Adaptive Testing Environment Generation for Connected and Automated Vehicles With Dense Reinforcement Learning

Jingxuan Yang, Ruoxuan Bai, Haoyuan Ji, Yi Zhang, *Senior Member, IEEE*, Jianming Hu, *Senior Member, IEEE*, and Shuo Feng, *Member, IEEE*

***Abstract*— The assessment of safety performance plays a pivotal role in the development and deployment of connected and automated vehicles (CAVs). A common approach involves designing testing scenarios based on prior knowledge of CAVs (e.g., surrogate models), conducting tests in these scenarios, and subsequently evaluating CAVs' safety performances. However, substantial differences between CAVs and the prior knowledge can significantly diminish the evaluation efficiency. In response to this issue, existing studies predominantly concentrate on the adaptive design of testing scenarios during the CAV testing process. Yet, these methods have limitations in their applicability to high-dimensional scenarios. To overcome this challenge, we develop an adaptive testing environment that bolsters evaluation robustness by incorporating multiple surrogate models and optimizing the combination coefficients of these surrogate models to enhance evaluation efficiency. We formulate the optimization problem as a regression task utilizing quadratic programming. To efficiently obtain the regression target via reinforcement learning, we propose the dense reinforcement learning method and devise a new adaptive policy with high sample efficiency. Essentially, our approach centers on learning the values of critical scenes displaying substantial surrogate-to-real gaps. The effectiveness of our method is validated in high-dimensional overtaking scenarios, demonstrating that our approach achieves notable evaluation efficiency.**

***Index Terms*— Adaptive testing environment generation, connected and automated vehicles, dense reinforcement learning.**

I. Introduction

TESTING and evaluating the safety performance of connected and automated vehicles presents notable challenges in their development and deployment. One suggested approach involves testing CAVs in the naturalistic driving environment (NDE), observing their behaviors, and statistically comparing the testing results with human drivers. However, the scarcity of safety-critical events in NDE necessitates an impractical amount of testing miles–sometimes in the hundreds of millions or even billions–to demonstrate CAVs' safety performance at the human-level, rendering the evaluation process intolerably inefficient [1]. To increase evaluation efficiency, recent years have seen rapid advancements in the field of testing scenario library generation [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26]. This involves deliberately generating safety-critical testing scenarios using prior knowledge of CAVs, such as surrogate models (SMs). Employing SMs is promising for significantly enhancing the evaluation efficiency [27], [28]. Nevertheless, due to the intricate nature and black-box characteristics of CAVs, substantial safety performance disparities exist between SMs and diverse CAVs, a phenomenon referred to as the surrogate-to-real gap, which is illustrated in Fig. 1. This mismatch could undermine the effectiveness of the generated testing scenario libraries, ultimately diminishing the evaluation efficiency for diverse CAVs.

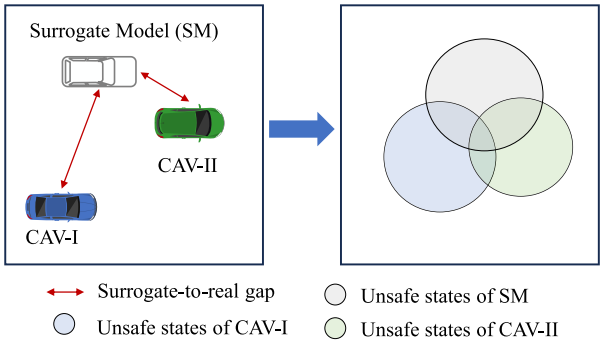


Fig. 1. Illustration of the surrogate-to-real gaps, i.e., the safety performance differences between SMs and various CAVs. The set of unsafe states indicates all traffic scenes (i.e., snapshots of traffic scenarios) in which CAVs or SMs may crash with background vehicles.

To tackle this issue, several adaptive testing methods have been proposed [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39]. The fundamental concept of these methods is to dynamically generate testing scenarios during the evaluation

Received 29 February 2024; revised 27 December 2024; accepted 24 January 2025. Date of publication 11 February 2025; date of current version 31 March 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62473224, in part by Beijing Natural Science Foundation under Grant 4244092, and in part by Beijing Nova Program under Grant 20230484259 and Grant 20240484642. The Associate Editor for this article was G. Orosz. *(Corresponding author: Shuo Feng.)*

Jingxuan Yang, Ruoxuan Bai, Haoyuan Ji, Jianming Hu, and Shuo Feng are with the Department of Automation, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China (e-mail: yangjx20@mails.tsinghua.edu.cn; brx22@mails.tsinghua.edu.cn; jihy21@mails.tsinghua.edu.cn; hujm@mail.tsinghua.edu.cn; fshuo@tsinghua.edu.cn).

Yi Zhang is with the Department of Automation, BNRist, Tsinghua University, Beijing 100084, China, and also with Jiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, Nanjing 210096, China (e-mail: zhyi@mail.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TITS.2025.3535866

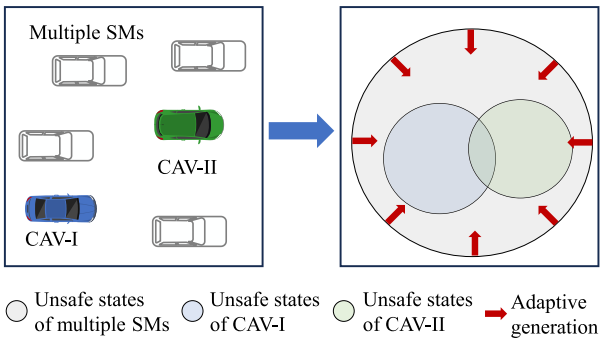


Fig. 2. Illustration of the adaptive testing environment generation method with multiple SMs.

process of CAVs. As more testing results accumulate, more posterior knowledge of CAVs is gained, enabling the customization of testing scenarios for the specific CAV under test. However, most existing methods often apply only to relatively simple scenarios, leaving the challenge of handling high-dimensional scenarios unsolved. The difficulty in adaptively generating high-dimensional scenarios stems from the compounded effects of the curse of rarity (CoR) and the curse of dimensionality (CoD) [40]. The CoR indicates that, due to the rarity of safety-critical events, the volume of data needed for sufficient information grows exponentially. The CoD pertains to the dimensionality of variables representing realistic scenarios, causing computational costs to escalate exponentially with the increase in scenario dimensions. Due to the CoR and CoD challenges, most existing scenario-based testing approaches are limited to short scenario segments with few background road users, involving low-dimensional decision variables that fail to capture the full complexity and variability of the real-world driving environment [27], [28], [41], [42], [43], [44]. Towards addressing this challenge, our previous work introduced the naturalistic and adversarial driving environment (NADE) method capable of generating high-dimensional highway driving scenarios [45]. However, the NADE overlooked the performance gap between diverse CAVs and the SM, potentially impeding the testing process.

To address this problem, we develop an adaptive testing environment (AdaTE) generation method that enhances evaluation robustness by employing multiple SMs, while optimizing the combination coefficients of these SMs to improve evaluation efficiency, as shown in Fig. 2. In NADE, if the set of unsafe states of the SM can not cover the set of unsafe states of the CAV under test, then the crash rate of this CAV might be severely underestimated. This is because NADE will rarely test crash scenarios containing unsafe states not covered by the SM. We will demonstrate such cases in Subsection V-B. Using multiple SMs can broaden the coverage of unsafe states, enabling AdaTE to test various CAVs unbaisedly. In the absence of any information about the particular CAV under test, using SMs with average combination coefficients might be the most suitable approach. However, this could reduce evaluation efficiency since the resulting NADE is not tailored to any specific CAV. To improve evaluation efficiency, we optimize the combination coefficients of SMs through adaptive testing, which is formulated as a regression task using quadratic programming (QP). However, efficiently obtaining the regression target through reinforcement learning (RL) is highly challenging, as the regression target represents the prediction of crash probabilities. This is primarily due to the CoR that critical information such as crash events is rare in NDE, resulting in extremely sparse rewards. To tackle this challenge, we introduce the dense reinforcement learning (DenseRL) method, which extends the dense deep reinforcement learning method proposed in [46] for deep reinforcement learning to the tabular setting. The DenseRL method, coupled with a newly designed adaptive policy, can efficiently learn the regression target. Essentially, our approach focuses on learning the values of critical state-action pairs exhibiting significant surrogate-to-real gaps. To validate our method, the high-dimensional overtaking scenarios are investigated. The results demonstrate that our approach achieves higher evaluation efficiency compared to both NDE and NADE.

The subsequent sections of this paper are structured as follows. Section II furnishes foundational knowledge for testing CAVs in NDE and NADE, and then formulates the problem of adaptive testing in high-dimensional scenarios as a regression problem that optimizes the combination coefficients of multiple SMs. Towards addressing this problem, the AdaTE is developed in Section III, where the DenseRL method is proposed to efficiently learn the regression target, and then the regression problem is solved using QP. The theoretical analysis for the convergence of DenseRL method is established in Section IV. To validate the effectiveness of the proposed method, Section V provides empirical results from testing CAVs in the high-dimensional overtaking scenarios. Finally, Section VI concludes the paper and discusses future research.

II. Problem Formulation

In this section, the preliminary knowledge for testing CAVs in NDE and NADE is provided in Subsection II-A and II-B, respectively. Then the adaptive testing problem will be formulated in Subsection II-C. The list of abbreviations is shown in Table I. Summary of notation is listed in Table II.

A. Naturalistic Driving Environment Testing

Let $\boldsymbol{x} := (\boldsymbol{s}_0, \boldsymbol{a}_0, \ldots, \boldsymbol{s}_{T-1}, \boldsymbol{a}_{T-1}, \boldsymbol{s}_T) \in \mathcal{X}$ denote the testing scenario, where $\boldsymbol{s}_t \in \mathcal{S}$ is the state of the CAV and background vehicles (BVs) at time t, $\boldsymbol{a}_t \in \mathcal{A}$ is the action of BVs at time t, T is the time horizon, and $\mathcal{X}$ is the set of all feasible scenarios. Consider the probability space $(\mathcal{X}, \mathcal{F}, \mathbb{P})$, where $\mathcal{F} := \mathcal{P}(\mathcal{X})$ is the σ-algebra, $\mathcal{P}(\mathcal{X}) := \{\mathcal{X}' : \mathcal{X}' \subseteq \mathcal{X}\}$ is the power set of $\mathcal{X}$, $\mathbb{P}(\{\boldsymbol{x}\}) := p(\boldsymbol{x}),\ \forall \boldsymbol{x} \in \mathcal{X}$ is the probability measure, and p is the naturalistic distribution of scenarios, which can be expressed as

$$p(\boldsymbol{x}) := \rho(\boldsymbol{s}_0) \prod_{t=0}^{T-1} \phi(\boldsymbol{a}_t|\boldsymbol{s}_t) P(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t, \boldsymbol{a}_t),\ \forall \boldsymbol{x} \in \mathcal{X}, \quad (1)$$

where ρ is the initial state distribution, ϕ is the naturalistic driving policy of BVs, and P is the state transition probability. Denote the crash event between the CAV and BVs as $F := \{\boldsymbol{x} \in \mathcal{X} : \boldsymbol{s}_T \in \mathcal{S}_{\text{crash}}\} \in \mathcal{F}$, where $\mathcal{S}_{\text{crash}}$ is the set of crash

TABLE I
LIST OF ABBREVIATIONS

Abbreviation	Definition
AAR	average acceleration ratio
AdaTE	adaptive testing environment
ASD	average sliding difference
AV	automated vehicle
BV	background vehicle
CAV	connected and automated vehicle
CoD	curse of dimensionality
CoR	curse of rarity
DenseRL	dense reinforcement learning
FVDM	full velocity difference model
IDM	intelligent driver model
IF	importance function
LV	leading vehicle
NDE	naturalistic driving environment
NADE	naturalistic and adversarial driving environment
PUCT	probabilistic upper confidence tree bound
QP	quadratic programming
RHW	relative half-width
RL	reinforcement learning
SM	surrogate model

states. Then the crash rate in NDE is given by $\mu := \mathbb{P}(F) = \mathbb{E}_p[\mathbb{I}_F(\boldsymbol{X})]$, where $\mathbb{I}_F$ is the indicator function of F, and $\boldsymbol{X} : \boldsymbol{x} \mapsto \boldsymbol{x}$, $\forall \boldsymbol{x} \in \mathcal{X}$ is the scenario random variable. According to Monte Carlo theory [47], the crash rate can be estimated in NDE as

$$\hat{\mu}_n := \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}_F(\boldsymbol{X}_i), \quad \boldsymbol{X}_i \sim p, \tag{2}$$

where n is the number of tests, and $\boldsymbol{X}_i$ are scenario random variables sampled i.i.d. from p.

B. Naturalistic and Adversarial Driving Environment Testing

The evaluation efficiency of NDE suffers severely from the CoR [40]. Using importance sampling method to replace the naturalistic distribution with the importance function (IF) is helpful to improve the evaluation efficiency [27], [28], [42], [43]. However, the importance sampling method can not be directly applied in high-dimensional scenarios because of the CoD [48]. Therefore, the NADE method has been proposed to only control critical variables at critical moments, while keeping other variables with their naturalistic distributions [45]. Specifically, the importance function is given by

$$q(\boldsymbol{x}) := \rho(\boldsymbol{s}_0) \prod_{t=0}^{T-1} \psi(\boldsymbol{a}_t|\boldsymbol{s}_t) P(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t, \boldsymbol{a}_t), \ \forall \boldsymbol{x} \in \mathcal{X}, \tag{3}$$

where ψ is the importance policy defined as

$$\psi(\boldsymbol{a}|\boldsymbol{s}) := \begin{cases} \phi(\boldsymbol{a}|\boldsymbol{s}), & \text{if } \boldsymbol{s} \notin \mathcal{S}_c, \\ \epsilon\phi(\boldsymbol{a}|\boldsymbol{s}) + (1-\epsilon)\dfrac{Q(\boldsymbol{s},\boldsymbol{a})\phi(\boldsymbol{a}|\boldsymbol{s})}{V(\boldsymbol{s})}, & \text{if } \boldsymbol{s} \in \mathcal{S}_c. \end{cases} \tag{4}$$

Here, $\mathcal{S}_c$ represents the set of safety-critical states, $\epsilon \in (0, 1)$ is a defensive parameter, $Q(\boldsymbol{s}, \boldsymbol{a}) \in [0, 1]$ is the maneuver challenge indicating the crash probability when BVs take action $\boldsymbol{a}$ in state $\boldsymbol{s}$, $V(\boldsymbol{s}) := \mathbb{E}_\phi[Q(\boldsymbol{s}, \boldsymbol{A})] \in [0, 1]$ is the criticality, and $\boldsymbol{A} : \boldsymbol{x} \mapsto \boldsymbol{a}$ for all $\boldsymbol{x} \in \mathcal{X}$ is the action random variable. According to importance sampling theory [47], the crash rate can be estimated in NADE as

$$\hat{\mu}_q := \frac{1}{n}\sum_{i=1}^{n} \frac{\mathbb{I}_F(\boldsymbol{X}_i)p(\boldsymbol{X}_i)}{q(\boldsymbol{X}_i)}, \ \boldsymbol{X}_i \sim q. \tag{5}$$

C. Adaptive Testing

Although the NADE has shown great potential for testing CAVs efficiently [27], [28], [42], [43], one crucial issue arises when testing diverse CAVs. The performance of NADE strongly relies on the selected importance function, which may not be suitable for various CAVs, leading to catastrophic failures (such as crash rate underestimation). Towards solving this issue, the goal of adaptive testing is to improve the robustness of NADE for diverse CAVs, while keeping the evaluation efficiency. One approach is to solve the optimization problem that minimizes the estimation variance of NADE over the function space $\mathcal{Q}$ that incorporates all possible importance functions, i.e.,

$$\min_{q\in\mathcal{Q}} \ \sigma_q^2 := \mathrm{Var}_q\left(\frac{\mathbb{I}_F(\boldsymbol{X})p(\boldsymbol{X})}{q(\boldsymbol{X})}\right). \tag{6}$$

By optimizing q in $\mathcal{Q}$, the importance function could be customized for the specific CAV under test, thus improving the evaluation efficiency for diverse CAVs.

Solving the optimization problem (6) is highly challenging, because the optimization space $\mathcal{Q}$ is a general function space. To address this issue, we propose to reduce the optimization space $\mathcal{Q}$ to the function space spanned by multiple importance functions, which can be formulated as $\mathcal{Q}_J := \{q_{\boldsymbol{\alpha}} \in \mathcal{Q} : \sum_{j=1}^{J} \alpha_j = 1, \ \alpha_j \geqslant 0\} \subset \mathcal{Q}$, where $q_{\boldsymbol{\alpha}}$ is the mixture importance function with mixture importance policy $\psi_{\boldsymbol{\alpha}} := \sum_{j=1}^{J} \alpha_j \psi_j$, ψ_j are importance policies, α_j are combination coefficients, $\boldsymbol{\alpha} := [\alpha_1, \ldots, \alpha_J]^\top$, and J is the number of importance functions. We note that these importance functions could be obtained from multiple SMs and other safety metrics [45]. With $\mathcal{Q}_J$ in place of $\mathcal{Q}$, the optimization problem (6) can be simplified as

$$\min_{q_{\boldsymbol{\alpha}}\in\mathcal{Q}_J} \ \sigma_{q_{\boldsymbol{\alpha}}}^2 := \mathrm{Var}_{q_{\boldsymbol{\alpha}}}\left(\frac{\mathbb{I}_F(\boldsymbol{X})p(\boldsymbol{X})}{q_{\boldsymbol{\alpha}}(\boldsymbol{X})}\right). \tag{7}$$

Then our goal is to optimize $q_{\boldsymbol{\alpha}}$ towards the optimal importance function q^*, which can be approximated by optimizing $\psi_{\boldsymbol{\alpha}}$ towards the optimal importance policy ψ^*. According to importance sampling theory [47], the optimal importance policy is given by $\psi^*(\boldsymbol{a}|\boldsymbol{s}) = Q^*(\boldsymbol{s}, \boldsymbol{a})\phi(\boldsymbol{a}|\boldsymbol{s})/V^*(\boldsymbol{s})$, where $Q^*(\boldsymbol{s}, \boldsymbol{a}) := \mathbb{P}(F|\boldsymbol{S} = \boldsymbol{s}, \boldsymbol{A} = \boldsymbol{a})$ is the optimal maneuver challenge that represents the crash probability given current state-action pair $(\boldsymbol{s}, \boldsymbol{a})$, $\boldsymbol{S} : \boldsymbol{x} \mapsto \boldsymbol{s}$, $\forall \boldsymbol{x} \in \mathcal{X}$ is the state random variable, and $V^*(\boldsymbol{s}) := \mathbb{E}_\phi[Q^*(\boldsymbol{s}, \boldsymbol{A})]$ is the optimal criticality. Then the optimization problem (7) can be simplified as a regression task via QP, i.e.,

$$\begin{aligned} \min_{\boldsymbol{\alpha}\in\mathbb{R}^J} \ & \frac{1}{2}\sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} \left[Q^*(\boldsymbol{s}, \boldsymbol{a}) - Q_{\boldsymbol{\alpha}}(\boldsymbol{s}, \boldsymbol{a})\right]^2 \\ \text{s.t.} \ & \mathbf{1}^\top\boldsymbol{\alpha} = 1, \ \boldsymbol{\alpha} \geqslant \mathbf{0}, \end{aligned} \tag{8}$$

TABLE II
SUMMARY OF NOTATION

Notation	Definition	Notation	Definition	Notation	Definition
$\boldsymbol{a}, \boldsymbol{a}_t$	action, action at time t	$\mathcal{Q}$	function space of all IFs	$\boldsymbol{x}, \boldsymbol{X}$	scenario, random variable of $\boldsymbol{x}$
$a_{\min}, a_{\max}$	min and max accelerations	$\mathcal{Q}_J$	function space spanned by J IFs	$v_{\text{LV}}, v_{\text{BV}}, v_{\text{AV}}$	velocities of LV, BV, AV
$\boldsymbol{A}_t$	action random variable	Q, Q^*	state-action value function, optimal Q	$x_{\text{LV}}, x_{\text{BV}}, x_{\text{AV}}$	positions of LV, BV, AV
$\mathcal{A}$	action space	$Q^{(t)}$	Q at t-th iteration	$\boldsymbol{\alpha}, \alpha_j$	combination coefficients
F	crash event	Q_j	j-th Q	γ	discount ratio
$\mathcal{F}$	σ-algebra	$Q_{\boldsymbol{\alpha}}$	$\boldsymbol{\alpha}$-combination of Q_j	δ_t	temporal difference error
g	surrogate-to-real gap	r, R	reward, random variable of r	Δ	sliding stride
$\mathbb{I}$	indicator function	R_1	range between LV and BV	η	adaptive policy
J	total number of IFs	R_2	range between BV and AV	μ	crash rate in NDE
n	total number of tests	$\dot{R}_1$	range rate between LV and BV	$\hat{\mu}_n$	estimation of μ in NDE
$N(\boldsymbol{s}, \boldsymbol{a})$	visit count of $(\boldsymbol{s}, \boldsymbol{a})$	$\dot{R}_2$	range rate between BV and AV	$\hat{\mu}_q$	estimation of μ in NADE
$\mathbb{N}$	set of natural numbers	$\mathbb{R}$	set of real numbers	ν_t	learning rate at time t
p	naturalistic distribution	$\boldsymbol{s}, \mathcal{S}$	state, state space	σ_q^2	asymptotic variance of $\hat{\mu}_q$
$\mathbb{P}$	probability measure	$\boldsymbol{s}_t, \boldsymbol{S}_t$	state at time t, random variable of s_t	$\sigma_{q_{\boldsymbol{\alpha}}}^2$	asymptotic variance of $\hat{\mu}_{q_{\boldsymbol{\alpha}}}$
P	state transition probability	t, T	time step, time horizon	ϕ	naturalistic policy
q, q_j, q^*	IF, j-th IF, optimal IF	V, V^*	state value function, optimal V	$\psi, \psi_j, \psi_{\boldsymbol{\alpha}}$	importance policies
$q_{\boldsymbol{\alpha}}$	mixture IF	$\mathcal{X}$	scenario space	ω, Ω	state-action pair and space

where $Q_{\boldsymbol{\alpha}} := \sum_{j=1}^{J} \alpha_j Q_j$ is the mixture maneuver challenge, and Q_j are maneuver challenges associated with importance policies ψ_j. The key for solving this optimization problem lies in efficiently obtaining Q^*, which can be formulated as a RL problem (see Subsection III-A). However, learning Q^* through RL in NDE is highly challenging due to the CoR that the critical information such as crash events is extremely rare. Moreover, since our goal is to optimize the combination coefficients, accurately computing Q^* across the entire state-action space is unnecessary, as it requires large number of tests and thereby compromises optimization efficiency. We will address these challenges in the forthcoming Section III.

III. METHODS

To address the CoR, we first propose in Subsection III-A the dense reinforcement learning method. To facilitate DenseRL method for adaptive testing, a new adaptive policy with high sample efficiency is designed in Subsection III-B. Then the regression problem that optimizes combination coefficients will be solved via QP in Subsection III-C. Finally, Subsection III-D summarizes the AdaTE generation algorithm.

A. Dense Reinforcement Learning

The problem to find Q^* can be formulated as a RL problem. Define $\mathcal{M} := (\mathcal{S}, \mathcal{A}, R, P, \gamma)$ as the Markov decision process, where R is the reward function, $R(\boldsymbol{s}) := \mathbb{I}_{\mathcal{S}_{\text{crash}}}(\boldsymbol{s})$, $\forall \boldsymbol{s} \in \mathcal{S}$, and $\gamma \in (0, 1]$ is the discount factor. The state-action value function is given by

$$Q_\phi(\boldsymbol{s}, \boldsymbol{a}) = \mathbb{E}_p \left[\sum_{\tau=t+1}^{T} \gamma^{\tau-t-1} R_\tau \middle| \boldsymbol{S}_t = \boldsymbol{s}, \boldsymbol{A}_t = \boldsymbol{a} \right], \quad (9)$$

where t is the time step of $(\boldsymbol{s}, \boldsymbol{a})$, $R_\tau := R$. If $\gamma = 1$, then $Q^* = Q_\phi$. To see this, write

$$\begin{aligned} Q^*(\boldsymbol{s}_t, \boldsymbol{a}_t) &= \mathbb{P}(F | \boldsymbol{S} = \boldsymbol{s}_t, \boldsymbol{A} = \boldsymbol{a}_t) \\ &= \mathbb{E}[\mathbb{I}_F((\boldsymbol{s}_t, \boldsymbol{a}_t, \ldots, \boldsymbol{S}_T)) | \boldsymbol{S} = \boldsymbol{s}_t, \boldsymbol{A} = \boldsymbol{a}_t] \\ &= \mathbb{E}[\mathbb{I}_{\mathcal{S}_{\text{crash}}}(\boldsymbol{S}_T) | \boldsymbol{S} = \boldsymbol{s}_t, \boldsymbol{A} = \boldsymbol{a}_t] \end{aligned}$$

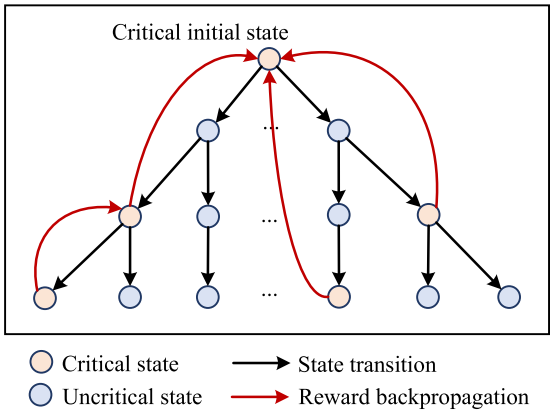


Fig. 3. Illustration of the dense reinforcement learning method.

$$= \mathbb{E}_p \left[\sum_{\tau=t+1}^{T} R_\tau \middle| \boldsymbol{S}_t = \boldsymbol{s}_t, \boldsymbol{A}_t = \boldsymbol{a}_t \right] = Q_\phi(\boldsymbol{s}_t, \boldsymbol{a}_t), \ \forall \boldsymbol{s}_t \in \mathcal{S}, \boldsymbol{a}_t \in \mathcal{A}. \quad (10)$$

Then Q^* can be learned by RL in NDE, which faces the CoR, because the informative data (i.e., critical states and actions) in NDE is rare and the rewards (i.e., crash events) are extremely sparse. To address this challenge, we propose the dense reinforcement learning method, following the similar ideas in [46]. As shown in Fig. 3, the core concept of DenseRL is to start with critical initial states, use off-policy learning mechanism, edit the Markov chains by removing the uncritical states and reconnecting the critical states, and then backpropagate the reward along the edited Markov chains.

Initially, we set $Q(\boldsymbol{s}, \boldsymbol{a}) = 0$, $\forall \boldsymbol{s} \in \mathcal{S}$, $\boldsymbol{a} \in \mathcal{A}$. To optimize Q towards Q^*, DenseRL tries to minimize the Bellman error $\bar{\delta} := \mathcal{B}_\phi Q - Q$, where $\mathcal{B}_\phi$ is the Bellman backup operator [49], [50] defined as

$$\mathcal{B}_\phi Q(\boldsymbol{S}_t, \boldsymbol{A}_t) := \mathbb{E}_\phi[R_{t+1} + \gamma Q(\boldsymbol{S}_{t+1}, \boldsymbol{A}_{t+1}) | \boldsymbol{S}_t, \boldsymbol{A}_t]. \quad (11)$$

Let $\mathcal{S}_c := \{\boldsymbol{s} \in \mathcal{S} : \bar{V}(\boldsymbol{s}) > 0\}$ denote the set of critical states, where $\bar{V}(\boldsymbol{s}) := \mathbb{E}_\phi[\bar{Q}(\boldsymbol{s}, \boldsymbol{A})]$ and $\bar{Q} := (1/J) \sum_{j=1}^{J} Q_j$. Then for each training iteration, the initial state will be sampled

uniformly from $\mathcal{S}_c$, thereafter following an appropriate behavior policy (e.g., the uniform policy). For each transition $(\boldsymbol{S}_t, \boldsymbol{A}_t, R_{t+1}, \boldsymbol{S}_{t+1})$, DenseRL learns Q by the following update rule:

$$Q(\boldsymbol{S}_t, \boldsymbol{A}_t) \leftarrow Q(\boldsymbol{S}_t, \boldsymbol{A}_t) + \nu_t \delta_t \mathbb{I}_{\mathcal{S}_c}(\boldsymbol{S}_t), \tag{12}$$

where ν_t is the learning rate, $\delta_t := \hat{\mathcal{B}}_\phi Q(\boldsymbol{S}_t, \boldsymbol{A}_t) - Q(\boldsymbol{S}_t, \boldsymbol{A}_t)$ is the temporal difference error, and $\hat{\mathcal{B}}_\phi$ is the Bellman evaluation operator defined as

$$\hat{\mathcal{B}}_\phi Q(\boldsymbol{S}_t, \boldsymbol{A}_t) := R_{t+1} + \gamma \mathbb{E}_\phi[Q(\boldsymbol{S}_{t+1}, \boldsymbol{A}_{t+1}) | \boldsymbol{S}_{t+1}]. \tag{13}$$

B. Adaptive Policy Design

In adaptive testing, our focus is primarily on state-action pairs that exhibit significant surrogate-to-real gaps. This emphasis is not effectively utilized when employing DenseRL with a uniform policy, which can result in diminished learning efficiency. Here, the surrogate-to-real gap aims to measure the gap between $Q_{\boldsymbol{\alpha}}$ and Q, which is defined as

$$g(Q||Q_{\boldsymbol{\alpha}}) := \begin{cases} \dfrac{|Q - Q_{\boldsymbol{\alpha}}|}{Q_{\boldsymbol{\alpha}}}, & \text{if } Q_{\boldsymbol{\alpha}} > 0, \\ 0, & \text{if } Q = Q_{\boldsymbol{\alpha}} = 0, \\ +\infty, & \text{if } Q > Q_{\boldsymbol{\alpha}} = 0. \end{cases} \tag{14}$$

To enhance learning efficiency, we propose a novel adaptive policy based on the probabilistic upper confidence tree (PUCT) bound [51], which has been successfully employed in the action selection stage of the Monte Carlo tree search algorithm by AlphaGo [52]. Specifically, the PUCT bound is defined as

$$U_1(\boldsymbol{s}, \boldsymbol{a}) := Q(\boldsymbol{s}, \boldsymbol{a}) + u(\boldsymbol{s}, \boldsymbol{a}), \tag{15}$$

where

$$u(\boldsymbol{s}, \boldsymbol{a}) := c\phi(\boldsymbol{a}|\boldsymbol{s}) \frac{\sqrt{\sum_{\boldsymbol{a}' \in \mathcal{A}} N(\boldsymbol{s}, \boldsymbol{a}')}}{1 + N(\boldsymbol{s}, \boldsymbol{a})}, \tag{16}$$

c is a constant that determines the degree of exploration, and $N(\boldsymbol{s}, \boldsymbol{a})$ is the visit count of $(\boldsymbol{s}, \boldsymbol{a})$, $\forall \boldsymbol{s} \in \mathcal{S}$, $\boldsymbol{a} \in \mathcal{A}$. The action selection policy that maximizing U_1 over the action space initially prefers actions with high exposure frequency $\phi(\boldsymbol{a}|\boldsymbol{s})$ and low visit count $N(\boldsymbol{s}, \boldsymbol{a})$, but asymptotically prefers actions with high state-action value $Q(\boldsymbol{s}, \boldsymbol{a})$. For adaptive testing, we modify the PUCT bound by replacing Q with $g(Q||Q_{\boldsymbol{\alpha}})\phi$, which accounts for both the surrogate-to-real gap $g(Q||Q_{\boldsymbol{\alpha}})$ and the exposure frequency ϕ. Specifically, the adaptive policy is defined as

$$\eta(\boldsymbol{a}|\boldsymbol{s}) := \begin{cases} 1, & \text{if } \boldsymbol{a} = \underset{\boldsymbol{a}' \in \mathcal{A}}{\operatorname{argmax}}\, U_2(\boldsymbol{s}, \boldsymbol{a}'), \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

for all $\boldsymbol{s} \in \mathcal{S}$ and $\boldsymbol{a} \in \mathcal{A}$, where

$$U_2(\boldsymbol{s}, \boldsymbol{a}) := g(Q||Q_{\boldsymbol{\alpha}})(\boldsymbol{s}, \boldsymbol{a})\phi(\boldsymbol{a}|\boldsymbol{s}) + u(\boldsymbol{s}, \boldsymbol{a}). \tag{18}$$

Algorithm 1 Adaptive Testing Environment Generation With Dense Reinforcement Learning

Input: naturalistic distribution p, maneuver challenges Q_j, $j = 1, \ldots, J$, max simulation time T
Output: crash rate estimate

Initialize $Q(\boldsymbol{s}, \boldsymbol{a}) = 0$, $N(\boldsymbol{s}, \boldsymbol{a}) = 0$, $\forall \boldsymbol{s} \in \mathcal{S}$, $\boldsymbol{a} \in \mathcal{A}$;
Initialize $i = 0$, $\Delta = 10$, $\boldsymbol{\alpha} = \mathbf{1}/J$;
Initialize $c = 2$, *termination* = False, $\ell = 1$, $\ell_{\text{th}} = 0.3$;
while *not termination* **do**
  Sample initial state $\boldsymbol{s}$ uniformly from $\mathcal{S}_c$;
  Set $r \leftarrow 0$, $t \leftarrow 0$;
  while $r = 0$ *and* $t < T$ **do**
    Set $t \leftarrow t + 1$;
    Sample $\boldsymbol{a}$ from the adaptive policy η given by Eq. (17);
    Set $N(\boldsymbol{s}, \boldsymbol{a}) \leftarrow N(\boldsymbol{s}, \boldsymbol{a}) + 1$;
    Take action $\boldsymbol{a}$, and observe $\boldsymbol{s}'$, r;
    Update $Q(\boldsymbol{s}, \boldsymbol{a})$ according to Eq. (12);
    Set $\boldsymbol{s} \leftarrow \boldsymbol{s}'$;
  end
  Update $\boldsymbol{\alpha}$ by solving the QP in Eq. (19) (e.g., via CVXOPT [53]);
  Update *termination* according to Eq. (20);
end
while $\ell > \ell_{th}$ **do**
  Set $i \leftarrow i + 1$;
  Sequencially sample a testing scenario $\boldsymbol{X}_i$ from $q_{\boldsymbol{\alpha}}$ and test the CAV in this scenario;
  Estimate crash rate $\hat{\mu}_{q_{\boldsymbol{\alpha}}}$ by Eq. (5);
  Set $\ell \leftarrow$ relative half-width [42] of $\hat{\mu}_{q_{\boldsymbol{\alpha}}}$;
end
Return $\hat{\mu}_{q_{\boldsymbol{\alpha}}}$;

C. Combination Coefficient Optimization

Let $\mathcal{D}$ denote the set of visited critical state-action pairs, then the combination coefficients can be optimized by solving the following regression problem:

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^J} \frac{1}{2} \sum_{(\boldsymbol{s}, \boldsymbol{a}) \in \mathcal{D}} \left[Q(\boldsymbol{s}, \boldsymbol{a}) - Q_{\boldsymbol{\alpha}}(\boldsymbol{s}, \boldsymbol{a}) \right]^2 \quad \text{s.t. } \mathbf{1}^\top \boldsymbol{\alpha} = 1, \ \boldsymbol{\alpha} \geqslant \mathbf{0}, \tag{19}$$

where Q is learned by DenseRL with the adaptive policy. This regression problem (19) is a QP, which can be solved by standard convex optimization tools such as CVXOPT [53].

D. Adaptive Testing Environment Generation Algorithm

By utilizing DenseRL, the combination coefficients can be optimized for the particular CAV under test, resulting in the generation of AdaTE. This process is outlined in Algorithm 1. The termination criterion is when the average sliding difference (ASD) falls below a predetermined threshold (e.g., 0.02), which is defined as

$$\text{ASD}(k) := \frac{1}{J} \sum_{j=1}^{J} \left| \sum_{k'=k-\Delta+1}^{k} \left[\alpha_j^{(k')} - \alpha_j^{(k'-\Delta)} \right] \right|, \tag{20}$$

where $k \in \mathbb{N}_{>0}$ is the number of tests, $\Delta \in \mathbb{N}_{>0}$ is the sliding stride (e.g., $\Delta = 10$), $\alpha_j^{(k)}$ are combination coefficients of the k-th iteration, and we set $\alpha_j^{(k)} := \alpha_j^{(1)}$ for $k < 1$.

IV. Theoretical Analysis

This section will provide theoretical analysis of the proposed DenseRL method. Specifically, we prove the convergence of DenseRL, i.e., $Q^{(t)}$ converges to Q^* with probability one, where $Q^{(t)}$ represents the Q function at t-th iteration and $t \in \mathbb{N}_{\geqslant 0}$. The proof is based on the following lemma [54].

Lemma 1: Consider the stochastic process (ν_t, Δ_t, F_t), $t \in \mathbb{N}_{\geqslant 0}$, where ν_t, Δ_t, $F_t : \Omega \to \mathbb{R}$ satisfy $\Delta_{t+1}(\omega) = [1 - \nu_t(\omega)]\Delta_t(\omega) + \nu_t(\omega)F_t(\omega)$, $\omega \in \Omega$. Let $\mathcal{F}_t$ be a sequence of increasing σ-fields such that ν_0 and Δ_0 are $\mathcal{F}_0$-measurable and ν_t, Δ_t and F_{t-1} are $\mathcal{F}_t$-measurable, $t \in \mathbb{N}_{>0}$. Then Δ_t converges to zero with probability one under the following Assumption 1, 2, 3 and 4.

Assumption 1: The set Ω is finite.

Assumption 2: $\nu_t(\omega) \in [0, 1]$, $t \in \mathbb{N}_{\geqslant 0}$, $\sum_t \nu_t(\omega) = \infty$, $\sum_t \nu_t^2(\omega) < \infty$, $\forall \omega \in \Omega$.

Assumption 3: $\|\mathbb{E}[F_t|\mathcal{F}_t]\|_\infty \leqslant \gamma \|\Delta_t\|_\infty$, where $\gamma \in (0, 1)$, $t \in \mathbb{N}_{\geqslant 0}$.

Assumption 4: $\mathrm{Var}(F_t(\omega)|\mathcal{F}_t) \leqslant C(1 + \|\Delta_t\|_\infty)^2$, $C > 0$, $t \in \mathbb{N}_{\geqslant 0}$.

Proof: See [54]. □

Leveraging Lemma 1, we are now in position to prove the following theorem.

Theorem 1: The DenseRL algorithm given by Subsection III-A converges with probability one to Q^* under the following Assumption 5, 6, 7 and 8.

Assumption 5: The sets $\mathcal{S}$ and $\mathcal{A}$ are finite.

Assumption 6: $\nu_t(\boldsymbol{s}, \boldsymbol{a}) \in [0, 1]$, $t \in \mathbb{N}_{\geqslant 0}$, $\sum_t \nu_t(\boldsymbol{s}, \boldsymbol{a}) = \infty$, $\sum_t \nu_t^2(\boldsymbol{s}, \boldsymbol{a}) < \infty$, $\forall \boldsymbol{s} \in \mathcal{S}_c$, $\boldsymbol{a} \in \mathcal{A}$.

Assumption 7: $Q^{(0)}(\boldsymbol{s}, \boldsymbol{a}) = 0$, $\forall \boldsymbol{s} \in \mathcal{S}$, $\boldsymbol{a} \in \mathcal{A}$.

Assumption 8: $\bar{V}(\boldsymbol{s}) > 0$ whenever $V^*(\boldsymbol{s}) > 0$.

Proof: The correspondence to Lemma 1 follows from associating Ω with the state-action space $\mathcal{S} \times \mathcal{A}$, ω with the state-action pair $(\boldsymbol{s}, \boldsymbol{a})$, $\nu_t(\omega)$ with the learning rate $\nu_t(\boldsymbol{s}, \boldsymbol{a})$, $\Delta_t(\omega)$ with $Q^{(t)}(\boldsymbol{s}, \boldsymbol{a}) - Q^*(\boldsymbol{s}, \boldsymbol{a})$, and $\mathcal{F}_t$ with the σ-field generated by random variables $\{Q^{(0)}, \boldsymbol{S}_0, \boldsymbol{A}_0, \nu_0, R_1, \ldots, \boldsymbol{S}_t, \boldsymbol{A}_t, \nu_t\}$. Then Theorem 1 can be proved by verifying Assumption 1, 2, 3 and 4 in Lemma 1 accordingly.

1) Verification of Assumption 1. Assumption 5 clearly confirms that $\Omega = \mathcal{S} \times \mathcal{A}$ is finite.
2) Verification of Assumption 2. Assumption 2 in Lemma 1 requires that all state-action pairs be visited infinitely often [55]. According to Assumption 7 and 8, we know that $Q^{(t)}(\boldsymbol{s}, \boldsymbol{a}) = Q^*(\boldsymbol{s}, \boldsymbol{a}) = 0$, $\forall \boldsymbol{s} \in \mathcal{S}_{-c}$, $\boldsymbol{a} \in \mathcal{A}$. In other words, the state-action values for uncritical states are already optimal values, and hence do not need to be visited. It is sufficient that all critical state-action pairs will be visited infinitely often, therefore the Assumption 2 can be verified by Assumption 6.
3) Verification of Assumption 3. Rewriting Eq. (12) we get
$$Q^{(t+1)}(\omega_t) = \big[1 - \nu_t(\omega_t)\big]Q^{(t)}(\omega_t) + \nu_t(\omega_t)\hat{\mathcal{B}}_\phi Q^{(t)}(\omega_t). \quad (21)$$
Subtracting from both sides the quantity $Q^*(\omega_t)$ yields
$$\Delta_{t+1}(\omega_t) = \big[1 - \nu_t(\omega_t)\big]\Delta_t(\omega_t) + \nu_t(\omega_t)F_t(\omega_t), \quad (22)$$
where $F_t := \hat{\mathcal{B}}_\phi Q^{(t)} - Q^*$. Since $\mathcal{B}_\phi$ is a γ-contraction mapping [56], we have
$$\begin{aligned}\|\mathbb{E}[F_t|\mathcal{F}_t]\|_\infty &= \|\mathcal{B}_\phi Q^{(t)} - Q^*\|_\infty \\ &= \|\mathcal{B}_\phi Q^{(t)} - \mathcal{B}_\phi Q^*\|_\infty \\ &\leqslant \gamma \|Q^{(t)} - Q^*\|_\infty = \gamma \|\Delta_t\|_\infty. \end{aligned} \quad (23)$$
4) Verification of Assumption 4. Due to the fact that the reward function is bounded, we have
$$\begin{aligned}\mathrm{Var}\big(F_t(\omega_t)|\mathcal{F}_t\big) &= \mathrm{Var}\big(\hat{\mathcal{B}}_\phi Q^{(t)}(\omega_t) - Q^*(\omega_t)|\mathcal{F}_t\big) \\ &= \mathrm{Var}\big(\hat{\mathcal{B}}_\phi Q^{(t)}(\omega_t)|\mathcal{F}_t\big) \\ &\leqslant C(1 + \|\Delta_t\|_\infty)^2, \end{aligned} \quad (24)$$
for some constant $C > 0$.

To verify the measurability requirements in Lemma 1, we note that $Q^{(t)}$ are $\mathcal{F}_t$-measurable, and thus both Δ_t and F_{t-1} are $\mathcal{F}_t$-measurable. Therefore, by Lemma 1 we know that Δ_t converges to zero with probability one, i.e., $Q^{(t)}$ converges to Q^* with probability one. □

Remark 1: The Assumption 5 can be satisfied if both the state space and the action space are discretized. Similar with Assumption 2, the Assumption 6 requires that all critical state-action pairs be visited infinitely often. The Assumption 7 is an initialization requirement that ensures all uncritical state-action values are optimal values (i.e., 0). Moreover, the Assumption 8 means that the critical states identified by all surrogate criticalities can cover the critical states of the CAV under test, since otherwise we would omit critical states that should be explored and learned, leading to misconvergence issues.

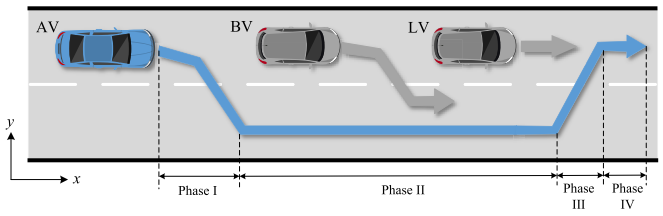
Remark 2: In adaptive testing, to fulfill the requirement of Assumption 8, the selected SMs should exhibit sufficient diversity to encompass the critical states of various CAVs.

V. Results

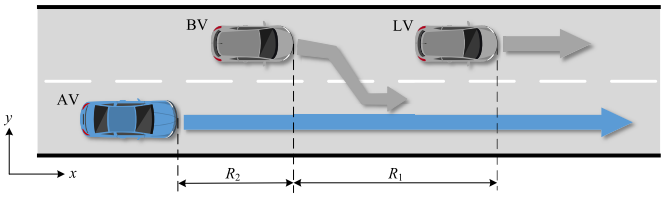
In this section, the high-dimensional overtaking scenarios will be elaborated in Subsection V-A. Then in Subsection V-B, the testing and evaluation results in NDE, NADE and AdaTE will be presented and analyzed.

A. Overtaking Scenarios

As shown in Fig. 4, we study the passing phase of the high-dimensional overtaking scenarios, where a relatively slow-moving leading vehicle (LV) travels in front of the BV, while the automated vehicle (AV) is going to overtake BV and LV. Meanwhile, BV can also overtake LV, then AV may rear-end with BV, resulting in a crash. Denote the longitudinal positions and velocities of LV, BV and AV as x_{LV}, x_{BV}, x_{AV}, v_{LV}, v_{BV}, v_{AV}, respectively, then the state of the overtaking scenarios can be formulated as $\boldsymbol{s} := [v_{\mathrm{BV}}, R_1, \dot{R}_1, R_2, \dot{R}_2]^\top$, where $R_1 := x_{\mathrm{LV}} - x_{\mathrm{BV}}$, $\dot{R}_1 := v_{\mathrm{LV}} - v_{\mathrm{BV}}$, $R_2 := x_{\mathrm{BV}} - x_{\mathrm{AV}}$, and $\dot{R}_2 := v_{\mathrm{BV}} - v_{\mathrm{AV}}$. The action of the overtaking scenarios is defined as the collection of accelerations of LV and BV, i.e.,



(a) Four phases of overtaking scenarios.



(b) Passing phase of overtaking scenarios (focus of this paper).

Fig. 4. Illustrations of the four phases of overtaking scenarios (a) and the passing phase (Phase II) of overtaking scenarios (b). In overtaking scenarios, the AV will overtake BV and LV. In the passing phase, the AV will pass BV and LV. While AV is passing, BV may overtake LV.

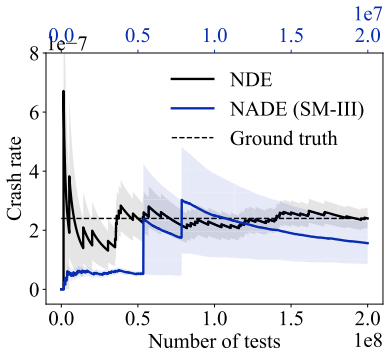
$\boldsymbol{a} := [a_{\text{LV}}, a_{\text{BV}}]^\top$. The maximum simulation time and time resolution are set to 20 s and 0.1 s, respectively.

The maximum dimension of overtaking scenarios will exceed 1400 dimensions (201 time steps, each with 5 state variables and 2 action variables). Due to constraints of vehicle dynamics, the intrinsic dimension of the overtaking scenarios is smaller than the extrinsic one. However, commonly used dimension reduction techniques typically cannot guarantee the preservation of all critical information during the reduction process. As a result, the safety-critical scenarios that are essential for evaluating the safety performance of CAVs may be excluded. Moreover, our approach is complementary to potential dimension reduction techniques that achieve no information loss. In this paper, we focus on generating an adaptive testing environment by directly dealing with the extrinsic dimension of overtaking scenarios. This extrinsic dimension of 1400 gives rise to the CoD, posing significant challenges to the testing environment generation process.

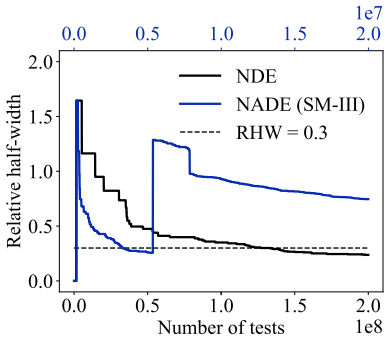
B. Testing and Evaluation Results

In this subsection, we present and analyze the testing and evaluation results in NDE, NADE and AdaTE. For the generation of NDE and NADE, we use the same way as in [45].[1] To investigate the generalizability of the proposed method, we test three diverse AVs: (1) the intelligent driver model (IDM) [57], denoted as AV-I; (2) the IDM calibrated in [58], denoted as AV-II; (3) the RL agent trained by proximal policy optimization [59], denoted as AV-III. We use three representative SMs involving normal, aggressive and conservative driving styles: (1) IDM, denoted as SM-I (same as AV-I); (2) the full velocity difference model (FVDM) [57] with $a_{\min} = -1$ m/s^2, denoted as SM-II; (3) FVDM with $a_{\min} = -6$ m/s^2, denoted as SM-III.

To demonstrate the failure cases of NADE due to surrogate-to-real gaps, we test AV-I in the NADE where the importance function is constructed from SM-III. Fig. 5(a) shows the crash rate estimated by NDE and NADE (SM-III), where the bottom x-axis represents the number of tests of NDE and the top x-axis stands for the number of tests of NADE (SM-III). The relative half-width (RHW) [42] is used as a proxy to measure the convergence of crash rate estimation, which is shown in Fig. 5(b). It can be seen that NADE (SM-III) fails to converge to the ground truth crash rate estimated by NDE. The spikes observed in Figs. 5(a) and 5(b) around 5.37×10^6 and 7.87×10^6 tests are caused by limitations in the SM used by NADE, specifically the SM-III, which fails to capture all unsafe states of the AV under test. When an uncovered crash scenario, $\boldsymbol{X}_u$, occurs–where the SM incorrectly classifies all unsafe states in this scenario as "safe"–NADE produces a testing result of $\mathbb{I}_F(\boldsymbol{X}_u)p(\boldsymbol{X}_u)/q(\boldsymbol{X}_u) = 1$. In most cases, the testing results for crash scenarios are small (around 10^{-3}), while non-crash scenarios yield testing results of 0. A testing result of 1 represents a significant deviation from the typical results, which causes the curves in Figs. 5(a) and 5(b) to show spikes. These spikes highlight that when the SM cannot fully capture the unsafe states for the AV under test, NADE is unable to accurately and efficiently estimate the crash rate. This illustrates the need for an adaptive testing environment to effectively and unbiasedly evaluate the safety performance of diverse AVs.



(a) AV-I



(b) AV-I

Fig. 5. (a) The crash rate estimations of AV-I in NDE and the NADE where the importance function is constructed from SM-III. (b) The RHW of crash rate estimations.

To bolster the evaluation robustness of NADE, we use three SMs with average combination coefficients (i.e., $\boldsymbol{\alpha} = [1/3, 1/3, 1/3]^\top$) to establish the importance function. Fig. 6 shows the crash rate estimated in this new NADE and the corresponding RHW for AV-I, AV-II and AV-III, respectively. It can be found that for all three AVs, NADE converges to

[1]Link to source code: https://github.com/michigan-traffic-lab/Naturalistic-and-Adversarial-Driving-Environment

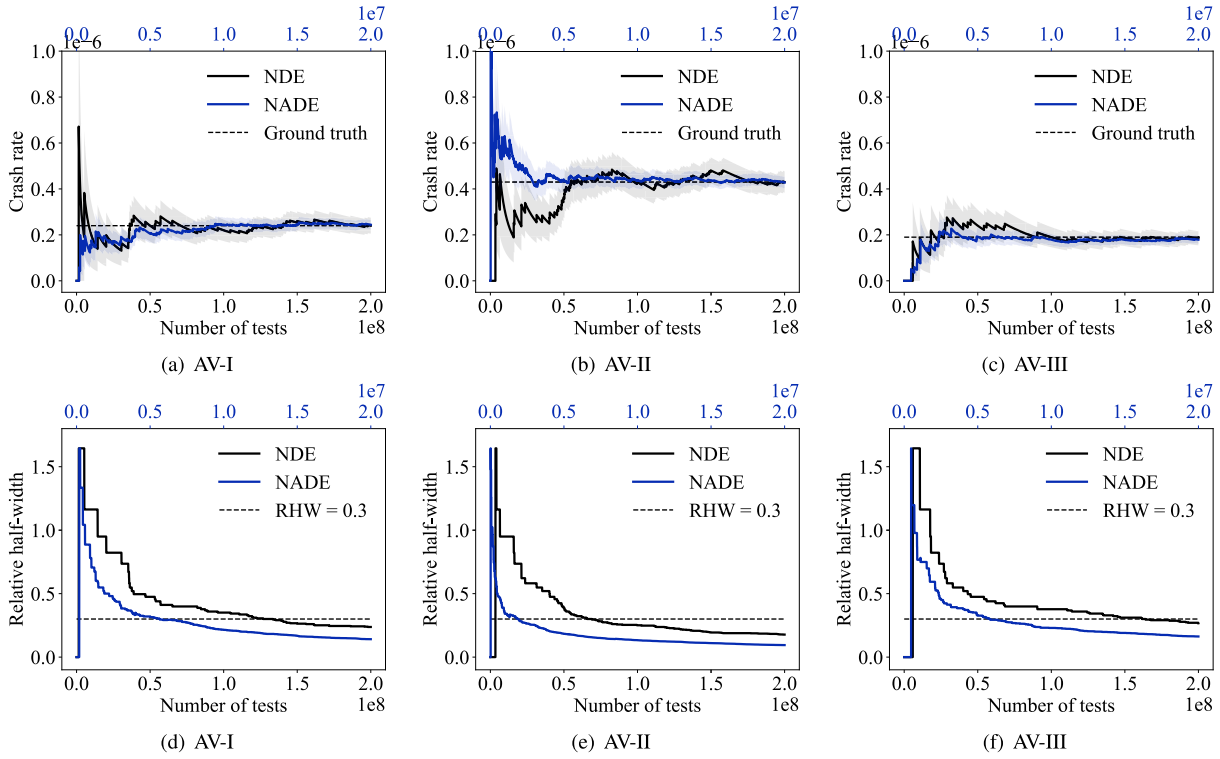


Fig. 6. The crash rate estimations for (a) AV-I, (b) AV-II and (c) AV-III in NDE and NADE, and corresponding RHW for (d) AV-I, (e) AV-II and (f) AV-III.

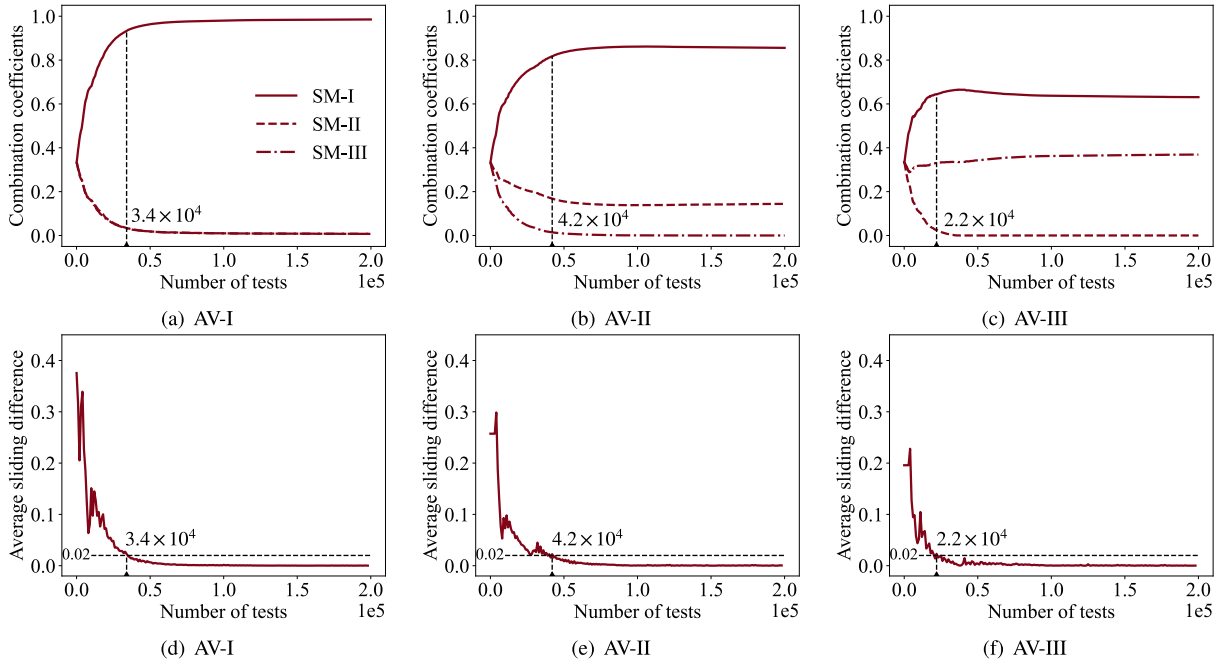


Fig. 7. The combination coefficients optimized by DenseRL with the adaptive policy for (a) AV-I, (b) AV-II and (c) AV-III, and the corresponding ASD for (d) AV-I, (e) AV-II and (f) AV-III.

the same crash rate estimation as NDE, while using much less number of tests for reaching the 0.3 RHW threshold.

Using multiple SMs with average combination coefficients could improve evaluation robustness of NADE, but the evaluation efficiency may be compromised, as such NADE is not customized for any specific AV under test. We optimize the combination coefficients by DenseRL. Figs. 7(a)-(c) uncover that DenseRL is able to optimize the combination coefficients effectively and efficiently. In particular, the optimized combination coefficients for AV-I, AV-II and AV-III are $\boldsymbol{\alpha}_{\text{AV-I}} = [0.94, 0.03, 0.03]^\top$ (the ground truth is $\boldsymbol{\alpha}^*_{\text{AV-I}} = [1, 0, 0]^\top$), $\boldsymbol{\alpha}_{\text{AV-II}} = [0.82, 0.17, 0.01]^\top$, and $\boldsymbol{\alpha}_{\text{AV-III}} = [0.64, 0.03, 0.33]^\top$, respectively. To reach the ASD threshold (0.02), the required number of tests for AV-I, AV-II and AV-III are 3.4×10^4, 4.2×10^4, and 2.2×10^4, respectively, as shown in Figs. 7(d)-(f). Then the AdaTE can be

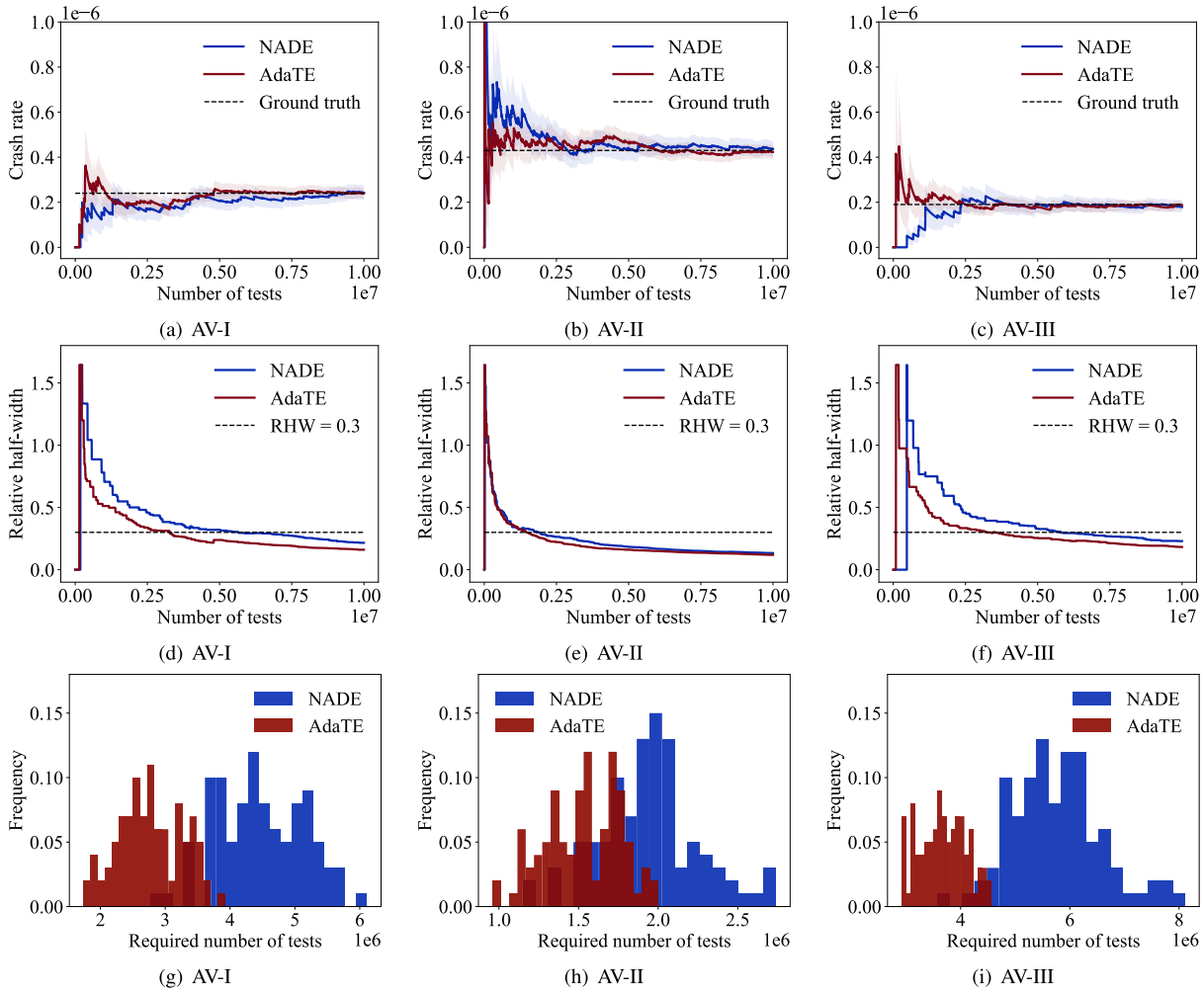


Fig. 8. The crash rate estimations for (a) AV-I, (b) AV-II and (c) AV-III in NADE and AdaTE, RHW of crash rate estimations for (d) AV-I, (e) AV-II and (f) AV-III, and frequency distributions of bootstrapped required number of tests for (g) AV-I, (h) AV-II and (i) AV-III.

TABLE III
AVERAGE REQUIRED NUMBER OF TESTS AND AVERAGE ACCELERATION RATIOS FOR AV-I, AV-II AND AV-III

Methods	AV-I (AAR)	AV-II (AAR)	AV-III (AAR)
NDE	1.23×10^8	7.01×10^7	1.57×10^8
NADE	4.46×10^6 (28)	1.94×10^6 (36)	5.74×10^6 (27)
AdaTE	2.78×10^6 (44)	1.52×10^6 (46)	3.67×10^6 (43)

generated by using three SMs with the optimized combination coefficients.

To investigate the performance of AdaTE, we compare its results with NADE, as shown in Fig. 8. It can be seen from Figs. 8(a)-(c) that AdaTE achieves the same crash rate estimation as NADE for all three AVs. To reach the 0.3 RHW threshold, AdaTE requires less number of tests than NADE, as shown in Figs. 8(d)-(f). To alleviate the stochasticity of experiments, we bootstrap the testing results by shuffling 100 times. The frequency distributions of required number of tests for AV-I, AV-II and AV-III are shown in Figs. 8(g)-(i), respectively. The average required number of tests and average acceleration ratios (AARs) of NDE, NADE and AdaTE for three AVs are shown in Table III, where AARs (presented in parentheses) are ratios of the average required number of tests in NADE and AdaTE with respect to NDE. Compared with NADE, AdaTE can reduce 37.67%, 21.64%, 36.06% number of tests for AV-I, AV-II and AV-III, respectively, revealing significant performance for increasing evaluation efficiency while enhancing evaluation robustness.

VI. CONCLUSION

This paper proposes the dense reinforcement learning approach, which is designed to facilitate adaptive testing for a wide range of CAVs. The key idea involves learning exclusively the values associated with critical state-action pairs that exhibit significant surrogate-to-real gaps. By integrating DenseRL with an adaptive policy for determining the regression target and employing QP for the regression of combination coefficients, the AdaTE can be generated for diverse CAVs. The effectiveness of our method is validated in high-dimensional overtaking scenarios, revealing AdaTE's superior evaluation efficiency compared to both NDE and NADE. One limitation of this work is that we only consider discretized state and action spaces. Extending our approach to continuous cases warrants further exploration. Additionally, we have solely concentrated on the adaptive generation of

testing scenarios, without delving into the adaptive evaluation of testing results. Future endeavors will aim to integrate both aspects.

REFERENCES

[1] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" *Transp. Res. A, Policy Pract.*, vol. 94, pp. 182–193, Dec. 2016.

[2] L. Li, W.-L. Huang, Y. Liu, N.-N. Zheng, and F.-Y. Wang, "Intelligence testing for autonomous vehicles: A new approach," *IEEE Trans. Intell. Veh.*, vol. 1, no. 2, pp. 158–166, Jun. 2016.

[3] L. Li et al., "Artificial intelligence test: A case study of intelligent vehicles," *Artif. Intell. Rev.*, vol. 50, no. 3, pp. 441–465, Oct. 2018.

[4] L. Li et al., "Parallel testing of vehicle intelligence via virtual-real interaction," *Sci. Robot.*, vol. 4, no. 28, p. 4106, Mar. 2019.

[5] G. E. Mullins, P. G. Stankiewicz, and S. K. Gupta, "Automated generation of diverse and challenging scenarios for test and evaluation of autonomous vehicles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1443–1450.

[6] T. Menzel, G. Bagschik, and M. Maurer, "Scenarios for development, test and validation of automated vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1821–1827.

[7] Y. Tian, K. Pei, S. Jana, and B. Ray, "DeepTest: Automated testing of deep-neural-network-driven autonomous cars," in *Proc. IEEE/ACM 40th Int. Conf. Softw. Eng. (ICSE)*, May 2018, pp. 303–314.

[8] J. Norden, M. O'Kelly, and A. Sinha, "Efficient black-box assessment of autonomous vehicle safety," 2019, *arXiv:1912.03618*.

[9] C. E. Tuncali, G. Fainekos, D. Prokhorov, H. Ito, and J. Kapinski, "Requirements-driven test generation for autonomous vehicles with machine learning components," *IEEE Trans. Intell. Vehicles*, vol. 5, no. 2, pp. 265–280, Jun. 2020.

[10] A. Sinha, M. O'Kelly, R. Tedrake, and J. C. Duchi, "Neural bridge sampling for evaluating safety-critical autonomous systems," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 6402–6416.

[11] A. Nonnengart, M. Klusch, and C. Müller, "CriSGen: Constraint-based generation of critical scenarios for autonomous vehicles," in *Proc. Formal Methods. FM Int. Workshops*, Porto, Portugal, Jan. 2020, pp. 233–248.

[12] B. Weng, L. Capito, U. Ozguner, and K. Redmill, "Towards guaranteed safety assurance of automated driving systems with scenario sampling: An invariant set perspective," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 3, pp. 638–651, Sep. 2022.

[13] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 8, pp. 87456–87477, 2020.

[14] H. Sun, S. Feng, X. Yan, and H. X. Liu, "Corner case generation and analysis for safety assessment of autonomous vehicles," *Transp. Res. Record, J. Transp. Res. Board*, vol. 2675, no. 11, pp. 587–600, Nov. 2021.

[15] A. Li, S. Chen, L. Sun, N. Zheng, M. Tomizuka, and W. Zhan, "SceGene: Bio-inspired traffic scenario generation for autonomous driving testing," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14859–14874, Sep. 2022.

[16] J. Wang et al., "AdvSim: Generating safety-critical scenarios for self-driving vehicles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 9909–9918.

[17] N. E. Chelbi, D. Gingras, and C. Sauvageau, "Worst-case scenarios identification approach for the evaluation of advanced driver assistance systems in intelligent/autonomous vehicles under multiple conditions," *J. Intell. Transp. Syst.*, vol. 26, no. 3, pp. 284–310, May 2022.

[18] D. Rempe, J. Philion, L. J. Guibas, S. Fidler, and O. Litany, "Generating useful accident-prone driving scenarios via a learned traffic prior," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jul. 2022, pp. 17305–17315.

[19] X. Yan, Z. Zou, S. Feng, H. Zhu, H. Sun, and H. X. Liu, "Learning naturalistic driving environment with statistical realism," *Nature Commun.*, vol. 14, no. 1, p. 2037, Apr. 2023.

[20] K. Ren, J. Yang, Q. Lu, Y. Zhang, J. Hu, and S. Feng, "Intelligent testing environment generation for autonomous vehicles with implicit distributions of traffic behaviors," *SSRN*, pp. 1–17, Jan. 2024. [Online]. Available: https://ssrn.com/abstract=4973349

[21] S. Li, J. Yang, H. He, Y. Zhang, J. Hu, and S. Feng, "Few-shot scenario testing for autonomous vehicles based on neighborhood coverage and similarity," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2024, pp. 620–626.

[22] S. Li, H. He, J. Yang, J. Hu, Y. Zhang, and S. Feng, "Few-shot testing of autonomous vehicles with scenario similarity learning," 2024, *arXiv:2409.14369*.

[23] R. Bai, J. Yang, W. Gong, Y. Zhang, Q. Lu, and S. Feng, "Accurately predicting probabilities of safety-critical rare events for intelligent systems," in *Proc. IEEE 20th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2024, pp. 3243–3249.

[24] H. He et al., "Knowmoformer: Knowledge-conditioned motion transformer for controllable traffic scenario simulation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshop Data-Driven Auto. Driving Simul.*, Jul. 2024, pp. 1–8.

[25] Y. Ma et al., "Evolving testing scenario generation and intelligence evaluation for automated vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 163, Jun. 2024, Art. no. 104620.

[26] W. Xu, H. Pei, J. Yang, Y. Shi, Y. Zhang, and Q. Zhao, "Exploring critical testing scenarios for decision-making policies: An LLM approach," 2024, *arXiv:2412.06684*.

[27] S. Feng, Y. Feng, C. Yu, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, part I: Methodology," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1573–1582, Mar. 2020.

[28] S. Feng, Y. Feng, H. Sun, S. Bao, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, part II: Case studies," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5635–5647, Sep. 2021.

[29] G. E. Mullins, P. G. Stankiewicz, R. C. Hawthorne, and S. K. Gupta, "Adaptive generation of challenging scenarios for testing and evaluation of autonomous vehicles," *J. Syst. Softw.*, vol. 137, pp. 197–215, Mar. 2018.

[30] M. Koren, S. Alsaif, R. Lee, and M. J. Kochenderfer, "Adaptive stress testing for autonomous vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jul. 2018, pp. 1–7.

[31] M. O'Kelly, A. Sinha, H. Namkoong, J. C. Duchi, and R. Tedrake, "Scalable end-to-end autonomous vehicle testing via rare-event simulation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, Jan. 2018, pp. 1–11.

[32] A. Corso, P. Du, K. Driggs-Campbell, and M. J. Kochenderfer, "Adaptive stress testing with reward augmentation for autonomous vehicle validatio," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 163–168.

[33] S. Feng, Y. Feng, H. Sun, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles: An adaptive framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1213–1222, Feb. 2022.

[34] J. Sun, H. Zhou, H. Xi, H. Zhang, and Y. Tian, "Adaptive design of experiments for safety evaluation of automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14497–14508, Sep. 2021.

[35] J. Yang, H. He, Y. Zhang, S. Feng, and H. X. Liu, "Adaptive testing for connected and automated vehicles with sparse control variates in overtaking scenarios," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2022, pp. 2791–2797.

[36] X. Wang, S. Zhang, and H. Peng, "Comprehensive safety evaluation of highly automated vehicles at the roundabout scenario," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 20873–20888, Nov. 2022.

[37] X. Gong, S. Feng, and Y. Pan, "An adaptive multi-fidelity sampling framework for safety analysis of connected and automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 14393–14405, Dec. 2023.

[38] J. Yang, H. Sun, H. He, Y. Zhang, H. X. Liu, and S. Feng, "Adaptive safety evaluation for connected and automated vehicles with sparse control variates," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 1761–1773, Feb. 2024.

[39] J. Zhou, L. Wang, and X. Wang, "Online adaptive generation of critical boundary scenarios for evaluation of autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 6, pp. 6372–6388, Jun. 2023.

[40] H. X. Liu and S. Feng, "Curse of rarity for autonomous vehicles," *Nature Commun.*, vol. 15, no. 1, p. 4808, Jun. 2024.

[41] S. Feng, Y. Feng, X. Yan, S. Shen, S. Xu, and H. X. Liu, "Safety assessment of highly automated driving systems in test tracks: A new framework," *Accident Anal. Prevention*, vol. 144, Sep. 2020, Art. no. 105664.

[42] D. Zhao et al., "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 595–607, Mar. 2017.

[43] D. Zhao, X. Huang, H. Peng, H. Lam, and D. J. LeBlanc, "Accelerated evaluation of automated vehicles in car-following maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 733–744, Mar. 2017.

[44] S. Zhang, H. Peng, D. Zhao, and H. E. Tseng, "Accelerated evaluation of autonomous vehicles in the lane change scenario based on subset simulation technique," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, Nov. 2018, pp. 3935–3940.
[45] S. Feng, X. Yan, H. Sun, Y. Feng, and H. X. Liu, "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment," *Nature Commun.*, vol. 12, no. 1, pp. 1–14, Feb. 2021.
[46] S. Feng et al., "Dense reinforcement learning for safety validation of autonomous vehicles," *Nature*, vol. 615, no. 7953, pp. 620–627, Mar. 2023.
[47] A. B. Owen, *Monte Carlo Theory, Methods and Examples*. Stanford, CA, USA: Stanford Univ. Press, 2013.
[48] S. K. Au and J. L. Beck, "Important sampling in high dimensions," *Struct. Saf.*, vol. 25, no. 2, pp. 139–163, Apr. 2003.
[49] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.
[50] S. E. Li, *Reinforcement Learning for Sequential Decision and Optimal Control*. Cham, Switzerland: Springer, 2023.
[51] C. D. Rosin, "Multi-armed bandits with episode context," *Ann. Math. Artif. Intell.*, vol. 61, no. 3, pp. 203–230, Mar. 2011.
[52] D. Silver et al., "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.
[53] M. Andersen, J. Dahl, and L. Vandenberghe. (2004). *Cvxopt: Python Software for Convex Optimization, Version 1.3.2*. [Online]. Available: https://cvxopt.org
[54] T. Jaakkola, M. Jordan, and S. Singh, "Convergence of stochastic iterative dynamic programming algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 6, 1993, pp. 1–8.
[55] F. S. Melo, "Convergence of Q-learning: A simple proof," Institue Syst. Robot., Lisbon, Portugal, Tech. Rep., 2001, pp. 1–4.
[56] C. Szepesvári, *Algorithms for Reinforcement Learning*. Cham, Switzerland: Springer, 2022.
[57] J. W. Ro, P. S. Roop, A. Malik, and P. Ranjitkar, "A formal approach for modeling and simulation of human car-following behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 639–648, Feb. 2018.
[58] J. Sangster, H. Rakha, and J. Du, "Application of naturalistic driving data to modeling of driver car-following behavior," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2390, no. 1, pp. 20–33, Jan. 2013.
[59] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

Jingxuan Yang received the bachelor's degree from the School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include adaptive testing and evaluation of connected and automated vehicles.

Ruoxuan Bai received the bachelor's degree from the School of Mechanical and Vehicular Engineering, Beijing Institute of Technology, Beijing, China, in 2022. She is currently pursuing the master's degree with the Department of Automation, Tsinghua University, Beijing. Her current research interests include testing and evaluation of intelligent systems.

Haoyuan Ji is currently pursuing the bachelor's degree with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include adaptive testing, reinforcement learning and computer vision.

Yi Zhang (Senior Member, IEEE) received the B.S. and M.S. degrees from Tsinghua University, China, in 1986 and 1988, respectively, and the Ph.D. degree from the University of Strathclyde, U.K., in 1995. He is currently a Professor of control science and engineering with Tsinghua University. His current research interests include intelligent transportation systems. His active research areas include intelligent vehicle-infrastructure cooperative systems, analysis of urban transportation systems, urban road network management, traffic data fusion and dissemination, and urban traffic control and management. His research fields also cover the advanced control theory and applications, advanced detection and measurement, and systems engineering.

Jianming Hu (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees in 1995, 1998, and 2001, respectively. He is currently an Associate Professor with the Department of Automation (DA), Tsinghua University. He has presided and participated in more than 20 research projects granted from the Ministry of Science and Technology of China, National Science Foundation of China, and other large companies with more than 30 journal articles and more than 100 conference papers. His research interests include networked traffic flow, large-scale traffic information processing, intelligent vehicle infrastructure cooperation systems (V2X or connected vehicles), and urban traffic signal control.

Shuo Feng (Member, IEEE) received the bachelor's and Ph.D. degrees from the Department of Automation, Tsinghua University, China, in 2014 and 2019, respectively. He was a Post-Doctoral Research Fellow with the Department of Civil and Environmental Engineering and also an Assistant Research Scientist with the University of Michigan Transportation Research Institute (UMTRI), University of Michigan, Ann Arbor. He is currently an Associate Professor with the Department of Automation, Tsinghua University. His research interests include the development and validation of safety-critical machine learning, particularly for connected and automated vehicles. He was a recipient of the Best Ph.D. Dissertation Award from the IEEE Intelligent Transportation Systems Society in 2020 and the ITS Best Paper Award from the INFORMS TSL Society in 2021. He is an Associate Editor of IEEE TRANSACTIONS ON INTELLIGENT VEHICLES and an Academic Editor of the *Automotive Innovation*.