Highlights

Intelligent testing environment generation for autonomous vehicles with implicit distributions of traffic behaviors

Kun Ren, Jingxuan Yang, Qiujing Lu, Yi Zhang, Jianming Hu, Shuo Feng

- Autonomous Vehicle (AV) testing faces challenges due to complex, high-dimensional environments and the black-box nature of AV models.
- Traditional methods, including naturalistic driving Environments and adversarial attacks, offer limited efficiency and scenario diversity.
- Importance sampling addresses the inefficiency challenge but struggles with implicit distributions of traffic behaviors in complex environments.
- We propose the Implicit Importance Sampling approach that generates intelligent testing environment under implicit distributions of traffic behaviors, improving testing efficiency.
- IIS significantly accelerates AV testing and maintains reliable evaluation results with controlled bias, as demonstrated by experiments on diverse NDE models.

Intelligent testing environment generation for autonomous vehicles with implicit distributions of traffic behaviors*

Kun Ren[a], Jingxuan Yang[a], Qiujing Lu[a], Yi Zhang[a], Jianming Hu[a,*] and Shuo Feng[a,*]

[a]Department of Automation, Tsinghua University, 100084, Beijing, China

ARTICLE INFO

Keywords:
Autonomous vehicles
Importance sampling
Accelerated testing

ABSTRACT

The advancement of autonomous vehicles hinges significantly on addressing safety concerns and obtaining reliable evaluation results. Testing the safety of autonomous vehicles is challenging due to the complexity of the high-dimensional traffic environment and the rarity of safety-critical events, often requiring billions of miles to achieve comprehensive validation, which is inefficient and costly. Current approaches, such as accelerated testing using importance sampling, aim to provide unbiased estimates of the performance of autonomous vehicles by generating a new distribution of background vehicles' behaviors based on an initial nominal distribution. However, these methods require knowledge of the original distribution of traffic behaviors, which is often difficult to obtain in practice. In response to these challenges, we introduce a novel methodology termed implicit importance sampling (IIS). Unlike traditional methods, IIS is designed to generate intelligent driving environments based on implicit distributions of traffic behaviors where the true distributions are unknown or not explicitly defined. IIS method leverages accept-reject sampling to construct an unnormalized proposal distribution, which increases the likelihood of sampling adversarial cases. Through applying importance sampling technique with unnormalized proposal distribution, IIS enhances testing efficiency and obtains reliable and representative evaluation results as well. The bias caused by unnormalization is also proved to be controlled and bounded.

1. Introduction

The development and deployment of autonomous vehicles (AVs) are expected to revolutionize transportation by enhancing safety and reducing traffic congestion. However, ensuring the safety and reliability of AVs remains a critical challenge due to several factors. First, the high-dimensionality, complexity, and stochastic nature of traffic environments can lead to the "curse of dimensionality" (Feng et al., 2021c), making it difficult to explicitly model a traffic environment. Second, the black-box nature of AV models makes their decision-making processes hard to predict and limits their ability to handle scenarios beyond their training experience (Filos et al., 2020). Third, the presence of long-tail events, which are rare but critical, plays a significant role, as these low-probability events are often the ones that can lead to accidents (Liu and Feng, 2024). These events are referred to as safety-critical cases (Ding et al., 2023) or corner cases (Sun et al., 2021). Traditional testing methods, which require AVs to drive billions of miles to encounter a wide range of scenarios, are prohibitively time-consuming and inefficient (Kalra and Paddock, 2016).

To address this problem, researchers have developed several advanced testing methods. The most commonly applied method is testing AVs in simulation of naturalistic driving environments (NDEs), which are often modeled through rules or naturalistic driving data (NDD) that sampled from the real world (Feng et al., 2021c; Yan et al., 2023). Many high-fidelity simulators integrate driving models as well, such as CARLA (Dosovitskiy et al., 2017), AirSim (Shah et al., 2018), AADS (Li et al., 2019), and SUMO (Krajzewicz, 2010). However, NDD and NDE alone are usually not sufficient to evaluate the performance of AVs due to the limitations in scenario diversity and the lack of rare, safety-critical events. Therefore, methods such as clustering (Kruber et al., 2018; Wang and Zhao, 2018; Sun et al., 2021) or random perturbation (Scanlon et al., 2021; Fang et al., 2020; Lu et al.; Liu and Feng, 2024) have been used to augment datasets to obtain more safety-critical data from NDE or NDD. While preserving the naturalism of generated scenarios, these methods face challenges in efficiency and diversity.

*Corresponding author
rk23@mails.tsinghua.edu.cn (K. Ren); yangjx@mails.tsinghua.edu.cn (J. Yang); qiujinglu@mails.tsinghua.edu.cn (Q. Lu); zhyi@mail.tsinghua.edu.cn (Y. Zhang); hujm@mail.tsinghua.edu.cn (J. Hu); fshuo@tsinghua.edu.cn (S. Feng)
ORCID(s): 0009-0005-4082-5220 (K. Ren); 0000-0001-9798-7347 (J. Yang); 0000-0002-3038-256X (Q. Lu); 0000-0001-5526-866X (Y. Zhang); 0000-0001-8065-7309 (J. Hu); 0000-0002-2117-4427 (S. Feng)

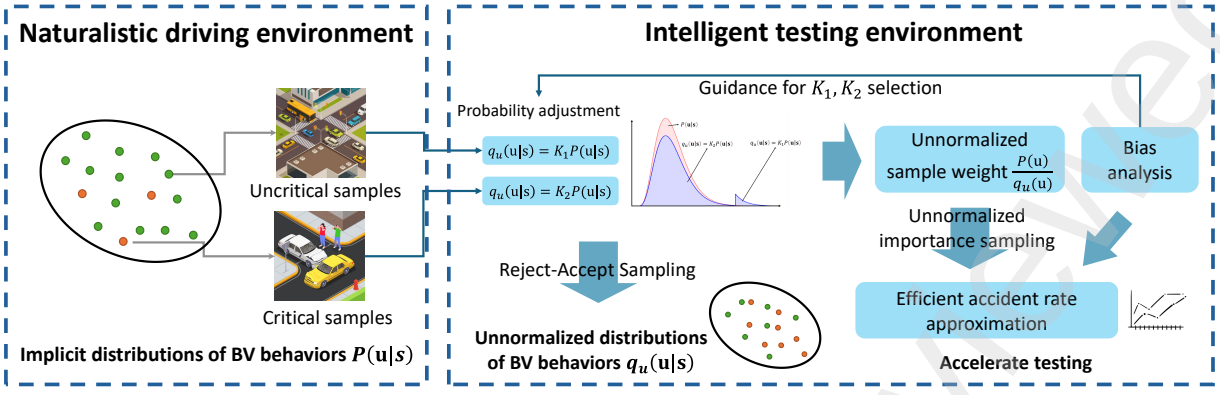


Figure 1: Overview of implicit importance sampling framework.

Adversarial attacks are another approach to generating data for AV testing. The main idea is to control the background vehicles (BVs) to challenge AVs purposely. Adaptive stress testing (Koren and Kochenderfer, 2019; Lee et al., 2020; Corso et al., 2019) aims to identify the most likely failure events for AVs by adaptively adjusting test conditions based on the AVs performance. This method uses reinforcement learning to explore the possible scenarios and focus on those that lead to AVs' failures. Many other RL-based methods have also been proposed to diversely search safety-critical traffic scenarios (Niu et al., 2023; Yang et al., 2024). Additionally, applying specific disturbances to BVs is also a feasible approach (Hanselmann et al., 2022; Rempe et al., 2022; Hao et al., 2023), in which techniques such as generative adversarial networks (GANs) (Goodfellow et al., 2014) and variational autoencoders (VAEs) (Kingma and Welling, 2013) have been employed to generate both naturalistic and adversarial driving scenarios. Furthermore, scenario library construction is also crucial. A recent study proposed a novel method, leveraging Genetic Algorithm (GA) to efficiently construct diverse and critical traffic scenarios, thus contributing to the development of a comprehensive scenario library (Zhao et al., 2023).

However, there is an expectation that we could efficiently evaluate the performance of AVs in contrast with human drivers, for example, the accident rate under the same NDE. The aforementioned methods cannot achieve this because only scenarios in a small area with a small number of vehicles can be generated instead of a continuous traffic flow. Recent studies have introduced innovative approaches to address these issues (Riedmaier et al., 2020; Nalic et al., 2020). For example, surrogate-based optimization methods leverage naturalistic driving data to create models that simulate real-world scenarios, improving risk assessment accuracy (Zhang et al., 2022, 2023). The splitting technique estimates small probability events by dividing the probability space into manageable segments, enhancing accuracy and efficiency in rare safety scenarios (Cancela et al., 2009). Importance sampling has shown significant potential for addressing the problem of rare events by focusing on generating critical cases that are more likely to lead to safety violations, thus reducing the number of miles needed for testing (Owen, 2013; Cancela et al., 2009; Morris et al., 1996). Works such as (Zhao et al., 2017; Feng et al., 2021c, 2023; Arief et al., 2022; Huang et al., 2019; Jiang et al., 2022) have demonstrated the capacity of importance sampling to provide unbiased safety estimates, using the NDE modeled as discrete distributions extracted from NDD. Feng et al. (2021c) introduced the concept of an intelligent testing environment, which utilizes importance sampling methods to create naturalistic and adversarial driving environment (NADE) for AV testing. This approach results in unbiased estimates of AV performance with greater efficiency than NDE. Building upon this foundation, Feng et al. (2023) further improve testing efficiency through using reinforcement learning to optimize the proposal distribution in importance sampling, thereby reducing sampling variance.

Despite the advantages of importance sampling, one of its key limitations is the requirement for a well-defined nominal distribution to compute the proposal distribution and associated weights. In practice, however, it is often challenging to obtain explicit models of the nominal distribution, especially in more complex NDEs where the distribution may be implicit. This issue arises because NDEs are usually difficult to model explicitly due to the high-dimensionality, complexity, and stochastic nature of traffic environments. In some simple cases, for example, a fixed road network, NDE can be modelled explicitly (Zhao et al., 2017; Feng et al., 2021c; Arief et al., 2022; Huang et al., 2019; Jiang et al., 2022). However, in most cases, NDEs are modelled using complex modesl, rules, or neural networks (Yan et al., 2023; Niu et al., 2023; Rempe et al., 2022). As a result, the exact probabilities of events or

behaviors are unknown, and we can only draw samples from the environments under an underlying yet unknown probability distribution. This challenge is not unique to autonomous driving but is common across various domains that involve testing of intelligent systems such as robotics.

In the domain of importance sampling, several approaches have been suggested to overcome this challenge, such as approximating the weights through convex quadratic optimization rather than relying on the exact likelihood ratio (O'Hagan, 1987; Henmi et al., 2007; Delyon and Portier, 2016; Liu and Lee, 2017; Oates et al., 2016). While These methods have the advantage of reducing the variance of importance sampling and producing unbiased estimates, they still depend on certain necessary information about the nominal distribution, such as the form of the distribution or the derivatives of the probability density function. Therefore, there is a need for more generalizable approaches that can handle the situations that only data can be sampled following the implicit distribution without any other information related to probability density. Due to the rarity of safety-critical events, it is also prohibitively inefficient to estimate the distribution by only sampling the data.

In response to these challenges, we introduce a novel approach called implicit importance sampling (IIS). Unlike traditional methods, IIS is designed to generate the intelligent testing envrionment under implicit nominal distributions. By identifying critical cases and leveraging accept-reject sampling (Neumann, 1951), the probability of sampling critical cases is increased. Our method constructs an unnormalized proposal distribution using accept-reject sampling, along with an associated unnormalized weight. This approach enables the efficient evaluation of AV performance, even in the absence of a fully specified nominal distribution. Although this introduces some bias due to unnormalization, we prove that this bias can be restricted to a controllable range, ensuring that the evaluation results remain reliable. This allows for significantly accelerating testing, reducing the required number of test miles by several orders of magnitude. Fig. 1 provides a schematic illustration of the proposed method.

Our contributions are threefold:

- We present a method capable of generating intelligent testing environments for NDEs with implicit distributions of traffic behaviors.
- We prove that our method facilitates accelerated testing with controllable bias, providing a reliable range for bias that ensures the reliability of evaluation results.
- We conducted experiments on two distinct NDE models to demonstrate the generalizability of our approach. The results demonstrated significant acceleration in testing while maintaining a controlled level of bias, thereby validating the effectiveness of IIS in diverse scenarios.

2. Preliminary work

This section provides an overview of existing methods related to NDE and NADE models, which are essential for understanding the subsequent development of our approach.

2.1. Naturalistic driving environment (NDE)

Our algorithm for generating NADE is based on existing NDE models for sampling. It is necessary to represent NDE first, as referred to in existing work (Feng et al., 2021c). NDE is represented by the combination of variables in a traffic scenario, which may include the position and velocity of vehicles and the parameters of road or weather. The NDE with N vehicles and T time steps can be represented as

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_{1,1} & \cdots & \mathbf{x}_{1,T} \\ \ddots & \cdots & \ddots \\ \mathbf{x}_{N,1} & \cdots & \mathbf{x}_{N,T} \end{bmatrix}, \mathbf{x} \in \mathbf{X}, \tag{1}$$

where $\mathbf{x}_{i,j}$ represents the variables of the i-th vehicle at the j-th time step. This representation results in an extremely high-dimensional variable. To handle the complexities associated with this high dimensionality, Markov Decision Process (MDP) is used to simplify the distribution of $\mathbf{x}$. In a dynamic traffic scenario, the state and action of the i-th vehicle at the j-th time step are written as $\mathbf{s}_i(j)$ and $\mathbf{u}_i(j)$ respectively. The state and action of all vehicles at the j-th time step are denoted as

$$\begin{aligned} \mathbf{s}(j) &= [\mathbf{s}_0(j), \mathbf{s}_1(j), \cdots, \mathbf{s}_N(j)], \\ \mathbf{u}(j) &= [\mathbf{u}_0(j), \mathbf{u}_1(j), \cdots, \mathbf{u}_N(j)]. \end{aligned} \tag{2}$$

So a scenario can be represented as a MDP (Puterman, 1990):

$$\mathbf{s}(0) \to \mathbf{u}(0) \to \mathbf{s}(1) \to \mathbf{u}(1) \cdots \to \mathbf{u}(T-1) \to \mathbf{s}(T). \tag{3}$$

The distribution $P(\mathbf{x})$ can be simplified as

$$P(\mathbf{x}) = P(\mathbf{s}(0)) \prod_{k=0}^{T} P(\mathbf{u}(k) | \mathbf{s}(k)) \mathcal{T}(\mathbf{s}(k+1) | \mathbf{s}(k), \mathbf{u}(k)), \tag{4}$$

where $\mathcal{T}(\cdot)$ is the state transition distribution.

Finally, our purpose is to evaluate the AV. To achieve this purpose, the performance of AV is measured by calculating accident rate

$$P(A) = \sum_{\mathbf{x} \in \mathbf{X}} P(A|\mathbf{x}) P(\mathbf{x}), \tag{5}$$

where A indicates the accident between AV and BVs. Given a driving environment $\mathbf{x}_i$, $P\left(A|\mathbf{x}_i\right)$ is estimated by counting the number of accident events occurring during the test.

2.2. Naturalistic and adversarial Driving Environment (NADE)

To obtain more accidents and accelerate the AV testing process, NADE has been proposed in (Feng et al., 2021c). The main idea is to increase the probability to sample $\mathbf{x}$ that may lead to accidents. A proposal distribution $q(\mathbf{x})$ is used to replace $P(\mathbf{x})$ to achieve this goal. And the key of this section is to design the distribution $q(\mathbf{x})$. The form of $q(\mathbf{x})$ resembles that of $P(\mathbf{x})$:

$$q(\mathbf{x}) = q(\mathbf{s}(0)) \prod_{k=0}^{T} q(\mathbf{u}(k) | \mathbf{s}(k)) \mathcal{T}(\mathbf{s}(k+1) | \mathbf{s}(k), \mathbf{u}(k)), \tag{6}$$

where $q(\mathbf{s}(0)) = P(\mathbf{s}(0))$.

To increase the likelihood of encountering accidents, we aim to sample $\mathbf{x}$ with a higher probability $p(\mathbf{x})$ when the conditional probability of an accident $P(A|\mathbf{x})$ is high, ideally close to 1. Accordingly, we aim to ensure that when $P(A|\mathbf{u}, \mathbf{s})$ is higher, the corresponding action $\mathbf{u}$ can be sampled with a higher probability, namely higher $q(\mathbf{u}|\mathbf{s})$.

So the first step in obtaining $q(\mathbf{x})$ is to calculate the criticality given state $\mathbf{s}$. The criticality (Feng et al., 2021a,b) is calculated as

$$C(\mathbf{s}) = \sum_{\mathbf{u}} P(A|\mathbf{u}, \mathbf{s}) P(\mathbf{u}|\mathbf{s}). \tag{7}$$

Then adjust distribution $P(\mathbf{u}|\mathbf{s})$ to obtain an expected $q(\mathbf{u}|\mathbf{s})$. Because $P(\mathbf{u}|\mathbf{s})$ is an explicit distribution whose probability can be obtained, $q(\mathbf{u}|\mathbf{s})$ can be designed by changing the value of probability directly according to the criticality $C(\mathbf{s})$.

2.3. AV performance evaluation in NADE

Accident rate is used to evaluate the performance of AV. The most direct method is testing an AV in NDE and estimating $P(A)$ by the Crude Monte Carlo (CMC) method (Mooney, 1997):

$$\begin{aligned} P(A) &= \mathbb{E}_P(P(A|\mathbf{x})) \\ &\approx \frac{1}{n} \sum_{i=1}^{n} P\left(A|\mathbf{x}_i\right) \approx \frac{m}{n}, \mathbf{x}_i \sim P(\mathbf{x}), \end{aligned} \tag{8}$$

where n is the number of tests, m is the number of accidents, and $\mathbf{x}_i \sim P(\mathbf{x})$ means that $\mathbf{x}_i$ is sampled from naturalistic distribution $P(\mathbf{x})$.

However, due to the extremely low accident rate in NDE, a huge number of tests are required to estimate $P(A)$. Based on NADE, $P(A)$ can also be estimated more efficiently through the importance sampling method:

$$\begin{aligned} P(A) &= \mathbb{E}_P\left(P(A|\mathbf{x})\right) \\ &= \mathbb{E}_q\left(P(A|\mathbf{x})\frac{P(\mathbf{x})}{q(\mathbf{x})}\right) \\ &\approx \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x}_i\right)\frac{P\left(\mathbf{x}_i\right)}{q\left(\mathbf{x}_i\right)}, \mathbf{x}_i \sim q\left(\mathbf{x}_i\right), \end{aligned} \tag{9}$$

where $\mathbf{x}_i \sim q\left(\mathbf{x}_i\right)$ means that $\mathbf{x}_i$ is sampled from naturalistic and adversarial distribution $q(\mathbf{x})$. Here proposal distribution $q(\mathbf{x})$ is referred to as the importance distribution. The ratio of $P(\mathbf{x})$ to $q(\mathbf{x})$ is called weight:

$$w(\mathbf{x}) = \frac{P(\mathbf{x})}{q(\mathbf{x})}. \tag{10}$$

$P(\mathbf{x})$ and $q(\mathbf{x})$ have been simplified through MDP, so equation (9) can also be simplified as

$$P(A) \approx \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x}_i\right)\prod_{k=1}^{T_i}\frac{P\left(\mathbf{u}(k)|\mathbf{s}(k)\right)}{q\left(\mathbf{u}(k)|\mathbf{s}(k)\right)}, \mathbf{x}_i \sim q\left(\mathbf{x}_i\right), \tag{11}$$

where T_i denotes the time-steps of the i-th test. While increasing the probability of sampling critical events, previous work has demonstrated that the estimated result in equation (9) is unbiased. According to importance sampling technique:

$$\mathbb{E}_q\left(\frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x}_i\right)\frac{P\left(\mathbf{x}_i\right)}{q\left(\mathbf{x}_i\right)}\right) = \mathbb{E}_q\left(P(A|\mathbf{x})\frac{P(\mathbf{x})}{q(\mathbf{x})}\right) = \mathbb{E}_P\left(P(A|\mathbf{x})\right). \tag{12}$$

So we can get an accurate testing result of AV performance in NADE.

3. Methodology

3.1. Generation of testing environment

It is easy to see that the preliminary method relies on explicit distribution $q(\mathbf{x})$, which may be limited in complex and high-dimensional scenarios. To address this, we introduce a method that operates with implicit distributions, allowing for greater flexibility and adaptability in generating NADE and accelerating testing.

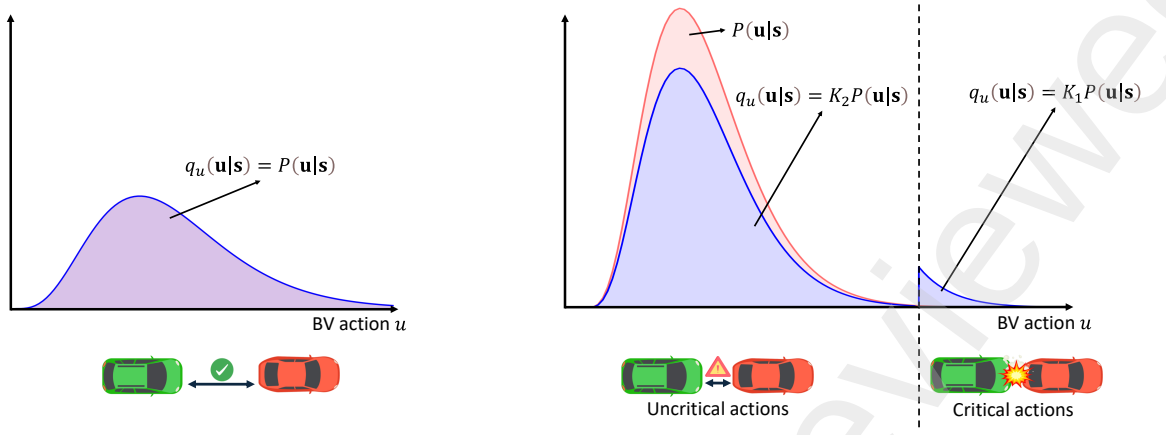
Because $P(\mathbf{u}|\mathbf{s})$ is an implicit distribution whose probability cannot be calculated, we design and sample from $q(\mathbf{u}|\mathbf{s})$ with the help of accept-reject sampling (Neumann, 1951). The distribution $q(\mathbf{u}|\mathbf{s})$ is designed as

$$\begin{aligned} q(\mathbf{u}|\mathbf{s}) &= \frac{q_u(\mathbf{u}|\mathbf{s})}{\int q_u(\mathbf{u}|\mathbf{s})\mathrm{d}\mathbf{u}}, \\ q_u(\mathbf{u}|\mathbf{s}) &= K(\mathbf{u},\mathbf{s})\,P(\mathbf{u}|\mathbf{s}), \\ K(\mathbf{u},\mathbf{s}) &= \begin{cases} K_0 = 1, \text{if } C(\mathbf{s}) = 0 \\ K_1 > 1, \text{if } C(\mathbf{s}) \neq 0 \text{ and } P(A|\mathbf{u},\mathbf{s}) > 0, \\ K_2 < 1, \text{if } C(\mathbf{s}) \neq 0 \text{ and } P(A|\mathbf{u},\mathbf{s}) = 0 \end{cases} \end{aligned} \tag{13}$$

where $q_u(\mathbf{u}|\mathbf{s})$ represents the unnormalized distribution because $\int K(\mathbf{u},\mathbf{s})\,P(\mathbf{u}|\mathbf{s})\mathrm{d}\mathbf{u}$ cannot be ensured to equal to 1. $C(\mathbf{s})$ denotes the criticality of state $\mathbf{s}$ and $C(\mathbf{s}) \neq 0$ means there is a possibility of an accident.

To sample from $q_u(\mathbf{u}|\mathbf{s})$ while $C(\mathbf{s}) \neq 0$, accept-reject sampling method is used to just continually sample $\mathbf{u} \sim P(\mathbf{u}|\mathbf{s})$, until a sample result is accepted with the accept probability:

$$\begin{aligned} P_{\text{acc}}(\mathbf{u}|\mathbf{s}) &= \frac{q_u(\mathbf{u}|\mathbf{s})}{K_1 P(\mathbf{u}|\mathbf{s})} = \frac{K(\mathbf{u}|\mathbf{s})}{K_1} \\ &= \begin{cases} 1, \text{if } C(\mathbf{s}) \neq 0 \text{ and } P(A|\mathbf{u},\mathbf{s}) > 0 \\ \dfrac{K_2}{K_1}, \text{if } C(\mathbf{s}) \neq 0 \text{ and } P(A|\mathbf{u},\mathbf{s}) = 0 \end{cases}. \end{aligned} \tag{14}$$



(a) Uncritical state

(b) Critical state

Figure 2: A illustration of how to design proposal distribution.

This can ensure that the sample results obey distribution $q_u(\mathbf{u}|\mathbf{s})$.

Fig. 2 illustrates how to adjust $P(\mathbf{u}|\mathbf{s})$ to obtain expected $q_u(\mathbf{u}|\mathbf{s})$. The red and blue curves in the figure illustrate the distributions $P(\mathbf{u}|\mathbf{s})$ and $q(\mathbf{u}|\mathbf{s})$, respectively. To ensure the driving environment is both adversarial and naturalistic, only if criticality $C(\mathbf{s})$ is not zero will the action $\mathbf{u}$ be sampled in $q(\mathbf{u}|\mathbf{s})$, else in $q(\mathbf{u}|\mathbf{s})$. So, as shown in Fig. 2a, $q_u(\mathbf{u}|\mathbf{s})$ is equal to $P(\mathbf{u}|\mathbf{s})$ at uncritical state. On the other hand, while $C(\mathbf{s}) \neq 0$, $K_1 > 1$ will act on the part of $P(A|\mathbf{u},\mathbf{s}) > 0$, just as shown in Fig. 2b. So the probability of $\mathbf{u}$ that satisfy $P(A|\mathbf{u},\mathbf{s}) > 0$ can be increased, meaning that the probability of event A can be increased.

Meanwhile, K_1 also causes the integral area of the probability density function to not be equal to 1. So $K_2 < 1$ are used for balancing. However, the value of $\int K(\mathbf{u},\mathbf{s}) P(\mathbf{u}|\mathbf{s})\mathrm{d}\mathbf{u}$ varies for different states $\mathbf{s}$. Coupled with the implicit feature of $P(\mathbf{u}|\mathbf{s})$, it can be challenging to choose appropriate values for K_1 and K_2. Therefore, while sampling $\mathbf{u}$, the unnormalized distribution $q_u(\mathbf{u}|\mathbf{s})$ is used to replace $q(\mathbf{u}|\mathbf{s})$, as shown in equation (14). Unnormalization affects only the accident rate statistics, not the sampling results, which will be discussed in the next section.

3.2. AV performance evaluation

Importance sampling relies on explicit distributions to calculate weights. But in this paper we can only get an unnormalized weight $w_u(\mathbf{x})$. The unnormalized weight at the i-th test is calculated as:

$$\begin{aligned} w_u(\mathbf{x}_i) &= \frac{P(\mathbf{x}_i)}{q_u(\mathbf{x}_i)} = \prod_{k=1}^{T_i} \frac{P(\mathbf{u}(k)|\mathbf{s}(k))}{q_u(\mathbf{u}(k)|\mathbf{s}(k))} = \prod_{k=1}^{T_i} w_{u,ik}, \\ w_{u,ik} &= \frac{P(\mathbf{u}(k)|\mathbf{s}(k))}{q_u(\mathbf{u}(k)|\mathbf{s}(k))} = \frac{1}{K(\mathbf{u}(k),\mathbf{s}(k))}, \end{aligned} \tag{15}$$

where T_i denotes the time-steps of the i-th test and w_{ik} represents the weight at k-th time step during i-th test.

At an uncritical state $\mathbf{s}$ whose criticality $C(\mathbf{s})$ is 0, we have $P(\mathbf{u}|\mathbf{s}) = q(\mathbf{u}|\mathbf{s})$ and so that its weight is 1. So only critical states need to be considered when calculating weight. $T_{i,C}$ denotes the set of critical time steps during i-th test, then

$$w_u(\mathbf{x}_i) = \prod_{k \in T_{i,C}} w_{u,ik}. \tag{16}$$

Therefore, we can get an estimation of $P(A)$:

$$\mu_{q_u} = \mathbb{E}_q\left(P(A|\mathbf{x})\, w_u(\mathbf{x}_i)\right). \tag{17}$$

We use $\hat{\mu}_{q_u}$ to represent importance sampling estimation of μ_{q_u}:

$$\hat{\mu}_{q_u} = \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x}_i\right) w_u\left(\mathbf{x_i}\right), \mathbf{x_i} \sim q\left(\mathbf{x}\right). \tag{18}$$

While sampling efficiency can be ensured to be higher than NDE, it cannot be ensured that $\hat{\mu}_{q_u}$ is an unbiased estimation of $P\left(A\right)$ because μ_{q_u} is obviously not equal to $P\left(A\right)$ in equation (9). But we have found that given suitable parameters K_1 and K_2, we can obtain evaluation results almost identical to equation (9) efficiently. The next part will elaborate on this and analyse the bias and variance of μ_{q_u}.

3.3. Theoretical analysis of bias

In this section, we discuss the bias between the estimation μ_{q_u} and $P\left(A\right)$ and how to choose suitable parameters K_1, K_2 to decrease the bias. The source of bias between $P\left(A\right)$ and μ_{q_u} is the difference between $q_u\left(\mathbf{x}\right)$ and $q\left(\mathbf{x}\right)$. We define the coefficient $c\left(\mathbf{x}\right)$ as the ratio of $q_u\left(\mathbf{x}\right)$ to $q\left(\mathbf{x}\right)$:

$$c\left(\mathbf{x}\right) = \frac{q_u\left(\mathbf{x}\right)}{q\left(\mathbf{x}\right)}. \tag{19}$$

Then we present the following theorems.

Theorem 1.

$$\mathbb{E}_q\left(\hat{\mu}_{q_u}\right) = \mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}\right). \tag{20}$$

Proof.

$$\begin{aligned}\mathbb{E}_q\left(\hat{\mu}_{q_u}\right) &= \mathbb{E}_q\left(\frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x_i}\right) w_u\left(\mathbf{x_i}\right)\right) = \mathbb{E}_q\left(P\left(A|\mathbf{x}\right) w_u\left(\mathbf{x}\right)\right) \\ &= \mathbb{E}_q\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}\frac{P\left(\mathbf{x}\right)}{q\left(\mathbf{x}\right)}\right) = \mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}\right).\end{aligned} \tag{21}$$

End of proof. □

Remark 1. Theorem 1 demonstrates that $\hat{\mu}_{q_u}$ provides an unbiased estimation of $\mathbb{E}\left(\frac{P(A|\mathbf{x})}{c(\mathbf{x})}\right)$ rather than $P\left(A\right)$. This indicates that while $\hat{\mu}_{q_u}$ may not exactly estimate $P\left(A\right)$, it estimates a quantity scaled by the ratio of $q_u\left(\mathbf{x}\right)$ to $q\left(\mathbf{x}\right)$. The bias here arises from the mismatch between the proposal distribution $q_u\left(\mathbf{x}\right)$ and the true distribution $q\left(\mathbf{x}\right)$. Ideally, to minimize this bias, $c\left(\mathbf{x}\right)$ should be as close to 1 as possible, ensuring that $\hat{\mu}_{q_u}$ provides a better approximation of $P\left(A\right)$.

Theorem 2. *The bound of $P\left(A\right)$ can be calculated as*

$$\mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}c_{\min}\left(\mathbf{x}\right)\right) \leq P\left(A\right) \leq \mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}c_{\max}\left(\mathbf{x}\right)\right), \tag{22}$$

where $c_{\min}\left(\mathbf{x}\right)$ and $c_{\max}\left(\mathbf{x}\right)$ denotes the bound of $c\left(\mathbf{x}\right)$ during test $\mathbf{x}$.

Proof.

$$\mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}c_{\min}\left(\mathbf{x}\right)\right) \leq P\left(A\right) = \mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}c\left(\mathbf{x}\right)\right) \leq \mathbb{E}_P\left(\frac{P\left(A|\mathbf{x}\right)}{c\left(\mathbf{x}\right)}c_{\max}\left(\mathbf{x}\right)\right). \tag{23}$$

End of proof. □

Remark 2. Theorem 2 establishes bounds on $P(A)$ based on the minimum and maximum values of the coefficient $c(\mathbf{x})$. This theorem provides a range within which the true value of $P(A)$ lies. The bounds are calculated by incorporating $c_{\min}(\mathbf{x})$ and $c_{\max}(\mathbf{x})$, which represent the extremal values of $c(\mathbf{x})$. By estimating these bounds through sampling results, we can obtain a reliable range for $P(A)$.

In the following content, we will detail the methods used to calculate the bound of $P(A)$ as outlined in Theorem 2, including how to estimate the bound of $c(\mathbf{x})$. Further, we will describe the process for selecting the parameters K_1 and K_2 to control the bias in the estimation. This involves ensuring that $c(\mathbf{x})$ remains close to 1, as discussed in Theorem 1, to ensure that the bound of $P(A)$ is both accurate and reliable for evaluating the performance of AV.

We estimate the bound of $P(A)$ using sampling results based on $q(\mathbf{x})$:

$$\begin{aligned}\mathbb{E}_P\left(\frac{P(A|\mathbf{x})}{c(\mathbf{x})}c_{\max}(\mathbf{x})\right) &= \mathbb{E}_q\left(\frac{P(A|\mathbf{x})}{c(\mathbf{x})}\frac{P(\mathbf{x})}{q(\mathbf{x})}c_{\max}(\mathbf{x})\right) = \mathbb{E}_q\left(P(A|\mathbf{x})\,w_u(\mathbf{x})\,c_{\max}(\mathbf{x})\right)\\ &\approx \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x_i}\right)w_u\left(\mathbf{x_i}\right)c_{\max}\left(\mathbf{x_i}\right), \quad \mathbf{x_i}\sim q(\mathbf{x}),\\ \mathbb{E}_P\left(\frac{P(A|\mathbf{x})}{c(\mathbf{x})}c_{\min}(\mathbf{x})\right) &\approx \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x_i}\right)w_u\left(\mathbf{x_i}\right)c_{\min}\left(\mathbf{x_i}\right), \quad \mathbf{x_i}\sim q(\mathbf{x}).\end{aligned} \tag{24}$$

And we record these estimations as

$$\begin{aligned}\hat{\mu}_{q_u,\min} &= \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x_i}\right)w_u\left(\mathbf{x_i}\right)c_{\min}\left(\mathbf{x_i}\right),\\ \hat{\mu}_{q_u,\max} &= \frac{1}{n}\sum_{i=1}^{n} P\left(A|\mathbf{x_i}\right)w_u\left(\mathbf{x_i}\right)c_{\max}\left(\mathbf{x_i}\right).\end{aligned} \tag{25}$$

Introduce confidence level λ and standard deviation function $\mathrm{std}(\cdot)$, then the bound of $P(A)$ can be estimated as

$$\begin{aligned}P(A) &\geq \mathbb{E}_P\left(\frac{P(A|\mathbf{x})}{c(\mathbf{x})}c_{\min}(\mathbf{x})\right)\\ &\geq \hat{\mu}_{q_u,\min} - \lambda\cdot\mathrm{std}\left(\hat{\mu}_{q_u,\min}\right) = \inf P(A),\\ P(A) &\leq \mathbb{E}_P\left(\frac{P(A|\mathbf{x})}{c(\mathbf{x})}c_{\max}(\mathbf{x})\right)\\ &\leq \hat{\mu}_{q_u,\max} - \lambda\cdot\mathrm{std}\left(\hat{\mu}_{q_u,\max}\right) = \sup P(A).\end{aligned} \tag{26}$$

For each test $\mathbf{x}_i$, $P\left(A|\mathbf{x}_i\right)w_u\left(\mathbf{x}_i\right)$ can be calculated based on the sample results. The key is to estimate the bound of $c\left(\mathbf{x_i}\right)$. For test $\mathbf{x}_i$, using c_{ik} to represent the normalization coefficient of $q_u\left(\mathbf{u}_k|\mathbf{s}_k\right)$ at the k-th time step:

$$c_{ik} = \int q_u\left(\mathbf{u}_k|\mathbf{s}_k\right)\mathrm{d}\mathbf{u}_\mathrm{k}. \tag{27}$$

It is straightforward to see that $q\left(\mathbf{u}_k|\mathbf{s}_k\right) = \frac{q_u(\mathbf{u}_k|\mathbf{s}_k)}{c_{ik}}$. At time steps that are not critical, i.e., $k \notin T_{i,C}$, we have $P\left(\mathbf{u}_k|\mathbf{s}_k\right) = q\left(\mathbf{u}_k|\mathbf{s}_k\right) = q_u\left(\mathbf{u}_k|\mathbf{s}_k\right)$ and therefore $c_{ik} = 1$. So for the entire sequence, $c\left(\mathbf{x}_i\right)$ is given by the product $\prod_{k\in T_{i,C}} c_{ik}$. Further, to estimate the bound of $\frac{1}{m}\sum_{i\in n_C}\frac{1}{c(\mathbf{x}_i)}$, we calculate c_{ik} as

$$\begin{aligned}c_{ik} &= K_1\cdot H_1(\mathbf{s}(k)) + K_2\cdot H_2(\mathbf{s}(k))\\ &= \left(K_1-1\right)\cdot H_1(\mathbf{s}(k)) + K_2,\end{aligned} \tag{28}$$

where

$$\begin{aligned}H_1(\mathbf{s}(k)) &= \int_{\mathbf{U}_C} q(\mathbf{u}|\mathbf{s}(k))\mathrm{d}\mathbf{u},\\ H_2(\mathbf{s}(k)) &= 1 - H_1(\mathbf{s}(k)).\end{aligned} \tag{29}$$

$\mathbf{U}_c$ denotes the set of critical actions. So H_1 represents the probability of critical action in the current state $\mathbf{s}_k$. $H_1(\mathbf{s}(k))$ represents the probability of critical actions in state $\mathbf{s}_k$, and $H_2(\mathbf{s}(k))$ is the coplementary probability. The geometric meaning of H_1 and H_2 can be illustrated as the area of the right and left part of red area in Fig. 2b respectively.

Due to c_{ik} being linear respect to H_1, so the minimum and maximum values of c_{ik} are respectively

$$\begin{aligned} \min c_{ik} &= \left(K_1 - 1\right) \cdot \min H_1 + K_2, \\ \max c_{ik} &= \left(K_1 - 1\right) \cdot \max H_1 + K_2. \end{aligned} \tag{30}$$

Then

$$\begin{aligned} c\left(x_i\right) &= \prod_{k \in T_{i,C}} c_{ik} \geq \prod_{k \in T_{i,C}} \min c_{ik} = \left(\min c_{ik}\right)^{T_{i,C}}, \\ c\left(x_i\right) &= \prod_{k \in T_{i,C}} c_{ik} \leq \prod_{k \in T_{i,C}} \max c_{ik} = \left(\max c_{ik}\right)^{T_{i,C}}, \end{aligned} \tag{31}$$

where superscript $T_{i,C}$ denote the size of set $T_{i,C}$. Define

$$\begin{aligned} c_{\min}\left(\mathbf{x}_i\right) &= \left(\min c_{ik}\right)^{T_{i,C}}, \\ c_{\max}\left(\mathbf{x}_i\right) &= \left(\max c_{ik}\right)^{T_{i,C}}. \end{aligned} \tag{32}$$

Then the bound of $P(A)$ in equation (26) can be calculated.

Equations (26), (30) and (32) provide practical methods for calculating the bound of $P(A)$. We will next specify H_1 and the parameters K_1 and K_2 in equation (30). H_1 represents the probability of critical actions given critical states, so it can be estimated within a range based on prior knowledge of the NDE. This paper provides different ranges for H_1 depending on the NDE models through the analysis of NDE sample results and NDE model parameters. More detials will be provided in the experimental chapter.

Given the range of H_1, it is crucial to select K_1, K_2 to control the bound of $P(A)$. As discussed in Theorem 1 and equation (32), the aim is to make c_{ik} as close to 1 as possible, and so do $c(\mathbf{x})$. Suitable K_1 and K_2 will be chosen according to the range of H_1 and equation (30), to make $\min c_{ik}$ and $\max c_{ik}$ close to 1. More details on the range of H_1 and the parameters K_1, K_2 used in the paper can also be found in the experimental chapter.

4. Experimental studies

To validate our method, we adapted our implicit importance sampling approach to two different NDE models based on existing works (Feng et al., 2023; Yan et al., 2023). We refer to these environments as NDE-I and NDE-II in this paper. Since the NDEs were generated based on naturalistic driving data, the evaluation results are representative of real-world scenarios. And the scalability and generalizability of our approach were verified, as different types of NDE models and varied road geometries were utilized in these experiments.

NDE-I is modelled with explicit and discrete distributions $P(\mathbf{u}|\mathbf{s})$, where the value of $P(\mathbf{u}|\mathbf{s})$ is known. Additionally, the true value of the distribution $q(\mathbf{u}|\mathbf{s})$ can be easily obtained through normalization, allowing us to calculate the true weight values. We compared the experimental results using both the true weights and the unnormalized estimated weights from IIS in NDE to demonstrate the validity of our method. All experiments on NDE-I were conducted on a two-lane highway with a 400-meter driving distance for the AV. Each test episode concluded either when the AV reached the 400-meter mark or an accident occurred.

On the other hand, NDE-II is based on a neural network, resulting in an implicit distribution $P(\mathbf{u}|\mathbf{s})$. As the exact distribution value is not accessible, only the sample results can be obtained, making implicit importance sampling the sole applicable approach. We compared the experimental results using unnormalized estimated weights from IIS with results in NDE to confirm that our method is effective for problems involving implicit distributions. We conducted experiments in a roundabout environment, where the AV was tested for a 36-second driving duration or until an accident occurred, or the AV exited the roundabout.

For performance metrics, we assessed both adversarialism and naturalism by comparing crash rates and key data distributions between IIS and NDE. In the case of NDE-I, only risky data, which constitutes a small proportion, was collected and analyzed to demonstrate that IIS generates more adversarial driving environments. For NDE-II, all driving

data was collected and analyzed to prove that, despite the small amount of risk data, the generated NADE closely resembles the original NDE. Also, we compared accident rates to verify the results of accelerated testing, showing that our method can estimate accident rates more efficiently with controllable and acceptable biases.

In summary, our experiments validated three key aspects of our method:

- IIS is capable of generating NADE that increase adversarial scenarios while preserving the naturalistic properties of the original driving environment.
- IIS achieves accelerated testing by efficiently estimating accident rates, particularly in cases where the nominal distribution $P(\mathbf{x})$ is implicit.
- Our method allows for controlled bias during estimation, and with appropriate parameter selection, we demonstrate that the bias can be kept within a reliable range. Additionally, we provide guidelines for choosing parameters to ensure reliable results.

4.1. Results analysis for NDE-I

Due to the discrete nature of NDE-I, we were able to directly obtain the probability of critical actions, namely H_1, directly during the sampling process. Based on data analysis from preliminary work (Feng et al., 2023), the range of H_1 was determined to be $\left(1 \times 10^{-7}, 5 \times 10^{-5}\right)$. We set the parameters as $K_1 = 100, K_2 = 0.99$ and $K_1 = 500, K_2 = 0.99$

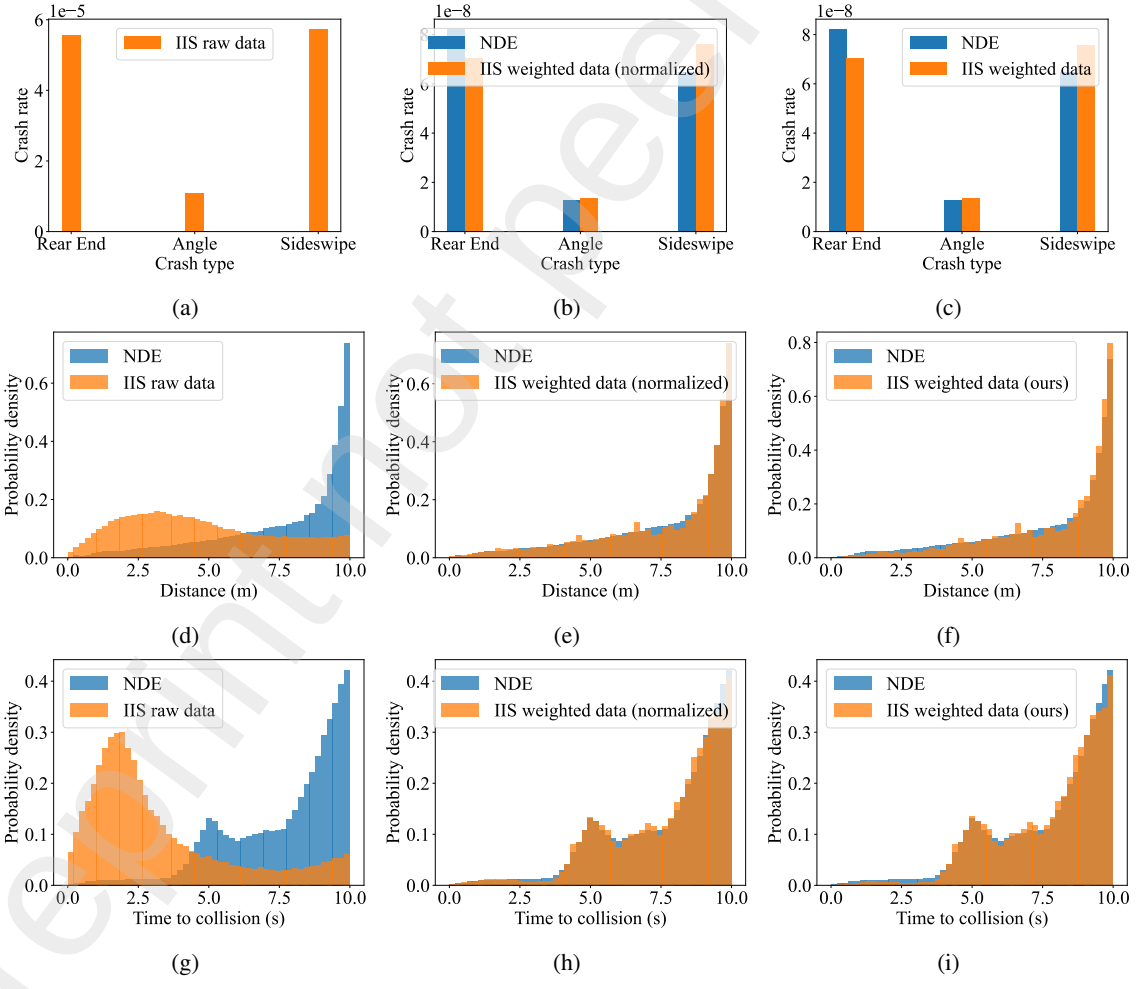


Figure 3: Naturalistic and adversarial driving environment generation for NDE-I

respectively, to ensure that c_{ik} approaches 1, as shown in equation (30). Although both K_1 and K_2 are involved, K_2 is consistently set to 0.99, so the analysis in the paper focuses primarily on the variations in K_1. For validation, we conducted approximately 5×10^6 test episodes for the parameter setting $K_1 = 100$, and 6×10^5 test episodes for the setting $K_1 = 500$, in the naturalistic and adversarial driving environments, while the corresponding number of episodes in NDE was approximately 2×10^8. Fig. 3 visualizes the key data distributions with $K_1 = 500$, $K_2 = 0.99$, comparing NDE with the NADE generated by IIS.

The first column of figures shows the distribution of crash types and near-miss incidents involving AV. We use time-to-collision (TTC) and bumper-to-bumper distance to assess proximity between AV and background vehicles (BVs). It is evident that the crash rate in NADE reached 2.45×10^{-05}, significantly higher than the NDE's crash rate of 1.58×10^{-7}. This result demonstrates that the environment generated by IIS is more adversarial. Additionally, the near-miss distances and TTC values in the IIS-generated environment tend to cluster at lower values, further explaining the increased frequency of accidents. Traffic flow distributions were altered, as the sample distribution $q(\mathbf{x})$ replaced the nominal distribution $P(\mathbf{x})$.

One advantage of importance sampling is its ability to provide unbiased estimates by applying weights to the samples. By applying the true weight $w(\mathbf{x})$, we modified the distributions as shown in the second column of figures, bringing them closer to the original NDE distribution. This indicates that importance sampling maintains unbiasedness within a certain level of accuracy. However, in our work, $P(\mathbf{x})$ is treated as an implicit distribution, and only approximate, unnormalized weights can be calculated. As a result, the estimated outcomes inherently carry bias. Despite this, our experiments show that the bias remains small under the given parameters, as demonstrated in the final column of figures.

We also evaluated the crash rate estimation using IIS. A confidence level of $\lambda = 0.95$ was chosen for calculating bounds and confidence intervals. Fig. 4 presents the evaluation results under two parameter settings. Fig. 4a and Fig. 4d depict the crash rate progression over time, with the shaded areas representing confidence intervals. Fig. 4b and Fig. 4e display the relative half-width (RHW) metric (Zhao et al., 2017), which is used to measure efficiency. The minimum number of test episodes required to reach a precision threshold (RHW=0.3) was computed. For NDE, approximately

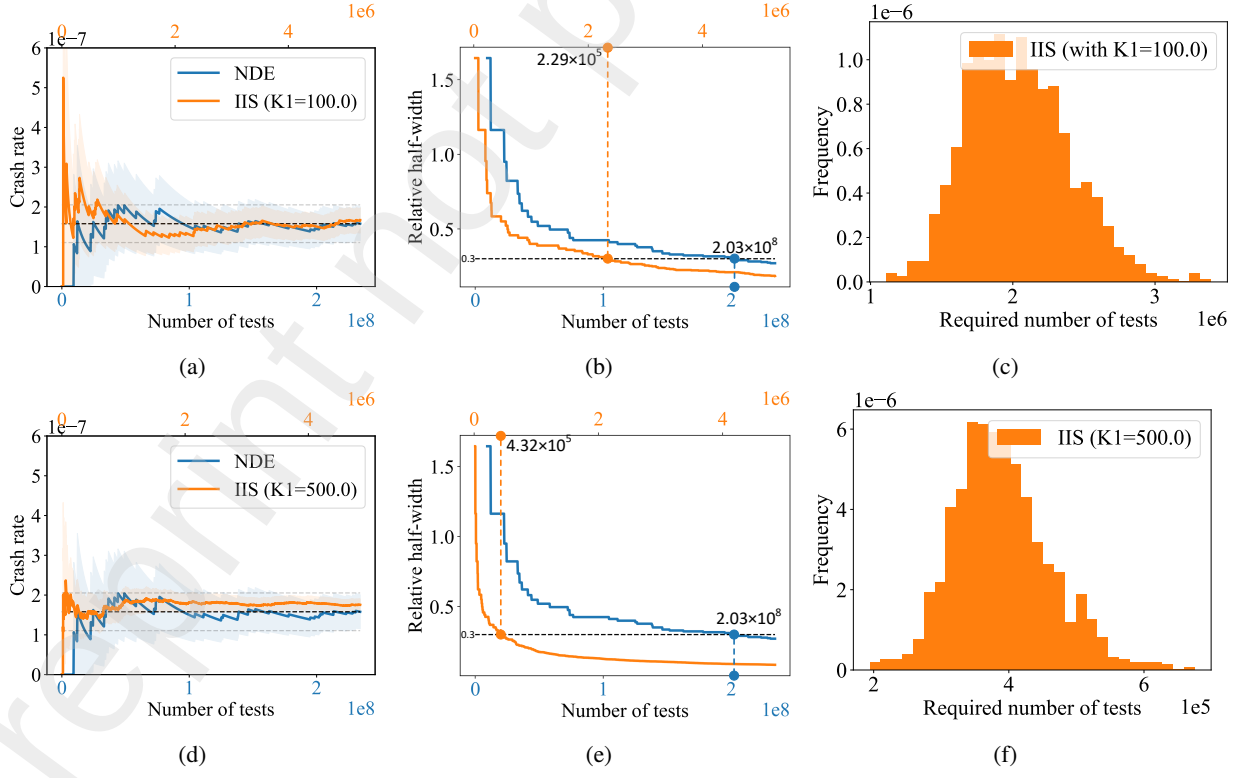


Figure 4: AV performance evaluation for NDE-I

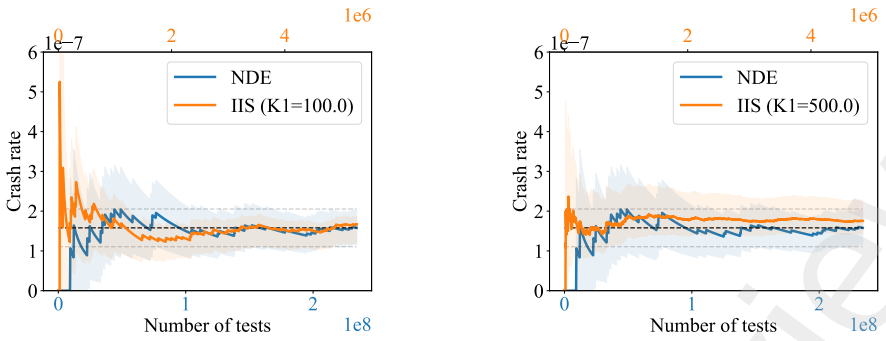


Figure 5: Bias analysis of AV performance evaluation results for NDE-I

2.03×10^8 episodes were required. With $K_1 = 100$, the number of tests required in the environment generated by IIS was reduced to 2.34×10^6, accelerating the evaluation by a factor of 87. And the experimental results showed that the estimated crash rate converges to the very similar value. When $K_1 = 500$, only 4.32×10^5 tests were needed, accelerating the evaluation by a factor of 470. However, this higher efficiency came at the cost of increased yet still controlled bias in the crash rate estimate.

The results above demonstrate that our method significantly accelerates the evaluation process while providing a reliable estimation of the crash rate, even with some degree of bias. Then we further analyze the efficiency, the range of bias, and the relationship between these outcomes and the parameter K_1. The range of H_1 lies between $\left(1 \times 10^{-7}, 5 \times 10^{-5}\right)$, leading to the corresponding values of $\left(\min c_{ik}, \max c_{ik}\right)$ being approximately $(0.9900099, 0.99495)$ for $K_1 = 100$ and $(0.9900499, 1.01495)$ for $K_1 = 500$. Given that c_{ik} remains close to 1, we can ensure that the bias is kept within a controllable range. We also computed the bounds for the estimated crash rate and displayed them in Fig. 5, where the shaded area represents the bounds of $P(A)$ calculated using equation (26). While a larger K_1 increases the probability of encountering riskier scenarios, it also results in a looser bound on $c(\mathbf{x})$. Thus, $K_1 = 500$ achieved greater efficiency but with a slightly larger bias. The optimal value of K_1 should be selected based on the trade-offs between efficiency and accuracy.

4.2. Results analysis for NDE-II

NDE-II is based on a neural network, so the probability of critical actions cannot be directly obtained. Based on an analysis of crash criticality provided by preliminary work (Yan et al., 2023), the range of H_1 was estimated to be $\left(4.5 \times 10^{-5}, 3 \times 10^{-3}\right)$. We then set the parameters as $K_1 = 50, K_2 = 0.99$ and $K_1 = 100, K_2 = 0.99$, respectively. For validation, we collected approximately 4×10^5 test episodes under both parameter settings in NADE, while about 1×10^7 episodes were tested in NDE. In Fig. 6, we visualized the distribution of key data with $K_1 = 50, K_2 = 0.99$.

The crash rate in NADE reached 4.72×10^{-4}, significantly higher than in NDE (5.21×10^{-6}), demonstrating the adversarial nature of the generated environment. After applying the weighting adjustments, the crash rate in NADE was modified to 4.66×10^{-6}, as shown in Fig. 6b. This modification results in a slight bias due to the use of unnormalized weights, but it remains within a reasonable range for practical purposes. Fig. 6c, Fig. 6b, and Fig. 6e show the distribution of distance between vehicles and vehicle speeds, respectively. The KL-divergence values of the distributions were 0.002, 0.004, and 0.003, respectively, indicating that the generated NADE closely resembles NDE. This is reasonable because only the behavior of critical vehicles at specific moments, representing an extremely small proportion of the data, was altered, while the overall driving environment remained consistent with naturalistic conditions.

Similar to the analysis in Chapter 4.1, we estimated the accident rates and plotted the results in Fig. 7. In NDE, approximately 5.87×10^6 tests were required to reach the desired precision. With $K_1 = 50$, our method required only 1.24×10^5 tests, accelerating the evaluation by a factor of 47. For $K_1 = 100$, only 8.05×10^4 tests were required, resulting in a speed-up of 73 times. Although a larger K_1 leads to higher evaluation efficiency, it also introduces a slight bias in the results. We further calculated the bounds of the estimated crash rates, which are visualized in Fig. 8, showing the trade-off between efficiency and accuracy when choosing different values of K_1.

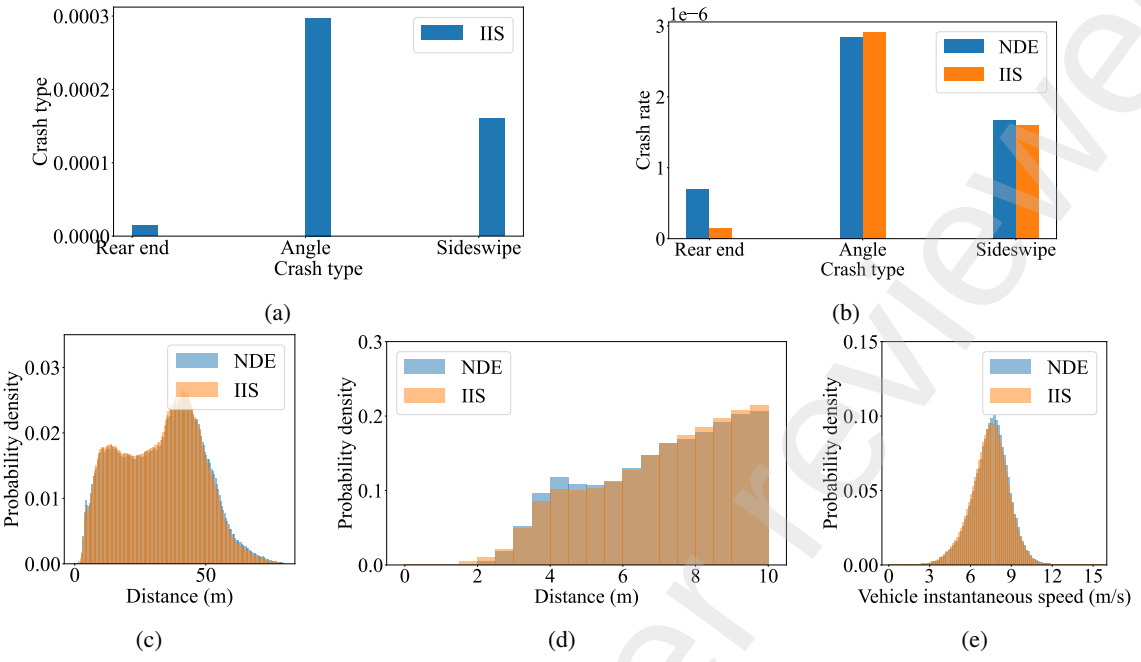


Figure 6: Naturalistic and adversarial driving rnvironment generation for NDE-II

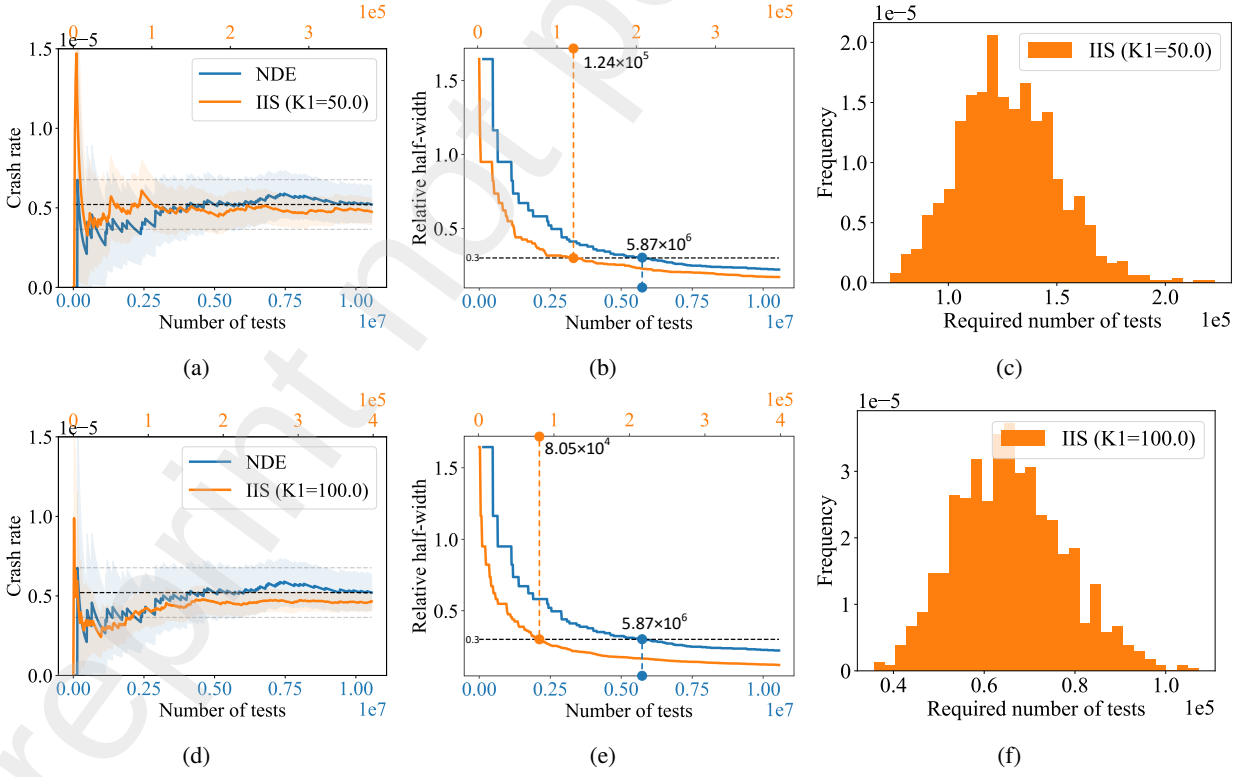


Figure 7: AV performance evaluation for NDE-II

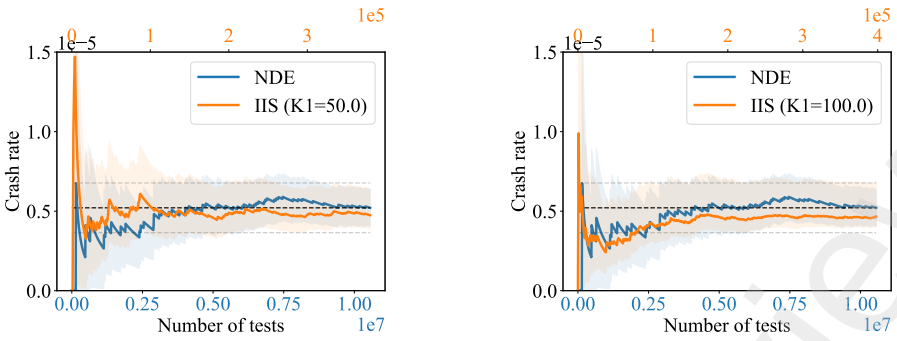


Figure 8: Bias analysis of AV performance evaluation results for NDE-II

5. Conclusion and Future Work

In this paper, we introduce a novel Implicit Importance Sampling (IIS) approach designed to enhance the intelligent testing environment for autonomous vehicles (AVs). Our method generates testing environments that are more adversarial, which allows for accelerated testing and provides reliable and representative evaluation results. We validated our approach through experiments on two different NDE models, demonstrating its scalability and effectiveness. Our method is effective for any NDE model that testing scenarios can be sampled, even if the NDE model is implicit or not well defined. This versatility highlights the potential of IIS to address common challenges in testing not only autonomous vehicles but also other intelligent agents in complex environments.

Future work will focus on reducing bias by improving our method's parameter selection process. Currently, our approach uses some prior knowledge-based assumptions to choose reasonable parameters and minimize bias. The next step is to develop adaptive parameter selection techniques to further reduce error and eliminate dependence on these assumptions. This enhancement will make our method more robust and accurate, leading to more reliable testing outcomes for autonomous vehicles and other intelligent systems.

CRediT authorship contribution statement

Kun Ren: Writing - Review & editing, Writing - Original draft, Validation, Software, Methodology, Formal analysis, Data curation, Investigation, Visualization. **Jingxuan Yang:** Writing - Review & editing, Methodology, Formal analysis. **Qiujing Lu:** Methodology, Formal analysis. **Yi Zhang:** Funding acquisition, Supervision, Conceptualization. **Jianming Hu:** Resources, Funding acquisition, Supervision, Conceptualization. **Shuo Feng:** Writing - Review & editing, Resources, Methodology, Funding acquisition, Formal analysis, Supervision, Project administration, Conceptualization.

Acknowledgement

This work is supported by Beijing Natural Science Foundation 4244092 and L231014, Beijing Nova Program 20230484259, and National Natural Science Foundation of China under Grant No. 62333015.

References

Arief, M., Cen, Z., Liu, Z., Huang, Z., Li, B., Lam, H., Zhao, D., 2022. Certifiable evaluation for autonomous vehicle perception systems using deep importance sampling (deep is), in: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), IEEE. pp. 1736–1742.

Cancela, H., El Khadiri, M., Rubino, G., 2009. Rare event analysis by monte carlo techniques in static models. Rare Event Simulation Using Monte Carlo Methods , 145–170.

Corso, A., Du, P., Driggs-Campbell, K., Kochenderfer, M.J., 2019. Adaptive stress testing with reward augmentation for autonomous vehicle validatio, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE. pp. 163–168.

Delyon, B., Portier, F., 2016. Integral approximation by kernel smoothing. Bernoulli 22, 2177 – 2208. URL: https://doi.org/10.3150/15-BEJ725, doi:10.3150/15-BEJ725.

Ding, W., Xu, C., Arief, M., Lin, H., Li, B., Zhao, D., 2023. A survey on safety-critical driving scenario generationa methodological perspective. IEEE Transactions on Intelligent Transportation Systems 24, 6971–6988.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V., 2017. Carla: An open urban driving simulator, in: Conference on robot learning, PMLR. pp. 1–16.

Fang, J., Zhou, D., Yan, F., Zhao, T., Zhang, F., Ma, Y., Wang, L., Yang, R., 2020. Augmented lidar simulator for autonomous driving. IEEE Robotics and Automation Letters 5, 1931–1938.

Feng, S., Feng, Y., Sun, H., Bao, S., Zhang, Y., Liu, H.X., 2021a. Testing scenario library generation for connected and automated vehicles, part ii: Case studies. IEEE Transactions on Intelligent Transportation Systems 22, 5635–5647. doi:10.1109/TITS.2020.2988309.

Feng, S., Feng, Y., Yu, C., Zhang, Y., Liu, H.X., 2021b. Testing scenario library generation for connected and automated vehicles, part i: Methodology. IEEE Transactions on Intelligent Transportation Systems 22, 1573–1582. doi:10.1109/TITS.2020.2972211.

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. URL: https://www.nature.com/articles/s41586-023-05732-2, doi:https://doi.org/10.1038/s41586-023-05732-2. publisher: Nature Publishing Group.

Feng, S., Yan, X., Sun, H., Feng, Y., Liu, H.X., 2021c. Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment. Nature communications 12, 748.

Filos, A., Tigkas, P., McAllister, R., Rhinehart, N., Levine, S., Gal, Y., 2020. Can autonomous vehicles identify, recover from, and adapt to distribution shifts?, in: International Conference on Machine Learning, PMLR. pp. 3145–3153.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. Advances in neural information processing systems 27.

Hanselmann, N., Renz, K., Chitta, K., Bhattacharyya, A., Geiger, A., 2022. King: Generating safety-critical driving scenarios for robust imitation via kinematics gradients, in: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVIII, Springer. pp. 335–352.

Hao, K., Cui, W., Luo, Y., Xie, L., Bai, Y., Yang, J., Yan, S., Pan, Y., Yang, Z., 2023. Adversarial safety-critical scenario generation using naturalistic human driving priors. IEEE Transactions on Intelligent Vehicles .

Henmi, M., Yoshida, R., Eguchi, S., 2007. Importance sampling via the estimated sampler. Biometrika 94, 985–991.

Huang, Z., Arief, M., Lam, H., Zhao, D., 2019. Evaluation uncertainty in data-driven self-driving testing, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE. pp. 1902–1907.

Jiang, Z., Pan, W., Liu, J., Dang, S., Yang, Z., Li, H., Pan, Y., 2022. Efficient and unbiased safety test for autonomous driving systems. IEEE Transactions on Intelligent Vehicles 8, 3336–3348.

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.

Kingma, D.P., Welling, M., 2013. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 .

Koren, M., Kochenderfer, M.J., 2019. Efficient autonomy validation in simulation with adaptive stress testing, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE. pp. 4178–4183.

Krajzewicz, D., 2010. Traffic simulation with sumo–simulation of urban mobility. Fundamentals of traffic simulation , 269–293.

Kruber, F., Wurst, J., Botsch, M., 2018. An unsupervised random forest clustering technique for automatic traffic scenario categorization, in: 2018 21st International conference on intelligent transportation systems (ITSC), IEEE. pp. 2811–2818.

Lee, R., Mengshoel, O.J., Saksena, A., Gardner, R.W., Genin, D., Silbermann, J., Owen, M., Kochenderfer, M.J., 2020. Adaptive stress testing: Finding likely failure events with reinforcement learning. Journal of Artificial Intelligence Research 69, 1165–1201.

Li, W., Pan, C., Zhang, R., Ren, J., Ma, Y., Fang, J., Yan, F., Geng, Q., Huang, X., Gong, H., et al., 2019. Aads: Augmented autonomous driving simulation using data-driven algorithms. Science robotics 4, eaaw0863.

Liu, H.X., Feng, S., 2024. Curse of rarity for autonomous vehicles. nature communications 15, 4808.

Liu, Q., Lee, J., 2017. Black-box importance sampling, in: Artificial Intelligence and Statistics, PMLR. pp. 952–961.

Lu, Q., Bai, R., Li, S., He, H., Feng, S., . Racl: Risk aware closed-loop agent simulation with high fidelity .

Mooney, C.Z., 1997. Monte carlo simulation. 116, Sage.

Morris, R., Descombes, X., Zerubia, J., 1996. The ising/potts model is not well suited to segmentation tasks, in: 1996 IEEE Digital Signal Processing Workshop Proceedings, IEEE. pp. 263–266.

Nalic, D., Mihalj, T., Bäumler, M., Lehmann, M., Eichberger, A., Bernsteiner, S., 2020. Scenario based testing of automated driving systems: A literature survey, in: FISITA web Congress, p. 1.

Neumann, V., 1951. Various techniques used in connection with random digits. Notes by GE Forsythe , 36–38.

Niu, H., Ren, K., Xu, Y., Yang, Z., Lin, Y., Zhang, Y., Hu, J., 2023. (re)2h2o: Autonomous driving scenario generation via reversely regularized hybrid offline-and-online reinforcement learning, in: 2023 IEEE Intelligent Vehicles Symposium (IV), pp. 1–8. doi:10.1109/IV55152.2023.10186559.

Oates, C.J., Girolami, M., Chopin, N., 2016. Control Functionals for Monte Carlo Integration. Journal of the Royal Statistical Society Series B: Statistical Methodology 79, 695–718. URL: https://doi.org/10.1111/rssb.12185, doi:10.1111/rssb.12185, arXiv:https://academic.oup.com/jrsssb/article-pdf/79/3/695/49215177/jrsssb_79_3_695.pdf.

O'Hagan, A., 1987. Monte carlo is fundamentally unsound. The Statistician , 247–249.

Owen, A.B., 2013. Monte carlo theory, methods and examples.

Puterman, M.L., 1990. Markov decision processes. Handbooks in operations research and management science 2, 331–434.

Rempe, D., Philion, J., Guibas, L.J., Fidler, S., Litany, O., 2022. Generating useful accident-prone driving scenarios via a learned traffic prior, in: Conference on Computer Vision and Pattern Recognition (CVPR).

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.

Scanlon, J.M., Kusano, K.D., Daniel, T., Alderson, C., Ogle, A., Victor, T., 2021. Waymo simulated driving behavior in reconstructed fatal crashes within an autonomous vehicle operating domain. Accident Analysis & Prevention 163, 106454.

Shah, S., Dey, D., Lovett, C., Kapoor, A., 2018. Airsim: High-fidelity visual and physical simulation for autonomous vehicles, in: Field and Service Robotics: Results of the 11th International Conference, Springer. pp. 621–635.

Sun, H., Feng, S., Yan, X., Liu, H.X., 2021. Corner case generation and analysis for safety assessment of autonomous vehicles. Transportation research record 2675, 587–600.

Wang, W., Zhao, D., 2018. Extracting traffic primitives directly from naturalistically logged data for self-driving applications. IEEE Robotics and Automation Letters 3, 1223–1229.

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, Z., Li, Z., Hu, J., Zhang, Y., 2024. Dynamically expanding capacity of autonomous driving with near-miss focused training framework, in: International Conference on Transportation and Development 2024, pp. 616–626.

Zhang, H., Sun, J., Tian, Y., 2023. Accelerated risk assessment for highly automated vehicles: Surrogate-based monte carlo method. IEEE Transactions on Intelligent Transportation Systems .

Zhang, H., Zhou, H., Sun, J., Tian, Y., 2022. Risk assessment of highly automated vehicles with naturalistic driving data: A surrogate-based optimization method, in: 2022 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 580–585.

Zhao, D., Lam, H., Peng, H., Bao, S., LeBlanc, D.J., Nobukawa, K., Pan, C.S., 2017. Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques. IEEE Transactions on Intelligent Transportation Systems 18, 595–607. doi:10.1109/TITS.2016.2582208.

Zhao, S., Duan, J., Wu, S., Gu, X., Li, C., Yin, K., Wang, H., 2023. Genetic algorithm-based sotif scenario construction for complex traffic flow. Automotive Innovation 6, 531–546.