Transportation Research Part C 141 (2022) 103744

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Reinforcement Learning based cooperative longitudinal control for reducing traffic oscillations and improving platoon stability

Liming Jiang [a], Yuanchang Xie [a,*], Nicholas G. Evans [b], Xiao Wen [a], Tienan Li [a], Danjue Chen [a]

[a] Department of Civil and Environmental Engineering, University of Massachusetts Lowell, 1 University Ave, Lowell, MA 01854, United States
[b] Department of Philosophy, University of Massachusetts Lowell, 1 University Ave, Lowell, MA 01854, United States

ARTICLE INFO

Keywords:
Cooperative Longitudinal Control
Adaptive Cruise Control
Ethics
String Stability
Car-following Models
Reinforcement Learning

ABSTRACT

Stop-and-go traffic poses significant challenges to the efficiency and safety of traffic operations. In this study, a cooperative longitudinal control based on Soft Actor Critic (SAC) Reinforcement Learning (RL) is proposed to address this issue. The reward function is carefully designed to consider vehicle cooperation and to achieve three main objectives: safety, efficiency, and oscillation dampening. A global performance metric for oscillation dampening is proposed to evaluate the developed RL and other baseline models. Depending on the number of preceding vehicles that can share maneuver information, two models *RL-1* and *RL-2* are proposed and compared with human driven (HD) and an adaptive cruise control (ACC) model using the *HighD* and simulated data. It is found that with information from additional preceding vehicles, *RL-2* can dampen shockwaves more efficiently. Specifically, *RL-1* and *RL-2* decrease traffic oscillation by 15%-36% and 15%-42%, respectively, while HD amplifies the oscillation by 14–37%. The ACC model can also dampen shockwaves but is not as effective as *RL-1* and *RL-2*. The two RL control methods are further evaluated based on data collected using a commercial Model X vehicle. Compared with the commercial Model X ACC vehicle in some controlled settings, the proposed RL methods can better dampen the stop-and-go waves by generating smaller oscillation growth, overshooting, and average acceleration/deceleration rate change, suggesting that they can generalize well in a new but similar environment. Finally, the RL methods are evaluated considering a platoon of vehicles with different RL penetration rates. The results show that they consistently outperform HD and ACC in dampening shockwaves.

1. Introduction

Stop-and-go traffic is a common phenomenon and has important impacts on traffic operations (Sugiyama et al., 2008). Small perturbations in a lead vehicle's speed profile could be amplified as they are passed on to following vehicles and this creates stop-and-go waves broadcast backwards (i.e., traveling upstream), which results in wasted fuel consumption, additional traffic emissions, increased likelihood of rear-end crashes, and congestion. It is concluded that shorter reaction time and better sharing of vehicle maneuver information are among the keys to address this issue (Li et al., 2014). Existing Adaptive Cruise Control (ACC) and Automated

* Corresponding author.
E-mail addresses: Liming_Jiang@student.uml.edu (L. Jiang), Yuanchang_Xie@uml.edu (Y. Xie), Nicholas_Evans@uml.edu (N.G. Evans), Xiao_Wen@student.uml.edu (X. Wen), Tienan_Li@student.uml.edu (T. Li), Danjue_Chen@uml.edu (D. Chen).

https://doi.org/10.1016/j.trc.2022.103744
Received 12 August 2021; Received in revised form 20 January 2022; Accepted 2 June 2022
Available online 9 June 2022
0968-090X/© 2022 Elsevier Ltd. All rights reserved.

Vehicles (AV) control systems are designed mostly to maximize the utility of the subject vehicle without considering cooperation and the utility of the traffic system. Although they can substantially improve the safety of the subject vehicle, such "self-centered" control algorithms are not designed for addressing the stop-and-go shockwave issue and the safety of other vehicles surrounding the subject vehicle.

AV with cooperative control have attracted much attention recently. They have been adopted in cooperative merging (Ren et al., 2020a; 2020b) and stabilizing traffic (Ge and Orosz, 2014; Stern et al., 2018; Wu et al., 2018). This study argues that future vehicle control algorithms should also consider traffic system utility in addition to individual vehicle utility. It further explores the potential of cooperative vehicle control in addressing stop-and-go traffic and proposes a novel cooperative longitudinal control approach based on the Soft Actor Critic (SAC) Reinforcement Learning (RL). The proposed RL-based algorithms implicitly take traffic flow stability into consideration. A combination of real-world and simulated data is used to demonstrate the proposed algorithms' capability to absorb stop-and-go shockwaves, reduce traffic oscillations, and improve platoon stability in a mixed autonomy environment with both human drivers and AV. The proposed cooperative longitudinal control concept and algorithms can be integrated into traditional ACC and future AV control, so that such vehicles can act as traffic stabilizers in the future to improve traffic system operations.

The main contributions of this study are: (1) instead of using a closed-loop ring road network and assuming the location and maneuver information of all vehicles to be known (Stern et al., 2018; Wu, 2018), we consider a more realistic straight roadway segment and take only the cooperative and automated vehicle (CAV) and its lead vehicles' states as the inputs; (2) the reward function is carefully designed to include terms specifically for dampening shockwaves; (3) the proposed RL control agents are also trained and tested considering field collected vehicle trajectories in naturalistic driving settings, instead of purely simulated data; (4) the proposed RL control agents are further compared with a commercial ACC system based on a combination of field and simulated data; and (5) the proposed RL agents are tested in a long vehicle platoon.

2. Background

2.1. Related work

Instead of using traditional control theory and formula-based analytic solutions for AV control, some studies (Desjardins and Chaib-draa, 2011; Khodayari et al., 2012; Morton et al., 2017; Wang and Chan, 2017; Zhu et al., 2019) adopted machine learning algorithms, especially Reinforcement Learning (RL). By carefully choosing the state representation and reward function, an RL agent (i.e., AV) is able to learn how to regulate its longitudinal behavior based on incentives (rewards) received during interactions with other vehicles. The first study using RL to control AV in a connected environment was conducted by Desjardins and Chaib-draa (2011). They concluded that RL-based control could be a promising approach to ensure safe longitudinal following behavior of AV. Later RL was adopted in many AV behavior modeling, such as longitudinal control (Zhu et al., 2019) and merge control (Ren et al., 2020b; Wang and Chan, 2017).

The most relevant studies to this paper are Wu (2018), Vinitsky et al. (2018), and Qu et al. (2020). In Vinitsky et al. (2018), four tasks including merging and intersection control were investigated considering AV controlled by RL. To study the potential of AV as traffic stabilizers, Wu (2018) and Qu et al. (2020) both considered a closed-loop ring road network, which was loaded with Human-Driven (HD) vehicles and AV controlled by RL. The RL-controlled AV in Wu (2018) was assumed to have a global (or complete) view of the environment (i.e., speeds and positions of all vehicles), and each AV learned to address the stop-and-go traffic by maximizing its reward function, which was defined based on all vehicles' speeds and headways. Although the concept and results of these studies are very interesting, assuming a global view is quite restrictive. Also, their RL controllers were trained based completely on simulated vehicle trajectories and on a ring road network, which may not accurately reflect vehicle maneuvers in practice.

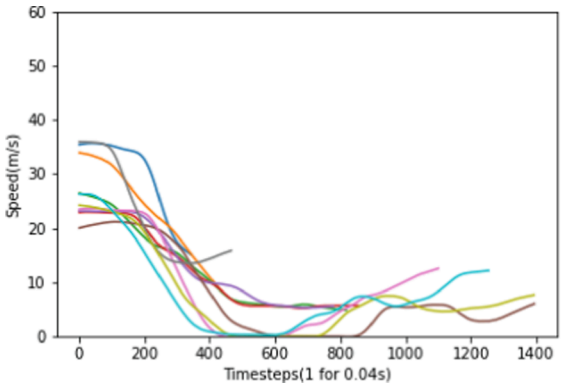
Some other relevant RL studies focused on Cooperative Adaptive Cruise Control (Chu and Kalabić, 2019; Desjardins and Chaib-draa, 2011). These studies adopted RL to train AV so that they can stably follow the lead vehicle. These studies did not explicitly consider using the RL-controlled AV to dampen the impacts of stop-and-go shockwave on vehicles following them. In other words, these AV use RL to improve their own longitudinal control performance and behave "selfishly" without considering vehicles behind them.

AV can be trained/designed to behave selfishly or a little cooperatively (or even altruistically). An interesting but not fully understood question is how and to what extent these different behaviors may affect traffic operations, which is also a key motivation of this study. To our best knowledge, this is among the few studies attempting to adopt Reinforcement Learning (RL) approach for AV control that incorporates cooperation or altruism into the control objective (e.g., dampening stop-and-go wave for the benefits of vehicles behind it) and the first study considering both real-world and simulated vehicle trajectories instead of purely simulated ones for model training and testing.

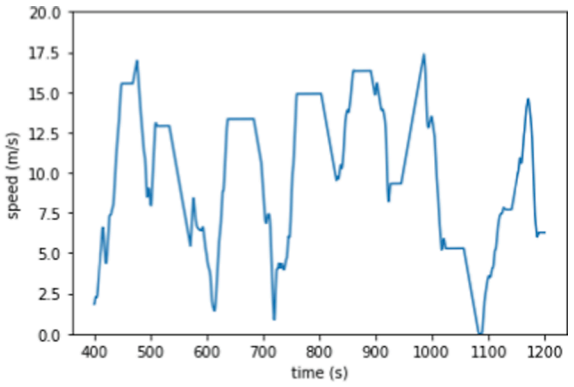
2.2. Soft Actor-Critic (SAC)

With Reinforcement Learning (RL) control, each AV is treated as an RL agent. The agent learns optimal control policies through its interactions with the environment (i.e., surrounding vehicles). Good control policies are rewarded while bad ones are penalized through the reward function. Over time the agent learns to adjust its behavior to maximize the cumulative reward or return.

In this research, our goal is to optimally adjust an AV's action (i.e., its acceleration/deceleration for the next time step). A continuous action space is considered, and the acceleration/deceleration can be any value between -3 to $2\ m/s^2$. Although the range is



(a) Speed Profiles of Randomly Selected Vehicles



(b) A Section of the Stitched HighD Speed Profile (total length is 11,000s)

Fig. 1. HighD Speed Profiles Used in this Study.

not very wide, the action space still is complicated given that it is continuous (e.g., 1.115 m/s^2 and 1.116 m/s^2 are two different actions). Therefore, the Soft Actor-Critic (SAC) algorithm (Haarnoja et al., 2018a) is adopted. SAC is an actor-critic and model-free RL algorithm. A brief introduction of SAC is provided below, and more detailed information can be found in the original paper.

Actor-critic RL methods combine the advantages offered by both value-based and policy-based methods by employing an actor (to execute an action) and a critic (to evaluate the action made by the actor). This allows actor-critic RL to be more sample efficient (Lillicrap et al., 2019).

The goal of traditional RL algorithm is to train an agent to maximize the expected sum of rewards $\sum_t \mathbb{E}_{(s_t,a_t)\ \rho_\pi}[r(s_t,a_t)]$, while SAC introduces an additional term into the objective: maximum entropy.

$$\sum_{t=0}^{T} \mathbb{E}_{(s_t,a_t)\ \rho_\pi}[r(s_t,a_t) + \lambda \mathscr{H}(\pi(\hat{\mathrm{A}} \cdot |s_t))] \tag{1}$$

where λ is the temperature parameter which controls the relative importance of entropy against reward. In other words, it controls the extent of exploration of the RL policy. The entropy term could be further expanded as:

$$\mathscr{H}(\pi(\hat{\mathrm{A}} \cdot |s_t)) = -\mathbb{E}_{a_t} \log \pi(a_t|s_t) \tag{2}$$

With this objective design, the policy is incentivized to explore the action space (i.e., to reach the next state with larger entropy). Unlike deterministic RL algorithms such as deep deterministic policy gradient (DDPG), the policy here gives similar importance to near-optimal actions that are approximately equally attractive, which encourages the exploration of the action space so that the agent is less likely to get stuck in local optimums.

There are two models (i.e., critic and actor) need to be trained for SAC: $Q_\theta(s_t,a_t)$ and $\pi_\phi(a_t|s_t)$. Neural networks are adopted for the two models. The Q-function $Q_\theta(s_t,a_t)$ is for critic, which takes (s_t,a_t) pair at time t and outputs its estimated long-term expected reward. The tractable policy $\pi_\phi(a_t|s_t)$ parameterized by ϕ takes state at time t and produces probabilities for each action. Since the action space (e.g., vehicle acceleration) in this study is continuous (which is well suited for SAC), the output of the policy network is a Gaussian distribution with mean and covariance. For actions in the vicinity of the optimal action, their probabilities are similar to the optimal action's probability but lower.

The optimization/training of both models usually adopts stochastic gradients. More details could be found in (Haarnoja et al., 2018b). The training of SAC behaves stably and is insensitive to different random seeds. It converges faster than traditional RL methods, encourages exploration, and ultimately produces higher convergence episode reward. It has been empirically shown that SAC outperforms other RL algorithms such as deterministic policy gradient (DDPG) and proximal policy optimization (PPO) (Haarnoja et al., 2018a) for continuous control problems.

3. Data and methodology

3.1. Data and simulation environment

The main purpose of this study is to design RL agents that can dampen traffic shockwaves generated by their preceding vehicles. Therefore, lane-changing behaviors are not considered. The proposed RL agents have to be trained using lead vehicles' trajectories. To show that the trained RL agents can tackle realistic car-following tasks, the lead vehicles' trajectories are sampled from the *HighD* dataset (Krajewski et al., 2018) (instead of using simulated trajectories), which was collected using drones from German highways and reflects real-world naturalistic driving behaviors. *HighD* trajectories cover both free-flow and congested traffic. Since free-flow traffic

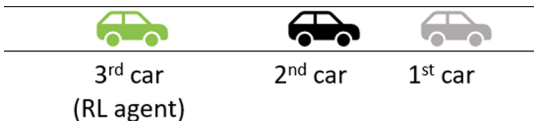


Fig. 2. Training scenario (1^{st} vehicle: speed profile from HighD training; 2^{nd} vehicle: HD or ACC; and 3^{rd} vehicle: RL agent).

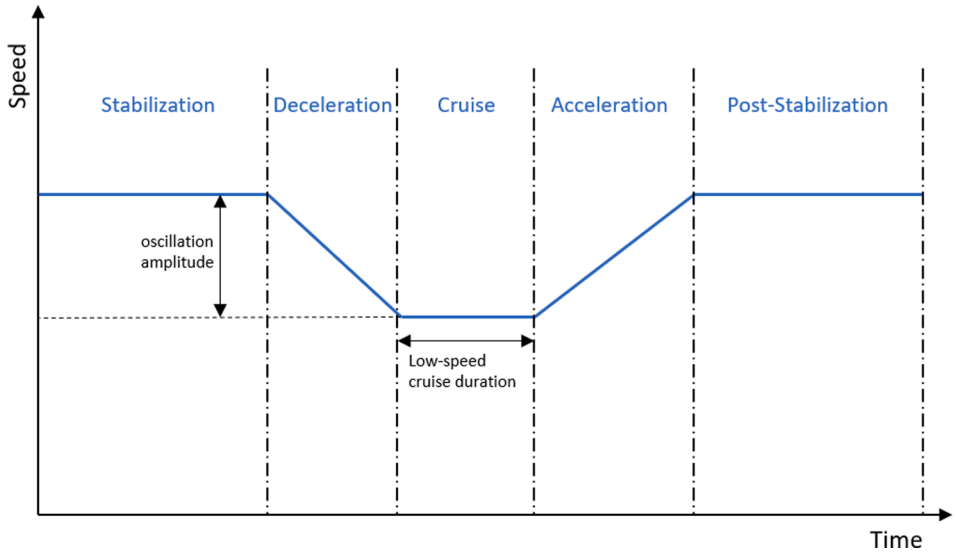


Fig. 3. Driving Cycle of the Lead Vehicle in Model X Data Collection.

does not tell us much about AV's capability of absorbing the speed oscillation of the lead vehicle, trajectories reflecting congested traffic (see Fig. 1) are used in this study. Specifically, only trajectories with a maximum speed less than $18m/s$ and a speed standard deviation more than $2m/s$ are selected. These trajectories are concatenated together to generate the speed profile of the lead vehicle considered in this study. The resultant lead vehicle speed profile is equivalent to a 70-kilometer single-lane road segment where vehicles follow and are constrained by their preceding vehicles.

The starting speed of a sampled *HighD* trajectory often does not precisely match the ending speed of the previous trajectory, although the difference may be small. To concatenate such trajectories, a gentle acceleration/deceleration rate ($\pm 0.2m/s^2$) is adopted to stitch these trajectories together. For instance, if the ending speed of the previous trajectory is smaller than the starting speed of the next trajectory, a positive acceleration rate of $0.2m/s^2$ is adopted to fill the speed gap of the two consecutive trajectories. Otherwise, a deceleration rate of $-0.2m/s^2$ is adopted. In addition, a stabilization phase of $10 \sim 50$ s (the phase length is randomly drawn) is inserted between two consecutive trajectories. In the stabilization phase, the vehicle maintains a constant speed which equals the beginning speed of the next sampled trajectory. The stitched *HighD* trajectory data (A sample section is shown in Fig. 1(b)) is further split into *HighD training* (8,000 s, starts from 0 s and ends at 8,000 s) and *HighD testing* (3,000 s, starts from 8,000 s and ends at 11,000 s). *HighD training* is utilized for RL agents training, and *HighD testing* is used for evaluation. For both the *HighD training* and *testing* datasets, the step length 0.2 s (i.e., there are 55,000 data points in total).

Since a main goal of this study is to research the impacts of stop-and-go shockwaves on vehicle platoons, having the trajectory of one vehicle is not enough. Therefore, SUMO (Simulation of Urban MObility) simulation is used to generate additional vehicle trajectories for RL agents training and testing. During the SUMO simulation, a 3-vehicle platoon is created as shown in Fig. 2. The 1st vehicle's trajectory is given by the 8,000 s *HighD training* data that creates stop-and-go traffic patterns. This lead vehicle is followed by the 2nd vehicle (i.e., a Human-Driven (HD) vehicle or an Adaptive Cruise Control (ACC) vehicle) and the 3rd vehicle (an RL-controlled agent). It is assumed that the 1st and 2nd vehicles may choose to share maneuver information with the RL agent via vehicle-to-vehicle (V2V) communications in real time. Based on the 1st and 2nd vehicles' states over time, the RL agent learns how to control its longitudinal behavior cooperatively with the purpose to dampen stop-and-go shockwaves.

The 2nd vehicle in Fig. 2 could be either an HD or ACC vehicle. Its trajectory is also used in the training and evaluation of RL agents as detailed later in the paper. Introducing this 2nd vehicle helps to diversify the training and testing scenarios. If it is an HD vehicle, the Wiedemann 99 model (Aghabayk et al., 2013) is adopted to simulate its behavior. For ACC vehicle, the ACC driving behavior model developed by Xiao et al. (2017) is used. The adopted ACC model allows human drivers' takeover and has collision-free property. The desired time headway is set to 2 s and the standstill distance is 2 m for both the Wiedemann 99 and ACC models. For other parameters, the default values (German Aerospace Center (DLR) and others, 2021) in SUMO are utilized.

In Fig. 2, the 1st vehicle's trajectory is sampled from *HighD* data, and the 2nd vehicle's trajectory is simulated using SUMO.

Theoretically, we could also simulate the 1st vehicle's trajectory using SUMO. It is unlikely that we can generate realistic trajectories as those sampled from the *HighD* dataset. The hybrid dataset (i.e., observed and simulated) is used to train and test the proposed RL agents. In addition, this study also considers experimental data generated by some commercial vehicles equipped with ACC (hereinafter referred to as Model X) (Li et al., 2021) to evaluate the RL agents. Both the Model X and *HighD* data were collected in the field rather than using a traffic simulator. The main difference between them is that the speed profiles of *HighD* data were collected in a naturalistic setting, while the speed profiles of Model X were designed to artificially create different driving patterns of the lead vehicle. Also, the Model X experiment collected data from a platoon of three vehicles (one HD vehicle followed by two Model X vehicles). If we replace the last Model X vehicle in the platoon by our RL agents, it is possible to compare the performance of the trained RL agents with Model X ACC system based on field data, which can be very interesting. To our best knowledge, we are not aware of any studies that have done this type of comparison to evaluate the performance of RL-based cooperative longitudinal control algorithms.

During the Model X data collection experiment, the lead Model X vehicle was controlled by a human driver, who was instructed to produce a set of pre-designed speed profiles reflecting different oscillation and stimulus. Specifically, the lead vehicle went through a five-phase driving cycle as in Fig. 3: stabilization, deceleration, cruise, acceleration, and post-stabilization. In the original study (Li et al., 2021), the authors considered many driving cycles characterized by different headway settings, stable speeds, oscillation amplitudes, low speed cruise patterns, and deceleration/acceleration maneuvers. To best mimic the stop-and-go traffic, this study only uses data for cases of *low* stable speed (15.6 m/s), *large* oscillation amplitude (4.5 m/s), *strong* deceleration/acceleration maneuvers with varying headway settings, and low-speed cruise patterns. The headway settings considered in this study include headway-1 (with an estimated time headway of 1.10 s) and headway-3 (with an estimated time headway of 1.45 s). The low-speed cruise patterns considered are "dip" and "long cruise". "dip" is for the lead vehicle to brake briefly followed immediately by acceleration (i.e., no cruise time between braking and acceleration), and "long cruise" is where the lead vehicle brakes and maintains its minimum speed for $10 \sim 15$ s before accelerating. Note that the Model X dataset is not utilized for RL agents training, but only for evaluating the trained RL agents' generalization ability.

3.2. Modeling of RL control

This section focuses on how the proposed SAC control is implemented and trained. After the RL model (i.e., SAC) has been decided, four critical aspects of an RL control algorithm can significantly affect its performance: system state representation, action space definition, reward function, and hyperparameters. These key topics are described in detail below.

3.2.1. State representation

The system state represents what an agent can sense regarding the surrounding environment and itself. In this study, the system state is characterized by the maneuver information of the lead vehicles and the ego vehicle (the RL-controlled AV). It is assumed that all lead and ego vehicles are connected via on-board devices to share movement dynamics information in real time. Depending on how many lead vehicles the ego vehicle is connected to, two kinds of RL-agents are proposed in this study: *RL-1* and *RL-2*. For *RL-1*, the ego vehicle is only connected to the immediate lead vehicle, and its state representation is defined as:

$$\{\Delta s_1, v_1, \mathrm{a}_1, v_0, a_0\}$$

where Δs_1 is the distance (i.e., space headway) between the ego vehicle and its immediate lead vehicle. v_1 is the speed of the lead vehicle. Subscript "1″ refers to the 1st vehicle downstream of the ego vehicle. v_0 and a_0 are the speed and acceleration of the ego vehicle itself, respectively. All those variables are collected from the previous time step. For *RL-2*, the ego vehicle is assumed to be connected with two lead vehicles right in front of it, and the state representation for *RL-2* is defined as:

$$\{\Delta s_2, v_2, \mathrm{a}_2, \Delta s_1, v_1, \mathrm{a}_1, v_0, a_0\}$$

The definitions of Δs_2 and v_2 are similar to those of Δs_1 and v_1. Subscript "2″ now refers to the 2nd vehicle downstream of the ego vehicle. Among vehicles "1" and "2", "1" is closer to the ego vehicle.

3.2.2. Action space

In this study, action space defines the range of acceleration actions the RL agents can execute at each time step. To reflect the realistic acceleration behaviors of typical vehicles, we restrict the acceleration rate (i.e., action space) to be continuous and in the range of (-3, 2) m/s^2.

3.2.3. Reward function

Reward function measures how an executed action can benefit the agent. It turns changes in the environment into quantitative measures that guide the training of RL agents' behavior. A well-designed reward function helps RL agents learn the intended behavior quickly and facilitates the training process to converge fast.

Our reward function design aims to achieve three main goals. The first goal is to ensure safety. Depending on the time headway between the lead and ego vehicles, the agent will be given the safety reward defined in Eq. (3). Whenever a crash occurs (i.e., time headway ≤ 0) during the RL agent training, a large penalty of -100 is given to the agent. When the time headway to the lead vehicle is larger than 1 s, no headway penalty is given to the agent (Note that in this case there could be other types of reward/penalty assigned to the agent). When the headway is between 0 s and 1 s, a circular function is adopted to gradually change the reward from -100 to 0.

Compared to only having one reward term to penalize the scenario of *headway* ≤ 0, this gradual transition of reward contributes to more efficient optimization of the RL agent's policy.

$$Reward_{headway} = \begin{cases} -100, (if h \leq 0) \\ -100 + \sqrt{100^2\left(1-(x-1)^2\right)}, \\ (if 0 < h \leq 1) \\ 0, (if h > 1) \end{cases} \tag{3}$$

The second goal is efficiency. It is obvious that one can design a safe AV control to dampen stop-and-go waves by letting the ego vehicle (i.e., the AV) travel at a constant but very low speed. As long as the ego vehicle's speed is much less than the lead vehicle's average speed, the ego vehicle most likely will not need to decelerate and can maintain a safe headway with the lead vehicle. However, this approach would constantly increase the ego vehicle's gap to the lead vehicle, thus making the ego vehicle a moving bottleneck and increasing the anxiety of drivers behind it. Therefore, the second goal is to ensure that the ego vehicle maintains a safe time headway but also a reasonably fast speed.

In our design, the efficiency or speed goal is broken down into two parts as in Eq. (4), in which $Reward_{speed}$ is the reward regarding the speed of the ego vehicle, v_{Ept} is the expected speed or speed limit, and v_{ego} is the speed of the ego vehicle. The first part of Eq. (4) v_{Ept} encourages the ego vehicle to maintain a speed close to the speed limit. The second part penalizes the ego vehicle for going slower than the expected speed but does not reward or penalize it for going faster than it. By combining the two parts, the goal is for the ego vehicle to drive as fast as it can but not to travel faster than the speed limit.

$$Reward_{speed} = v_{Ept} - Max\left(0, v_{Ept} - v_{ego}\right) \tag{4}$$

The third goal is to reduce speed variation or dampen the stop-and-go traffic. In other words, when the lead vehicle executes a hard deceleration, the ego vehicle should be able to predict that and take proactive actions (e.g., maintain a large time headway in anticipation of the hard deceleration) so that a hard deceleration is not needed for the ego vehicle. Also, the following vehicles behind the ego vehicle would not need to brake hard either.

To achieve this goal, a speed penalty term defined in Eq. (5) is considered if the time headway h is smaller than a critical headway value h_c. The rationale behind this is that the ego vehicle is not supposed to travel faster than its lead vehicle when it is already very close to the lead vehicle ($h < h_c$). If it does (e.g., in the situation that the lead vehicle decelerates), the ego vehicle should be penalized by having a larger speed relative to its lead vehicle.

$$Reward_{speeddiff} = \begin{cases} (v_{ego} - v_{lead})^*(h - h_c), if v_{ego} > v_{lead} and h < h_c \\ 0, otherwise \end{cases} \tag{5}$$

where $v_{ego} - v_{lead}$ is speed difference between the ego vehicle and its lead vehicle, h is the current time headway of the ego vehicle to its lead vehicle, and h_c is the critical headway set to be 2 s in this study. This reward is calculated only if $v_{ego} > v_{lead}$ and $h < h_c$. Based on Eq. (5), more penalty is given to the RL agent when it drives faster than its lead vehicle and keeps a shorter than critical gap to its lead vehicle.

Another reward function term for achieving the third goal is to penalize large acceleration rates. Large acceleration rates (either negative or positive) should be penalized to ensure smooth driving and help to reduce speed oscillation. This term is defined as follows:

$$Reward_{acc} = -a_{ego}^2 \tag{6}$$

As in Eq. (6), to further penalize large accelerations, the ego vehicle acceleration is squared. The following Eq. (7) is the complete reward function that includes all previously discussed reward terms covering three goals: safety, effiency, and oscillation dampening.

$$Reward = \alpha^* Reward_{headway} + \beta^* Reward_{speed} + \gamma^* Reward_{speeddiff} + \delta^* Reward_{acc} \tag{7}$$

where α, β, γ, and δ are weights for different components of the reward function. The following values are used for them after careful parameter tuning: $\alpha = 1$, $\beta = 1$, $\gamma = 4$, and $\delta = 4$.

3.2.4. Training setup and hyperparameters

As described earlier, two RL agents are trained in this study: *RL-1* and *RL-2*. During the training, HD and ACC are both considered to control the 2nd vehicle in Fig. 2, and the 1st vehicle follows the stitched trajectories sampled from the *HighD* dataset. At the start of an episode training of *RL-2*, the type of the 2nd vehicle in Fig. 2 is randomly selected to be either HD or ACC. The HD or ACC control model is then used to generate its trajectory. The *RL* agent is the 3rd vehicle in the platoon. For *RL-1*, it learns how to control its longitudinal behavior based on the trajectory of only the 2nd vehicle, while for *RL-2*, its learning is based on the trajectories of both lead vehicles (i.e., the 1st and 2nd vehicles in Fig. 2).

Since SAC follows the actor-critic architecture, an actor model and a critic model need to be specified. The goal of the actor model is to optimize policy. It takes state representation as the input and generates a corresponding action. The critic model seeks to optimize the value function that evaluates each state's expected return. In this study, a multilayer perceptron (MLP) with 2 hidden layers (each has 64 neurons) is adopted for both the critic and the actor models.

During the model training and evaluation, the RL agents collect information from the SUMO simulation platform to identify system state, calculate reward, and send driving instructions to SUMO for the ego vehicle. Some important hyperparameters used in training include discount factor$\gamma = 0.99$, learning rate=0.0003, and training batch size = 64. The size of memory replay (Schaul et al., 2016)

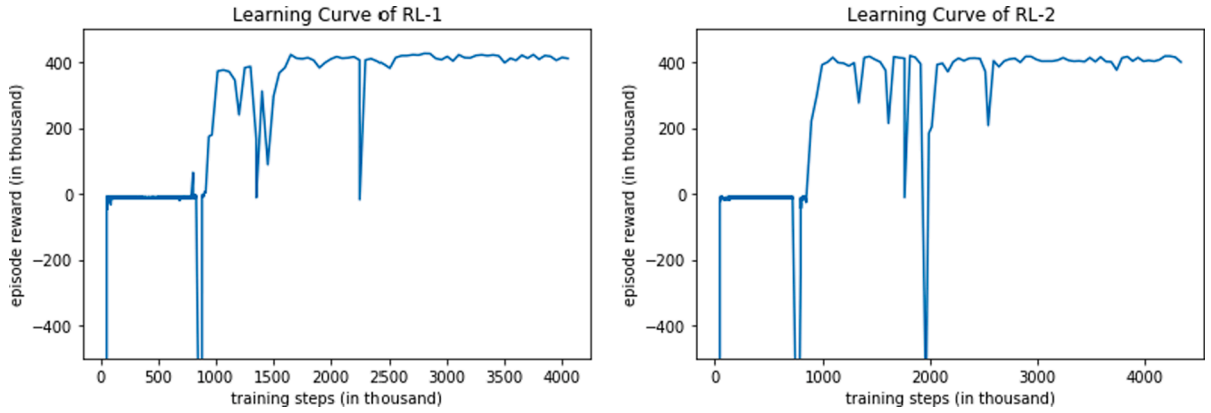


Fig. 4. Learning Curve for RL Agents.

buffer is set to 50,000. The temperature coefficient λ in the objective function of SAC is automatically learned through a dual objective (Haarnoja et al., 2018a).

3.3. Model evaluation

Three evaluations are conducted to thoroughly investigate the RL agents' capability to reduce traffic oscillations and improve platoon stability. The first evaluation is based on the speed profiles in *HighD training* and *HighD testing*. The 1st vehicle (see Fig. 2) follows the *HighD* speed profile. It is followed by the 2nd vehicle controlled by either HD or ACC logic. The 3rd vehicle in the platoon is controlled by *RL-1* or *RL-2* to dampen stop-and-go shockwaves. In the first evaluation, two baseline models are included for comparison. These two baseline models use HD and ACC, respectively, to control the 3rd vehicle (instead of *RL-1* or *RL-2*) to demonstrate the superiority of the proposed RL-controlled agents.

The second evaluation is based on the Model X experimental data, which was collected using a platoon of three vehicles. Among them, the lead vehicle was an HD and the other two vehicles were Model X equipped with ACC (Li et al., 2021). Unlike the *HighD* data, the speed profile of the 1st HD vehicle does not reflect human drivers' naturalistic behavior. Its speed profile was carefully designed to follow certain patterns of oscillation. Four patterns in the Model X dataset are considered in this study. In the second evaluation, the 3rd Model X vehicle is assumed to be controlled by the RL agents (*RL-1* or *RL-2*), so that the behavior of the 3rd Model X vehicle and our proposed RL agents can be compared.

The third evaluation focuses on the impacts of the proposed RL control on a long vehicle platoon. It again utilizes the *HighD testing* data, and the first vehicle in the platoon follows the *HighD* speed profile. Different from the first evaluation, the third evaluation simulates a platoon of 101 vehicles instead of 3 vehicles. In the first evaluation, only the last vehicle is controlled by RL. While in the third evaluation multiple specialized agents (SA, they could be HD, ACC, *RL-1*, or *RL-2*) are inserted into the long platoon to absorb shockwaves. The purpose is to understand how SA of different types and penetration rates can affect the propagation of stop-and-go shockwaves in a long platoon.

The following rolling mean and standard deviation of speed are proposed to measure the capability of *RL-1* and *RL-2* to absorb hard accelerations/decelerations of the lead vehicle. These two metrics consider a time window of size T and are time dependent. In other words, their values vary with time and can be calculated at each time step k.

$$\bar{x}_k^T = \frac{1}{T}\sum_{i=k}^{k+T} x_i \tag{8}$$

$$s_k^T = \sqrt{\frac{1}{T-1}\sum_{i=k}^{k+T}(x_i - \bar{x}_k^T)^2} \tag{9}$$

where T is the rolling time window length, $\bar{x}_k^T$ is the mean speed starting at the k^{th} time step of length T, and s_k^T is the rolling standard deviation of speeds starting at the k^{th} time step. To measure the overall local speed variation over the entire evaluation period, the average of all rolling standard deviations $\bar{s}^T$ is utilized as defined below.

$$\bar{s}^T = \frac{1}{end - start - T - 1}\sum_{i=start}^{end-T} s_i^T \tag{10}$$

In addition to $\bar{s}^T$, four more metrics are adopted specifically for the Model X data (i.e., the second evaluation study) to analyze each control model's oscillation amplification/dampening effects. The four metrics are *average deceleration rate change* $\Delta\bar{d}$, *average acceleration rate change* $\Delta\bar{a}$, *oscillation growth* ϕ, and *overshooting* ψ. Before introducing these metrics, some critical points (CP) need to be

Table 1
Oscillations of 1st, 2nd and 3rd Vehicles in the *HighD* training data.

1st veh.	$\bar{s}^T$ (1st veh.)	2nd veh. type	$\bar{s}^T$ (2nd veh.)	$\bar{s}^T$ change pct. (2nd)	3rd veh. type	$\bar{s}^T$ (3rd veh.)	$\bar{s}^T$ change pct. (3rd)
HighD training	0.22	HD	0.33	+50.0%	HD	0.30	−9.1%
					ACC	0.24	−27.3%
					RL-1	0.21	−36.4%
					RL-2	0.19	−42.4%
		ACC	0.20	−9.1%	HD	0.32	+60.0%
					ACC	0.19	−5.0%
					RL-1	0.17	−15.0%
					RL-2	0.17	−15.0%

Table 2
Oscillations of 1st, 2nd and 3rd Vehicles in the *HighD* testing data.

1st veh.	$\bar{s}^T$ (1st veh.)	2nd veh. type	$\bar{s}^T$ (2nd veh.)	$\bar{s}^T$ change pct. (2nd)	3rd veh. type	$\bar{s}^T$ (3rd veh.)	$\bar{s}^T$ change pct. (3rd)
HighD testing	0.23	HD	0.32	+43.5%	HD	0.30	−6.3%
					ACC	0.23	−28.1%
					RL-1	0.21	−34.4%
					RL-2	0.19	−40.6%
		ACC	0.21	−13.0%	HD	0.32	+52.4%
					ACC	0.19	−9.5%
					RL-1	0.17	−19.1%
					RL-2	0.17	−19.1%

defined, which are deceleration start time, deceleration end time, minimum speed time, acceleration start time, and acceleration end time. The minimum speed time is when the lowest speed is reached in a driving cycle. The remaining points are identified using wavelet analysis (Zheng et al., 2011) and some thresholds (Li et al., 2021). $\Delta\bar{d}$ is the average deceleration of the following vehicle (the sign of deceleration is always positive) minus the average deceleration of the lead vehicle. The two average decelerations are calculated based on deceleration start and end times of the lead and following vehicles, respectively. $\Delta\bar{a}$ is defined similarly as $\Delta\bar{d}$. ϕ equals the minimum speed of the leader minus the minimum speed of the follower in one driving cycle. ψ equals the speed of the follower at the end of its acceleration phase minus the speed of the leader at its acceleration end time. If a follower can dampen the oscillation produced by its leader, $\Delta\bar{a}$, $\Delta\bar{d}$, ϕ, and ψ should all (or mostly) be negative.

4. Analysis of results

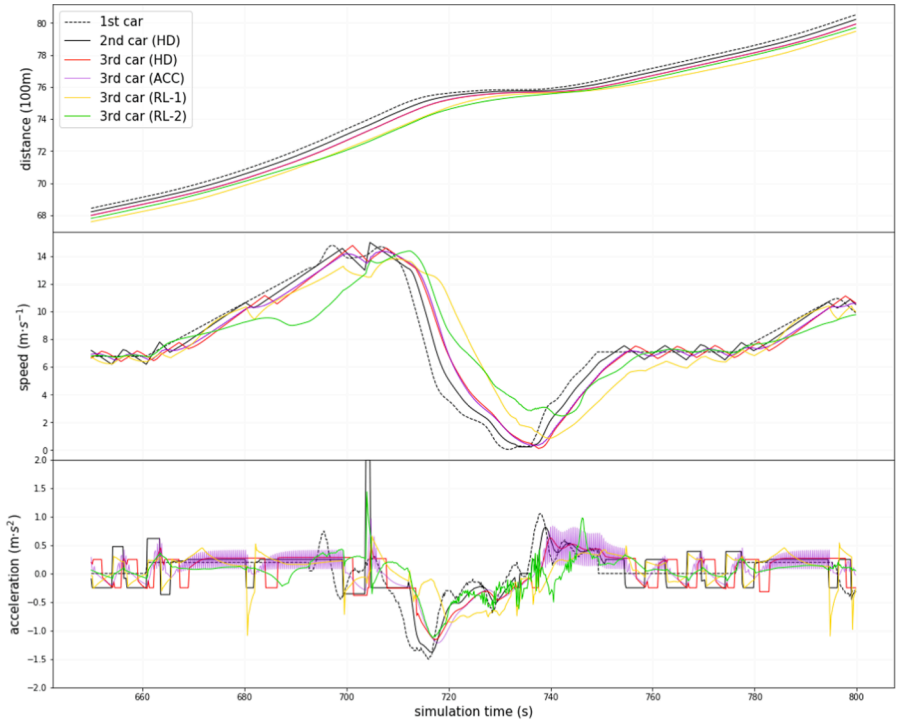
4.1. Results based on HighD data

The learning/training curves for *RL-1* and *RL-2* are presented in Fig. 4. The *y* value of each data point is the cumulative reward at the end of an episode or simulation run, and the *x* value is measured in thousand seconds and indicates when the corresponding episode is completed. Fig. 4 suggests that the RL training processes stabilize after approximately 2,500,000 training steps (each training step equals 0.2 s), but still see small fluctuations (reflected by episode reward changes). A possible reason is that in each episode the second vehicle in Fig. 2 was randomly assigned to be either an HD or ACC. Such randomness and explorative nature of RL agents in the training phase could have contributed to the fluctuations in episode rewards. The occasional sudden drops indicate that one or more crashes occurred during those episodes as a result of executing the acceleration/decelerations generated by the proposed RL algorithms, which are normal especially in the early stage of the training process. With more training steps, the RL algorithms become more reliable and are less likely to generate risky or suboptimal accelerations/decelerations. Therefore, the learning curve (episode reward) increases and stabilizes as the number of training steps increases.

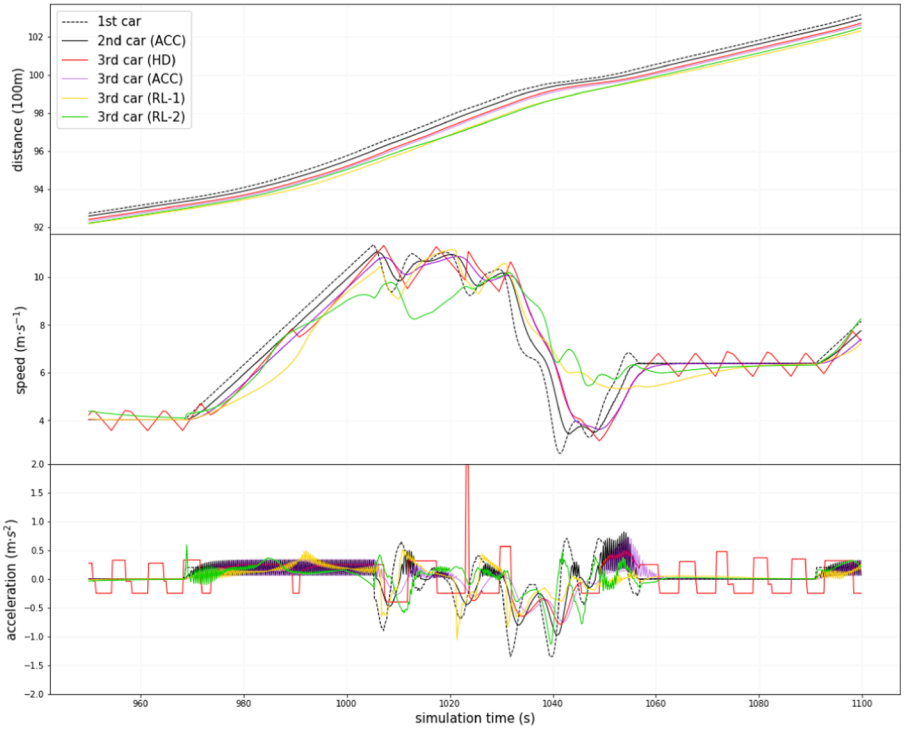
To find the best trained agents, an evaluation was carried out every 10 training episodes and the best agent would be updated if the evaluation resulted in a higher reward compared to all existing ones. The evaluation was done separately from the training process by averaging the rewards of 5 episodes. After 4,000,000 training steps, the best *RL-1* and *RL-2* agents were selected for model testing later in a different environment (i.e., the *HighD* testing data).

To quantify the effects of stop-and-go waves on traffic flow considering HD, ACC, *RL-1* and *RL-2*, the average rolling speed standard deviations for vehicles at different positions in the training data are calculated and presented in Table 1. Since the 2nd vehicle can be controlled by either HD or ACC, and the 3rd vehicle can be HD, ACC, *RL-1* and *RL-2*, this results in a total of 8 different scenarios as in Table 1.

$\bar{s}^T$ measures a vehicle's speed oscillation throughout the simulation (more than 2 h). Table 1 suggests that when the $\bar{s}^T$ of the lead vehicle is relatively small (about 0.2), HD clearly amplifies the oscillation of its lead vehicle regardless of the lead vehicle type and the position in the platoon. When the lead vehicle's $\bar{s}^T$ is larger (e.g., 0.33 in Table 1), its following HD tends to dampen the oscillation. By contrast, ACC can digest the oscillation and consistently has negative $\bar{s}^T$ change percentage, especially when its leader is an HD with



(a) *the 2nd car is an HD*



(b) *the 2nd car is an ACC*

Fig. 5. Illustration of HD, ACC, RL-1 and RL-2 Performance on HighD Testing Data.

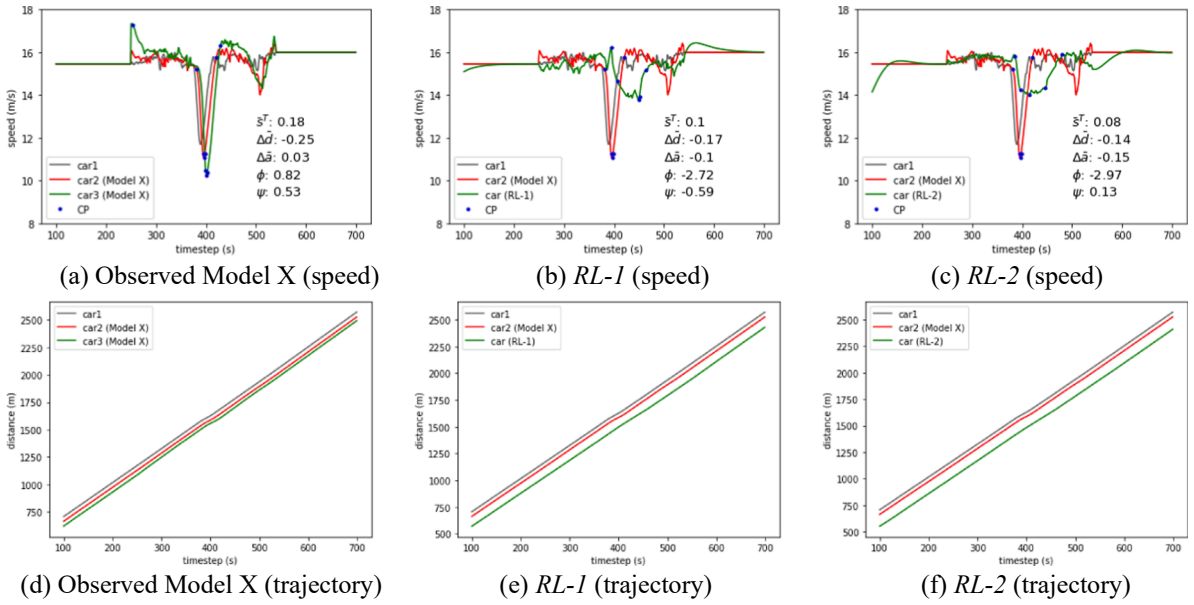


Fig. 6. Pattern 1: dip with headway-1.

high $\bar{s}^T$. In addition, *RL-1* and *RL-2* can even better dampen the oscillation created by its lead vehicle and generate a much smaller $\bar{s}^T$. In the scenario that the 2nd vehicle is an HD, *RL-2* performs clearly better than *RL-1* by producing a smaller $\bar{s}^T$.

Given the above comparison results, a valid question is that will RL control increase the overall vehicle travel time or not. For example, the AV could travel at a low and constant speed. Although this can lead to a very smooth trajectory, it will take the AV a much longer time to travel the same distance than an HD vehicle with a stop-and-go trajectory. Based on the simulation results in this study (see the Platoon Analysis section later), the RL-controlled AVs are able to not only absorb the stop-and-go shockwaves created by the lead vehicle, but also travel at about the same average journey speed as HD and ACC vehicles. Note that all the HD, ACC and RL-controlled vehicle speeds are constrained by the lead vehicle. Overall, RL-controlled AVs tend to keep a longer distance (about 60 m on average with a standard deviation of 30 m) to their lead vehicles than HD and ACC (about 30 m). Although this distance is not explicitly defined by the reward function in our design, the RL agents figure it out through training to avoid harsh brakes and/or frequent accelerations/decelerations.

To test how the RL agents perform in a new environment, they are evaluated on the *HighD* testing data and the results are given in Table 2. The results in Table 1 and Table 2 are highly consistent, suggesting that the performances of *RL-1* and *RL-2* are stable and they generalize well to a slightly different environment (*HighD* training and testing datasets are different, but are from the same *HighD* source).

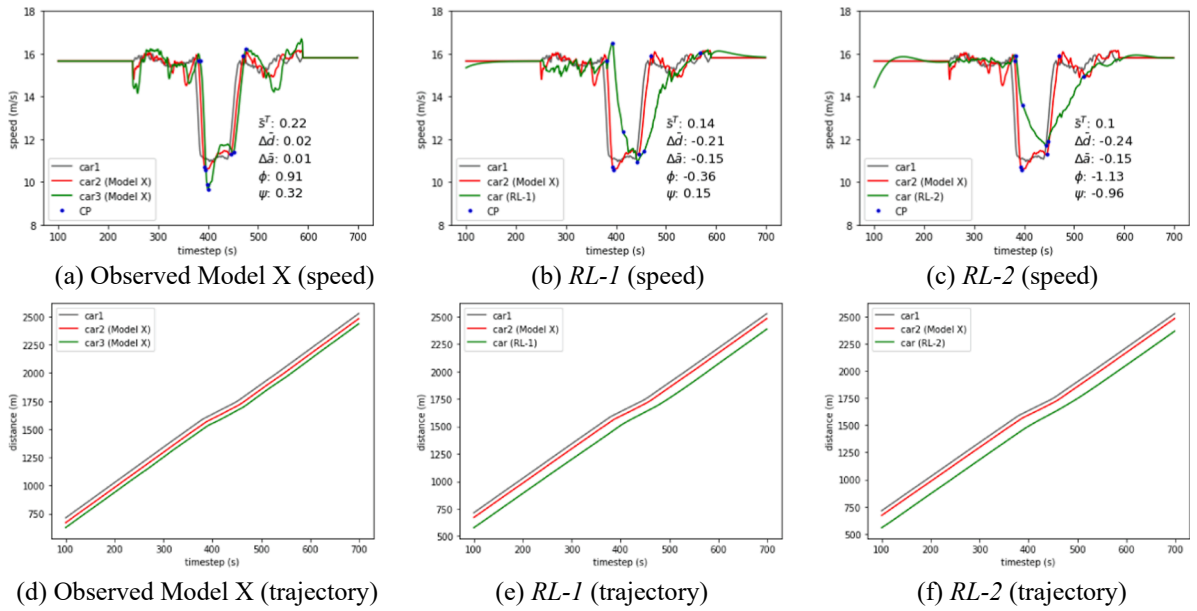
To further compare HD, ACC, *RL-1*, and *RL-2*, their trajectory, speed, and acceleration profiles for two 150-second intervals are plotted in Fig. 5, in which the 1st car's trajectory is from the *HighD* testing dataset, the 2nd car follows HD behavior, and the 3rd car is controlled by four different methods (i.e., HD, ACC, *RL-1*, and *RL-2*). Fig. 5(a) illustrates how different vehicles react to the pattern of the 1st vehicle (*HighD testing*) when the 2nd vehicle is an HD. Fig. 5(b) is for a slightly different scenario where the 2nd vehicle is an ACC.

Fig. 5(a) shows that HD as the 3rd car (in red) tends to slightly amplify the speed oscillations from its preceding HD vehicle and generates many local speed variations; ACC is better than HD as it seems to maintain the magnitude of oscillation of its leader and its speed profile is smooth; and *RL-1* and *RL-2* are much better than HD and ACC at dampening oscillations. Because they generate a larger minimum speed during the oscillation (from 700 s to 760 s) so that the traffic shockwave is less likely to be amplified by their followers. *RL-2* behaves better than *RL-1* because 1) it generates a larger minimum speed during the oscillation; and 2) it leads to smoother speed profiles and does not overly react to the 2nd car's local behavior. Another interesting phenomenon is that compared to HD and ACC, RL agents keep larger gaps with their preceding vehicles, especially when the preceding vehicles are speeding up. Such gaps are used as buffer zones for RL agents to decelerate more gradually when the preceding vehicles brake abruptly later (i.e., dampen the shockwave). The difference between the two RL control is that *RL-2* closes the gap sooner when its leader is in low-speed stabilization phase (740 s to 800 s). It seems additional state variables acquired by RL-2 enable it to behave more intelligently.

When the 2nd vehicle is an ACC, similar findings can be drawn from Fig. 5(b). ACC as the 2nd vehicle tends to generate many high-frequency local acceleration oscillations, which affect the acceleration behavior of *RL-1* and *RL-2*. Both *RL-1* and *RL-2* also generate frequent local acceleration oscillations by following the 2nd ACC vehicle, although the magnitude of acceleration oscillations for *RL-2* is smaller than that for *RL-1*. A more detailed analysis of this subject is provided in the next section based on the Model X experimental data.

Table 3
Explanation of Five Variables Critical for Shockwave Dampening.

Performance Metric	Meaning	Theoretic range	Desired value for dampening shockwaves	Applicability
$\Delta\bar{a}$	Average acceleration rate change	$[-\infty, +\infty]$	The smaller, the better	One driving cycle/oscillation
$\Delta\bar{d}$	Average deceleration rate change	$[-\infty, +\infty]$		
ϕ	Oscillation growth	$[-\infty, +\infty]$		
ψ	Overshooting	$[-\infty, +\infty]$		
$\bar{s}^T$	Average rolling Std.	$[0, +\infty]$	Close to zero	One/multiple driving cycles



(a) Observed Model X (speed) (b) *RL-1* (speed) (c) *RL-2* (speed)

(d) Observed Model X (trajectory) (e) *RL-1* (trajectory) (f) *RL-2* (trajectory)

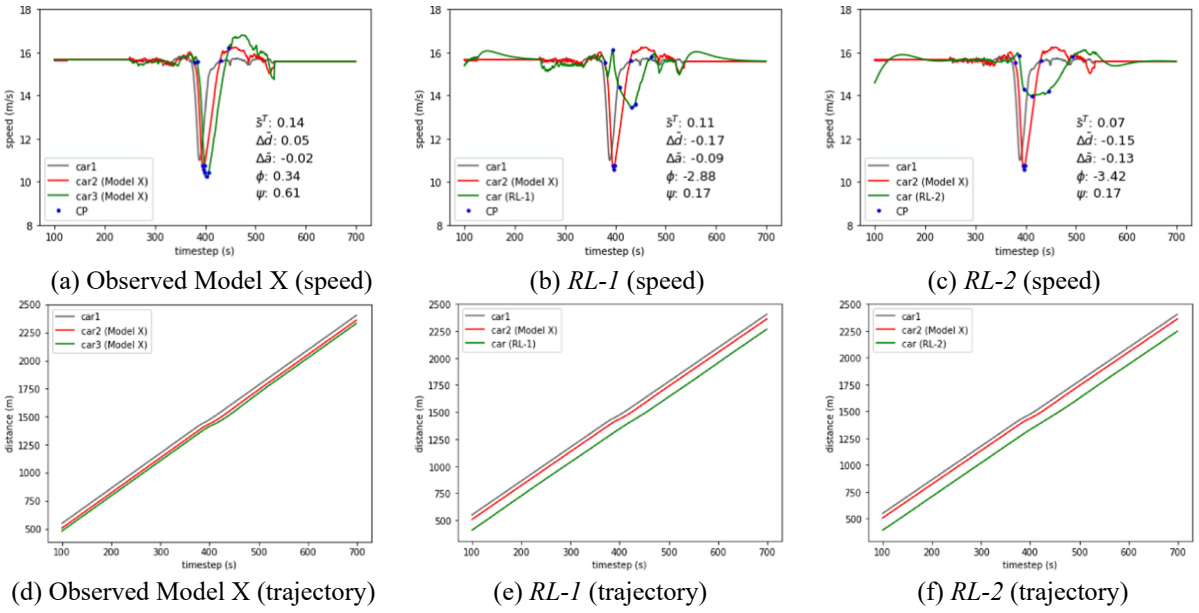
Fig. 7. Pattern 2: long-cruise with headway-1.

4.2. Results based on Model X Data

The proposed RL agents are further evaluated based on experimental data collected using a platoon of three commercial vehicles (Li et al., 2021). In this experiment, the 1st vehicle was an HD followed by two Model X vehicles with ACC capability. The HD vehicle's trajectory was carefully designed to follow certain patterns. Each pattern contained a major oscillation and some negligible small oscillations due to imprecise human control. Four patterns in the Model X dataset are considered in this study. When applying the RL control, the 3rd car in the Model X experiment is replaced by our RL agents (*RL-1* or *RL-2*). The application results of *RL-1* and *RL-2* are described below in Fig. 6 through Fig. 9. In each of these four figures, the values of five performance metrics related to dampening shockwaves for the 3rd vehicle are also provided. The definitions of these performance metrics and the implications of their values are given in Table 3.

The first pattern is characterized by a major dip in speed of the 1st HD vehicle and a small headway for all three vehicles (referred to as headway-1) as shown in Fig. 6. Fig. 6(a) and 6(d) show the field data collected during the Model X experiment. The remaining four subfigures in Fig. 6 illustrate the results of *RL-1* and *RL-2*. CP in Fig. 6 means critical point, which could be deceleration start time, deceleration end time, etc. The beginnings (around 100 s) of the speed profiles for *RL-1* (Fig. 6(b)) and *RL-2* (Fig. 6(c)) are very different from those of the 1st and 2nd vehicles. This is because when RL agents are initially inserted into the SUMO simulation network, they are placed in the position of the 3rd Model X vehicle, which is much closer to its leader (i.e., the 2nd Model X vehicle) than an RL agent would normally do. Therefore, these RL agents all start with a deceleration phase to increase spacing followed by an acceleration phase to catch up their leaders. The three speed profiles and the values of five performance metrics ($\bar{s}^T$, $\Delta\bar{a}$, $\Delta\bar{d}$, ϕ, and ψ) all suggest that both RL agents are better than the 3rd Model X vehicle in dampening traffic oscillation. However, the cost is that RL agents need larger gaps (see Fig. 6(d), 6(e) and 6(f)) to absorb shockwaves. Compared to *RL-1*, *RL-2* results in smaller $\Delta\bar{a}$, ϕ, but larger $\Delta\bar{d}$, ψ.

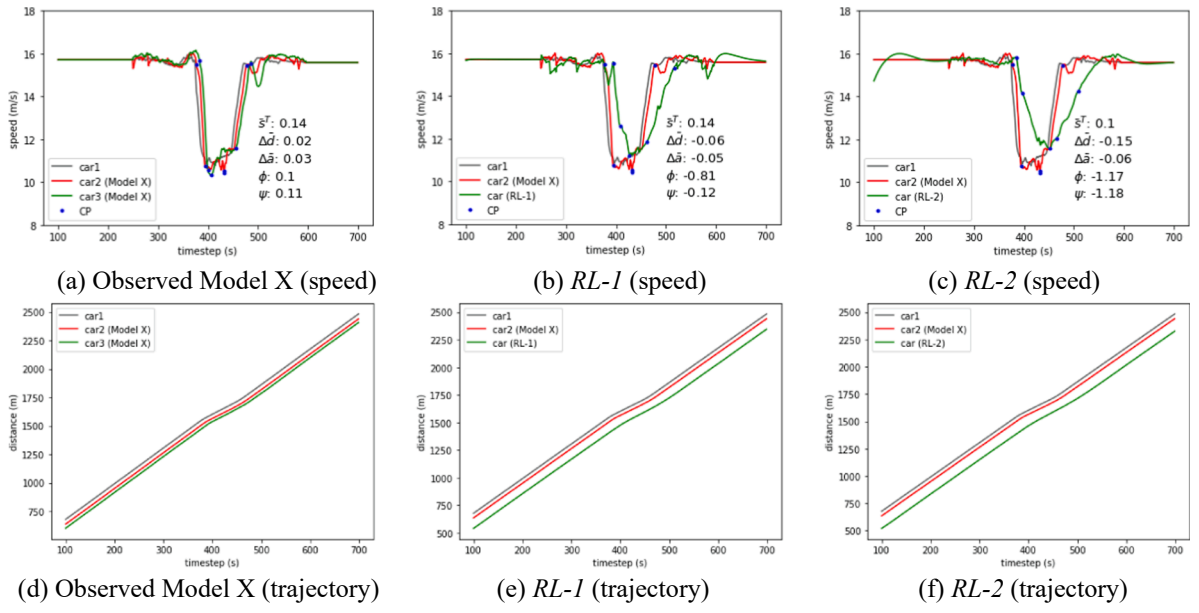
Fig. 7 shows the results for Pattern 2 with a long low-speed cruise period and small headways. The main difference between Patterns 1 and 2 is that Pattern 2 replaces the dip with an extended low-speed cruise period. All three oscillation growths (ϕ) for the 3rd vehicle in Pattern 2 are greater than those in Pattern 1, suggesting that the benefits of Model X ACC, *RL-1*, and *RL-2* are more significant for Pattern 1. The overall findings from Fig. 7 are consistent with those in Fig. 6 except that now *RL-2* is clearly better than *RL-1* based on the five performance metrics.



(a) Observed Model X (speed) (b) *RL-1* (speed) (c) *RL-2* (speed)

(d) Observed Model X (trajectory) (e) *RL-1* (trajectory) (f) *RL-2* (trajectory)

Fig. 8. Pattern 3: dip with headway-3.



(a) Observed Model X (speed) (b) *RL-1* (speed) (c) *RL-2* (speed)

(d) Observed Model X (trajectory) (e) *RL-1* (trajectory) (f) *RL-2* (trajectory)

Fig. 9. Pattern 4: long-cruise with headway-3.

Pattern 3 in Fig. 8 is similar to Pattern 1 except that a longer headway (referred to as headway-3) is considered. Similarly, Pattern 4 in Fig. 9 is based on Pattern 2 by considering headway-3 instead of headway-1. With a larger headway, Model X ACC now introduces less oscillations to the platoon (see Fig. 8(a) and Fig. 9(a)). However, its oscillation growth (ϕ), and overshooting (ψ) are still positive and cannot be ignored, which may yield unsafe consequences as the oscillation is further amplified by following vehicles (Chen et al., 2012). For RL agents (both *RL-1* and *RL-2*), they can clearly dampen the oscillations and generate smaller average acceleration/deceleration rate ($\Delta\bar{a}$, $\Delta\bar{d}$) and larger minimum speed than its lead vehicle (i.e., the 2nd Model X vehicle).

$\bar{s}^T$ is a metric proposed in this study to measure local speed variation at the global scale. It is proved to be effective in capturing traffic oscillations and is generally consistent with the outcomes of $\Delta\bar{a}$, $\Delta\bar{d}$, ϕ, and ψ. For example, in Fig. 7 $\Delta\bar{a}$, $\Delta\bar{d}$, ϕ, and ψ of *RL-1* are all smaller than those of Model X, which indicates *RL-1* is better than Model X ACC for dampening oscillations under this specific traffic condition. In the meantime, $\bar{s}^T$ of *RL-1* (0.14) is also smaller than that of Model X ACC (0.22). The advantage of $\bar{s}^T$ is that it can be easily

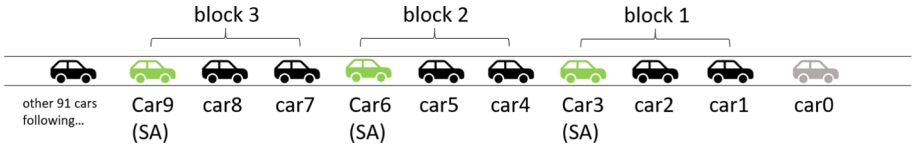


Fig. 10. A Platoon with n = 2 (SA's PR = 33.3%).

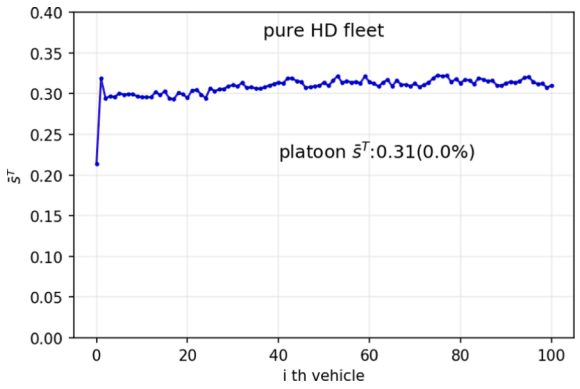


Fig. 11. $\bar{s}^T$ of vehicles at different locations in a pure HD Platoon.

calculated and serves as a global indicator especially when many oscillations exist.

4.3. Platoon analysis

In this section, the RL agents are tested in a long platoon consisting of 101 vehicles (from the 0th to the 100th vehicle). The speed profile sampled from the *HighD* testing data is adopted for the 0th vehicle. For every n vehicles starting from the 1st vehicle, a "specialized" agent (SA) is inserted right behind it. This SA could be controlled by *RL-1*, *RL-2*, HD, or ACC. However, in one simulation run all SAs are of the same type. Fig. 10 illustrates the concept of SA with $n = 2$, which means the Penetration Rate (PR) of SA is 33.3%. For vehicles other than the SAs in the network, they all follow either the HD (i.e., Wiedemann 99) or ACC model, generating two types of platoons: HD Platoon and ACC Platoon. The comparison of *RL-1*, *RL-2*, HD, and ACC is also divided into two groups below based on the platoon type.

4.3.1. HD platoon

Fig. 11 shows the $\bar{s}^T$ results for a platoon of 100 HD vehicles following a lead HD vehicle. The lead HD vehicle's trajectory is sampled from the *HighD* testing dataset. Each data point in Fig. 11 represents the $\bar{s}^T$ of a vehicle at a specific location in the platoon (i.e., the i^{th} vehicle). It can be seen that traffic oscillation gets significantly amplified at the 1st car and becomes gently worsened in the upstream direction afterwards.

To test the effects of SA, three main scenarios are considered. In each scenario, different SA penetration rates (i.e., different n values) are considered to replace some vehicles in the above HD platoon with 1) ACC, 2) *RL-1*, or 3) *RL-2*. These hybrid platoons are simulated using SUMO and the results are given in Fig. 12, in which the first column is for SA controlled by ACC, and the second and third columns are for *RL-1* and *RL-2*, respectively. Fig. 12 has seven rows and each of them is for the results of a different n value (taking 1, 2, 4, 9, 13, 19, and 32). In each subfigure, the light blue line shows each vehicle's $\bar{s}^T$ and its position in the 101-vehicle platoon; the green line connects the $\bar{s}^T$ of each SA; and the red line consists of the $\bar{s}^T$ of each SA's immediate lead vehicle. The average $\bar{s}^T$ of all 101 vehicles (i.e., platoon $\bar{s}^T$) is also calculated and shown in each subfigure. The percentage change next to each platoon $\bar{s}^T$ indicates its oscillation dampening effects compared to the 100% HD platoon (with a platoon $\bar{s}^T$ of 0.31).

Fig. 12 shows that ACC SA could dampen the overall oscillation compared to the pure HD Platoon (platoon $\bar{s}^T$ % change ranges from 0% to −16.1%). When considering *RL-1* as SA, it can bring the lower bound of $\bar{s}^T$ down to 0.2–0.25 (compared to 0.31 for the 100% HD platoon) depending on the value of *n*. *RL-2* SA further improves the performance of *RL-1* by reducing the lower bound of $\bar{s}^T$ to 0.1–0.2. From the perspective of the entire platoon, *RL-1* and *RL-2* both have lower platoon $\bar{s}^T$ than ACC under various scenarios. More importantly, the red and green trend lines of $\bar{s}^T$ for *RL-1* and *RL-2* do not amplify (i.e., going up) along the upstream direction as in the ACC SA case. The two trend lines remain stable or slightly decrease along upstream direction regardless of a vehicle's position in the platoon. Such a difference between RL agents and ACC is critical, which suggests ACC SA cannot fully digest the oscillation and will pass it on to upstream vehicles (vehicles behind them), while RL agents can effectively stop the propagation of oscillation.

To better illustrate the impacts of various types of SA on the whole HD platoon, trajectories of four 101-vehicle platoons with

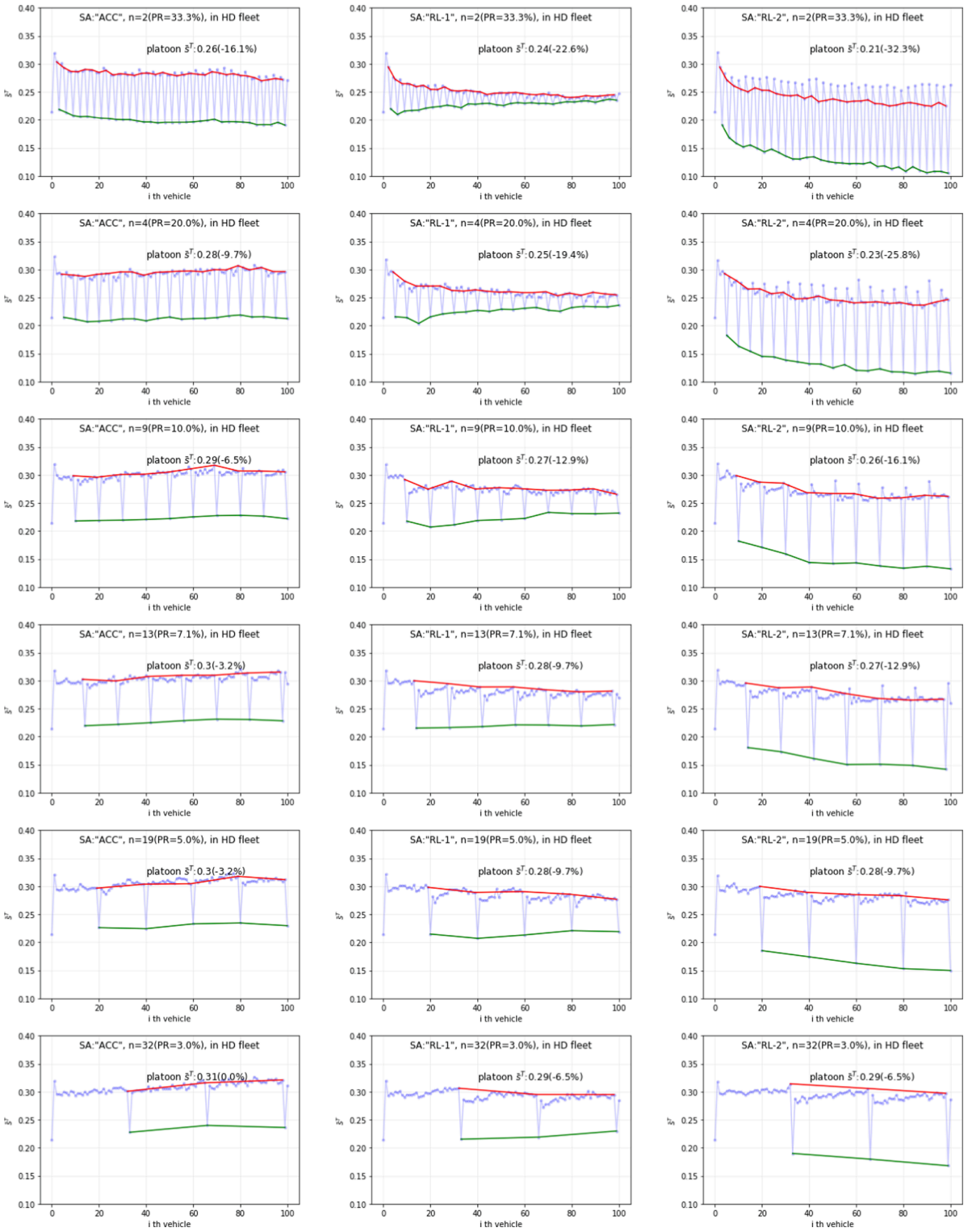


Fig. 12. Impacts of Different Types of SA on HD Platoon Stability.

different SA are plotted in Fig. 13. When HD and ACC are adopted as SA as in Fig. 13(a) and Fig. 13(b), respectively, low-speed strips clearly emerge, indicating vehicles experience stop-and-go shockwaves. While for *RL-1* (Fig. 13(c)) and *RL-2* Fig. 13(d), first there are fewer low-speed strips, which means vehicles in the platoon now experience less shockwaves. Second, there are less variations in the

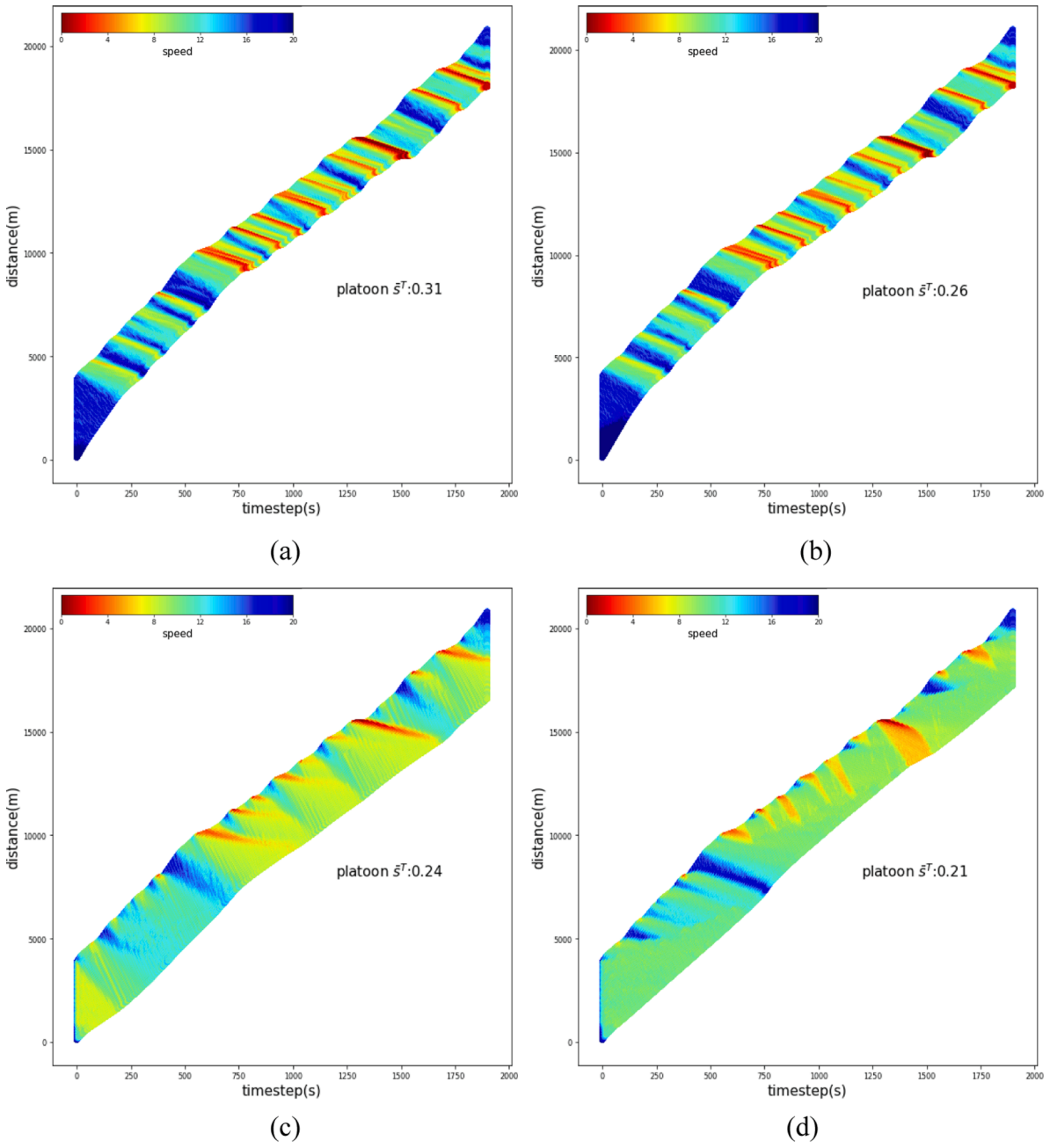


Fig. 13. Trajectories of 101 vehicles in HD Platoon. (a):pure HD, (b):ACC as SA, (c):RL-1 as SA, (d):RL-2 as SA. n = 2.

platoon color and extreme colors (i.e., dark blue and dark red) are "smoothed out", suggesting that vehicles trajectories are smoother and vehicles make abrupt accelerations/decelerations less frequently. A close look at Fig. 13(c) and Fig. 13(d) reveals that *RL-2* absorbs shockwaves quicker than *RL-1*, since the red strips for the RL-2 case are shorter in the dimension of x-axis (time). *RL-2* also absorbs shockwaves more efficiently than *RL-1*, because in many cases when shockwaves (red areas in Fig. 13(c) and Fig. 13(d)) propagate upstream, *RL-2* can completely digest them in the middle of the platoon, and a good portion of the vehicles at the end of the platoon only experience one slowdown (see Fig. 13(d)) during the sample segment.

The results so far all suggest *RL-1* and *RL-2* can effectively dampen the stop-and-go traffic. However, this might come at a cost, e.g., introducing extra delay to the traffic flow and/or reducing road capacity. The trade-offs among dampening shockwaves, road capacity and average delay for HD platoon is plotted in Fig. 14. Capacity is denoted by $flow^*$, which measures the maximum flow of the platoon when the 1st vehicle travels at a constant speed 15 m/s (not sampled from the *HighD testing* speed profile). Average delay is denoted by $delay^*$, which is calculated as the average travel time of all 101 vehicles crossing a 10 km segment (the lead vehicle speed profile is

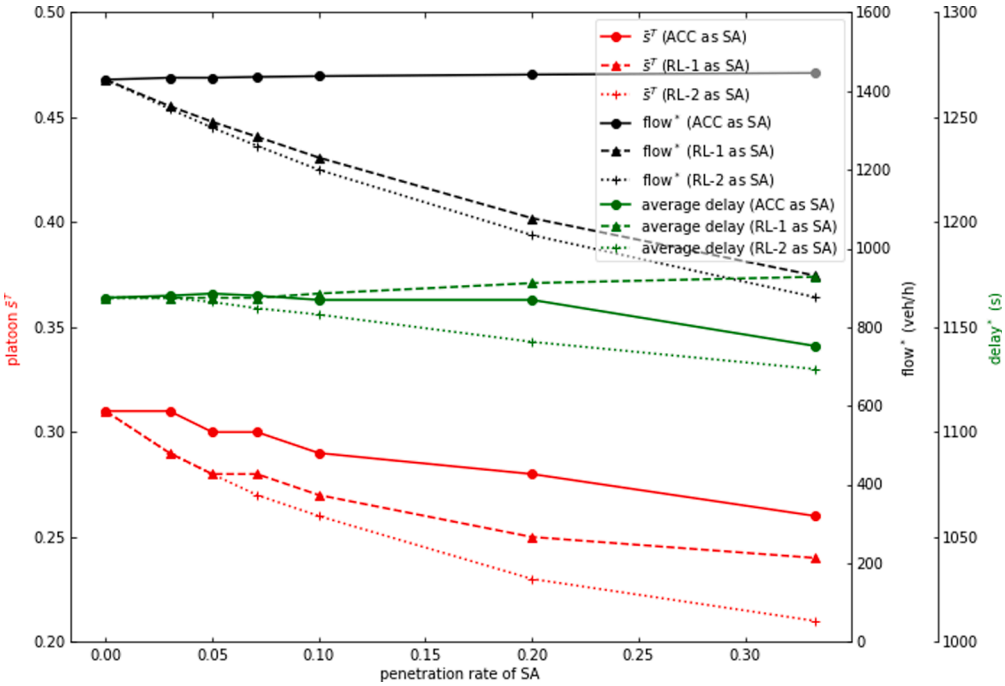


Fig. 14. Trade-offs among Oscillation Dampening Effects $\bar{s}^T$, Capacity (*flow**) and average delay *delay** for different Types of SA on the Whole HD Platoon.

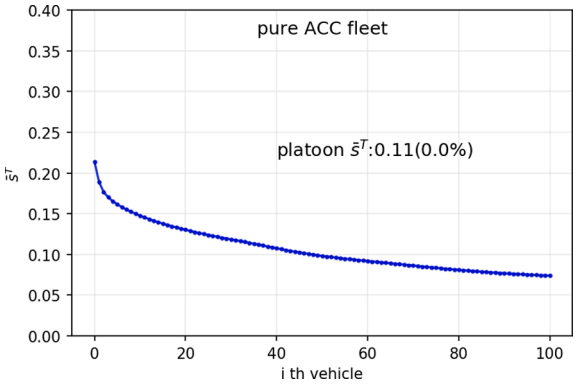


Fig. 15. $\bar{s}^T$ of vehicles at different locations in a pure ACC Platoon.

sampled from the *HighD* dataset). Fig. 14 suggests that as the penetration rates of *RL-1* and *RL-2* go up, the shockwave dampening benefits become more significant. However, the road capacity (when the lead vehicle travels at a constant speed) drops under both *RL-1* and *RL-2*. Additionally, *RL-1* slightly increases the average travel time, while *RL-2* is able to reduce it. This travel time finding is interesting but not surprising, which suggests that following a lead vehicle closely (such as what HD and ACC typically do) in stop-and-go traffic does not necessarily reduce a driver's travel time. The fact that *RL-2* reduces both travel time and capacity may seem contradictory. This is because travel time and capacity are measured under different conditions with stop-and-go traffic for travel time and steady flow for capacity. Since RL-controlled AV tend to keep a larger distance to the lead vehicle than HD and ACC vehicles and use that distance as a buffer zone, it is not surprising to see a reduced capacity. Note that here the theoretical capacity for HD/ACC vehicles is overestimated assuming that the lead vehicle travels at a constant speed. When the lead vehicle is affected by stop-and-go shockwaves, the actual capacity will certainly be less than the theoretical one, and the capacity gap between RL and HD/ACC cases will be smaller.

4.3.2. ACC platoon

ACC vehicles behave differently from HD vehicles. They may amplify or dampen traffic shockwaves depending on the speed profile of its leader and headway settings (Li et al., 2021). Similar to Fig. 11, Fig. 15 shows the $\bar{s}^T$ of vehicles at different locations in a 100%

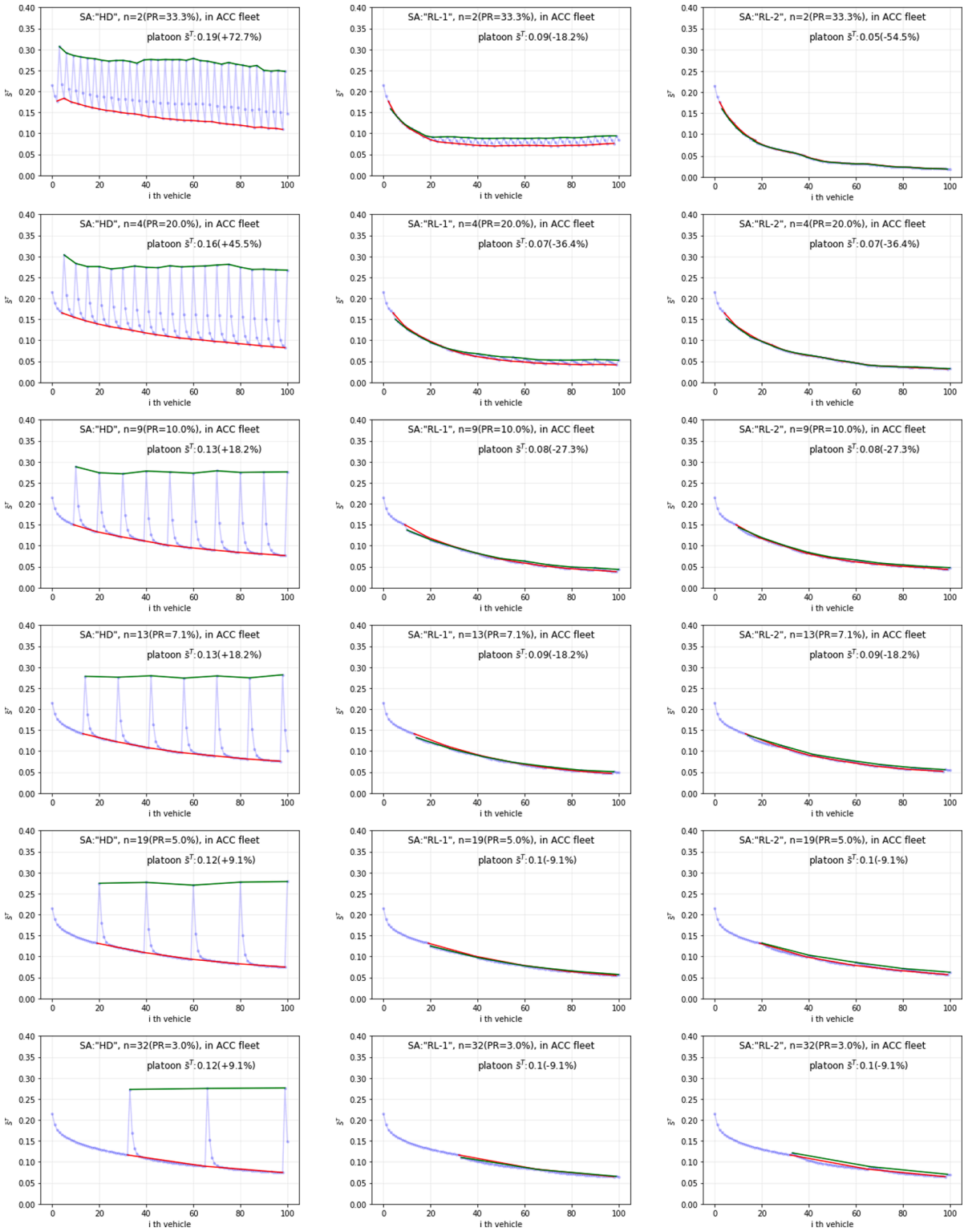


Fig. 16. Impacts of Different Types of SA on ACC Platoon Stability.

ACC platoon. A comparison of the two figures suggests that ACC platoon is more stable than HD platoon. Along the upstream direction, the $\bar{s}^T$ goes down quickly, indicating that this ACC model is capable of dampening shockwave. Based on the 100% ACC platoon, we also create three new scenarios as in the previous section (for the HD platoon). In each scenario, some ACC vehicles in the platoon are

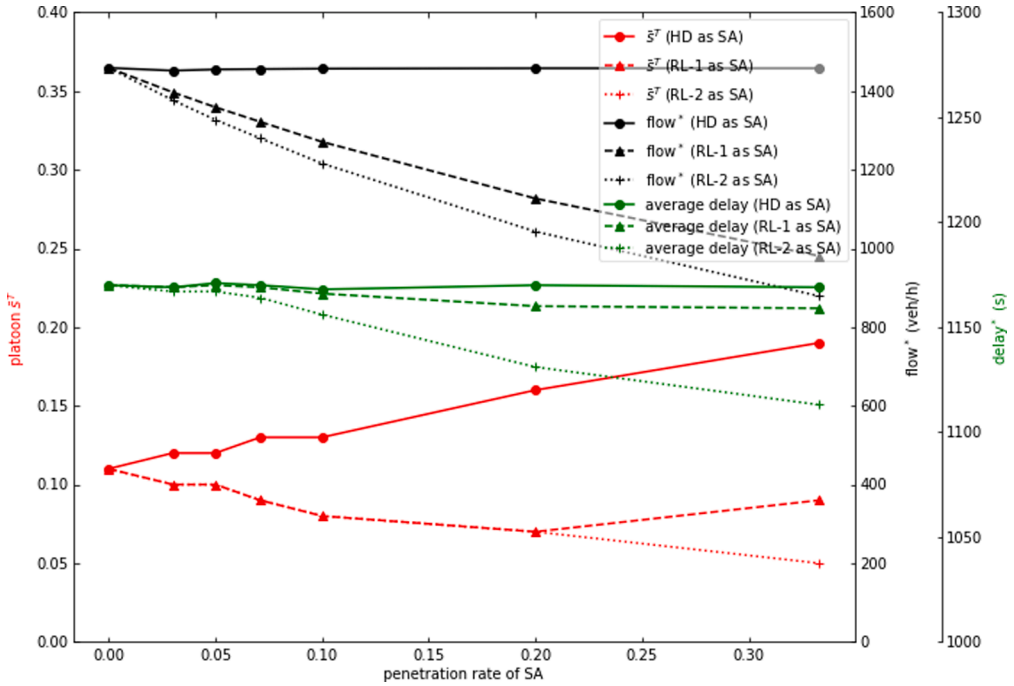


Fig. 17. Trade-offs among Oscillation Dampening Effects $\bar{s}^T$, Capacity (*flow**) and average delay for different Types of SA on the Whole HD Platoon.

replaced with 1) HD, 2) *RL-1*, or 3) *RL-2*. For all three scenarios, the 0th vehicle's trajectory is taken from the *HighD* testing dataset.

The first column in Fig. 16 shows the results for using HD as SA. Obviously, adding HD worsens the oscillation of the original ACC platoon. The green line connecting the oscillations of individual HD SA ($\bar{s}^T$) now sits above the red line (which is for the oscillations of the lead ACC vehicles of each HD SA). Also, the $\bar{s}^T$ of each hybrid ACC platoon with HD SA is larger than that of the 100% ACC platoon (0.11). Different from the HD SA, the *RL-1* and *RL-2* SAs consistently demonstrate improved performance in hybrid ACC platoons. With a 3% penetration rate, *RL-1* and *RL-2* could cut the platoon oscillation by 9.1% compared to the 100% ACC platoon. More dampening effects of *RL-1* and *RL-2* are found under scenarios with higher penetration rates.

Another interesting phenomenon is that *RL-1* and *RL-2* sometimes introduce additional minor oscillations when the oscillations of their leaders are relatively small (e.g., $\bar{s}^T$ less than 0.1). One example is when $n = 2$ for SA being *RL-1* in Fig. 16. In this case, RL agents at the beginning of the platoon are able to effectively reduce oscillations. However, when the $\bar{s}^T$ is below 0.1 (around the 19th vehicle in the platoon), *RL-1* increases $\bar{s}^T$ compared to ACC. There are two possible reasons for this. First, ACC performs very well in terms of dampening oscillation when its lead vehicle's oscillation is already low. Second, the RL agents are trained based on unstable traffic flow data, and additional data for stable traffic flow data may be needed to train them.

The trade-offs among dampening shockwaves, road capacity and average delay for ACC platoon are also summarized in Fig. 17. Overll the trends in Figs. 14 and 17 are similar. The platoon analysis suggests the following ranking of oscillation dampening capability: *RL-2* > *RL-1* > ACC ≫ HD. Although adding ACC to a 100% HD platoon can help to dampen oscillation, when penetration rate is relatively low, ACC SA cannot fully absorb the oscillation received and will pass part of that to vehicles behind it. On the other hand, *RL-1* and *RL-2* SAs can effectively digest the oscillation produced by their lead vehicles, and further reduce the $\bar{s}^T$ compared to the SA before them (see the downward trends exhibited by the green lines in Fig. 12). Also, the analysis tradeoffs among capacity, delay and shockwave dampening effects illustrates that the benefits of introducing RL-1 and RL-2 may come at a price of flow reduction.

5. Conclusions and discussion

This study shows that RL is a promising approach to dampen stop-and-go traffic, which can lead to significant safety and environmental benefits. Having information of one additional preceding vehicle (i.e., *RL-2*) can further improve the oscillation dampening effects of the *RL-1* agent. These benefits are not only for the ego CAV vehicle, but also for other vehicles following it (as demonstrated in the platoon analysis). Although the two RL agents are trained on the *HighD* data, they are able to generalize well to the Model X data and outperform the commercial Model X ACC under certain stop-and-go traffic conditions. Note that the RL agents are designed specifically for absorbing stop-and-go traffic. Therefore, they are not evaluated under other traffic conditions based on the Model X dataset.

In this study, cooperation is implicitly built into CAV's behavior without much debate to dampen stop-and-go traffic. Such behavior turns out to be beneficial to both the CAV and its followers. An interesting ethical question for future research is how cooperative or

courteous should CAV be. There might be a trade-off between the benefits to CAV and its followers, and this trade-off could depend on the level of cooperation of the CAV.

Also, we simplify the problem and assume that the vehicle sensors and control are perfect, so we can focus on addressing the main challenges, which are reducing traffic oscillations and improving platoon stability. In future research, we plan to take sensor measurement errors, communications delay, and control delay of AV into consideration. Additionally, this study takes information of the last time step as state representation, expanding the state space by considering more historical information (e.g., two or more time steps) is likely to further improve the performance of RL agents.

Funding

This work was supported by the NSF under Grants No. 1734521, No. 1932921, and No. 1826162.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Aghabayk, K., Sarvi, M., Young, W., Kautzsch, L., 2013. A novel methodology for evolutionary calibration of Vissim by multi-threading. Presented at the Australasian Transport Research Forum 1–15.

Chen, D., Laval, J., Zheng, Z., Ahn, S., 2012. A behavioral car-following model that captures traffic oscillations. Transportation Research Part B: Methodological 46 (6), 744–761. https://doi.org/10.1016/j.trb.2012.01.009.

Chu, T., Kalabić, U., 2019. Model-based deep reinforcement learning for CACC in mixed-autonomy vehicle platoon, in: 2019 IEEE 58th Conference on Decision and Control (CDC). Presented at the 2019 IEEE 58th Conference on Decision and Control (CDC), pp. 4079–4084. https://doi.org/10.1109/CDC40024.2019.9030110.

Desjardins, C., Chaib-draa, B., 2011. Cooperative Adaptive Cruise Control: A Reinforcement Learning Approach. IEEE Trans. Intell. Transport. Syst. 12 (4), 1248–1260. https://doi.org/10.1109/TITS.2011.2157145.

Ge, J.I., Orosz, G., 2014. Dynamics of connected vehicle systems with delayed acceleration feedback. Transportation Research Part C: Emerging Technologies 46, 46–64. https://doi.org/10.1016/j.trc.2014.04.014.

German Aerospace Center (DLR) and others, 2021. car-following model parameters.

Haarnoja, T., Zhou, A., Abbeel, P., Levine, S., 2018a. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. arXiv: 1801.01290 [cs, stat].

Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., 2018b. Soft actor-critic algorithms and applications. arXiv preprint arXiv:1812.05905.

Khodayari, A., Ghaffari, A., Kazemi, R., Braunstingl, R., 2012. A Modified Car-Following Model Based on a Neural Network Model of the Human Driver Effects. IEEE Trans. Syst., Man Cybern. A 42 (6), 1440–1449. https://doi.org/10.1109/TSMCA.2012.2192262.

Krajewski, R., Bock, J., Kloeker, L., Eckstein, L., 2018. The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). Presented at the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE, Maui, HI, pp. 2118–2125. https://doi.org/10.1109/ITSC.2018.8569552.

Li, T., Chen, D., Zhou, H., Laval, J., Xie, Y., 2021. Car-following behavior characteristics of adaptive cruise control vehicles based on empirical experiments. Transportation Research Part B: Methodological 147, 67–91. https://doi.org/10.1016/j.trb.2021.03.003.

Li, X., Cui, J., An, S., Parsafard, M., 2014. Stop-and-go traffic analysis: Theoretical properties, environmental impacts and oscillation mitigation. Transportation Research Part B: Methodological 70, 319–339. https://doi.org/10.1016/j.trb.2014.09.014.

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D., 2019. Continuous control with deep reinforcement learning. arXiv: 1509.02971 [cs, stat].

Morton, J., Wheeler, T.A., Kochenderfer, M.J., 2017. Analysis of Recurrent Neural Networks for Probabilistic Modeling of Driver Behavior. IEEE Trans. Intell. Transport. Syst. 18 (5), 1289–1298. https://doi.org/10.1109/TITS.2016.2603007.

Qu, X., Yu, Y., Zhou, M., Lin, C.-T., Wang, X., 2020. Jointly dampening traffic oscillations and improving energy consumption with electric, connected and automated vehicles: A reinforcement learning based approach. Appl. Energy 257, 114030. https://doi.org/10.1016/j.apenergy.2019.114030.

Ren, T., Xie, Y., Jiang, L., 2021. New England merge: a novel cooperative merge control method for improving highway work zone mobility and safety. Journal of Intelligent Transportation Systems 25 (1), 107–121. https://doi.org/10.1080/15472450.2020.1822747.

Ren, T., Xie, Y., Jiang, L., 2020b. Cooperative Highway Work Zone Merge Control Based on Reinforcement Learning in a Connected and Automated Environment. Transp. Res. Rec. 2674 (10), 363–374.

Schaul, T., Quan, J., Antonoglou, I., Silver, D., 2016. Prioritized Experience Replay. Presented at the ICLR (Poster).

Stern, R.E., Cui, S., Delle Monache, M.L., Bhadani, R., Bunting, M., Churchill, M., Hamilton, N., Haulcy, R., Pohlmann, H., Wu, F., Piccoli, B., Seibold, B., Sprinkle, J., Work, D.B., 2018. Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments. Transportation Research Part C: Emerging Technologies 89, 205–221. https://doi.org/10.1016/j.trc.2018.02.005.

Sugiyama, Y., Fukui, M., Kikuchi, M., Hasebe, K., Nakayama, A., Nishinari, K., Tadaki, S.-I., Yukawa, S., 2008. Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam. New J. Phys. 10 (3), 033001. https://doi.org/10.1088/1367-2630/10/3/033001.

Vinitsky, E., Kreidieh, A., Le Flem, L., Kheterpal, N., Jang, K., Wu, C., Wu, F., Liaw, R., Liang, E., Bayen, A.M., 2018. Benchmarks for reinforcement learning in mixed-autonomy traffic. Conference on Robot Learning. PMLR 399–409.

Wang, P., Chan, C.-Y., 2017. Formulation of deep reinforcement learning architecture toward autonomous driving for on-ramp merge, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). Presented at the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), IEEE, Yokohama, pp. 1–6. https://doi.org/10.1109/ITSC.2017.8317735.

Wu, C., Bayen, A.M., Mehta, A., 2018. Stabilizing Traffic with Autonomous Vehicles, in: 2018 IEEE International Conference on Robotics and Automation (ICRA). Presented at the 2018 IEEE International Conference on Robotics and Automation (ICRA), IEEE, Brisbane, QLD, pp. 1–7. https://doi.org/10.1109/ICRA.2018.8460567.

Wu, Cathy, 2018. Learning and Optimization for Mixed Autonomy Systems-A Mobility Context.

Xiao, L., Wang, M., van Arem, B., 2017. Realistic Car-Following Models for Microscopic Simulation of Adaptive and Cooperative Adaptive Cruise Control Vehicles. Transp. Res. Rec. 2623 (1), 1–9. https://doi.org/10.3141/2623-01.

Zheng, Z., Ahn, S., Chen, D., Laval, J., 2011. Applications of wavelet transform for analysis of freeway traffic: Bottlenecks, transient traffic, and traffic oscillations. Transportation Research Part B: Methodological 45 (2), 372–384.

Zhu, M., Wang, Y., Pu, Z., Hu, J., Wang, X., Ke, R., 2019. Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving. Transportation Research Part C: Emerging Technologies 117, 102662. https://doi.org/10.1016/j.trc.2020.102662.