A Leading Cruise Controller for Autonomous Vehicles in Mixed Autonomy Based on Preference-Based Reinforcement Learning

Xiao (Luke) Wen
Department of Civil and Environmental Engineering
The Hong Kong University of Science and Technology
Kowloon, Hong Kong, SAR
xwenan@connect.ust.hk

Sisi Jian
Department of Civil and Environmental Engineering
The Hong Kong University of Science and Technology
Kowloon, Hong Kong, SAR
cesjian@ust.hk

Dengbo He*
Intelligent Transportation Thrust, Systems Hub
The Hong Kong University of Science and Technology (Guangzhou)
Guangdong 511400, China
dengbohe@hkust-gz.edu.cn

***Abstract*—Previous studies on car-following controllers for autonomous vehicles (AVs) in mixed traffic have a narrow focus on maximizing the AV's utility, neglecting the utility of the entire traffic flow. This leads to self-centered AVs that may not be beneficial to surrounding vehicles. Thus, this study aims to develop a leading cruise controller for AVs that considers not only the AV's behaviors, but also the behaviors of both the lead human-driven vehicle (LHDV) and the following human-driven vehicle (FHDV). To achieve this, the study uses real-world data from the Waymo Open Dataset to approximate the behaviors of human-driven vehicles (HDVs) through an inverse reinforcement learning (IRL) approach. The study then proposes a preference-based soft actor-critic (PbSAC) algorithm to optimize the speed of AVs in a three-vehicle car-following scenario, while also considering safety, efficiency, and string stability for both AV and FHDV in the reward function. To further improve the control algorithm, the study develops a preference-adjusting module that adaptively updates the weights of the reward function based on expert evaluation. Experimental results show that the proposed algorithm can significantly improve safety, efficiency, and string stability for both AV and FHDV.**

Keywords—autonomous vehicles, reinforcement learning, safety, efficiency, car-following

I. Introduction

AV technologies are advancing quickly, but experts in transportation, road engineering, and AV manufacturing recognize that there will be a gradual transition period as AVs are introduced [1]. During the transition period, there will be a mixed traffic environment where both AVs and human-driven vehicles (HDVs) will be present [2]. Given this context, the research question that arises is: how should the control logic of AVs be designed to enhance safety, efficiency, and string stability in mixed traffic [3].

The focus of this study is on the car-following model, which illustrates how vehicles interact longitudinally with one another in the same lane. Considerable efforts have been devoted to developing car-following models for AVs in mixed traffic. These approaches can generally be categorized into three types: (1) linear or non-linear AV controllers; (2) AV controllers based on model predictive control (MPC); and (3) AV controllers based on deep reinforcement learning (DRL) [4]. There are two primary advantages of DRL-based methods over the first two categories. Firstly, DRL is a learning-based and model-free approach that does not rely on predefined rules or stochastic system modeling. Secondly, the computational cost of DRL is considerably lower than MPC, as a trained DRL model can be implemented in real-time.

Previous studies have generally found that DRL-based longitudinal controllers for AVs can improve traffic flow performance. However, current DRL-based models suffer from two significant drawbacks. Firstly, many of these studies have focused solely on optimizing AVs and ignored interactions between AVs and surrounding HDVs, particularly HDVs following AVs. For instance, some literature has only considered a two-vehicle car-following scenario where the following vehicle is an AV controlled by a DRL algorithm. As a result, the impacts of developed AV controllers on the following HDV are not explored. Although other studies have concentrated on mixed platoons with three or more vehicles, the reward functions of developed DRL algorithms only consider the benefits of AVs. Consequently, the learned AV controllers may exhibit egocentric and aggressive driving behaviors, leading to serious traffic situations, e.g., congestion and crashes. Besides, many studies have assumed that AVs can communicate with surrounding vehicles in mixed traffic environments. However, this may not be realistic at the current stage since AVs deployed on public roads, such as Waymo, cannot communicate with surrounding vehicles.

Second, previous studies treated car-following control as a multi-objective reinforcement learning (MORL) problem by incorporating multiple terms in the reward function. They converted the multi-objective reward vector into a scalar weighted-sum reward and then determined the optimal weight combination through extensive experiment trials [5]. However, manually selecting weights for each objective may result in two significant drawbacks: (1) determining the appropriate scalarization is complex and time-consuming because objectives are often defined in different units and scales and there may be trade-offs among objectives; and (2) optimizing the DRL model based on a fixed weight combination can only generate an optimal policy for specific preferences. Consequently, the learned optimal policy may have limited adaptability to different user preferences and driving scenarios. To address these issues, this paper proposes an innovative algorithm - preference-based soft actor-critic (PbSAC) for AV longitudinal control. PbSAC can adjust the weight combination adaptively based on any user-specified preference and produce the optimal policy.

Given AVs' advanced onboard sensors that can gather real-time and precise information about their surroundings, this study is one of the few to develop a DRL-based leading cruise control algorithm for AVs in a three-vehicle car-following scenario. This algorithm allows AVs to "look behind" mixed traffic, enabling them to make car-following decisions that adapt to both the lead human-driven vehicle (LHDV) and the FHDV. This improves the safety, efficiency, and string stability of mixed traffic. To handle the multi-objective optimization problem, PbSAC is developed as the AV car-following control model, and a preference generator is proposed to dynamically determine the relative importance in the reward function based on expert evaluation. FHDV car-following behaviors are acquired through inverse reinforcement learning (IRL) using the Waymo Open Dataset.

The main contributions of this study are:

1. The study proposes a DRL-based AV controller that can "look behind" the mixed traffic flow. This controller can enhance the mixed traffic flow performance by considering the safety, efficiency, and string stability of both AV and FHDV in the reward function.
2. To address the multi-objective optimization problem, the study adopts PbSAC as the AV controller and proposes a preference learning module to determine the optimal policy by adjusting the weights in the reward function based on human preferences.

The paper will be structured as follows. Section II outlines the methodologies and scenarios of the proposed AV controller. Section III presents the experimental settings, assesses the model's performance, and analyzes the results. Finally, Section IV concludes the paper and proposes directions for future research.

II. Methodology

A. *Problem Formulation*

Fig. 1 illustrates that this study focuses on the three-vehicle car-following scenario, representing a common segment of mixed traffic. The leading vehicle is an LHDV that serves as an external input to achieve the speed profile extracted from the Waymo Open Dataset. The second vehicle is an AV controlled by PbSAC, which can gather real-time information on the states of both the LHDV and FHDV. Additionally, the HDV-following-AV model is utilized to replicate FHDV car-following behaviors.

During a car-following scenario, the following vehicle modifies its acceleration to follow the leading vehicle. Therefore, the car-following problem is converted to a Markov Decision Process (MDP) defined by a tuple: (S, A, R, P, γ), where S is the state space, A is the action space, $R(s_t, a_t)$ is the reward function representing the reward resulting from the interaction with the environment, the transition function $P(s_{t+1}|s_t, a_t)$ determines the next state given the current state and action, and γ is the discount factor. At each time step t, the vehicle adopts an action a_t according to the policy $\pi(a_t|s_t)$. Afterwards, the state s_t is updated to s_{t+1} based on the transition function, and a scalar reward $R(s_t, a_t)$ is returned. The objective in an MDP is to find the optimal policy to maximize the expected return: $\pi^* = arg\max_{\pi} \sum_{t=1}^{\infty} E_{(s_t,a_t)\sim\rho_\pi}[R(s_t, a_t)]$.

In the following, basic DRL concepts in the scenarios that are under investigation are provided. Note that “AV” and “agent” will be used interchangeably.

1) State: This study assumes that AVs collect real-time and accurate movement information on LHDV and FHDV, e.g., speed and position, via onboard sensors. Let s_t denote the state of AV at time step t. The state vector s_t involves two aspects of information: (1) the speed of AV $V_1(t)$, the gap between AV and FHDV $S_{1,2}(t)$, the gap between LHDV and AV $S_{0,1}(t)$, the relative speed between AV and FHDV $\Delta V_{1,2}(t)$ and the relative speed between LHDV and AV $\Delta V_{0,1}(t)$; and (2) the weight parameters in the reward function.

2) Action: The action that an AV needs to take at time step t is the longitudinal acceleration $a_1(t)$ which is bounded between -3 and $3m/s^2$.

3) Transition function: After the AV takes an action at time step t, the three-vehicle platoon state will be updated using the kinematic point-mass function:

$$V_2(t+1) = V_2(t) + \Delta T * a_2(t) \quad (1)$$

$$V_1(t+1) = V_1(t) + \Delta T * a_1(t) \quad (2)$$

$$\Delta V_{1,2}(t+1) = V_2(t+1) - V_1(t+1) \quad (3)$$

$$\Delta V_{0,1}(t+1) = V_1(t+1) - V_0(t+1) \quad (4)$$

$$S_{1,2}(t+1) = S_{1,2}(t) - \Delta T * \frac{\Delta V_{1,2}(t)+\Delta V_{1,2}(t+1)}{2} \quad (5)$$

$$S_{0,1}(t+1) = S_{0,1}(t) - \Delta T * \frac{\Delta V_{0,1}(t)+\Delta V_{0,1}(t+1)}{2} \quad (6)$$

where ΔT is the simulation time interval (which is $0.1s$ in this study); $V_0(t)$ and $V_2(t)$ are the speed of LHDV and FHDV at time step t; $a_2(t)$ is the acceleration of FHDV at time step t.

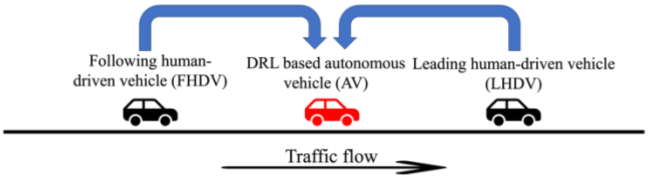

Following human-driven vehicle (FHDV)
DRL based autonomous vehicle (AV)
Leading human-driven vehicle (LHDV)
Traffic flow

Fig. 1. The proposed “look behind” control system has two HDVs and one AV in the platoon. The black vehicles are HDVs and the red ones are AVs. The blue arrows show the information collected by the AV.

B. *Reward Function*

Inspired by [6], the reward function will account for both control efficiency and string stability. The control efficiency reward $Rcontrol$ is expressed as a quadratic function that regulates the vehicle to maintain the predefined state:

$$R_{control} = -(\boldsymbol{x}(t))^T \boldsymbol{Q} \boldsymbol{x}(t) \quad (7)$$

$$\boldsymbol{x}(t) = [\Delta\tilde{S}(t), \Delta\tilde{V}(t)] \quad (8)$$

$$\Delta\tilde{S}(t) = S_{n-1,n}(t) - t^* * V_n(t) \quad (9)$$

$$\Delta\tilde{V}(t) = V_n(t) - V_{n-1}(t) \quad (10)$$

$$\boldsymbol{Q} = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix}, \alpha_1, \alpha_2 > 0 \quad (11)$$

where $\boldsymbol{x}(t)$ refers to the difference from the predetermined state. $\Delta\tilde{S}(t)$ indicates the discrepancy between the actual gap and the target gap, and $S_{n-1,n}(t)$ denotes the actual gap between consecutive vehicles. The target gap is defined as $t^* * V_n(t)$ where t^* is the target time gap. In this study, t^* is set to be $1.5s$, according to [7]. $\Delta\tilde{V}(t)$ represents the relative speed between consecutive vehicles while $V_n(t)$ and $V_{n-1}(t)$ are the speed of the vehicle and its leader. α_1 and α_2 are fixed at 0.04 and 0.08 based on massive experiments.

As the control efficiency reward $R_{control}$ is optimized to be maximum, $\Delta\tilde{S}(t)$ and $\Delta\tilde{V}(t)$ will both converge to zero. This indicates that the actual gap is approaching the target gap. Besides, note that the commonly-used safety indicator – TTC is calculated in Eq. (12):

$$TTC(t) = \frac{S_{n-1,n}(t)}{V_n(t) - V_{n-1}(t)} \tag{12}$$

When $\Delta\tilde{V}(t) = V_n(t) - V_{n-1}(t) \to 0$, $TTC(t) \to \infty$. Since the proposed AV controller considers the benefits of AV and FHDV, both $R_{control\_AV}$ and $R_{control\_FHDV}$ are incorporated into the reward function.

The capacity of a vehicle to dampen the speed oscillations of its leader, known as string stability, is a crucial attribute. If the magnitude of the leader's perturbation is greater than the magnitude of the vehicle's perturbation, the vehicle is deemed string stable. By referring to [8], it is found that string stability is negatively correlated with acceleration behaviors, meaning that sharp and frequent acceleration behaviors result in the string instability issue. Hence, this study uses the negative square of the acceleration as the string stability reward $R_{stability}$ and another term $-\frac{(a_n(t)-a_n(t-1))^2}{(a_{max}-a_{min})^2}$ is added to ensure driving comfort:

$$R_{stability} = -a_n(t)^2 - \frac{(a_n(t)-a_n(t-1))^2}{(a_{max}-a_{min})^2} \tag{13}$$

where a_{max} and a_{min} are the maximum and minimum acceleration (which are $3m/s2$ and $-3m/s2$ in this study), respectively. Similarly, both $R_{stability\_AV}$ and $R_{stability\_FHDV}$ are considered in the reward function.

Finally, Eq. (14) shows the completed reward function:

$$R_1 = \omega_1 R_{control_{AV}} + \omega_2 R_{control_{FHDV}} + \omega_3 R_{stability\_AV} + \omega_4 R_{stability\_FHDV} \tag{14}$$

$$s.t. \quad \forall i: \omega_i > 0$$

$$\sum_{i=1}^{4} \omega_i = 1$$

where ω_1, ω_2, ω_3, and ω_4 are the weight parameters that need to be tuned.

C. Preference-based Soft Actor-Critic (PbSAC)

The reward function in Eq. (14) has four terms, and setting the weight parameters manually can be difficult and time-consuming. But this study suggests a new preference generator that can optimize the weights for each reward term automatically using expert knowledge. For instance, AV algorithm developers can evaluate the average time gap in AV car-following trajectories to assess them. Based on their preferences, the weights will be adjusted. Then, by considering the updated weights in the reward function, the control policy can be enhanced.

```
Algorithm 1 Preference-based Soft Actor-Critic (PbSAC)
Initialize memory buffer M.
Initialize system buffer S.
Initialize trajectory buffer T.
Initialize weight parameters w.
Initialize critic $Q_\theta$ and actor $\pi_\phi$ networks.
for $ep$ in $\{1,2,3, \dots N\}$ do
  for t in {1,2,3,…T} do
    Sample action $a_t$ based on actor network, given current state $s_t$.
    Implement $a_t$ and transfer to the next state $s_{t+1}$.
    Obtain reward vector [$R_{safety_{AV}}, R_{safety_{FHDV}}, R_{stability_{AV}}, R_{stability_{FHDV}}, R_{comfort\_AV}, R_{comfort\_FHDV}$].
    Save the reward vector into trajectory buffer T.
    Calculate reward $r_t$ based on Eq. (14) and current weights w.
    Save transition $(s_t, a_t, r_t, s_{t+1})$ into M.
    for k in {1,2,3,…K} do
      Sample random batch of B transitions from M
      Update $\theta$ and $\phi$
    end for
  end for
  Save T into system buffer S.
  if $ep$ mod L ≡ 0 then
    Sample L trajectory pairs from S and evaluate them upon expert knowledge.
    Calculate likelihood of w using Eq. (16).
    Update weight parameters w through MCMC using Eq. (18).
    Clear trajectory buffer T.
  end if
end for
```

According to [9], the weight parameters are updated upon human preferences using a Bayesian learning framework. Let τ_i denote the ith car-following trajectory of an AV. The total rewards obtained from the trajectory τ_i is:

$$\boldsymbol{r}(\tau_i) = (r_1^i, r_2^i, \dots, r_T^i) \tag{15}$$

where T is the length of the car-following trajectory.

Then any trajectory pair (i.e., τ_i and τ_j) can be compared according to expert evaluation, e.g., a pair of car-following trajectories can be compared based on safety and efficiency indicators. Following [9], a probabilistic model over expert preferences is employed to compare two trajectories:

$$p(\tau_i > \tau_j | w) = \frac{\exp(w^T r(\tau_i))}{\exp(w^T r(\tau_i)) + \exp(w^T r(\tau_j))} \tag{16}$$

where $\tau_i > \tau_j$ means the preference of τ_i over τ_j based on expert evaluation. As a result, trajectories that achieve a higher linearly scalarized reward will be given a higher ranking.

After each policy improvement, a pair of trajectories (τ_i, τ_j) will be randomly sampled and evaluated based on the removal of posterior volume (as shown in Eq. (17)). (τ_i, τ_j)

that removes the most volume among the previous pairs since the last posterior update will be saved into the volume buffer and prepared for the next comparison.

$$\max_{(\tau_i,\tau_j)} \min(E_w[1 - p(\tau_i > \tau_j|w), E_w[1 - p(\tau_j > \tau_i|w)]) \tag{17}$$

Let q_m represent the comparison between two trajectory samples: $q_m = (\tau_i > \tau_j)_m$. After multiple pairwise comparisons $\{q_1, q_2, \ldots, q_n\}$, the posterior of the weights can be updated in a Bayesian fashion:

$$p(w|q_1, q_2, \ldots, q_n) \propto p(w) \prod_{m=1}^{n} p(q_m|w) \tag{18}$$

where $p(\boldsymbol{w})$ is the prior which is set to be normally distributed. This Bayesian model can prioritize the reward components in each iteration. Then an expert can regularize the agent's behaviors by providing pairwise preferences. Finally, the weight parameters can be updated using Markov Chain Monte Carlo (MCMC), and the Softmax function is implemented to normalize weight parameters.

This weight-adjusting module can be combined with most off-the-shelf DRL methods. Soft actor-critic (SAC) [10] is selected as the backbone DRL algorithm for multi-objective car-following control. The implementation specifics of PbSAC are outlined in Algorithm 1.

D. Study Scenarios

This paper suggests using a PbSAC-based leading cruise control model for AVs in a three-vehicle car-following situation. To confirm the effectiveness of this approach, four different scenarios are compared:

1. *Scenario 1*: The AV control algorithm based on PbSAC utilizes data from the three vehicles and takes into account the benefits of AV and FHDV. As a result, the agent's state is defined as $[V_1(t), S_{1,2}(t), S_{0,1}(t), \Delta V_{1,2}(t), \Delta V_{0,1}(t), \omega_1, \omega_2, \omega_3, \omega_4]$ and the reward function is specified in Eq. (14). According to [7], there is a trade-off among safety, efficiency and stability where pursuing less efficiency (e.g., larger time gaps) can lead to improvements in safety and stability. Thus, expert preferences are applied in the following way: given two car-following trajectories (τ_i, τ_j), if the discrepancy between the average time gap and $1.5s$ in τ_i is smaller than that in τ_j , $(\tau_i > \tau_j)$ is returned.
2. *Scenario 2*: The AV control algorithm based on PbSAC uses data from both LHDV and AV and only takes into account the benefits of AV. Consequently, the state of the agent is $[V_1(t), S_{0,1}(t), \Delta V_{0,1}(t), \omega_5, \omega_6]$and the reward function is defined in Eq. (19). Expert preferences are provided in a manner similar to scenario 1.

$$R_2 = \omega_5 R_{control\_\mathrm{AV}} + \omega_6 R_{stability\_AV} \tag{19}$$

3. *Scenario 3*: The sole difference between scenario 2 and scenario 3 is that the agent is controlled by SAC, resulting in the manual determination of weight parameters through extensive experiments: [0.5,0.5].
4. *Scenario 4*: The agent is an AV with an MPC based adaptive cruise control (ACC) module. It also does not use FHDV's information and ignores the benefits of FHDV. The constrained linear-quadratic MPC model developed in [11] will be adopted.

III. Experimental Results

A. Data Processing

1) Data extraction: This study uses real-world autonomous driving data from the Waymo Open Dataset [12], including trajectories collected by 5 LiDARs and 5 cameras installed in Waymo AVs. The data was collected at a frequency of 10 Hz on public roads in the U.S. The rules defined in [1][3] are followed to extract qualified car-following events, resulting in 253 HDV-following-AV and 1,301 HDV-following-HDV events.

2) Data denoising: To improve the quality of car-following trajectories, a two-step data denoising approach is implemented. Firstly, an optimization-based outlier removal approach is developed to identify trajectory points with anomalous acceleration and replace them with smooth trajectories using a linear optimization model. The smoothness is determined by the variation between the maximum and minimum acceleration within the optimization horizon. Secondly, a Savitky-Golay filter is applied to remove unusual fluctuations in the position and speed data and further denoise the car-following trajectories.

3) Car-following model fitting: This study adopts Inverse soft-Q Learning [13], an IRL method, to recover the reward functions of human drivers in HDV-following-AV scenarios. By approximating the Q-function, which represents both the reward and policy, this method simplifies the complex min-max game in traditional IRL methods into a simple minimization problem. The recovered reward functions are then used to train an SAC model to mimic human driving policies when following AVs, following similar procedures as in our previous studies [2][3]. Finally, the obtained HDV-following-AV model is adopted to control FHDV in this study.

B. Simulation Setup and PbSAC Training

As mentioned earlier, 1,301 HDV-following-HDV events have been identified in the Waymo Open Dataset. Out of these car-following events, 459 three-HDV platoons are extracted. These three-HDV platoons are randomly divided into training and testing sets, where 80% (367) of them are used for training while the rest of them (92) are used for testing.

In scenario 1, the PbSAC based AV controller framework is illustrated in **Fig. 2**. The LHDV is regarded as an external input while the AV takes its own state and that of the FHDV as inputs. It outputs the longitudinal acceleration and considers the benefits of both itself and the FHDV. On the other hand, the AV control models in scenarios 2, 3, and 4 only focus on AVs and do not take into account FHDVs. In all scenarios, the HDV-following-AV model acquired through IRL is used to control the FHDVs.

In this study, a three-vehicle car-following event is considered an episode during the training process. At the start of each episode, a three-HDV platoon is randomly selected from the training set and the speed profile of the platoon leader is assigned to LHDV in Fig. 2. The initial states of the AV and FHDV in Fig. 2 are set to match the two followers in the platoon. The AV and FHDV then take actions based on their respective controllers and their states are updated according to Eqs. (1)-(6). If a traffic crash occurs during the episode, the training is terminated and a reward of −10 is given. Otherwise, the AV control algorithm ends the current episode at the maximum time step of the LHDV speed profile. Another car-following event is randomly selected from the training set and the states are re-initialized with the new empirical data.

To determine the model convergence, the study uses the rolling mean episode reward as the evaluation metric. The PbSAC model with the highest rolling mean episode reward is chosen as the AV controller in scenario 1. The same training and selection strategy is applied to scenarios 2 and 3.

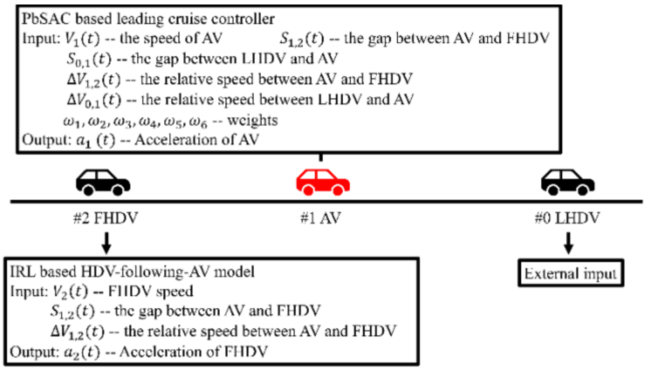

PbSAC based leading cruise controller
Input: $V_1(t)$ -- the speed of AV
$S_{1,2}(t)$ -- the gap between AV and FHDV
$S_{0,1}(t)$ -- the gap between LHDV and AV
$\Delta V_{1,2}(t)$ -- the relative speed between AV and FHDV
$\Delta V_{0,1}(t)$ -- the relative speed between LHDV and AV
$\omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6$ -- weights
Output: a_1 (t) -- Acceleration of AV
#2 FHDV
#1 AV
#0 LHDV
External input
IRL based HDV-following-AV model
Input: $V_2(t)$ -- FHDV speed
$S_{1,2}(t)$ -- the gap between AV and FHDV
$\Delta V_{1,2}(t)$ -- the relative speed between AV and FHDV
Output: $a_2(t)$ -- Acceleration of FHDV

Fig. 2. PbSAC based AV control framework in scenario 1.

C. *Evaluation of the Proposed Approach*

To demonstrate the superior capabilities of the proposed leading cruise controller in safely, efficiently, and stably following LHDVs, the study compares the trajectories of three-vehicle platoons generated in the four scenarios. All results are obtained using the testing set, which is similar to the training set but uses the speed profile of LHDV as an external input. The testing process involves sequentially extracting car-following events from the testing set and selecting actions for the agent using the learned model. The state of the three-vehicle platoon is then updated using Eqs. (1)-(6) until the maximum time step of the LHDV speed profile is reached.

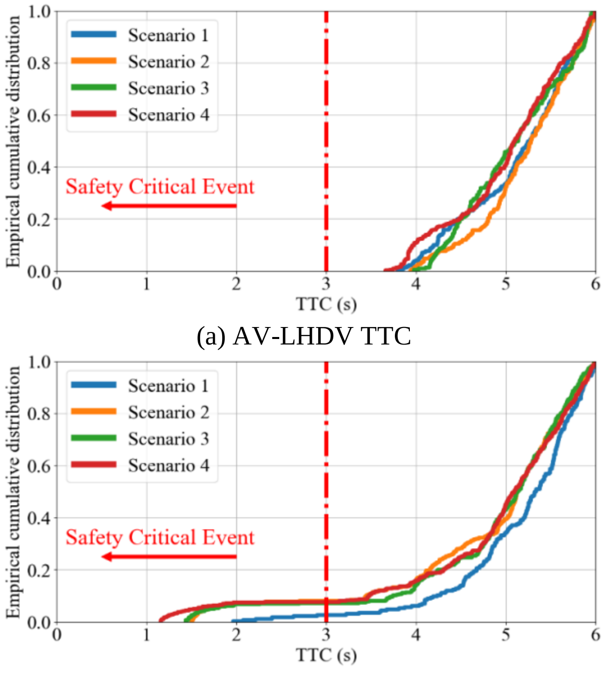
1). Car-following safety: To evaluate car-following safety, the study analyzes critical TTC values in the testing set. **Fig. 3** shows the empirical cumulative distributions of TTCs in the four scenarios. Upon visual inspection, it is evident that none of the scenarios (1, 2, 3, and 4) have critical TTCs for AV-LHDV. Additionally, it is clear that scenario 1 is the safest for FHDV-AV out of all the scenarios, as the blue line consistently remains below the other lines.

To strengthen the statistical validity of the findings, the Kolmogorov-Smirnov test is conducted to compare the AV-LHDV and FHDV-AV TTC distributions in scenario 1 with those in the other scenarios. The results indicate that, except for AV-LHDV TTCs between scenarios 1 and 2 and scenarios 1 and 3, other comparison results are all statistically significant with $p - values$ lower than 0.05. This means that the proposed AV controller, which takes into account the benefits of AV and FHDV, can reduce the risks of rear-end crashes for both AV and FHDV, compared to AV controllers that do not consider FHDV.

2). Traffic efficiency: To evaluate traffic efficiency, this study analyzes the average speeds of AV and FHDV. Table I presents the average speeds for both vehicles. The results suggest that scenarios 1-4 have similar average speeds, which is supported by the Mann-Whitney U test. This is because scenarios 1-4 use either DRL or MPC controllers to regulate the AV, which enables it to follow LHDV closely.

3). String stability: Table I presents the average standard deviation of speed for AV and FHDV. It is observed that the proposed control strategy has a lower average standard deviation of speed for both AV and FHDV compared to the three baseline methods. This indicates that the proposed AV control model serves as a virtual regulator that improves FHDV driving decisions, ultimately reducing traffic oscillations for the entire mixed traffic stream. Additionally, scenario 2 is able to lower the average standard deviation of speed for AV by 3% compared to scenario 3, suggesting that the preference adjusting module can enhance the DRL-based AV controller. By adjusting the weights towards the human-preferred time gap, the time gap distribution becomes more centralized, resulting in the dampening of speed perturbations. This demonstrates the capability of incorporating human prior-knowledge to handle multiple objectives in car-following control.

To sum up, the main findings are as follows: (1) The AV control model proposed in scenario 1 improves safety and string stability while maintaining comparable efficiency to other scenarios; (2) Scenario 2, which uses PbSAC with a preference adjusting module, is more efficient than scenario 3 (SAC) and scenario 4 (MPC) in improving the string stability of the AV; and (3) SAC shows superior string stability compared to MPC.



(a) AV-LHDV TTC

(b) FHDV-AV TTC

Fig. 3. TTC empirical cumulative distributions.

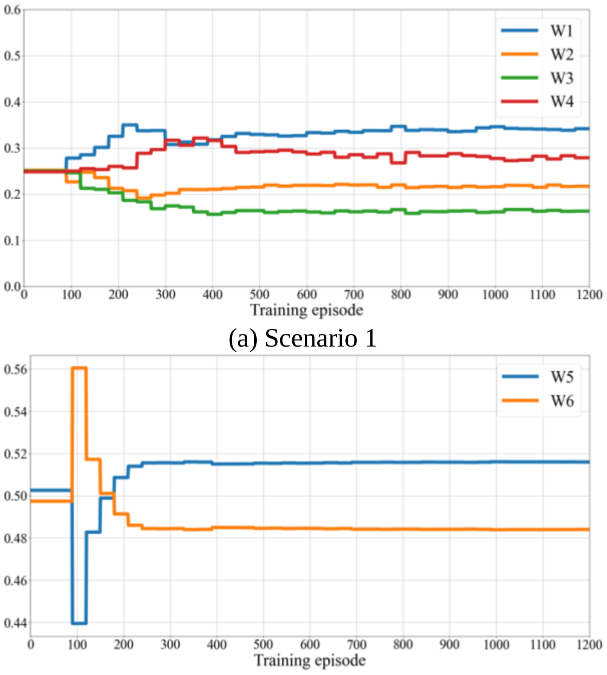
TABLE I

TRAFFIC FLOW PERFORMANCE ON THE WAYMO OPEN DATASET

Scenarios	Average Speed (m/s)		Average Std of speed (m/s)	
	AV	AV	AV	FHDV
1	9.80	9.78	**1.26**	**1.16**
2	**9.83**	**9.81**	1.30	1.17
3	9.82	9.79	1.34	1.20
4	9.80	9.77	1.38	1.23

D. Weight Learning Curves

Fig. 4 shows the convergence of weight parameters during the model training. It is revealed that the proposed preference generator can efficiently determine the stable weight combination, even if there are four terms in the reward function in scenario 1. In addition, experimental results show that weighting rewards with preference weights does not significantly induce instability for policy improvement.



(a) Scenario 1

(b) Scenario 2

Fig. 4. Convergence of the weights in the reward function.

IV. Conclusions

This study proposes a leading cruise controller for a mixed platoon of three vehicles using PbSAC. Unlike previous studies that only focused on optimizing the AV, this approach also considers the benefits of FHDV. Real-world car-following trajectories from the Waymo Open Dataset are extracted, processed, and analyzed. The study uses an IRL approach to model car-following behaviors of HDVs when following AVs. To address the multi-objective car-following control problem, the study introduces a PbSAC-based AV control model that integrates SAC with a preference generator. This generator can dynamically optimize multiple car-following objectives based on expert evaluation. Simulation results demonstrate that this approach can improve the safety, efficiency, and string stability of the mixed traffic stream compared to other AV controllers that do not take FHDV into account or without such a preference generator.

References

[1] 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.

[2] X. Wen, S. Jian, and D. He, “Modeling Human Driver Behaviors When Following Autonomous Vehicles: An Inverse Reinforcement Learning Approach,” in *Proc. 25th International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2022, pp. 1375- 1380.

[3] X. Wen, S. Jian, and D. He, “Modeling the Effects of Autonomous Vehicles on Human Driver Car-Following Behaviors using Inverse Reinforcement Learning,” *IEEE Transactions on Intelligent Transportation Systems*, August. 2023.

[4] X. Wen, C. Huang, S. Jian, and D. He, “Analysis of discretionary lane-changing behaviours of autonomous vehicles based on real-world data,” *Transportmetrica A: Transport Science*, pp. 1–24.

[5] J. Wang, and L. Sun, “Multi-objective multi-agent deep reinforcement learning to reduce bus bunching for multiline services with a shared corridor,” *Transportation Research Part C: Emerging Technologies*, vol. 155, Oct. 2023, Art. no. 104309.

[6] H. Shi, Y. Zhou, X. Wang, S. Fu, S. Gong, and B. Ran, “A deep reinforcement learning - based distributed connected automated vehicle control under communication failure,” *Computer - Aided Civil and Infrastructure Engineering*, vol. 37, pp. 2033-2051, Dec. 2022.

[7] X. Shi and X. Li, “Empirical study on car-following characteristics of commercial automated vehicles with different headway settings,” *Transportation Research Part C: Emerging Technologies*, vol. 128, Jul. 2021, Art. no. 103134.

[8] L. Jiang, Y. Xie, N. G. Evans, X. Wen, T. Li, and D. Chen, “Reinforcement Learning based cooperative longitudinal control for reducing traffic oscillations and improving platoon stability,” *Transportation Research Part C: Emerging Technologies*, vol. 141, Aug. 2022, Art. no. 103744.

[9] M. Peschl, A. Zgonnikov, F.A. Oliehoek, and L.C. Siebert. “MORAL: Aligning AI with human norms through multi-objective reinforced active learning,” *in Proc. 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2022, pp. 1038–1046.

[10] 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 (ICML)*, 2018, pp. 1861–1870.

[11] M. Zhu, Y. Wang, Z. Pu, J. Hu, X. Wang, and R. Ke, “Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving,” *Transportation Research Part C: Emerging Technologies*, vol. 117, Aug. 2020, Art. no. 102662.

[12] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. Qi, Y. Zhou et al., “Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset,” in *Proc. IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2021, pp. 9710-9719.

[13] 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 (NeurIPS)*, 2021, pp. 4028-403