem as a min-max game between reward and policy, which makes these two models sensitive to hyperparameter choices or minor implementation details [11]. The IQ-Learn model adopted in this study learns a single Q-function that optimizes both reward and policy simultaneously. Therefore, the min-max game is converted to a relatively simple minimization problem over the Q-function. We then introduce the model formulation and corresponding algorithms of the IQ-Learn approach.

The Q-function is denoted as $Q(s_t, a_t)$, which represents the amount of future cumulative rewards obtained by taking action a_t under state s_t. The update rules to learn $Q(s_t, a_t)$ in the IQ-Learn algorithm are listed as follows:

1) For a fixed policy π_ϕ, optimize $Q(s_t, a_t)$ by maximizing the objective function $\mathcal{J}(\pi_\phi, Q)$ using the gradient descent:

$$\mathcal{J}(\pi_\phi, Q) = E_{\rho_E}\left[\phi\left(Q - \gamma E_{S'\sim P(\cdot|s_t,a_t)} V^{\pi_\phi}(s_{t+1})\right)\right] - (1-\gamma)\, E_{\rho_0}[V^{\pi_\phi}(s_0)] \quad (4)$$

where ρ_E and $P(\cdot\,|s_t, a_t)$ are the occupancy measure of the optimal policy and the transition function; $\phi(\cdot)$ is a concave function; γ represents the discount factor; $V^{\pi_\phi}(s) = E_{a\sim\pi_\phi(\cdot|s)}[Q(s,a) - \log(\pi_\phi(a|s))]$.

2) For a fixed Q-function $Q(s_t, a_t)$, the SAC actor (which will be introduced in the next subsection) will be updated to minimize the following equation using the gradient descent:

$$\min_{\pi_\phi} E_{s\sim D, a\sim\pi_\phi(\cdot|s)}[Q(s_t, a_t) - \log(\pi_\phi(a_t|s_t))] \quad (5)$$

where D is the replay buffer. In this step, the policy π_ϕ will be optimized towards the optimal policy and Eq. (4) will be minimized. The detailed calculation process is presented in Algorithm 1.

Algorithm 1 IQ-Learn

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

Given the learned Q-function $Q(s_t, a_t)$, IQ-Learn recovers the reward function for each transition (s_t, a_t, s_{t+1}) as Eq. (6) reveals. The detailed calculation process is presented in Algorithm 2.

$$r(s_t, a_t, s_{t+1}) = Q(s_t, a_t) - \gamma V^{\pi_\phi}(s_{t+1}) \quad (6)$$

Algorithm 2 Recover policy and reward

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

C. Soft Actor-Critic

The optimal policies of human drivers are learned during the training process of IQ-Learn using SAC since SAC has the following advantages: (1) the policy is incentivized to explore more widely; (2) it is highly sample-efficient; and (3) it is suitable for continuous action space [12]. SAC is an off-policy MaxEnt DRL algorithm with the actor-critic framework. The soft policy iteration which alternates between policy evaluation and policy improvement is used to maximize the MaxEnt objective. In the policy evaluation step, the soft Q-function Q_{soft} is given as follows:

$$Q_{soft}(s_t, a_t) = r(s_t, a_t) + \gamma E_{s_{t+1}, a_{t+1}}[Q(s_{t+1}, a_{t+1}) - \alpha log(\pi_\emptyset(a_{t+1}|s_{t+1}))]] \quad (7)$$

where α is the temperature parameter; $\pi_\emptyset$ is the adopted policy with the distribution $\emptyset$.

In the policy improvement step, the policy is updated according to the function as follows:

$$\pi_{new} = arg\min_{\pi_\emptyset\in\Pi} D_{KL}(\pi_\emptyset(\cdot\,|s_t)||\frac{\exp(Q(s_t,\cdot)/\alpha)}{Z(s_t)}) \quad (8)$$

where Π is the feasible set of the policy function; D_{KL} is the Kullback-Leibler (KL) divergence; the partition function $Z(s_t)$ does not contribute to the gradient of the new policy and thus can be ignored.

The action of SAC is obtained from the policy network:

$$a_t = f_\emptyset(\epsilon_t; s_t) = f_\emptyset^\mu(s_t) + \epsilon_t \odot f_\emptyset^\sigma(s_t) \quad (9)$$

where ϵ_t is the input noise vector following the multivariate Gaussian distribution; $f_\emptyset^\mu$ is the mean of the action; $f_\emptyset^\sigma$ is the standard deviation of the action.

The value network Q_θ and the policy network π_ϕ are built using DNNs. A target network Q' is developed to stabilize the training process for the continuous control problem. As the replay buffer is adopted during the training of SAC, the sampled state transitions will be stored into the experience pool:

$$D_t = [s_t, a_t, r(s_t, a_t), s_{t+1}] \quad (10)$$

The parameters of the soft Q-function are updated by minimizing the soft Bellman residual:

$$J_Q(\theta) = E_{s_{t,a_t\sim D}}[\frac{1}{2}(Q_\theta(s_t,a_t) - (r(s_t,a_t) + \gamma Q'\left(s_{t+1},\pi_\phi(s_{t+1})\right) - \alpha\log\left(\pi_\emptyset(a_{t+1}|s_{t+1})\right)))^2] \quad (11)$$

The parameters of the target network perform a soft update:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'} \quad (12)$$

where τ is the soft update factor.

Then the update of parameters of the policy network is given by:

$$J_\pi(\emptyset) = D_{KL}(\pi_\phi(\cdot\,|s_t)||\exp\left(\frac{Q(s_t,\cdot)}{\alpha} - \log\left(Z(s_t)\right)\right)) = E_{s_{t\sim D}}[\log\left(\pi_\emptyset(a_t|s_t)\right) - \frac{Q_\theta\left(s_t,\pi_\phi(s_t)\right)}{\alpha} + \log\left(Z(s_t)\right)] \quad (13)$$

D. Benchmark models

Intelligent driver model (IDM): The IDM [13] was proposed to model the highway bottleneck congestions. The acceleration/deceleration generated by IDM considers both the desired speed and the desired distance, which is presented in the following equation:

$$a_n(t) = a_{max}^{(n)}\left(1 - \left(\frac{V_n(t)}{\widetilde{V_n}(t)}\right)^\beta - \left(\frac{\widetilde{S_n}(t)}{S_n(t)}\right)^2\right) \quad (14)$$

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

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

$$\widetilde{S_n}(t) = S_{jam}^{(n)} + \max\left(0, V_n(t)\widetilde{T_n}(t) + \frac{V_n(t)\Delta V_n(t)}{2\sqrt{a_{max}^{(n)}a_{comfort}^{(n)}}}\right) \quad (15)$$

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

To find the optimal parameter set of the IDM, the most commonly-used gradient-free algorithm – Genetic Algorithm (GA) [35] is implemented. The gradient-free algorithms are heuristic methods which do not require any information on the gradient [18]. A great advantage of this algorithm is that it can avoid the local minima and reach the global optimum. The relevant GA parameters for calibrating the IDM are specified as follows: population size 300, maximum number of generations 300, and number of stall generations 100. For the IDM, the desired speed $\widetilde{V_n}$ is set to be within the range [1, 30] m/s; the desired time gap $\widetilde{T_n}$ is set to be [0.1, 3] s; the minimum distance S_{jam} is set to be [0.1, 5] m; and the maximum acceleration a_{max} and the comfortable deceleration $a_{comfort}$ are set to be [0.1, 3] and [0.1, 5] m/s^2, respectively.

Long short-term memory (LSTM) neural network: The LSTM based car-following model is similar to [23] which takes the inputs including vehicle speed, inter-vehicle spacing and relevant speed at the current time step and outputs the longitudinal acceleration of the following vehicle. Then, the state for the next time step will be updated using Eqs. (1)-(3). The objective function of the LSTM model is described as follows:

$$C(W,B) = \frac{(S_{n-1,n}(t) - S_{n-1,n}^{obs}(t))^2}{(S_{n-1,n}^{obs}(t))^2} \quad (16)$$

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 minimizes the objective function by back-propagating a small update in the direction of optimizing the weights and biases.

Generative adversarial imitation learning (GAIL): [9] proposed GAIL based on generative adversarial networks (GAN) to mimic expert demonstration. A discriminator (D_ψ) parametrized by ψ is trained to distinguish whether a trajectory is from expert demonstration (π_E) or synthetic demonstration 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):

$$\min_\theta\max_\psi E_{\pi_E}[logD_\psi(s_t,a_t)] + E_{\pi_\theta}[log(1 - D_\psi(s_t,a_t))] \quad (17)$$

In order to fit π_θ, a surrogate reward function is calculated:

$$\tilde{r}(s_t,a_t;\psi) = -\log\left(1 - D_\psi(s_t,a_t)\right) \quad (18)$$

As the state-actions pairs (s_t,a_t) sampled from π_θ become similar to the pairs 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) [36] is used to update the policy parameters.

Adversarial inverse reinforcement learning (AIRL): [10] developed AIRL which is based on the Guided Cost Learning (GCL) and adversarial training strategy. The essential assumption of AIRL is that demonstration likelihood is proportional to the exponential of rewards. The discriminator (D_ψ) is formulated as follows:

$$D_\psi(s_t,a_t) = \frac{\exp(f_\psi(s_t,a_t))}{\exp\left(f_\psi(s_t,a_t)\right) + \pi_\theta(a_t|s_t)} \quad (19)$$

where $f_\psi(s_t,a_t)$ is the learned function and trained to infer the reward function; $\pi_\theta(a_t|s_t)$ is the current policy.

The discriminator (D_ψ) is updated by maximizing the cross-entropy loss given by Eq. (20) to tell expert demonstration apart from generated demonstration; while the policy (π_θ) is updated towards the maximization of the reward function given by $f_\psi(s_t,a_t)$.

TABLE I
DRIVERS CLUSTERING RESULTS

Features	HV-following-AV				HV-following-HV			
	Non-aggressive		Aggressive		Non-aggressive		Aggressive	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
maximum vehicle speed (m/s)	12.407	5.005	12.153	3.331	11.299	4.089	13.227	3.145
minimum vehicle speed (m/s)	8.881	4.511	6.288	2.557	7.019	3.402	6.430	2.702
Mean vehicle speed (m/s)	10.768	4.825	9.025	2.764	9.140	3.573	9.702	2.597
Standard deviation of vehicle speed (m/s)	1.019	0.486	1.787	0.727	1.272	0.738	2.047	0.831
Vehicle acceleration (m^2/s)	0.287	0.140	0.494	0.187	0.338	0.158	0.521	0.196
Vehicle deceleration (m^2/s)	0.240	0.129	0.471	0.245	0.295	0.210	0.569	0.299
Spacing (m)	21.657	9.442	14.094	5.722	18.164	8.255	12.913	2.961
Time headway (s)	2.745	0.888	2.239	0.580	2.685	0.813	1.976	0.532
Number of drivers	89		112		892		358	

$$\max_{\psi} E_{\pi_E}\left[logD_{\psi}(s_t,a_t)\right]+E_{\pi_\theta}\left[log(1-D_{\psi}(s_t,a_t))\right] \quad (20)$$

To alleviate the reward ambiguity, f_ψ is further decomposed to a reward estimator g_ψ and a potential shaping function $h_\emptyset$:

$$f_{\psi,\emptyset}(s_t,a_t,s_{t+1}) = g_\psi(s_t,a_t)+\gamma h_\emptyset(s_{t+1})-h_\emptyset(s_t) \quad (21)$$

where ψ and $\emptyset$ are parameters trained to maximize the objective function shown in Eq. (20); γ is the discount factor. Similarly, PPO [36] is adopted as the policy optimization algorithm using the estimated reward function.

E. Performance Comparison

According to [37][38][39], spacing is selected as the measure of performance (MoP) to compare different car-following models. The normalized root mean square error of spacing $NRMSE(S)$ is adopted as the goodness-of-fit function (GoF) for model evaluation. The calculation process of $NRMSE(S)$ is presented in Eqs. (22)-(23):

$$RMSE(S) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(S_i^{sim}-S_i^{obs})^2} \quad (22)$$

$$NRMSE(S) = \frac{RMSE(S)}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(S_i^{obs})^2}} \quad (23)$$

where S_i^{sim} is the i^{th} simulated spacing; S_i^{obs} is the i^{th} observed spacing; N is the number of total observations.

IV. EXPERIMENTAL RESULTS

A. Data Description

Car-following events used to train the models are extracted from the Waymo Open Dataset released by Waymo LLC. Waymo has been conducting road tests using SAE Level 4 AVs without any communication systems for more than 32 million km (kilometers) on public roads in many U.S. cities [40][41]. 5 Lidar and 5 cameras mounted on a fleet of AVs collected high-resolution data on AV's movements and the environments surrounding the AVs at 10-Hz frequency. The Waymo Open Dataset is constituted of two parts: the perception and motion parts, both of which have been used in this study.

The perception part contains 1,000 20-second video clips (as of March 2020), each of which is composed of well-synchronized and calibrated high-resolution Lidar and camera data recorded in urban and suburban areas. The Lidar data contains 12 million annotated 3D ground truth bounding boxes and the camera data contains 12 million annotated 2D fitting bounding boxes, which generates around 113k Lidar object tracks and around 250k camera image tracks [40].

The motion part consists of 103,354 20-second video clips representing 574 hours of driving data collected over 1,750 km of U.S. roadways. Each clip in the motion part contains the high-quality 3D ground truth bounding box and the speed vector for each road user (e.g., vehicles, pedestrians, and cyclists). Compared to the perception part, the motion part additionally provides a high-resolution map for each video clip as a set of polylines and polygons created from curves sampled at 0.5 meters [41].

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). Considering the sample size of the Waymo Open Dataset, timespan of each video clip and sensor detection range, this study extracts car-following events which satisfy the following criteria [2]:

1) The leading and following vehicles were driving in the same lane on a straight highway segment;
2) Neither the leading vehicle nor following vehicle changed lanes in the event;
3) The inter-vehicle spacing 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 the effects of traffic congestion;
5) The duration of car-following event should be at least 15 seconds long.

After data screening, 264 HV-following-AV and 1,376 HV-following-HV events are extracted from the dataset. For each car-following event, the second-order Savitzky–Golay filter [42] is used to filter the speed data to remove noises. Then, the acceleration/ deceleration is derived based on the filtered speed profiles and further smoothed to eliminate the remaining noises using the Savitzky–Golay filter again.

B. Driver Behavior Classification

Previous research shows that incorporating the inter-driver heterogeneity into the car-following modeling process can depict realistic car-following behaviors [33][34]. In this study,

human drivers' car-following styles are captured by the hierarchical clustering. By conducting a thorough literature view [2], critical features are chosen to reflect heterogeneous car-following preferences, including maximum speed, minimum speed, speed mean, speed standard deviation, acceleration, deceleration, spacing, and time headway. However, clustering multivariate data may lead to two problems: (1) the clusters are difficult to be visualized and assigned with specific driving patterns; and (2) features which do not vary across the samples make few contributions to differentiating driving styles. To this end, the principal component analysis (PCA) is adopted to reduce the dimension of the features to three principal components. Afterwards, the agglomerative hierarchical clustering method with the weighted linkage and Euclidean distance function is used to classify drivers in HV-following-AV and HV-following-HV scenarios. Readers are referred to our previous study [2] for more details on car-following behavior clustering results.

For the HV-following-AV scenario, 201 drivers are identified as either non-aggressive or aggressive drivers based on acceleration, deceleration, spacing and time headway while the remaining 63 drivers belong to smaller clusters. Table I summarizes the statistics of non-aggressive and aggressive driver groups in HV-following-AV. Similarly, 1,250 out of 1,376 drivers in HV-following-HV are categorized into non-aggressive or aggressive groups using the same clustering strategy. Two additional clusters consist of 126 drivers but they cannot be assigned with specific car-following patterns according to the above metrics. The summary statistics of driver features in HV-following-HV are also displayed in Table I. It should be noted that the presented values are the aggregation and average of vehicular kinematics of corresponding human driver groups.

Note that in the following study, we focus on the non-aggressive and aggressive driver groups in HV-following-AV and HV-following-HV, while the minority driver groups are not analyzed. This is because (1) those minority driver groups have very limited sample sizes; and (2) only two typical clusters are identified in HV-following-HV, suggesting that the smaller clusters in HV-following-AV have no counterparts in HV-following-HV. Due to these reasons, the following experiment is conducted using only the non-aggressive and aggressive driver groups in HV-following-AV and HV-following-HV. For each group of drivers, 80% of the car-following events are randomly selected for training and the remaining 20% car-following events are included in the testing set.

C. Car-following Scenario Comparison

We then demonstrate the effects of AVs on the following HVs by calibrating car-following models. The objective is to identify if there are any significant differences in the calibrated parameters of IDM between HV-following-AV and HV-following-HV. The IDM is calibrated for each car-following event to compare the behavioral characteristics of the drivers between HV-following-AV and HV-following-HV. Five representative parameters including $a_{max}^{(n)}$, $a_{comfort}^{(n)}$, $\widetilde{V_n}(t)$, $\widetilde{T_n}(t)$ and $S_{jam}^{(n)}$ are used as a data point to represent each car-following event. Inter-vehicle spacing is chosen as the MoP and the root mean square error of inter-vehicle spacing $RMSE(S)$ is selected as the GoF of GA. The distribution of parameters calibrated for each car-following event are shown in Fig. 1.

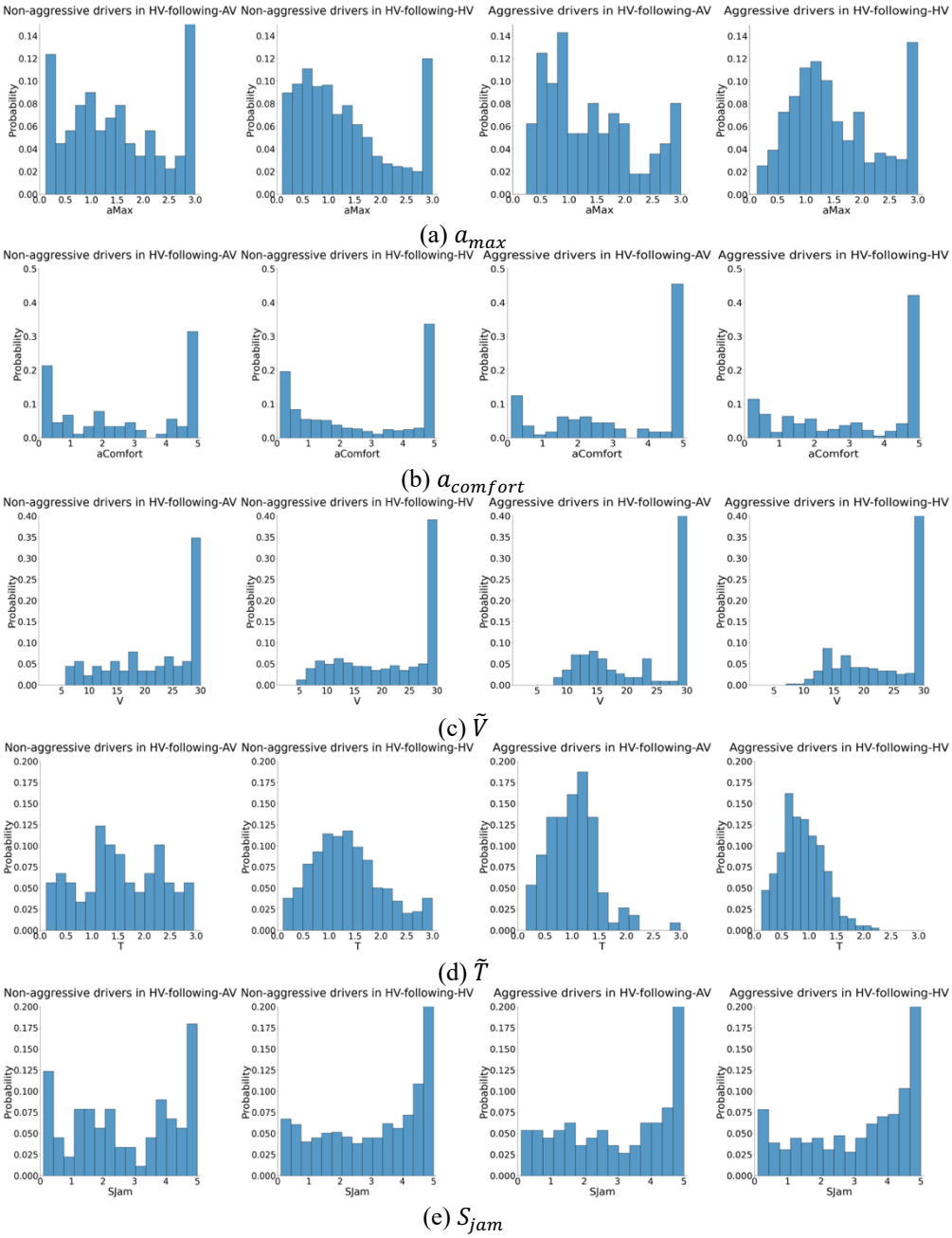
To determine if there are any significant differences in the calibrated IDM parameters between HV-following-AV and HV-following-HV scenarios, the unpaired two sample test is used since car-following events in these two scenarios are independent. Before performing the unpaired two sample test, the Shapiro-Wilk test is conducted to examine if any of these parameters in two scenarios follows the normal distribution. TABLE II shows that the p-values for all parameters are lower than the significance level of 0.05, suggesting that the null hypothesis should be rejected, thus no parameters are normally distributed. Based on this fact, the Mann-Whitney U test is conducted to identify if the differences in the calibrated parameters are statistically significant between two scenarios. In TABLE II, one can observe that at a 95% confidence level, the differences in a_{max} and $\tilde{T}$ are always significant while the differences in other parameters are not always significant. This observation implies that the leading AVs have considerable impacts on a_{max} and $\tilde{T}$ while the other parameters are insensitive to AVs. This finding is consistent with [43] which investigated the contributions of IDM parameters to the total output variance of the GoF. They concluded that $\tilde{T}$ explains the most share of the variance of $RMSE(S)$, followed by a_{max} and the other parameters make negligible contributions. In summary, the comparison results indicate that the leading AVs can indeed affect the car-following behaviors of the following HVs.

TABLE II
SHAPIRO-WILK AND MANN-WHITNEY U TEST RESULTS

	Non-aggressive drivers		Aggressive drivers	
	Shapiro-Wilk test	Mann-Whitney U test	Shapiro-Wilk test	Mann-Whitney U test
a_{max}	**< 0.001**	**0.042**	**< 0.001**	**0.028**
$a_{comfort}$	**< 0.001**	0.930	**< 0.001**	0.866
$\tilde{V}$	**< 0.001**	0.868	**< 0.001**	**0.049**
$\tilde{T}$	**< 0.001**	**0.022**	**< 0.001**	**0.003**
S_{jam}	**< 0.001**	0.070	**< 0.001**	0.883

D. IQ-Learn Training and Model Comparison

IQ-Learn with SAC is conducted to reproduce car-following trajectories for both non-aggressive and aggressive drivers in HV-following-AV. For SAC, the frameworks of value and policy networks are similar in two scenarios. For the value network, two soft Q-networks and two target networks are built and their architectures are the same. In each network, there are four layers: an input layer to input the state and action, an output layer to output the Q_{soft} as the evaluation of the state-action pair, and two hidden layers each containing 64 neurons. One policy network is constructed, which also includes four layers: an input layer to input the state, an output layer to output the mean and the standard deviation of a Gaussian distribution, and two hidden layers each containing 64 neurons. The Adam optimizer is used to update the value and policy networks. The critical hyperparameters which can significantly affect IQ-Learn and SAC performance have been tuned and provided in Table III [11].



(a) a_{max}

(b) $a_{comfort}$

(c) $\tilde{V}$

(d) $\tilde{T}$

(e) S_{jam}

Fig. 1. The distributions of the optimal parameters of the calibrated IDM.

For non-aggressive drivers, the IQ-Learn model is trained with 150 episodes where each episode represents a car-following event. During the training process, car-following events are fed into the model sequentially. To detect if there are overfitting issues, $NRMSE(S)$ for the entire training and testing datasets is computed whenever a training episode ends. The same training strategy is applied to the aggressive drivers with 180 episodes.

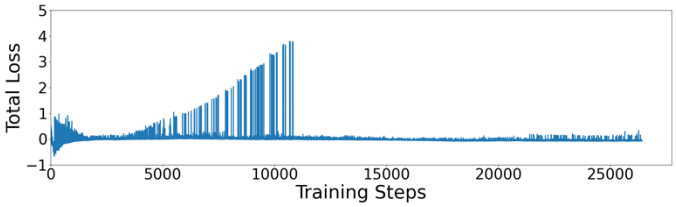
Fig. 2 and Fig. 3 show the IQ-Learn training loss ($\mathcal{J}(\pi_\phi, Q_\theta)$ in Eq. (4)) and $NRMSE(S)$ of training and testing datasets for non-aggressive and aggressive drivers, respectively. In Fig. 2(a), it is observed that at round 11,000 steps, the training loss has converged to zero. The convergence speed is even faster for aggressive drivers (as shown in Fig. 3(a)). It should be noted that there are periodical jumps of the training loss which may be attributed to the fact that the environment will be re-initialized and the state will change abruptly at the end of each training episode. One can still identify the overall tendency where the training loss keeps decreasing and then stabilizes. Fig. 2(b) demonstrates the model performance improvement in terms of $NRMSE(S)$. Similarly, it can be seen that $NRMSE(S)$ fluctuates before 60 episodes and no significant improvement is observed after 60 episodes. Finally, the IQ-Learn model that generates the smallest testing $NRMSE(S)$ is selected. The same model selection strategy is applied to the aggressive drivers as shown in Fig. 3(b).

Note that the same model training and selection strategies are also utilized for calibration of other car-following models.

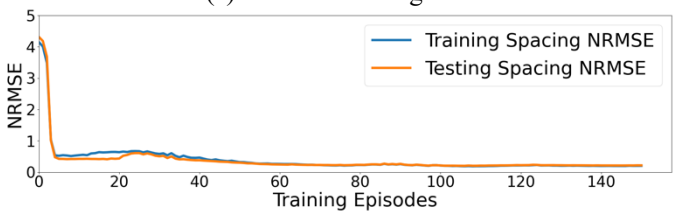
Specifically, IDM is calibrated using the corresponding training set and $NRMSE(S)$ is chosen as the GoF. The objective functions of LSTM, GAIL and AIRL are defined by Eqs. (16), (17) and (20), respectively. After each training episode, these three car-following models are evaluated based on $NRMSE(S)$ for the testing set, and the best-performed models will be retained.

TABLE III
HYPERPARAMETERS USED FOR TRAINING IQ-LEARN

PARAMETERS	DESCRIPTION	Non-aggressive	Aggressive
N_D	replay buffer size	10000	6000
α_0	initial temperature	0.1	0.3
lr^a, lr^c	learning rates of actor and critic networks	0.00001, 0.00001	0.00001, 0.00001
γ	discount factor	0.975	0.99
n_b	minibatch size	64	64
τ	soft update factor	0.005	0.005

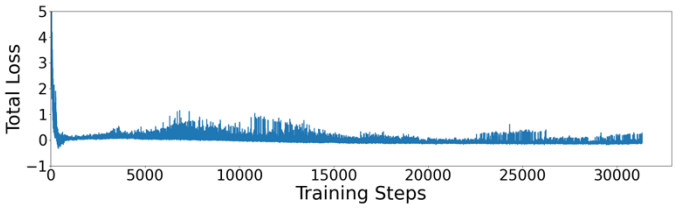


(a) Track of training loss

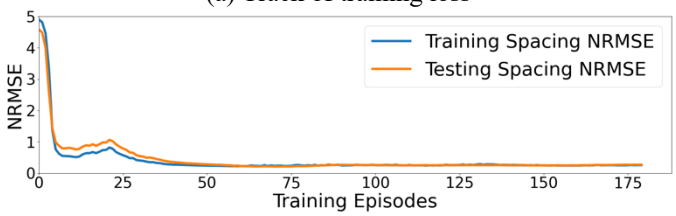


(b) Track of $NRMSE(S)$

Fig. 2. Training process for non-aggressive drivers.



(a) Track of training loss



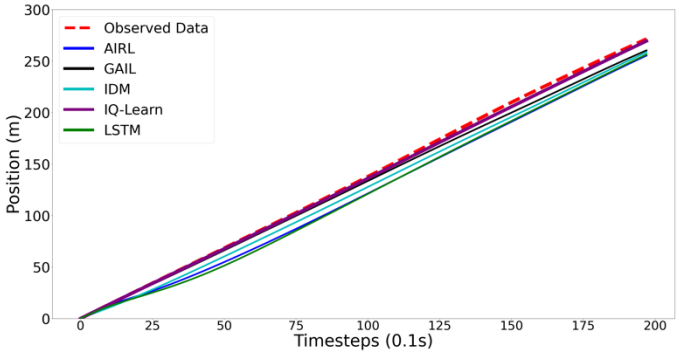
(b) Track of $NRMSE(S)$

Fig. 3. Training process for aggressive drivers.

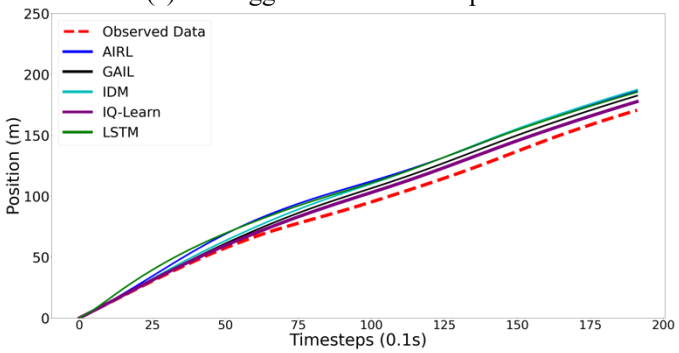
The optimal car-following policy is learned during the training process of the IQ-Learn. After the best-performed IQ-Learn models are determined, the corresponding SAC models are used to simulate the trajectory data based on the testing set. The details of the simulation are presented below. First, the following vehicle's speed, spacing and relative speed are initialized with a car-following event extracted from the testing set. Second, the following vehicle implements an acceleration based on sampling from the learned optimal policy and the state is updated using Eqs. (1)-(3). Next, when one simulation run ends, the state will be re-initialized using the next car-following even data. Finally, the simulated speed and spacing are compared to the empirical data to calculate simulation errors. The same simulation process is applied to other trained car-following models, i.e., IDM, LSTM, GAIL and AIRL. Table IV shows the performance of the proposed IQ-Learn model compared to benchmark models based on the testing set. On can observed that IQ-Learn outperforms the other investigated algorithms in terms of $NRMSE(S)$ for both non-aggressive and aggressive drivers in HV-following-AV. It is noteworthy to mention that from the microscopic perspective, there is not a "perfect" car-following model which can completely solve the discrepancy between the simulated and empirical data [44]. This is because there are always stochasticity or randomness in drivers' behaviors (i.e., some driver behaviors may reveal no perceptible patterns) [25][34].

TABLE IV
$NRMSE(S)$ ON THE TESTING SET

	Non-aggressive	Aggressive	NGSIM
IDM	0.273	0.259	0.288
LSTM	0.317	0.299	0.398
GAIL	0.254	0.227	0.269
AIRL	0.326	0.288	0.343
IQ-Learn	**0.204**	**0.212**	**0.252**



(a) Non-aggressive driver sample #1



(b) Aggressive driver sample #5

Fig. 4. Vehicle position generated by the IDM, LSTM, GAIL, AIRL, IQ-Learn.

Timespan of a car-following event is relatively short in this study and may not contain sufficient driving regimes for car-following model calibration [45]. Thus, these car-following models are fitted using the reconstructed NGSIM I80-1 dataset [46][47] to explore the model accuracy in reproducing HV-following-HV dynamics. Based on the aforementioned criteria, 1,345 car-following events have been extracted, of which 80% (1,076) are randomly selected for training and the remaining 20% (269) are used for testing. Table IV presents the testing $NRMSE(S)$ values of different car-following models. One can see that IQ-Learn still outperforms the other models and achieves a decent degree of predictive accuracy.

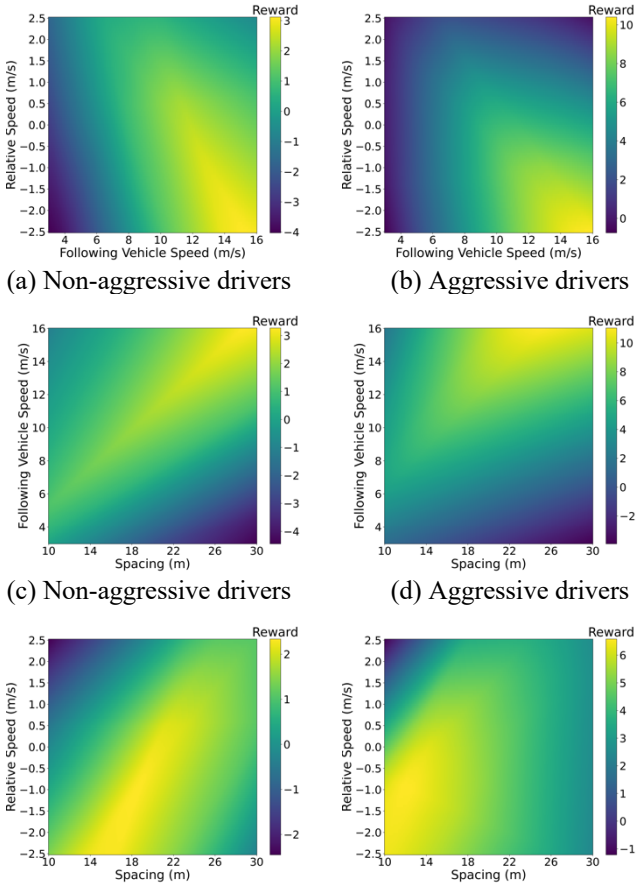
To demonstrate that the proposed IQ-Learn model can accurately reproduce human driver behaviors in dynamic traffic environments, two car-following events in the testing set are chosen (one is an non-aggressive driver while another is an aggressive driver). Fig. 4 displays the position of the observed and simulated data by the IDM, LSTM, GAIL, AIRL and IQ-Learn. It is indicated that the position simulated by IQ-Learn (in purple) is always the closest to the observed (in red) data, which suggests that the IQ-Learn model can predict the vehicle position the most accurately.

E. Recovery of HV-following-AV Reward Functions

Human drivers' car-following preferences can be inferred through the recovered reward functions, which can provide insights into their car-following behaviors when interacting with AVs. Each driver's state, on which the reward function is based, is defined by vehicle speed, spacing and relative speed while the action is described as the longitudinal acceleration. The recovered reward functions for HV-following-AV are visualized in Fig. 5. Similar to [48], the reward functions are presented as bivariate feature spaces where the other features are held at their mean values. For instance, in Fig. 5(a), the spacing is fixed at its mean value while the following vehicle speed and relative speed change within their ranges. The brighter the color is, the higher the reward is. Higher rewards indicate that human drivers prefer to stay at corresponding states. It should be noted that the behavior preferences inferred from the reward functions are correlated with the mean values of the other features, i.e., some preferences may differ if the values of the other features change.

For non-aggressive drivers, as depicted on the left panel of Fig. 5, one can observe that when the speed is within the range of $12m/s$ and $16m/s$, drivers who are following AVs have a tendency to be slower than AVs (Fig. 5(a)). Moreover, there is a clear decreasing tendency for the preferred relative speed, suggesting that as AVs accelerate, the following human drivers prefer to increase the speed discrepancies. Second, the reward function based on inter-vehicle spacing and following vehicle speed indicates that the preferred spacing increases linearly as the following HV speed increases (Fig. 5(c)). Third, the reward function which is presented using inter-vehicle spacing and relative speed reveals that the increment of the inter-vehicle spacing is correlated with the increased preferred relative speed (Fig. 5(e)). This can be explained as that the following HVs may need to catch up with the leading AVs if AVs drive away from them.

On the right panel of Fig. 5, Fig. 5(b) which is displayed using the following vehicle speed and relative speed, reveals that aggressive drivers prefer to drive at high speeds, ranging from $13m/s$ to $16m/s$, but are $2.0m/s$ to $2.5m/s$ slower than the leading AVs. In terms of the reward function shown using the spacing and following vehicle speed, the desired inter-vehicle spacing increases with the increment of the following vehicle speed (Fig. 5(d)). If the comparisons are made between non-aggressive and aggressive drivers, one can identify that the latter keeps shorter spacing to the AVs than the former given the same speed, which is consistent with the definition of aggressiveness. Fig. 5(f) demonstrates a positive relationship between inter-vehicle spacing and relative speed. When the inter-vehicle spacing is between $10m$ and $14m$ and the relative speed ranges from $-2.5m/s$ to $0m/s$, aggressive drivers gain the highest rewards.



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

(c) Non-aggressive drivers (d) Aggressive drivers

(e) Non-aggressive drivers (f) Aggressive drivers

Fig. 5. Reward functions for HV-following-AVs.

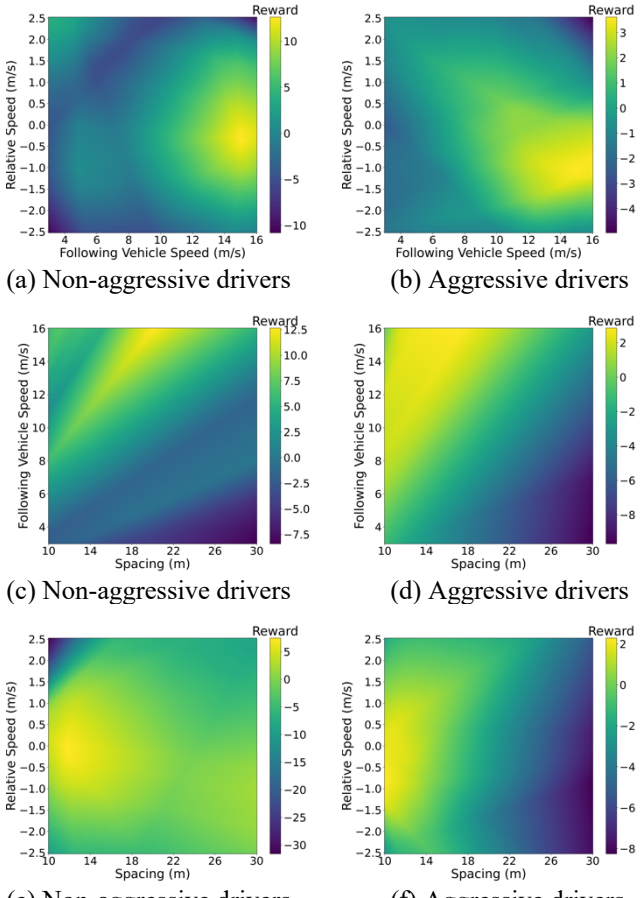
F. Comparison of Reward Functions

The reward functions of non-aggressive and aggressive drivers in HV-following-HV have also been recovered in Fig. 6. Comparing the reward functions of non-aggressive drivers in HV-following-AV (see the left panel of Fig. 5) and HV-following-HV (see the left panel of Fig. 6) scenarios, one can observe that:

1) Fig. 5(a) demonstrates that non-aggressive drivers following AVs prefer the speed between 12 and $16m/s$ and relative speed between -2.5 and $-0.5m/s$; while Fig. 6(a) shows similar preferred speed (i.e.,

$14-16m/s$) and intermediate preferred relative speed (i.e., $-0.5-0.5m/s$) for non-aggressive drivers following HVs. More importantly, the decreasing trend in Fig. 5(a) reveals that non-aggressive drivers in HV-following-AV have more flexibility than those in HV-following-HV in the car-following process.

2) Both Fig. 5(c) and Fig. 6(c) indicate an apparent linear relationship between the preferred inter-vehicle spacing and following vehicle speed. But one can see that when drivers are driving at $12-16m/s$, the preferred inter-vehicle spacing is $22-28m$ and $14-20m$ in HV-following-AV and HV-following-HV, respectively. This suggests that non-aggressive drivers in HV-following-AV prefer to stay farther away from AVs compared to those in HV-following-HV.
3) In Fig. 6(e), one can see that there is one hotspot when the spacing is within the range of $10-14m$ and the relative speed falls between -0.5 and $0.5m/s$. In contrast, Fig. 5(e) shows an increasing tendency and the highest rewards occur when the spacing is between 14 m and $22m$ and the relative speed ranges from $-2.5m/s$ to $0.5m/s$. The comparison between two scenarios again confirms that non-aggressive drivers who are following AVs are more flexible than those led by HVs.



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

(c) Non-aggressive drivers (d) Aggressive drivers

(e) Non-aggressive drivers (f) Aggressive drivers

Fig. 6. Reward functions for HV-following-HVs.

Aggressive drivers in HV-following-HV (see the right panel of Fig. 6) have been compared to those behind AVs (see the right panel of Fig. 5). The findings are listed below:

1) Comparing Fig. 5(b) with Fig. 6(b), one can see that aggressive drivers show similar speed preferences in HV-following-AV and HV-following-HV scenarios (i.e., $13-16m/s$). However, the former group are inclined to be $2.0-2.5m/s$ slower than the leading AVs. On the contrary, the latter group prefer relatively intermediate speed differences ($0.5-1.5m/s$).
2) Fig. 5(d) and Fig. 6(d) show that although the linear relationship between aggressive drivers' preferred spacing and speed exists in both scenarios, the slope in Fig. 5(d) is less than that in Fig. 6(d). This suggests that at similar speed, aggressive drivers following AVs prefer to keep longer spacing than those who are following HVs.
3) Comparing Fig. 5(f) with Fig. 6(f), one can find that aggressive drivers show similar inter-vehicle spacing preferences (i.e.,$10-14m$). But the difference is that in HV-following-AV, spacing preferences are linearly related to the preferred relative speed; while those who are following HVs prefer to maintain similar speed as the leading HVs (i.e., $-1-0.5m/s$).

V. Conclusions

This study bridges the gap in identifying the adaptations in human drivers' car-following behaviors when they are interacting with AVs. HV-following-AV and HV-following-HV events are extracted from the real-world dataset released by Waymo. Statistical test results reveal that the type of leading vehicle (i.e., AV versus HV) has significant effects on the following drivers' behaviors. IQ-Learn has been proposed to reproduce human driver trajectories when following AVs on highway segments. 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 $NRMSE(S)$.

Moreover, the recovered reward functions based on IQ-Learn display the preferences of human drivers in HV-following-AV and HV-following-HV. The results show that there are significant differences in the preferred states of drivers in two scenarios. This paper highlights the needs to consider drivers' heterogeneous car-following behaviors in response to the existence of AVs. Besides, it can provide feedbacks for the design of AV controllers, improve the inference ability of AVs and reflect the social acceptance of AVs.

This study investigates on car-following events on highways where the traffic flow is uninterrupted. It would be interesting to identify how AVs affect HVs in more complicated settings such as urban streets. Future study can also investigate other interactions (e.g., lane-changing, merging, diverging and turning) between HVs and AVs. Furthermore, the benefits of AVs identified in this research may depend on the AV control algorithms – it is possible that different AV controllers may lead to different car-following behaviors of HVs. Thus, it would be interesting to analyze real-world datasets released by other AV technique companies such as Lyft and compare the results to testify the generalization of such findings. Third, since the

Waymo Open Dataset is composed of a number of 20-second discontinuous segments, the duration of car-following events is relatively short and may not contain sufficient information for car-following model calibration [45]. Future research needs to collect more completed car-following trajectories to analyze the interactions between AVs and HVs.

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. 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.

[3] X. Zhao, Z. Wang, Z. Xu, Y. Wang, X. Li, and X. Qu, "Field experiments on longitudinal characteristics of human driver behavior following an autonomous vehicle," *Transportation research part C: emerging technologies*, vol. 114, May 2020, pp. 205-224.

[4] I. Mahdinia, A. Mohammadnazar, R. Arvin, and A. J. Khattak, "Integration of automated vehicles in mixed traffic: Evaluating changes in performance of following human-driven vehicles," *Accident Analysis & Prevention*, vol. 152, Mar. 2021, Art. no. 106006.

[5] X. Wen, S. Jian, and D. He, "Modeling Human Driver Behaviors When Following Autonomous Vehicles: An Inverse Reinforcement Learning Approach," in *Proc. IEEE International Conference on Intelligent Transportation Systems*, 2022, pp. 1375-1380.

[6] 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.

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

[8] S. Schaal, "Learning from demonstration," in *Proc. Advances in neural information processing systems*, 1997, pp. 1040–1046.

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

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

[11] 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, pp. 4028-4039.

[12] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *in Proc. International conference on machine learning*, 2018, pp. 1861-1870.

[13] 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.

[14] P. G. Gipps, "A behavioural car-following model for computer simulation," *Transportation Research Part B: Methodological*, vol. 15, Apr. 1981, Art. no. 2.

[15] R. Jiang, Q. Wu, and Z. Zhu, "Full velocity difference model for a car-following theory," *Physical Review E*, vol. 64, Jun. 2001, Art. no. 1.

[16] J. Zhang, T. Tang, and S. Yu, "An improved car-following model accounting for the preceding car's taillight," *Physica A: Statistical Mechanics and its Applications*, vol. 492, Feb. 2018, pp. 1831-1837.

[17] J. Zhang, T. Tang, and T. Wang, "Some features of car-following behaviour in the vicinity of signalised intersection and how to model them," *IET Intelligent Transport Systems*, vol. 13, Nov. 2019, pp. 1686-1693.

[18] Z. Mo, R. Shi, and X. Di, "A physics-informed deep learning paradigm for car-following models," *Transportation research part C: emerging technologies*, vol. 130, Sep. 2021, Art. no. 103240.

[19] X. Wen, Y. Xie, L. Wu, and L. Jiang, "Quantifying and comparing the effects of key risk factors on various types of roadway segment crashes with LightGBM and SHAP," *Accident Analysis & Prevention*, vol. 159, Jun. 2021, Art. no. 106261.

[20] X. Wen, Y. Xie, L. Jiang, Z. Pu, and T. Ge, "Applications of machine learning methods in traffic crash severity modelling: current status and future directions," *Transport reviews*, vol. 41, Jul. 2021, Art. no. 6.

[21] X. Wen, Y. Xie, L. Jiang, Y. Li, and T. Ge, "On the interpretability of machine learning methods in crash frequency modeling and crash modification factor development," *Accident Analysis & Prevention*, vol. 168, Apr. 2022, Art. no. 106617.

[22] S. Panwai and H. Dia, "Neural agent car-following models," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, Mar. 2007, Art. no. 1.

[23] 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.

[24] T. Tang, Y. Gui, J. Zhang, and T. Wang, "Car-Following model based on deep learning and Markov theory," *Journal of Transportation Engineering, Part A: Systems*, vol. 146, Jul. 2020, Art. no. 04020104.

[25] M. Zhu, X. Wang, and Y. Wang, "Human-like autonomous car-following model with deep reinforcement learning," *Transportation research part C: emerging technologies*, vol. 97, Dec. 2018, pp. 348-368.

[26] T. Tang, Y. Gui, and J. Zhang, "ATAC-Based Car-Following Model for Level 3 Autonomous Driving Considering Driver's Acceptance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, Aug. 2022, pp. 10309–10321.

[27] R. Tian, S. Li, N. Li, I. Kolmanovsky, A. Girard, and Y. Yildiz, "Adaptive game-theoretic decision making for autonomous vehicle control at roundabouts," *in Proc. IEEE Conference on Decision and Control (CDC)*, 2018, pp. 321-326.

[28] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions," *in Proc. Robotics: Science and Systems*, 2016, pp. 1-9.

[29] 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.

[30] 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.

[31] 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*, vol. 23, Aug. 2022, pp. 10239–10251.

[32] S. Ossen, and S.P. Hoogendoorn, "Heterogeneity in car-following behavior: Theory and empirics," *Transportation research part C: emerging technologies*, vol. 19, no. 2, Apr. 2011, pp. 182-195.

[33] V. Punzo, and M. Montanino, "A two-level probabilistic approach for validation of stochastic traffic simulations: impact of drivers' heterogeneity models," *Transportation research part C: emerging technologies*, vol. 121, Dec. 2020, Art. no. 102843.

[34] X. Chen, J. Sun, Z. Ma, J. Sun, and Z. Zheng, "Investigating the long-and short-term driving characteristics and incorporating them into car-following models," *Transportation research part C: emerging technologies*, vol. 117, Aug. 2020, Art. no. 102698.

[35] M. Kumar, M. Husian, N. Upreti, and D. Gupta, "Genetic algorithm overview: Review and application," *Int. J. Inf. Technol. Knowl. Manage.*, vol. 2, Dec. 2010, Art. no. 2.

[36] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[37] A. Kesting, and M. Treiber, "Calibrating car-following models by using trajectory data: Methodological study," *Transportation Research Record*, vol. 2088, no. 1, Jan. 2008, pp. 148-156.

[38] 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.

[39] V. Punzo, Z. Zheng, and M. Montanino, "About calibration of car-following dynamics of automated and human-driven vehicles: Methodology, guidelines and codes," *Transportation Research Part C: Emerging Technologies*, vol. 128, Jul. 2021, Art. no. 103165.

[40] 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.

[41] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, 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.

[42] A. Savitzky and M. JE Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical chemistry*, vol. 36, Jul. 1964, Art. no. 8.

[43] V. Punzo, M. Montanino, and B. Ciuffo, "Do we really need to calibrate all the parameters? Variance-based sensitivity analysis to simplify microscopic traffic flow models," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, Jul. 2014, Art. no. 1.

[44] E. Brockfeld, R. D. Kühne, and P. Wagner, "Calibration and validation of microscopic models of traffic flow," *Transportation Research Record*, vol. 1934, Jan. 2005, pp. 179-187.

[45] A. Sharma, Z. Zheng, and A. Bhaskar, "Is more always better? The impact of vehicular trajectory completeness on car-following model calibration and validation," *Transportation research part B: methodological*, vol. 120, Feb. 2019, pp. 49-75.

[46] V. Punzo, M. T. Borzacchiello, and B. Ciuffo, "On the assessment of vehicle trajectory data accuracy and application to the Next Generation SIMulation (NGSIM) program data," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 6, Dec. 2011, pp. 1243-1262.

[47] M. Montanino, and V. Punzo, "Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns," *Transportation Research Part B: Methodological*, vol. 80, Oct. 2015, pp. 82-106.

[48] R. Alsaleh, and T. Sayed, "Markov-game modeling of cyclist-pedestrian interactions in shared spaces: A multi-agent adversarial inverse reinforcement learning approach," *Transportation research part C: emerging technologies*, vol. 128, Jul. 2021, Art. no. 103191.