

Highlights

Adaptive safety performance testing for autonomous vehicles with adaptive importance sampling

Jingxuan Yang, Zihang Wang, Daihan Wang, Yi Zhang, Qiuqing Lu, Shuo Feng

- Efficiently and accurately assess the safety performance of various autonomous vehicles.
- Develop an adaptive testing framework that continuously refines testing policies during large-scale testing.
- Demonstrate significant evaluation efficiency in overtaking scenarios.

Adaptive safety performance testing for autonomous vehicles with adaptive importance sampling

Jingxuan Yang^a, Zihang Wang^b, Daihan Wang^c, Yi Zhang^{a,d}, Qiuqing Lu^a and Shuo Feng^{a,*}

^aDepartment of Automation, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, 100084, China

^bWeiyang College, Tsinghua University, Beijing, 100084, China

^cGuangzhou Automobile Group R&D Center, Guangzhou, 511434, China

^dJiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, Nanjing, 210096, China

ARTICLE INFO

Keywords:

Adaptive testing
Autonomous vehicles
Adaptive importance sampling
Dense reinforcement learning

ABSTRACT

Efficient and accurate safety testing and evaluation are crucial for autonomous vehicles (AVs). Recent studies have utilized prior information, such as surrogate models, to enhance testing efficiency by deliberately generating safety-critical scenarios. However, discrepancies between this prior knowledge and actual AV performance can undermine their effectiveness. To address this challenge, adaptive testing methods dynamically adjust testing policies based on posterior information of AVs, such as testing results. Most existing approaches focus on adaptively optimizing testing policies during pre-tests, yet neglecting how to adapt the testing policies in the large-scale testing process that is required for unbiased safety performance evaluation. To fill this gap, we propose an adaptive testing framework that continuously optimizes testing policies throughout large-scale testing. Our approach iteratively learns AV dynamics through deep learning and optimizes testing policies based on the learned dynamics using reinforcement learning. To tackle the challenge posed by the rarity of safety-critical events, our method focuses exclusively on learning safety-critical states in both the dynamics learning and the policy optimization processes. Additionally, we enhance evaluation robustness by integrating multiple pre-trained testing policies and optimizing their combination coefficients. To accurately assess safety performance, we evaluate testing results obtained from various testing policies using adaptive importance sampling. Experimental validation in overtaking scenarios demonstrates the significant evaluation efficiency of our method.

1. Introduction

Safety testing and evaluation are critical components in the development and deployment of AVs. One recommended method for assessing the safety performance of AVs is to test them in naturalistic driving environment (NDE), observe their behaviors, and statistically compare it to that of human drivers. However, the rarity of safety-critical events in NDE necessitates large-scale testing to evaluate AV safety comprehensively, often requiring testing mileage to extend into the billions of miles (Kalra and Paddock, 2016). In recent years, substantial research advances have been made in improving the efficiency of AV testing and evaluation (Li et al., 2016, 2018, 2019, 2020; Menzel et al., 2018; Tian et al., 2018; Riedmaier et al., 2020; Sun et al., 2021a; Li et al., 2021; Wang et al., 2021; Rempe et al., 2022; Feng et al., 2023; Yan et al., 2023; Ren et al., 2024; Li et al., 2024a,b; Bai et al., 2024; He et al., 2024). A key concept in these studies is leveraging prior knowledge of AVs, such as surrogate models (SMs), to generate testing scenarios rich in safety-critical events. These SMs effectively capture the overall characteristics of AVs, thus significantly enhancing testing efficiency (Feng et al., 2020a,c, 2021). However, the high complexity and black-box nature of AVs lead to discrepancies between SMs and real AV performance. This surrogate-to-real gap can severely reduce the effectiveness of the generated testing scenarios in evaluating the safety performance of diverse AVs.

To address this challenge, the core concept of existing adaptive testing methods (Zhao et al., 2016; Mullins et al., 2018; Koren et al., 2018; Feng et al., 2020b; Sun et al., 2021b; Yang et al., 2022, 2023, 2024; Gong et al., 2023) is to dynamically adjust the testing policy (i.e., the strategy for generating testing scenarios) based on posterior information

*Corresponding author

✉ yangjx20@mails.tsinghua.edu.cn (J. Yang); zihang-w21@mails.tsinghua.edu.cn (Z. Wang); wangdaihan@gacrnd.com (D. Wang); zhyi@mail.tsinghua.edu.cn (Y. Zhang); qiuqinglu@mail.tsinghua.edu.cn (Q. Lu); fshuo@tsinghua.edu.cn (S. Feng)
ORCID(s): 0000-0001-9798-7347 (J. Yang); 00009-0006-4038-2754 (Z. Wang); 0009-0006-7257-3870 (D. Wang); 0000-0001-5526-866X (Y. Zhang); 0000-0002-3038-256X (Q. Lu); 0000-0002-2117-4427 (S. Feng)

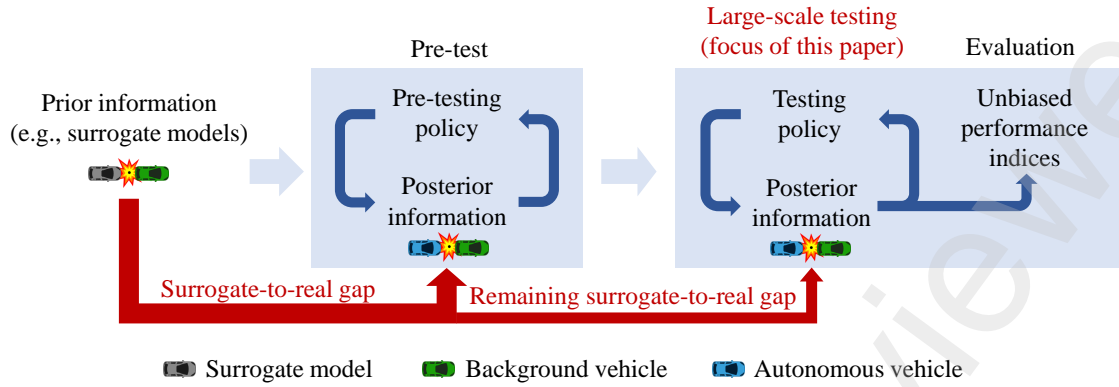


Figure 1: Illustration of three adaptive testing paradigms, each targeting a different stage: pre-test, large-scale testing, and evaluation. This paper focuses on the large-scale testing stage, introducing an adaptive testing framework that continuously optimizes testing policies.

of AVs, such as testing results. As more testing results are gathered, the posterior information of AVs is progressively enriched, enabling the optimization of testing policies that are better tailored to the specific AV under test. Most existing adaptive testing approaches focus on efficiently optimizing testing policies through a few pre-tests, and then unbiasedly estimate the safety performance of AVs through large-scale testing, utilizing techniques such as importance sampling (Zhao et al., 2016; Feng et al., 2020b; Gong et al., 2023; Yang et al., 2024). In importance sampling, the testing policies (i.e., importance functions) must meet certain criteria to ensure evaluation unbiasedness (Owen, 2013). With the large-scale testing results, the estimation efficiency of performance indices can be further improved during the evaluation stage (Yang et al., 2022, 2023). The three adaptive testing paradigms mentioned are illustrated in Fig. 1. Most existing methods have not addressed adaptive testing during the large-scale testing stage. The primary distinction between adaptive testing in the pre-test stage and the large-scale testing stage lies in how posterior information is collected. In the pre-test stage, posterior information is actively gathered through specially designed pre-testing policies aimed at minimizing the surrogate-to-real gap and optimizing testing policies with minimal tests. In contrast, during large-scale testing, posterior information is gathered reactively through testing policies that aim to efficiently and accurately evaluate the safety performance of AVs. Consequently, it is challenging to leverage such information to further optimize the testing policies during the large-scale testing stage, while maintaining the unbiasedness of the performance indices. This optimization is particularly challenging due to the curse of rarity (CoR) (Liu and Feng, 2024), as the informative scenarios (e.g., crashes) sampled by the testing policies are usually rare in the high-dimensional scenario spaces involving numerous sequential decision variables of road participants.

To address this challenge, we propose a novel adaptive testing framework that continuously optimizes testing policies during large-scale testing, as shown in Fig. 2. The key idea is to learn AV dynamics based on posterior information (i.e., testing results), use the learned dynamics to generate simulation testing results, and then optimize testing policies based on these results through reinforcement learning. Two main issues arise. First, since testing policies must ensure unbiased evaluation, the safety-critical dynamics data in testing scenarios are rare, making the use of ordinary deep learning to learn AV dynamics highly inefficient. Second, due to the limited safety-critical dynamics data, the learned AV dynamics may be more accurate in certain states but lack robustness across the entire state space. To overcome these issues, we first focus on learning AV dynamics from safety-critical dynamics data rather than from all data. We then optimize the testing policies using dense reinforcement learning (Feng et al., 2023; Yang et al., 2024), which exclusively learns from safety-critical states during the temporal-difference learning process. To enhance robustness, we utilize multiple pre-trained policies and optimize their combination coefficients. Finally, to estimate performance indices from the testing results of different testing policies, we employ adaptive importance sampling (Bugallo et al., 2017) to effectively aggregate these results. Validation in overtaking scenarios demonstrates that our method significantly improves evaluation efficiency for various AVs.

The subsequent sections of this paper are structured as follows. In Section 2, we elaborate on the preliminary methods for testing AVs in NDE and formulate the adaptive testing problem in large-scale testing stage. Section 3 then addresses this problem by proposing an adaptive testing framework. Next, Section 4 provides a theoretical analysis

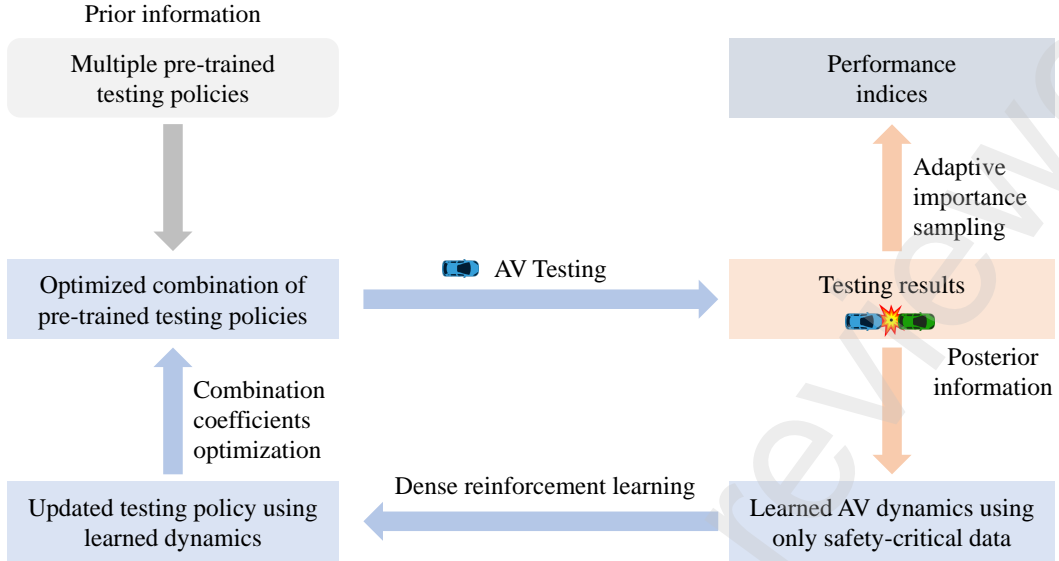


Figure 2: Illustration of the adaptive testing framework. The AV dynamics is learned using only safety-critical states, and the testing policy is then optimized based on the learned dynamics through dense reinforcement learning. Evaluation robustness is improved by utilizing multiple pre-trained testing policies and optimizing their combination coefficients. The testing results from different testing policies are evaluated using adaptive importance sampling.

Table 1

List of Abbreviations.

Abbreviation	Definition
AAR	average acceleration ratio
AV	autonomous vehicle
BV	background vehicle
CoD	curse of dimensionality
CoR	curse of rarity
LV	leading vehicle
MSE	mean squared error
NADE	naturalistic and adversarial driving environment
NDE	naturalistic driving environment
NeuDyM	neural dynamics model
RHW	relative half-width
SM	surrogate model

of the convergence, consistency, and efficiency of our method. Finally, Section 5 validates the proposed method's effectiveness through testing various AVs in overtaking scenarios.

2. Problem formulation

This section introduces the foundational concepts for testing AVs in NDE in Subsections 2.1. To improve the evaluation efficiency of NDE, the naturalistic and adversarial driving environment (Feng et al., 2021) is introduced in Subsection 2.2. The adaptive testing problem is subsequently formulated in Subsection 2.3. Table 1 presents the list of abbreviations, while Table 2 summarizes the notation used.

2.1. Naturalistic driving environment testing

Let $\mathcal{E} := (S, \mathcal{A}, \mathcal{A}^{AV}, \phi, \pi, \rho)$ represent the NDE, where S is the state space of the AV and background vehicles (BVs), \mathcal{A} and \mathcal{A}^{AV} are the action spaces for BVs and AV, respectively, ϕ and π denote the driving policies of BVs and

Table 2
 Summary of Notation.

Notation	Definition	Notation	Definition
\mathbf{a}, \mathbf{a}_t	BVs' action, BVs' action at time t	R_1, R_2	range between LV and BV, BV and AV
\mathbf{a}^{AV}	action of AV	\dot{R}_1, \dot{R}_2	range rate between LV and BV, BV and AV
a_{\min}, a_{\max}	min and max accelerations	\mathbb{R}	set of real numbers
A_t	action random variable of BVs	s, S	state, state space
A^{AV}	action random variable of AV	s_t, S_t	state at time t , random variable of s_t
\mathcal{A}	action space of BVs	S_{crash}	set of crash states
\mathcal{A}^{AV}	action space of AV	S_c	set of safety-critical states
B	batch size	t, T	time step, time horizon
$D^{(k)}$	set of testing results at update step k	\mathcal{T}_c	set of critical time steps
\mathcal{E}	naturalistic driving environment	V, V^*	criticality, optimal criticality
F	crash event	w, W	importance policy weight, importance weight
\mathcal{F}	σ -algebra	$W^{(k)}$	importance weight at update step k
$\mathbb{1}_F$	indicator function of F	\mathcal{X}	scenario space
J	number of pre-trained maneuver challenges	\mathbf{x}, \mathbf{X}	scenario, scenario random variable
k	update step in adaptive testing	$v_{\text{LV}}, v_{\text{BV}}, v_{\text{AV}}$	longitudinal velocities of LV, BV, AV
\mathcal{L}	loss function	$x_{\text{LV}}, x_{\text{BV}}, x_{\text{AV}}$	longitudinal positions of LV, BV, AV
n_k	number of tests in update step k	α, α_j	combination coefficients
$n^{(k)}$	total number of tests up to update step k	$\alpha^{(k)}$	combination coefficients at update step k
\mathbb{N}	set of natural numbers	γ	discount ratio
p	naturalistic distribution	δ_t	temporal difference error at time t
\mathbb{P}_p	probability measure	μ	crash rate in NDE
P_π	state transition probability	$\hat{\mu}_p$	estimation of μ in NDE
\mathcal{P}	power set	$\hat{\mu}_q$	estimation of μ in NADE
q	importance function	$\tilde{\mu}^{(k)}$	estimation of μ by adaptive testing at update step k
q^*	optimal importance function	v_t	learning rate at time t
$q^{(k)}$	importance function at update step k	σ_q^2	asymptotic variance of $\hat{\mu}_q$
\mathcal{Q}	function space of importance functions	ϕ	driving policy of BVs
Q	maneuver challenge	ψ	importance policy
Q^*	optimal maneuver challenge	ψ^*	optimal importance policy
$Q^{(k)}$	maneuver challenge at update step k	π	driving policy of AV
Q_j	pre-trained maneuver challenges	$\pi^{(k)}$	NeuDyM policy at update step k
Q_α	α -combination of Q_j	ρ	initial state distribution
r, R	reward, reward random variable	c	defensive parameter

AV, respectively, and ρ is the initial state distribution. The scenario can then be defined as

$$\mathbf{x} := (s_0, \mathbf{a}_0, \dots, s_{T-1}, \mathbf{a}_{T-1}, s_T) \in \mathcal{X}, \quad (1)$$

where $s_t \in \mathcal{S}$ represents the state at time t , $\mathbf{a}_t \in \mathcal{A}$ is the action of BVs at time t , T is the time horizon, and \mathcal{X} is the set of all feasible scenarios. In NDE, the naturalistic distribution p of scenarios is

$$p(\mathbf{x}) := \rho(s_0) \prod_{t=0}^{T-1} \phi(\mathbf{a}_t | s_t) P_\pi(s_{t+1} | s_t, \mathbf{a}_t), \quad \forall \mathbf{x} \in \mathcal{X}, \quad (2)$$

where P_π is the state transition probability associated with the AV policy π .

Consider the probability space $(\mathcal{X}, \mathcal{F}, \mathbb{P}_p)$, 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} , and $\mathbb{P}_p(\{\mathbf{x}\}) := p(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}$ is the probability measure. Denote the crash event between the AV and BVs as $F := \{\mathbf{x} \in \mathcal{X} : s_T \in S_{\text{crash}}\} \in \mathcal{F}$, where S_{crash} represents the set of crash states. The crash rate in NDE is then given by

$$\mu := \mathbb{P}_p(F) = \mathbb{E}_p[\mathbb{1}_F(\mathbf{X})], \quad (3)$$

where $\mathbf{X} : \mathbf{x} \mapsto \mathbf{x}$ for all $\mathbf{x} \in \mathcal{X}$ is the scenario random variable, and $\mathbb{1}_F$ is the indicator function of F ,

$$\mathbb{1}_F(\mathbf{X}) = \begin{cases} 1, & \text{if } \mathbf{X} \in F, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

According to Monte Carlo theory (Owen, 2013), the crash rate can be estimated in NDE as

$$\hat{\mu}_p := \frac{1}{n} \sum_{i=1}^n \mathbb{1}_F(\mathbf{X}_i), \mathbf{X}_i \sim p, \quad (5)$$

where n is the number of tests, and \mathbf{X}_i are scenario random variables sampled independently and identically distributed (i.i.d.) from the naturalistic distribution p .

2.2. Naturalistic and adversarial driving environment testing

The evaluation efficiency of crash rate in NDE is severely hindered due to the CoR (Liu and Feng, 2024), as the rarity of crash events requires an impractically large number of testing miles-often reaching hundreds of millions or even billions (Kalra and Paddock, 2016). Using importance sampling to replace the naturalistic distribution p with an importance function q can enhance the evaluation efficiency (Zhao et al., 2016, 2017; Feng et al., 2020a,c). However, the importance sampling approach cannot be directly applied in high-dimensional scenarios because of the curse of dimensionality (CoD) (Au and Beck, 2003), as the estimation variance of the crash rate using non-optimal importance functions increases exponentially with the number of scenario dimensions. To address this issue, the naturalistic and adversarial driving environment (NADE) has been proposed (Feng et al., 2021), which applies importance sampling only to critical variables at critical time steps, while retaining the naturalistic distribution for other variables. Specifically, the importance function is

$$q(\mathbf{x}) := \rho(s_0) \prod_{t=0}^{T-1} \psi(\mathbf{a}_t | s_t) P_{\pi}(s_{t+1} | s_t, \mathbf{a}_t), \forall \mathbf{x} \in \mathcal{X}, \quad (6)$$

where ψ is the importance policy defined as

$$\psi(\mathbf{a} | s) := \begin{cases} \phi(\mathbf{a} | s), & \text{if } s \notin S_c, \\ \epsilon \phi(\mathbf{a} | s) + (1 - \epsilon) \frac{Q(s, \mathbf{a}) \phi(\mathbf{a} | s)}{V(s)}, & \text{if } s \in S_c. \end{cases} \quad (7)$$

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

$$\hat{\mu}_q := \frac{1}{n} \sum_{i=1}^n \frac{\mathbb{1}_F(\mathbf{X}_i) p(\mathbf{X}_i)}{q(\mathbf{X}_i)} = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_F(\mathbf{X}_i) W(\mathbf{X}_i), \mathbf{X}_i \sim q, \quad (8)$$

where $W(\mathbf{x}) := p(\mathbf{x})/q(\mathbf{x}) = \prod_{t \in \mathcal{T}_c} w(\mathbf{a}_t | s_t)$ for all $\mathbf{x} \in \mathcal{X}$ is the importance weight, $w(\mathbf{a} | s) := \phi(\mathbf{a} | s)/\psi(\mathbf{a} | s)$ is the importance policy weight, and $\mathcal{T}_c := \{t \in \{0, \dots, T-1\} : s_t \in S_c\}$ denotes the set of critical time steps.

2.3. Adaptive testing

While NADE have demonstrated great potential for efficient testing and evaluation of AVs using the importance function derived from a single SM (Feng et al., 2021, 2023), its evaluation efficiency is significantly affected by the surrogate-to-real gap. In NADE, this gap refers to the discrepancies between the designed importance function and the optimal importance functions for different AVs. The adaptive testing method aims to address this issue. Mathematically, the objective of adaptive testing is to minimize the variance of the crash rate estimate in NADE over the function space \mathcal{Q} , which contains all probability distributions q that satisfy

$$q(\mathbf{x}) > 0, \forall \mathbf{x} \in \{\mathbf{x} \in \mathcal{X} : \mathbb{1}_F(\mathbf{x}) p(\mathbf{x}) > 0\}. \quad (9)$$

This optimization problem can be formulated as

$$\min_{q \in \mathcal{Q}} \sigma_q^2 := \text{Var}_q \left(\frac{\mathbb{1}_F(\mathbf{X}) p(\mathbf{X})}{q(\mathbf{X})} \right). \quad (10)$$

By optimizing q within the function space \mathcal{Q} , the importance function can be customized for specific AVs, thereby improving the evaluation efficiency.

In the pre-test stage, one solution to the adaptive testing problem (10) is to employ the Bayesian optimization framework (Snoek et al., 2012), where acquisition functions can be designed to identify the next most informative scenarios. The primary goal of these acquisition functions is to minimize the difference between q and the optimal importance function q^* with as few tests as possible. For example, the acquisition function can be designed to sample the next testing scenario \mathbf{x}' such that the absolute gap between $q(\mathbf{x}')$ and $q^*(\mathbf{x}')$ is maximized across the scenario space. In contrast, during large-scale testing, to accurately and efficiently evaluate performance indices, the testing scenarios are sampled from importance functions (either initially designed or optimized through adaptive testing) that meet the condition in Eq. (9). As a result, the testing scenarios cannot be deliberately selected during large-scale testing, and we must rely on the testing scenarios and results generated by the importance functions to optimize them. This optimization is highly challenging due to the CoR, as the informative scenarios (e.g., crashes) sampled by the importance functions are so rare that an impractically large number of tests is needed to optimize q effectively.

From Eq. (6), it is clear that optimizing q is equivalent to optimizing the importance policy ψ . According to importance sampling theory (Owen, 2013), the optimal importance policy is given by $\psi^*(\mathbf{a}|\mathbf{s}) := \mathcal{Q}^*(\mathbf{s}, \mathbf{a})\phi(\mathbf{a}|\mathbf{s})/V^*(\mathbf{s})$ for all $\mathbf{s} \in \mathcal{S}$ and $\mathbf{a} \in \mathcal{A}$, where $\mathcal{Q}^* := \mathbb{P}_\rho(F|\mathcal{S}, \mathbf{A})$ is the optimal maneuver challenge, and $V^* := \mathbb{P}_\rho(F|\mathcal{S})$ is the optimal criticality. Thus, the key to adaptive testing is optimizing \mathcal{Q} toward \mathcal{Q}^* based on the currently available testing results. Let $\mathcal{D}^{(k)}$ represent the testing scenarios and results accumulated up to update step k , and $\mathcal{Q}^{(k)}$ denote the optimized maneuver challenge at update step k , for $k = 1, 2, \dots$. The optimization of $\mathcal{Q}^{(k)}$ based on $\mathcal{D}^{(k-1)}$ can be formulated as a reinforcement learning problem. Let $\mathcal{M} := (\mathcal{S}, \mathcal{A}, R, P_\pi, \rho, \gamma)$ denote the Markov decision process, where R is the reward function defined as $R(\mathbf{s}) := \mathbb{1}_{\mathcal{S}_{\text{crash}}}(\mathbf{s})$ for all $\mathbf{s} \in \mathcal{S}$, and $\gamma \in (0, 1)$ is the discount factor. The optimal maneuver challenge \mathcal{Q}^* can then be expressed as the state-action value function, i.e., $\mathcal{Q}^* := \mathbb{E}_{(\phi, \pi)}[R_{t:T}|\mathcal{S}_t, \mathbf{A}_t]$, where t is any time step, and $R_{t:T} := \sum_{\tau=t+1}^T \gamma^{\tau-t-1} R_\tau$ is the discounted sum of future rewards, with $R_\tau := R$ for all $\tau = t+1, \dots, T$.

The maneuver challenges $\mathcal{Q}^{(k)}$ can then be trained using reinforcement learning based on $\mathcal{D}^{(k-1)}$. However, this process encounters the CoR, as safety-critical states and actions are rare, and rewards (i.e., crash events) are highly sparse. As the rarity of the crash events increases, the amount of training data for reinforcement learning to effectively learn $\mathcal{Q}^{(k)}$ increases significantly (Liu and Feng, 2024). To mitigate this issue, our previous work introduced the dense reinforcement learning method (Feng et al., 2023; Yang et al., 2024), which learns exclusively the safety-critical states during the temporal difference learning process. However, during testing, the number of safety-critical states in $\mathcal{D}^{(k-1)}$ is far less than what is required for $\mathcal{Q}^{(k)}$ to converge to \mathcal{Q}^* . We address this challenge in the following section.

3. Methods

This section presents our adaptive testing method. The core idea is to learn AV dynamics based on testing scenarios and results $\mathcal{D}^{(k-1)}$, and then optimize maneuver challenges $\mathcal{Q}^{(k)}$ based on the learned dynamics. Two primary issues emerge. First, the rarity of safety-critical dynamics data in testing scenarios renders ordinary deep learning for learning AV dynamics highly inefficient. Second, due to the limited availability of safety-critical dynamics data, the learned AV dynamics may not generalize well across the entire state space. To address the first issue, Subsection 3.1 introduces an adaptive dense reinforcement learning method to effectively learn maneuver challenges. To address the second, Subsection 3.2 formulates a quadratic programming for optimizing the combination coefficients of multiple maneuver challenges pre-trained with different SMs. In Subsection 3.3, we evaluate the crash rate using testing results obtained from continuously updated importance functions based on adaptive importance sampling. Finally, Subsection 3.4 summarizes the complete adaptive testing algorithm.

3.1. Adaptive dense reinforcement learning

To effectively learn maneuver challenges $\mathcal{Q}^{(k)}$, we propose an adaptive dense reinforcement learning method, which iteratively learns both the environment dynamics—specifically, the dynamics of the AV under test—through policy functions $\pi^{(k)}$ and state-action value functions (i.e., maneuver challenges) $\mathcal{Q}^{(k)}$. Here, $\mathcal{Q}^{(k)} := \mathbb{E}_{(\phi, \pi^{(k)})}[R_{t:T}|\mathcal{S}_t, \mathbf{A}_t]$ represents the optimal maneuver challenges under the AV policies $\pi^{(k)}$. To capture the dynamics $\pi^{(k)}$, we train a deep neural network using the dynamics data collected during testing, referred to as the neural dynamics model (NeuDyM). The training of NeuDyM faces the CoR, as safety-critical dynamics data is so rare that ordinary training methods (i.e., using all dynamics data) fail to capture meaningful information about safety-critical maneuvers. To resolve this challenge, we propose to use exclusively the safety-critical dynamics data to train NeuDyM. Specifically, the gradient

of the loss function \mathcal{L} with respect to the NeuDyM parameters θ is

$$\tilde{g} := \frac{1}{B} \sum_{i=1}^B \frac{\partial \mathcal{L}(\mathcal{S}_i, \mathbf{A}_i^{\text{AV}})}{\partial \theta} \Big|_{\mathcal{S}_c}(\mathcal{S}_i), \quad (11)$$

where $B \in \mathbb{N}_{>0}$ is the batch size, and $\mathbf{A}_i^{\text{AV}} : \mathbf{x} \mapsto \mathbf{a}^{\text{AV}} \in \mathcal{A}^{\text{AV}}$ is the action random variable of AV. By focusing solely on safety-critical dynamics data, the variance in gradient estimation for the loss function is significantly reduced, enabling NeuDyM to effectively learn AV's safety-critical maneuvers.

We then utilize the NeuDyM policies $\pi^{(k)}$ to learn the maneuver challenges $Q^{(k)}$ through dense reinforcement learning (Feng et al., 2023; Yang et al., 2024). The process begins by initializing $\hat{Q}(s, \mathbf{a}) = 0$ for all $s \in \mathcal{S}$ and $\mathbf{a} \in \mathcal{A}$. In each training iteration, the initial state is uniformly sampled from \mathcal{S}_c , after which the BVs follow the uniform behavior policy while the AV follows the NeuDyM policy $\pi^{(k)}$. In dense reinforcement learning, the update rule for \hat{Q} is

$$\hat{Q}(\mathcal{S}_t, \mathbf{A}_t) \leftarrow \hat{Q}(\mathcal{S}_t, \mathbf{A}_t) + v_t \delta_t \Big|_{\mathcal{S}_c}(\mathcal{S}_t), \quad (12)$$

where $\delta_t := R_{t+1} + \gamma \mathbb{E}_{\phi}[\hat{Q}(\mathcal{S}_{t+1}, \mathbf{A}_{t+1}) | \mathcal{S}_{t+1}] - \hat{Q}(\mathcal{S}_t, \mathbf{A}_t)$ is the temporal difference error, and v_t is the learning rate. Under mild assumptions, dense reinforcement learning is guaranteed to converge almost surely to $Q^{(k)}$ (see Theorem 1 in Yang et al. (2024)). However, the learned $Q^{(k)}$ may not generalize well to out-of-distribution state-action pairs. Specifically, the $Q^{(k)}$ values are typically more accurate within a subspace of the entire state-action space—primarily the space covered by the safety-critical dynamics data in $\mathcal{D}^{(k-1)}$ —while values in other subspaces may degrade, potentially becoming less accurate than before. As a result, directly applying the importance function derived from $Q^{(k)}$ for AV testing and evaluation can be risky. We address this issue in the following subsection.

3.2. Combination coefficient optimization

To improve the evaluation robustness of $Q^{(k)}$, we propose using a convex combination of multiple maneuver challenges Q_j for $j = 1, \dots, J$, pre-trained with diverse SMs, to create a constrained optimization space. The combination coefficients are then optimized by solving the following regression problem:

$$\begin{aligned} \min_{\alpha \in \mathbb{R}^J} \quad & \frac{1}{2} \sum_{(s, \mathbf{a}) \in \mathcal{S}_c \times \mathcal{A}} [Q^{(k)}(s, \mathbf{a}) - Q_{\alpha}(s, \mathbf{a})]^2 \\ \text{s.t.} \quad & \mathbf{1}^T \alpha = 1, \alpha \geq \mathbf{0}, \end{aligned} \quad (13)$$

where $Q_{\alpha} := \sum_{j=1}^J \alpha_j Q_j$ is the α -combination of Q_j , and $\alpha = [\alpha_1, \dots, \alpha_J]^T$ is the vector of combination coefficients. This regression problem is a quadratic programming and can be solved using standard convex optimization tools such as CVXOPT (Andersen et al., 2004). Let $\alpha^{(k)}$ represent the solution to Eq. (13). The importance functions $q^{(k)}$ can then be derived from Eqs. (6) and (7) using $Q_{\alpha^{(k)}}$ in place of Q . By iteratively updating the importance functions based on testing results and sampling new testing scenarios from the updated importance functions to gather more results, we establish the adaptive testing process, except for crash rate estimation, which is elaborated in the next subsection.

3.3. Adaptive importance sampling

To estimate the crash rate based on the testing results obtained from different importance functions, we propose applying adaptive importance sampling techniques (Bugallo et al., 2017) to aggregate these results effectively. The crash rate can be estimated using adaptive importance sampling as

$$\tilde{\mu}^{(k)} = \frac{1}{n^{(k)}} \sum_{\kappa=1}^k \sum_{i=1}^{n_{\kappa}} \mathbb{I}_F(\mathbf{X}_{\kappa, i}) W^{(k)}(\mathbf{X}_{\kappa, i}), \quad \mathbf{X}_{\kappa, i} \sim q^{(k)}, \quad (14)$$

where $n^{(k)} := \sum_{\kappa=1}^k n_{\kappa}$ is the total number of tests conducted, n_{κ} is the number of testing scenarios sampled from $q^{(k)}$, and $W^{(k)} := p/q^{(k)}$, for $\kappa = 1, \dots, k$ and $k = 1, 2, \dots$. It is worth noting that various adaptive importance sampling methods are available (see Bugallo et al. (2017) and references therein). We use Eq. (14) due to its simplicity, effectiveness, and because it provides theoretical guarantees of consistency and approximate normality—properties often lacking in most adaptive importance sampling approaches (Bugallo et al., 2017). In Section 4, we will show that our method ensures both consistency and approximate normality under mild assumptions.

Algorithm 1: Adaptive testing with adaptive importance sampling

Input: naturalistic distribution p , surrogate maneuver challenges $Q_j, j = 1, \dots, J$, number of testing scenarios n_k at each update step k

Output: crash rate estimate

```

1 Initialize  $Q_{\alpha^{(1)}} = (1/J) \sum_{j=1}^J Q_j, \ell = 1, \ell_{th} = 0.3, k = 1;$ 
2 while  $\ell > \ell_{th}$  do
3   Compute  $q^{(k)}$  based on Eqs. (6) and (7) with  $Q_{\alpha^{(k)}}$  for  $Q$ ;
4   Sample  $n_k$  testing scenarios from  $q^{(k)}$ ;
5   Estimate crash rate  $\tilde{\mu}^{(k)}$  using adaptive importance sampling via Eq. (14);
6   Set  $\ell \leftarrow$  relative half-width of  $\tilde{\mu}^{(k)}$ ;
7   Set  $\mathcal{D}^{(k)} \leftarrow$  testing scenarios and results up to now;
8   Set  $k \leftarrow k + 1;$ 
9   Train NeuDyM policy  $\pi^{(k)}$  based on Eq. (11);
10  Apply dense reinforcement learning to learn  $Q^{(k)}$  using policy  $\pi^{(k)}$ ;
11  Solve the quadratic programming in Eq. (13) to update  $\alpha^{(k)}$  (e.g., via CVXOPT (Andersen et al., 2004));
12 end
13 Return  $\tilde{\mu}^{(k)}$ ;
```

3.4. Adaptive testing algorithm

The adaptive testing process is summarized in Algorithm 1. The pipeline for update step k (where $k > 1$) is

$$\mathcal{D}^{(k-1)} \xrightarrow{(11)} \pi^{(k)} \xrightarrow{(12)} Q^{(k)} \xrightarrow{(13)} Q_{\alpha^{(k)}} \xrightarrow{(6) \text{ and } (7)} q^{(k)} \xrightarrow{(14)} (\mathcal{D}^{(k)}, \tilde{\mu}^{(k)}). \quad (15)$$

The core idea is to train the NeuDyM policies $\pi^{(k)}$ using only safety-critical dynamics data and apply dense reinforcement learning to learn the maneuver challenges $Q^{(k)}$. Next, the combination coefficients $\alpha^{(k)}$ are optimized through solving a quadratic programming problem. Following this, the importance functions $q^{(k)}$ are updated, and testing scenarios are sampled from $q^{(k)}$. Then the crash rate is estimated using adaptive importance sampling. This iterative process continues until the termination criteria are met. We use the relative half-width (RHW) (Zhao et al., 2016) as the stopping criterion, with the threshold set at 0.3.

4. Theoretical analysis

This section presents a theoretical analysis of the proposed adaptive testing method, covering the convergence analysis of adaptive dense reinforcement learning in Subsection 4.1, the consistency analysis of adaptive testing in Subsection 4.2, and the efficiency analysis of adaptive testing in Subsection 4.3.

4.1. Convergence analysis

First, we prove the convergence of adaptive dense reinforcement learning.

Assumption 1. *The NeuDyM policies $\pi^{(k)}$ converge to $\pi^\dagger \in \Pi$ almost surely, i.e., $\mathbb{P}_p(\lim_{k \rightarrow \infty} \pi^{(k)} = \pi^\dagger) = 1$, denoted as $\pi^{(k)} \xrightarrow{a.s.} \pi^\dagger$.*

Assumption 2. *The assumptions of Theorem 1 in Yang et al. (2024) hold for all $\pi^{(k)}$, $k = 1, 2, \dots$.*

Theorem 1. *Suppose that Assumptions 1 and 2 hold, then $Q^{(k)} \xrightarrow{a.s.} Q^\dagger$, where $Q^\dagger := \mathbb{E}_{(\phi, \pi^\dagger)}[R_{t:T} | \mathcal{S}_t, \mathbf{A}_t]$ and t is any time step.*

PROOF. From Assumption 1 we have $\pi^{(k)} \xrightarrow{a.s.} \pi^\dagger$. Under Assumption 2, the dense reinforcement learning algorithm will converge to $Q^{(k)}$ for any NeuDyM policy $\pi^{(k)}$. Then, by the Lebesgue dominated convergence theorem, it follows that $Q^{(k)} \xrightarrow{a.s.} Q^\dagger$. \square

Corollary 1. If $\pi^{(k)} \xrightarrow{a.s.} \pi$ and Assumption 2 holds, then $Q^{(k)} \xrightarrow{a.s.} Q^*$.

PROOF. This follows directly from Theorem 1. \square

Remark 1. As shown in Theorem 1, if NeuDyM policies $\pi^{(k)}$ can converge to some policy π^\dagger (not necessarily optimal) almost surely and Assumption 2 holds, then $Q^{(k)}$ will converge to Q^\dagger almost surely. Assumption 2 is necessary to ensure the convergence of dense reinforcement learning. Since NeuDyM is a deep neural network, it can approximate AV's driving policy π well given infinite training data, though convergence to π cannot be guaranteed. Nonetheless, Corollary 1 shows that if $\pi^{(k)}$ converges to π almost surely and Assumption 2 holds, then $Q^{(k)}$ will converge to Q^* almost surely.

4.2. Consistency analysis

Next, we prove the consistency of the proposed adaptive testing method, i.e., $\tilde{\mu}^{(k)} \xrightarrow{a.s.} \mu$ as $k \rightarrow \infty$. The proof is based on Lemma 1 (Oh and Berger, 1992). To elaborate on this lemma, we define some notations used in Oh and Berger (1992). Suppose the goal is to estimate $\eta = \mathbb{E}_f[\varphi]$, where f is a probability distribution. Denote the parametric family of importance functions as $\mathcal{G} := \{g_\lambda : \lambda \in \Lambda\}$, where Λ is the parameter space. At each update step $k = 1, 2, \dots$, denote the parameter as $\lambda^{(k)}$ and the corresponding importance function as $g^{(k)} := g_{\lambda^{(k)}}$. The estimate for η is given by

$$\hat{\eta}^{(k)} = \frac{1}{n^{(k)}} \sum_{\kappa=1}^k \sum_{i=1}^{n_\kappa} \frac{\varphi(\mathbf{X}_{\kappa,i}) f(\mathbf{X}_{\kappa,i})}{g^{(\kappa)}(\mathbf{X}_{\kappa,i})}, \quad \mathbf{X}_{\kappa,i} \sim g^{(\kappa)}. \quad (16)$$

Assumption 3. The importance functions $g^{(k)}$, $k = 1, 2, \dots$, have the same support as f .

Assumption 4. The expectation $\eta = \mathbb{E}_f[\varphi]$ exists.

Assumption 5. The importance weights $f/g^{(k)}$ are bounded for all $k = 1, 2, \dots$.

Lemma 1. Suppose that Assumptions 3, 4 and 5 hold and φ has finite second moment, then $\hat{\eta}^{(k)} \xrightarrow{a.s.} \eta$.

PROOF. This is the Theorem 3.1 in Oh and Berger (1992). \square

Theorem 2. Let $\tilde{\mu}^{(k)}$ be given by Eq. (14). Then we have $\tilde{\mu}^{(k)} \xrightarrow{a.s.} \mu$.

PROOF. The correspondence to Lemma 1 is established by associating φ with $\mathbb{1}_F$, f with p , and g with q . We then verify Assumptions 3, 4 and 5 accordingly.

- (1) Verification of Assumption 3. From Eq. (7), we know that $\psi^{(k)}(\mathbf{a}|\mathbf{s}) > 0$ whenever $\phi(\mathbf{a}|\mathbf{s}) > 0$ for all $k = 1, 2, \dots$. Therefore, $q^{(k)}(\mathbf{x}) > 0$ whenever $p(\mathbf{x}) > 0$ for all $k = 1, 2, \dots$, meaning that all $q^{(k)}$ share the same support as p .
- (2) Verification of Assumption 4. Clearly, $\eta = \mathbb{E}_f[\varphi] = \mathbb{E}_p[\mathbb{1}_F] = \mu$ exists.
- (3) Verification of Assumption 5. The importance policy weight for any critical state $\mathbf{s} \in \mathcal{S}_c$ is

$$\begin{aligned} w(\mathbf{a}|\mathbf{s}) &= \frac{\phi(\mathbf{a}|\mathbf{s})}{\epsilon \phi(\mathbf{a}|\mathbf{s}) + (1-\epsilon) \frac{Q(\mathbf{s}, \mathbf{a}) \phi(\mathbf{a}|\mathbf{s})}{V(\mathbf{s})}} \\ &\leq \frac{\phi(\mathbf{a}|\mathbf{s})}{\epsilon \phi(\mathbf{a}|\mathbf{s})} \\ &= \frac{1}{\epsilon}, \quad \forall \mathbf{a} \in \mathcal{A}. \end{aligned} \quad (17)$$

Thus, the importance weight for all $\mathbf{x} \in \mathcal{X}$ is given by

$$W(\mathbf{x}) = \prod_{t \in \mathcal{T}_c} w(\mathbf{a}_t|\mathbf{s}_t) \leq \frac{1}{\epsilon^L}, \quad (18)$$

where L is the maximum number of critical time steps. Therefore, the importance weights $W^{(k)}$ are bounded by $1/\epsilon^L$ for all $k = 1, 2, \dots$.

Note that $(\|_F)^2 = \|_F$, which implies that $\|_F$ has finite second moment. Thus, the theorem follows. \square

Remark 2. As shown in Theorem 2, our adaptive testing method converges to the true crash rate almost surely. This is achieved primarily by employing the defensive importance sampling paradigm in Eq. (7), where we incorporate the naturalistic policy ϕ with probability ϵ , which ensures that Assumptions 3 and 5 are satisfied. Furthermore, since $\eta = \mathbb{E}_f[\varphi] = \mathbb{E}_p[\|_F] = \mu$ represents the crash rate we aim to estimate, Assumption 4 is clearly met.

4.3. Efficiency analysis

Finally, we analyze the efficiency of the adaptive testing method. In addition to Assumptions 3, 4 and 5, we also require the following two additional assumptions.

Assumption 6. The parameters $\lambda^{(k)}$ converge to $\lambda^\dagger \in \Lambda$ almost surely, i.e., $\lambda^{(k)} \xrightarrow{a.s.} \lambda^\dagger$.

Assumption 7. Given Assumption 6, the importance functions $g^{(k)}$ converge to g_{λ^\dagger} almost surely, i.e., $g^{(k)} \xrightarrow{a.s.} g_{\lambda^\dagger}$.

Lemma 2. Suppose that Assumptions 3, 4, 5, 6 and 7 hold, φ has finite fourth moment, and $\sum_{\kappa=1}^k n_\kappa^2 / (n^{(k)})^2 \rightarrow 0$ as $k \rightarrow \infty$. Then

$$\sqrt{n^{(k)}} (\hat{\eta}^{(k)} - \eta) \xrightarrow{d} \mathcal{N} \left(0, \sigma_{g_{\lambda^\dagger}}^2 \right), \quad (19)$$

where \xrightarrow{d} denotes convergence in distribution, and $\sigma_{g_{\lambda^\dagger}}^2 := \text{Var}_{g_{\lambda^\dagger}}(\varphi f / g_{\lambda^\dagger})$.

PROOF. This is the Theorem 3.2 in Oh and Berger (1992). \square

Theorem 3. Let $\tilde{\mu}^{(k)}$ be given by Eq. (14). Suppose that Assumptions 1 and 2 hold, and $\sum_{\kappa=1}^k n_\kappa^2 / (n^{(k)})^2 \rightarrow 0$ as $k \rightarrow \infty$. Then

$$\sqrt{n^{(k)}} (\tilde{\mu}^{(k)} - \mu) \xrightarrow{d} \mathcal{N} \left(0, \sigma_{q^\dagger}^2 \right), \quad (20)$$

where $\sigma_{q^\dagger}^2 := \text{Var}_{q^\dagger}(\|_F p / q^\dagger)$ and q^\dagger is given by Eqs. (6) and (7) with Q^\dagger in place of Q .

PROOF. The correspondence to Lemma 2 follows from associating φ with $\|_F$, f with p , g with q , and λ with Q . In the proof of Theorem 2, the Assumptions 3, 4 and 5 have already been verified. Therefore, we only need to verify the remaining Assumptions 6 and 7.

- (1) Verification of Assumption 6. Given Assumptions 1 and 2, Theorem 1 establishes that $Q^{(k)} \xrightarrow{a.s.} Q^\dagger$. Hence, Assumption 6 holds.
- (2) Verification of Assumption 7. Since $Q^{(k)} \xrightarrow{a.s.} Q^\dagger$, by the Lebesgue dominated convergence theorem, we have $q^{(k)} \xrightarrow{a.s.} q^\dagger$.

Since $(\|_F)^4 = \|_F$, it is clear that $\|_F$ has finite fourth moment, which concludes the proof of this theorem. \square

Remark 3. Note that if $n_k \leq N$ for some $N > 0$ and for all $k = 1, 2, \dots$, then we have

$$\sum_{\kappa=1}^k \left(\frac{n_\kappa}{n^{(k)}} \right)^2 \leq k \left(\frac{N}{k} \right)^2 = \frac{N^2}{k} \rightarrow 0, \quad k \rightarrow \infty. \quad (21)$$

This can be easily achieved if n_k are the same for all $k = 1, 2, \dots$. Theorem 3 shows that if $q^{(k)}$ converge almost surely to an importance function q^\dagger (not necessarily optimal), then the distributions of $\tilde{\mu}^{(k)}$ converge to $\mathcal{N} \left(\mu, \sigma_{q^\dagger}^2 / n^{(k)} \right)$. This implies that as the importance functions are iteratively optimized, the asymptotic estimation variances of crash rates decrease and eventually converge to $\sigma_{q^\dagger}^2$. The closer the importance functions are optimized toward q^* , the more the estimation variance is reduced, and the greater the improvement in estimation efficiency.

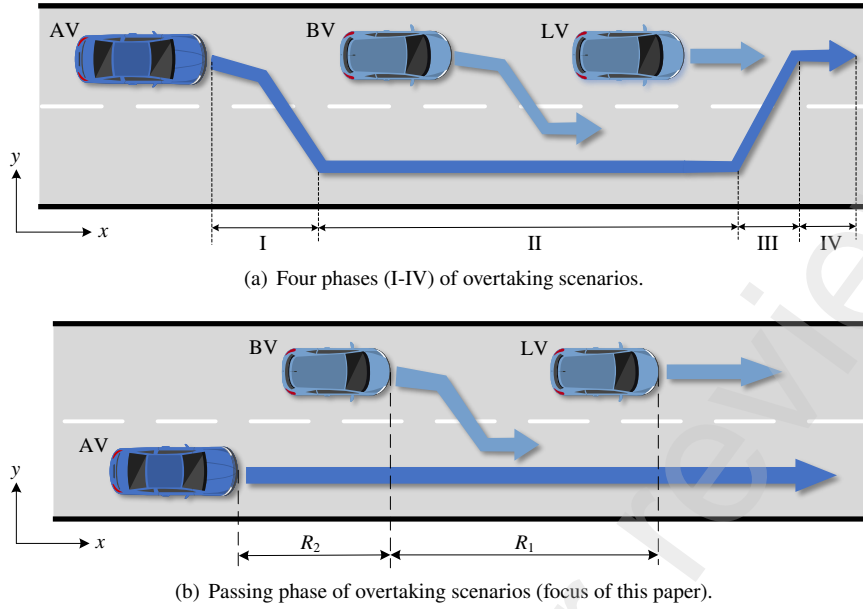


Figure 3: Illustrations of (a) the four phases of overtaking scenarios and (b) the passing phase (Phase II). In overtaking scenarios, the AV overtakes both BV and LV. While AV is engaged in passing, BV may overtake LV.

5. Results

This section begins by detailing the overtaking scenarios in Subsection 5.1. Following that, Subsection 5.2 presents and analyzes the testing and evaluation results across NDE, NADE, and adaptive testing.

5.1. Overtaking scenarios

As illustrated in Fig. 3, we focus on the passing phase of overtaking scenarios, where a slower-moving lead vehicle (LV) travels ahead of the background vehicle (BV), and the AV is attempting to overtake both BV and LV. During this process, BV may also attempt to overtake LV, which could lead to a rear-end crash between AV and BV. The state of the overtaking scenario is defined as $s := [v_{BV}, R_1, \dot{R}_1, R_2, \dot{R}_2]^T$, where $R_1 := x_{LV} - x_{BV}$, $\dot{R}_1 := v_{LV} - v_{BV}$, $R_2 := x_{BV} - x_{AV}$, and $\dot{R}_2 := v_{BV} - v_{AV}$. The action is defined as the accelerations of LV and BV, $a := [a_{LV}, a_{BV}]^T$. Here, x , v and a refer to the longitudinal position, velocity, and acceleration, respectively, with the subscripts corresponding to each specific vehicle. The simulation runs for a maximum of 20 seconds with a time resolution of 0.1 seconds. Typically, overtaking scenarios involve more than 1,400 dimensions (201 time steps, each with 5 state variables and 2 action variables), presenting the high-dimensionality challenge.

5.2. Testing and evaluation results

In this subsection, we analyze the testing and evaluation results of NDE, NADE and adaptive testing. The generation of NDE and NADE adheres to the methodology outlined in Feng et al. (2021)¹. To show the generalizability of our method, we select three distinct AVs:

- (1) AV-I: the intelligent driver model (Ro et al., 2017);
- (2) AV-II: the intelligent driver model calibrated in Sangster et al. (2013);
- (3) AV-III: the agent trained by proximal policy optimization (Schulman et al., 2017).

We use three archetypes of SMs that capture a range of driving behaviors:

- (1) SM-I: the intelligent driver model (same as AV-I);
- (2) SM-II: the full velocity difference model (Ro et al., 2017) with $a_{\min} = -1 \text{ m/s}^2$;
- (3) SM-III: the full velocity difference model with $a_{\min} = -6 \text{ m/s}^2$.

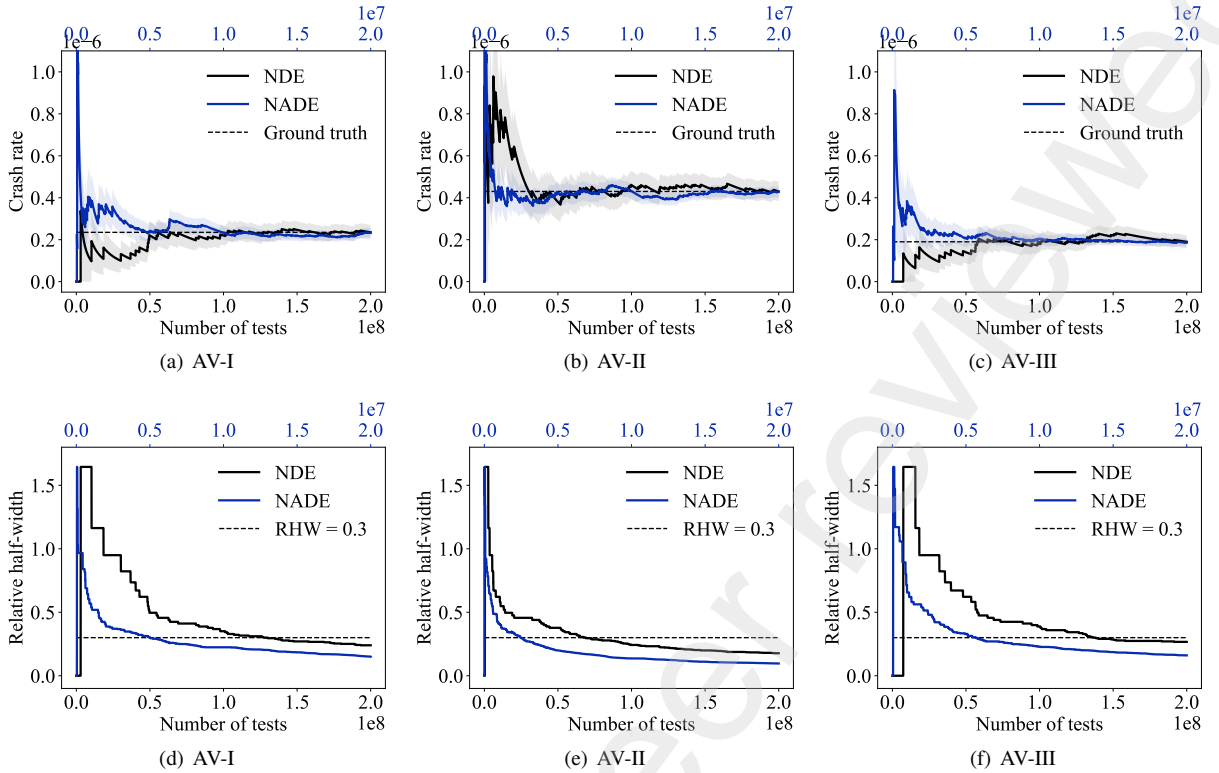


Figure 4: The crash rate estimations for (a) AV-I, (b) AV-II, and (c) AV-III in NDE and NADE, and corresponding RHWs for (d) AV-I, (e) AV-II, and (f) AV-III.

To enhance the robustness of crash rate estimation in NADE, we use three maneuver challenges, Q_1 , Q_2 , and Q_3 , pre-trained with SM-I, SM-II, and SM-III, respectively, with equal combination coefficients to establish the importance function. Specifically, the importance function is given by Eqs. (6) and (7) with $(Q_1 + Q_2 + Q_3)/3$ in place of Q . Fig. 4 displays the crash rate estimations and the corresponding RHWs for AV-I, AV-II, and AV-III in NDE and NADE. It can be seen that, across all three AVs, NADE converges to the same crash rate estimate as NDE, while requiring far fewer tests to reach the 0.3 RHW threshold. Although using multiple pre-trained maneuver challenges with average combination coefficients can enhance the evaluation robustness of NADE, it may reduce evaluation efficiency since such a configuration is not tailored for any specific AV under test.

To tackle this challenge, we continuously optimize the importance functions through the adaptive testing process. The number of tests is set to $n^{(k)} = 10^5$ for each update step $k = 1, 2, \dots$. The NeuDyM is implemented as a multilayer perceptron (MLP) with three hidden layers, each containing 256 neurons. We employ the Adam optimizer and the mean squared error (MSE) loss function, using the default hyperparameters provided in PyTorch 1.13.1 (Paszke et al., 2019), which are widely adopted in practice. Fig. 5 illustrates the MSEs in the NeuDyM training process. While the MSEs for both the training and testing datasets show a downward trend, only our learning method manages to reduce the MSE in safety-critical states, whereas the ordinary learning method fails to do so, with its MSE actually increasing. Based on the learned NeuDyM policies, the maneuver challenges are then learned using dense reinforcement learning, and the combination coefficients are optimized accordingly. Figs. 6(a)-(c) reveal that the combination coefficients are effectively optimized. Notably, at 10^7 tests, the optimized coefficients for AV-I, AV-II, and AV-III are $\alpha_{AV-I} = [0.94, 0.03, 0.03]^T$ (with the ground truth $\alpha_{AV-I}^* = [1, 0, 0]^T$), $\alpha_{AV-II} = [0.82, 0.17, 0.01]^T$, and $\alpha_{AV-III} = [0.64, 0.03, 0.33]^T$, respectively. As the combination coefficients are optimized, the importance functions are updated, and the testing results obtained from these importance functions are aggregated through adaptive importance sampling.

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

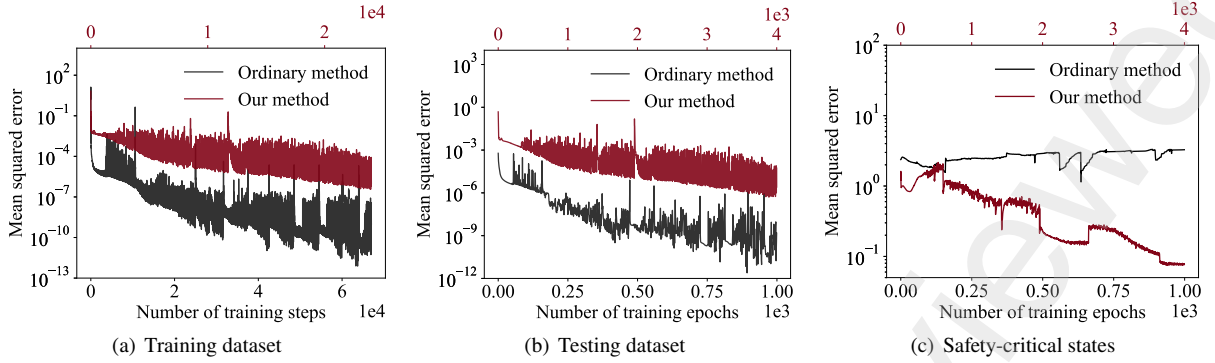


Figure 5: The (a) MSE on the training dataset, (b) MSE on the testing dataset and (c) MSE on safety-critical states throughout the NeuDyM training process. The black curves represent the MSEs from the ordinary learning method, which uses all dynamics data, while the red curves correspond to the MSEs from our method that learns with only safety-critical dynamics data.

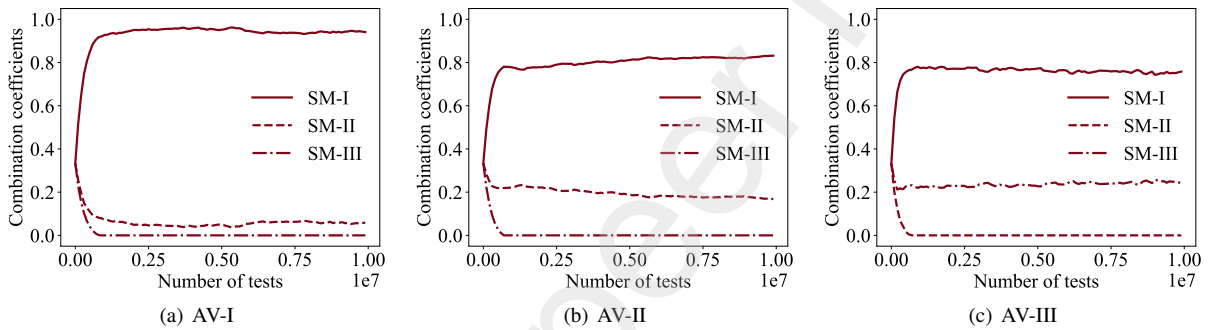


Figure 6: The combination coefficients optimized during adaptive testing for (a) AV-I, (b) AV-II, and (c) AV-III.

Table 3

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	1.32×10^8	7.20×10^7	1.59×10^8
NADE	5.11×10^6 (26)	2.09×10^6 (34)	5.63×10^6 (28)
Adaptive testing	2.75×10^6 (48)	1.49×10^6 (48)	3.68×10^6 (43)

To evaluate the performance of the adaptive testing method, we compare its results with those of NADE, as illustrated in Fig. 7. Figs. 7(a)-(c) show that adaptive testing produces the same crash rate estimates as NADE for all three AVs. However, as seen in Figs. 7(d)-(f), adaptive testing requires fewer tests than NADE to reach the 0.3 RHW threshold. To mitigate experimental stochasticity, we bootstrap the testing results by shuffling them 100 times. The frequency distributions of the required number of tests are shown in Figs. 7(g)-(i), respectively. Table 3 presents the average required number of tests and the average acceleration ratios (AARs) for NDE, NADE, and adaptive testing across the three AVs, where AARs (given in parentheses) represent the ratio of the average number of tests required by NADE and adaptive testing compared to NDE. Compared with NADE, adaptive testing reduces the required number of tests by 46.17%, 29.01%, and 34.67% for AV-I, AV-II, and AV-III, respectively, demonstrating its significant improvement in evaluation efficiency while maintaining robustness for diverse AVs.

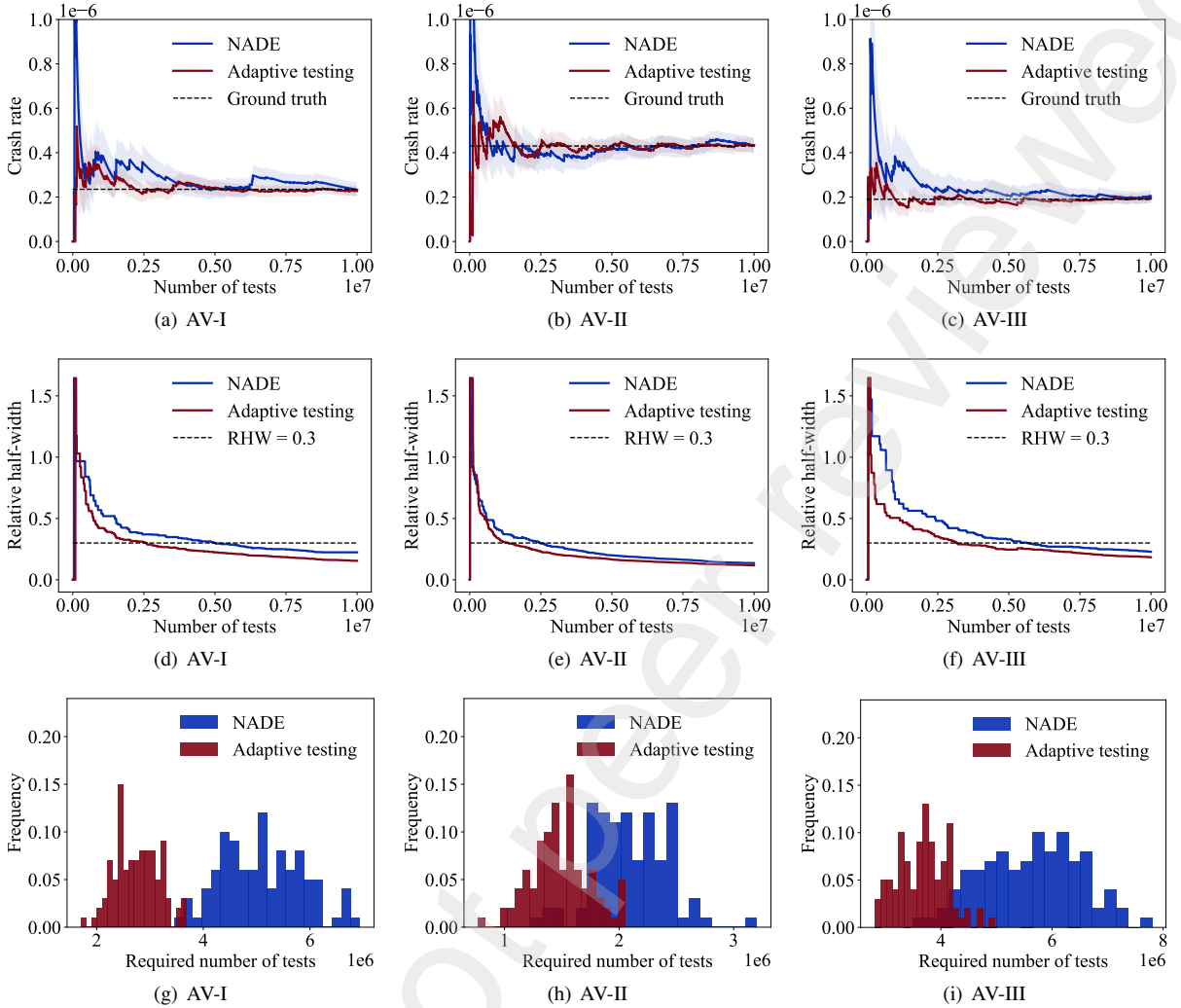


Figure 7: The crash rate estimations for (a) AV-I, (b) AV-II and (c) AV-III of NADE and adaptive testing, RHW of crash rate estimations for (d) AV-I, (e) AV-II and (f) AV-III, and frequency distributions of bootstrapped required number of tests for (g) AV-I, (h) AV-II and (i) AV-III.

6. Conclusion

This paper proposes an adaptive testing framework designed for continuous optimization of importance functions throughout the large-scale testing process. Our method centers on learning NeuDyM policies from exclusively the safety-critical dynamics data, then using dense reinforcement learning to optimize maneuver challenges based on the learned dynamics. The adaptive testing framework is further enhanced by combining multiple pre-trained maneuver challenges, optimizing their combination coefficients, and utilizing adaptive importance sampling techniques to assess testing results across different importance functions. Experimental results, demonstrated in overtaking scenarios, show that the proposed method significantly improves evaluation efficiency compared to both NDE and NADE. Future research will explore expanding this approach to driving environments with continuous state and action spaces. This work focuses primarily on the adaptive testing in the large-scale testing stage, highlighting the need for future research on integrating all three adaptive testing stages.

CRedit authorship contribution statement

Jingxuan Yang: Conceptualization, Formal analysis, Methodology, Software, Validation, Visualization, Writing - original draft, Writing - review and editing. **Zihang Wang:** Data curation, Software, Writing - review and editing. **Daihan Wang:** Investigation, Project administration, Resources, Writing - review and editing. **Yi Zhang:** Conceptualization, Funding acquisition, Supervision, Writing - review and editing. **Qiuqing Lu:** Conceptualization, Funding acquisition, Resources, Supervision. **Shuo Feng:** Conceptualization, Formal analysis, Funding acquisition, Methodology, Resources, Supervision, Writing - review and editing.

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.

Acknowledgements

This work is supported by National Natural Science Foundation of China No. 62473224.

References

- Andersen, M., Dahl, J., Vandenberghe, L., 2004. Cvxopt: Python software for convex optimization, version 1.3.2. Available at <https://cvxopt.org>.
- Au, S.K., Beck, J., 2003. Important sampling in high dimensions. *Structural safety* 25, 139–163.
- Bai, R., Yang, J., Gong, W., Zhang, Y., Lu, Q., Feng, S., 2024. Accurately predicting probabilities of safety-critical rare events for intelligent systems, in: 2024 IEEE 20th International Conference on Automation Science and Engineering (CASE), IEEE. pp. 1–7.
- Bugallo, M.F., Elvira, V., Martino, L., Luengo, D., Míguez, J., Djuric, P.M., 2017. Adaptive importance sampling: The past, the present, and the future. *IEEE Signal Processing Magazine* 34, 60–79.
- Feng, S., Feng, Y., Sun, H., Bao, S., Zhang, Y., Liu, H.X., 2020a. Testing scenario library generation for connected and automated vehicles, part ii: Case studies. *IEEE Transactions on Intelligent Transportation Systems* 22, 5635–5647.
- Feng, S., Feng, Y., Sun, H., Zhang, Y., Liu, H.X., 2020b. Testing scenario library generation for connected and automated vehicles: An adaptive framework. *IEEE Transactions on Intelligent Transportation Systems* 23, 1213–1222.
- Feng, S., Feng, Y., Yu, C., Zhang, Y., Liu, H.X., 2020c. Testing scenario library generation for connected and automated vehicles, part i: Methodology. *IEEE Transactions on Intelligent Transportation Systems* 22, 1573–1582.
- Feng, S., Sun, H., Yan, X., Zhu, H., Zou, Z., Shen, S., Liu, H.X., 2023. Dense reinforcement learning for safety validation of autonomous vehicles. *Nature* 615, 620–627.
- Feng, S., Yan, X., Sun, H., Feng, Y., Liu, H.X., 2021. Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment. *Nature Communications* 12, 1–14.
- Gong, X., Feng, S., Pan, Y., 2023. An adaptive multi-fidelity sampling framework for safety analysis of connected and automated vehicles. *IEEE Transactions on Intelligent Transportation Systems* 24, 14393–14405.
- He, H., Li, S., Yang, J., He, L., Lu, Q., Zhang, Y., Feng, S., 2024. Knowmoformer: Knowledge-conditioned motion transformer for controllable traffic scenario simulation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshop on Data-Driven Autonomous Driving Simulation, IEEE. pp. 1–8.
- Kalra, N., Paddock, S.M., 2016. Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? *Transportation Research Part A: Policy and Practice* 94, 182–193.
- Koren, M., Alsaif, S., Lee, R., Kochenderfer, M.J., 2018. Adaptive stress testing for autonomous vehicles, in: 2018 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 1–7.
- Li, A., Chen, S., Sun, L., Zheng, N., Tomizuka, M., Zhan, W., 2021. Scogene: Bio-inspired traffic scenario generation for autonomous driving testing. *IEEE Transactions on Intelligent Transportation Systems* 23, 14859–14874.
- Li, L., Huang, W.L., Liu, Y., Zheng, N.N., Wang, F.Y., 2016. Intelligence testing for autonomous vehicles: A new approach. *IEEE Transactions on Intelligent Vehicles* 1, 158–166.
- Li, L., Lin, Y.L., Zheng, N.N., Wang, F.Y., Liu, Y., Cao, D., Wang, K., Huang, W.L., 2018. Artificial intelligence test: A case study of intelligent vehicles. *Artificial Intelligence Review* 50, 441–465.
- Li, L., Wang, X., Wang, K., Lin, Y., Xin, J., Chen, L., Xu, L., Tian, B., Ai, Y., Wang, J., Cao, D., Liu, Y., Wang, C., Zheng, N., Wang, F.Y., 2019. Parallel testing of vehicle intelligence via virtual-real interaction. *Science Robotics* 4, eaaw4106.
- Li, L., Zheng, N., Wang, F.Y., 2020. A theoretical foundation of intelligence testing and its application for intelligent vehicles. *IEEE Transactions on Intelligent Transportation Systems* 22, 6297–6306.
- Li, S., He, H., Yang, J., Hu, J., Zhang, Y., Feng, S., 2024a. Few-shot testing of autonomous vehicles with scenario similarity learning. *arXiv preprint arXiv:2409.14369*.
- Li, S., Yang, J., He, H., Zhang, Y., Hu, J., Feng, S., 2024b. Few-shot scenario testing for autonomous vehicles based on neighborhood coverage and similarity, in: 2024 IEEE 35th Intelligent Vehicles Symposium (IV), IEEE. pp. 620–626.
- Liu, H.X., Feng, S., 2024. Curse of rarity for autonomous vehicles. *Nature Communications* 15, 4808.

- Menzel, T., Bagschik, G., Maurer, M., 2018. Scenarios for development, test and validation of automated vehicles, in: 2018 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 1821–1827.
- Mullins, G.E., Stankiewicz, P.G., Hawthorne, R.C., Gupta, S.K., 2018. Adaptive generation of challenging scenarios for testing and evaluation of autonomous vehicles. *Journal of Systems and Software* 137, 197–215.
- Oh, M.S., Berger, J.O., 1992. Adaptive importance sampling in monte carlo integration. *Journal of statistical computation and simulation* 41, 143–168.
- Owen, A.B., 2013. Monte Carlo theory, methods and examples. Stanford.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library, in: *Advances in Neural Information Processing Systems*, Curran Associates, Inc.. pp. 8024–8035.
- Rempe, D., Phillion, J., Guibas, L.J., Fidler, S., Litany, O., 2022. Generating useful accident-prone driving scenarios via a learned traffic prior, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17305–17315.
- Ren, K., Yang, J., Lu, Q., Zhang, Y., Hu, J., Feng, S., 2024. Intelligent testing environment generation for autonomous vehicles with implicit distributions of traffic behaviors. Available at SSRN 4973349 .
- Riedmaier, S., Ponn, T., Ludwig, D., Schick, B., Diermeyer, F., 2020. Survey on scenario-based safety assessment of automated vehicles. *IEEE access* 8, 87456–87477.
- Ro, J.W., Roop, P.S., Malik, A., Ranjitkar, P., 2017. A formal approach for modeling and simulation of human car-following behavior. *IEEE Transactions on Intelligent Transportation Systems* 19, 639–648.
- Sangster, J., Rakha, H., Du, J., 2013. Application of naturalistic driving data to modeling of driver car-following behavior. *Transportation research record* 2390, 20–33.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* .
- Snoek, J., Larochelle, H., Adams, R.P., 2012. Practical bayesian optimization of machine learning algorithms, in: *Neural Information Processing Systems*, pp. 2951–2959.
- Sun, H., Feng, S., Yan, X., Liu, H.X., 2021a. Corner case generation and analysis for safety assessment of autonomous vehicles. *Transportation research record* 2675, 587–600.
- Sun, J., Zhou, H., Xi, H., Zhang, H., Tian, Y., 2021b. Adaptive design of experiments for safety evaluation of automated vehicles. *IEEE Transactions on Intelligent Transportation Systems* 23, 14497–14508.
- Tian, Y., Pei, K., Jana, S., Ray, B., 2018. Deeptest: Automated testing of deep-neural-network-driven autonomous cars, in: *Proceedings of the 40th International Conference on Software Engineering*, pp. 303–314.
- Wang, J., Pun, A., Tu, J., Manivasagam, S., Sadat, A., Casas, S., Ren, M., Urtasun, R., 2021. Advsim: Generating safety-critical scenarios for self-driving vehicles, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9909–9918.
- Yan, X., Zou, Z., Feng, S., Zhu, H., Sun, H., Liu, H.X., 2023. Learning naturalistic driving environment with statistical realism. *Nature Communications* 14, 2037.
- Yang, J., Bai, R., Ji, H., Zhang, Y., Hu, J., Feng, S., 2024. Adaptive testing environment generation for connected and automated vehicles with dense reinforcement learning. *arXiv preprint arXiv:2402.19275* .
- Yang, J., He, H., Zhang, Y., Feng, S., Liu, H.X., 2022. Adaptive testing for connected and automated vehicles with sparse control variates in overtaking scenarios, in: *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE. pp. 2791–2797.
- Yang, J., Sun, H., He, H., Zhang, Y., Liu, H.X., Feng, S., 2023. Adaptive safety evaluation for connected and automated vehicles with sparse control variates. *IEEE Transactions on Intelligent Transportation Systems* 25, 1761–1773.
- Zhao, D., Huang, X., Peng, H., Lam, H., LeBlanc, D.J., 2017. Accelerated evaluation of automated vehicles in car-following maneuvers. *IEEE Transactions on Intelligent Transportation Systems* 19, 733–744.
- Zhao, D., Lam, H., Peng, H., Bao, S., LeBlanc, D.J., Nobukawa, K., Pan, C.S., 2016. Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques. *IEEE Transactions on Intelligent Transportation Systems* 18, 595–607.