

Testing Scenario Library Generation for Connected and Automated Vehicles, Part II: Case Studies

Shuo Feng[✉], Yiheng Feng[✉], Haowei Sun, Shan Bao, Yi Zhang[✉], *Member, IEEE*,
and Henry X. Liu[✉], *Member, IEEE*

Abstract—Testing scenario library generation (TSLG) is a critical step for the development and deployment of connected and automated vehicles (CAVs). In Part I of this study, a general method for TSLG is proposed, and theoretical properties are investigated regarding the accuracy and efficiency of CAV evaluation. This paper aims to provide implementation examples and guidelines, and to enhance the proposed methodology under high-dimensional scenarios. Three typical cases, including cut-in, highway-exit, and car-following, are designed and studied in this paper. For each case, the process of library generation and CAV evaluation is elaborated. To address the challenges brought by high dimensionality, the proposed method is further enhanced by reinforcement learning technique. For all three cases, results show that the proposed method can accelerate the CAV evaluation process by multiple magnitudes with same evaluation accuracy, if compared with the on-road test method.

Index Terms—Connected and automated vehicles, testing scenario library, safety, functionality, reinforcement learning.

I. INTRODUCTION

TESTING and evaluation is a critical step in the development and deployment of connected and automated vehicles (CAVs). In the past few years, increasing research efforts have been made to solve the testing scenario library generation (TSLG) problem [1]–[7] (see Part I of this study [8] for more details). However, all previous methods have limitations in either scenario types that can be handled (e.g., low-dimensional scenarios only), CAV models that can be applied (e.g., a specific CAV only), or performance metrics that can be evaluated (e.g., safety evaluation only).

To overcome these limitations, in Part I of this study [8], we propose a general method for the TSLG problem for

different scenario types, CAV models, and performance metrics. Testing scenario is evaluated by a newly proposed measure, scenario criticality, which can be computed as a combination of maneuver challenge and exposure frequency. The new measure is fundamentally different from most existing studies, which usually overvalue the worst-case scenarios [5], [9]. In our proposed method, scenarios with higher occurrence probability in the real-world and higher maneuver challenges will have higher priority for CAV evaluation.

Part I paper lays out the methodological foundation and proves the statistical accuracy and efficiency theoretically. To implement the proposed method, however, there exist several gaps:

First, although the proposed framework is generic, some sub-problems vary case by case, e.g., auxiliary objective function design, naturalistic driving data analysis, and surrogate model construction. Carefully selected case studies will provide examples for the implementation of the overall framework.

Second, it is significant to show the ability of the proposed method in handling different performance metrics. Most existing studies focus only on safety evaluation, which is essential but insufficient for a deployable CAV. Besides safety, functionality is another important performance metric, which shows the CAV's ability to complete driving tasks in a timely manner. Designing and implementing testing scenarios for functionality evaluation are necessary.

Finally, applying the proposed method directly to high-dimensional cases can be problematic, as the computational complexity of critical scenario searching increases exponentially with the increase of dimensionality. However, most of the driving scenarios are naturally high-dimensional. So how to deal with high dimensional testing scenarios becomes an important issue that needs to be addressed. We should note that most existing studies also suffer from the “curse of dimensionality”. For example, the PEGASUS project [3] applied an exhaustive searching method to find all scenarios, which is impossible for high-dimensional scenarios. The accelerated evaluation method proposed in [4] also has difficulty in calibrating importance functions, where computational complexity grows exponentially with the dimensionality.

This paper aims to fill in these gaps. Three common testing cases, including cut-in, highway exit, and car-following, will be discussed in this paper (see Fig. 1). The cut-in case illustrates each step of the scenario library generation process

Manuscript received October 21, 2019; revised February 4, 2020; accepted April 10, 2020. This work was supported in part by the U.S. Department of Transportation (USDOT) Region 5 University Transportation Center: Center for Connected and Automated Transportation (CCAT) of the University of Michigan. The Associate Editor for this article was H. A. Rakha. (Corresponding author: Henry X. Liu.)

Shuo Feng is with the Department of Automation, Tsinghua University, Beijing 100084, China, and also with the Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: s-feng14@mails.tsinghua.edu.cn).

Yiheng Feng and Shan Bao are with the University of Michigan Transportation Research Institute, Ann Arbor, MI 48109 USA (e-mail: yhfeng@umich.edu; shanbao@umich.edu).

Haowei Sun and Henry X. Liu are with the Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: haowei@umich.edu; henryliu@umich.edu).

Yi Zhang is with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: zhyi@mail.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TITS.2020.2988309

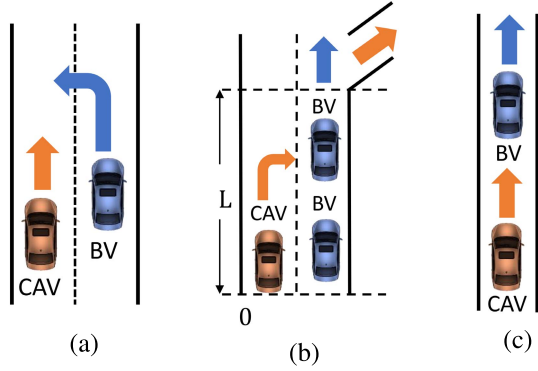


Fig. 1. An illustration of the three cases: (a) cut-in, (b) highway exit, and (c) car-following. BV denotes a background vehicle.

and our evaluation framework. Because the cut-in case is low dimensional, it is convenient to visualize the results and help readers better understand the proposed method. The highway exit case focuses on the functionality evaluation. Compared with safety evaluation, the major difference lies in the design of auxiliary objective function for the library generation, i.e., how to quantify the maneuver challenge regarding functionality. The car-following case is designed to show the ability of the proposed method under high-dimensional scenarios. To this end, the proposed method in Part I is enhanced by reinforcement learning (RL) techniques. The RL-enhanced method shows the powerful ability of the framework proposed in Part I in handling high-dimensional scenarios. For all three cases, results show that the proposed framework can effectively generate the testing scenario libraries and accelerate the CAV evaluation process by multiple magnitudes with the same evaluation accuracy, if compared with the on-road test method.

The rest of the paper is organized as follows. For the convenience of readers, Section II briefly revisits the proposed method in Part I. Section III studies the cut-in case for safety evaluation. Section IV studies the highway exit case for functionality evaluation. In Section V, RL-enhanced method is developed for the high-dimensional car-following case. The major advantages and limitations of the method are discussed in Section VI. Finally, Section VII concludes the paper.

II. REVISIT THE PROPOSED METHOD IN PART I

For the convenience of readers, in this section, we briefly revisit the proposed method in Part I [8] including problem formulation, library generation, and CAV evaluation. Notations of related variables are listed in Table I.

A. Problem Formulation

The goal of the proposed method is to generate a set of critical scenarios, which can be used to evaluate CAVs for certain performance indices. If an event of interest with CAVs is denoted as A (e.g., accident event), the performance of CAVs can be quantitatively evaluated by its occurrence

TABLE I
NOTATIONS OF VARIABLES

Variable	Notation
θ	Pre-determined parameters of scenarios in the operational design domain.
x	Decision variables of testing scenarios.
A	Event of interest (e.g., accident) with a CAV model.
S	Event of interest (e.g., accident) with a surrogate model.
\mathbb{X}	Feasible set of the decision variables.
Φ	Set of decision variables of critical testing scenarios.
$V(x \theta)$	Criticality value of a scenario determined by x and θ .
$N(\mathbb{X}), N(\Phi)$	Total number of scenarios of the set \mathbb{X}, Φ .
$\bar{P}(x_i \theta)$	Probability of sampling the scenario x_i in the generated library with pre-determined parameters θ .
ϵ	Exploration probability of ϵ -greedy sampling policy.
$\hat{P}(A \theta)$	Estimated probability of the event A with pre-determined parameters θ .
n	Total number of sampled testing scenarios.
R, \dot{R}	Range and range rate at the cut-in moment between the background vehicle and test CAV.
$R(t), \dot{R}(t)$	Range and range rate at time t between the background vehicle and test CAV.
ω	Weight parameter.
$d(x, \Omega)$	Normalized distance between scenario x and a high exposure frequency zone Ω .
W	Normalization factor.

probability (e.g., accident rate):

$$P(A|\theta) = \sum_{x \in \mathbb{X}} P(A|x, \theta) P(x|\theta), \quad (1)$$

where x denotes decision variables of scenarios, \mathbb{X} denotes the feasible set determined by the operational design domain (ODD), and θ denotes the pre-determined parameters under the ODD, such as road type, number of lanes, weather conditions, etc.

Essentially the on-road test is to measure $P(A|\theta)$ in the naturalistic driving environment as

$$\begin{aligned} P(A|\theta) &\approx \frac{1}{n} \sum_{i=1}^n P(A|x_i, \theta), \quad x_i \sim P(x|\theta), \\ &\approx \frac{m}{n}, \end{aligned} \quad (2)$$

where n denotes the total number of tests and m denotes the number of occurrences for event A . Here the scenario variables x_i follow the distribution from naturalistic driving data (NDD), i.e., $x_i \sim P(x|\theta)$. In this paper, either on-road tests or simulation of on-road tests with the naturalistic driving environment is referred as NDD evaluation. Because the event A in NDD evaluation is usually rare, the required number of tests is intolerably large with reasonable precision [10].

To mitigate this issue, importance sampling techniques were applied by [4] as

$$\begin{aligned} P(A|\theta) &= \sum_{x \in \mathbb{X}} \frac{P(A|x, \theta) P(x|\theta)}{q(x)} q(x), \\ &\approx \frac{1}{n} \sum_{i=1}^n \frac{P(A|x_i, \theta) P(x_i|\theta)}{q(x_i)}, \quad x_i \sim q(x), \end{aligned} \quad (3)$$

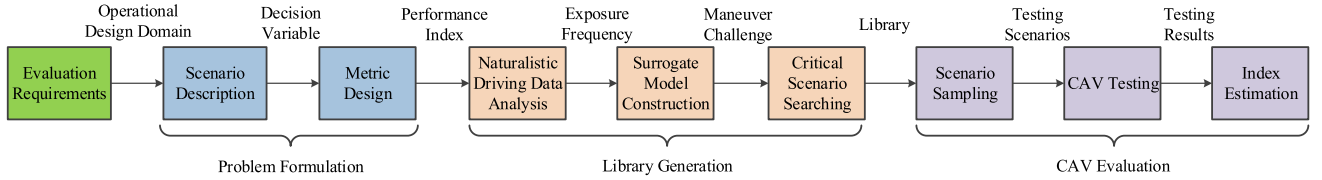


Fig. 2. An illustration of the general procedure for CAV evaluation.

where $q(x)$ denotes an importance function. According to importance sampling theory [11], to obtain a certain estimation precision, the required number of tests (n) is determined by the importance function, and how to construct a proper importance function remains a challenge. A properly constructed importance function is essential to achieving the goal of library generation.

B. Library Generation

The basic idea of library generation is to define the criticality of scenarios and search critical scenarios to construct the library. The criticality of a scenario is defined as

$$V(x|\theta) \stackrel{\text{def}}{=} P(S|x, \theta)P(x|\theta), \quad (4)$$

where S denotes the event of interest with a surrogate model (SM) of CAVs, and $P(S|x, \theta)$ is the occurrence probability of S in scenario (x, θ) . As shown in Eq. (4), the scenario criticality is expressed as a combination of maneuver challenge ($P(S|x, \theta)$) and exposure frequency ($P(x|\theta)$). Since too many local optimal solutions exist in the criticality function, naive search for the critical scenarios is inefficient. To solve this issue, the multi-start optimization and seed-fill based searching method is applied, where an auxiliary objective function is designed to provide searching directions. The SM and the auxiliary objective function will be discussed case by case.

C. CAV Evaluation

After the generation of library, testing scenarios are sampled from the library with ϵ -greedy policy, and the performance index ($P(A|\theta)$) is estimated based on the testing results. A minimal number of tests are required for achieving certain estimation precision.

The sampling distribution with ϵ -greedy policy is derived as

$$\bar{P}(x_i|\theta) = \begin{cases} (1 - \epsilon)V(x_i|\theta)/W, & x_i \in \Phi \\ \epsilon/(N(\mathbb{X}) - N(\Phi)), & x_i \in \mathbb{X} \setminus \Phi \end{cases} \quad (5)$$

where $N(\mathbb{X})$ denotes the total number of feasible scenarios, Φ denotes the set of critical scenarios, the selection of ϵ is theoretically analyzed (see Corollary 1 in Part I paper), and W is a normalization factor as

$$W = \sum_{x_i \in \Phi} V(x_i|\theta). \quad (6)$$

After testing the CAV with sampled scenarios, the index $P(A|\theta)$ can be estimated as

$$\hat{P}(A|\theta) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^n \frac{P(x_i|\theta)}{\bar{P}(x_i|\theta)} P(A|x_i, \theta), \quad (7)$$

where $P(A|x_i, \theta)$ is estimated by the testing results.

The minimal number of tests is shown as follows. The testing process stops if the relative half-width of the estimation is less than a pre-determined threshold β [4], [12], [13] as

$$\frac{z_\alpha}{\hat{\mu}} \sqrt{\text{Var}(\hat{\mu})} \leq \beta, \quad (8)$$

where z_α is a constant with the confidence level at $100(1 - \alpha)\%$, $\hat{\mu} = \hat{P}(A|\theta)$ is the estimation of the index, and $\text{Var}(\hat{\mu}) = \sigma^2/n$ is the estimation variance, which decreases with increasing number of tests, i.e., n .

III. CUT-IN CASE STUDY

Before we present the cut-in case study, a general implementation procedure for CAV testing is provided in Fig. 2. For all three case studies, we will present in the order of problem formulation, library generation, and CAV evaluation.

For the cut-in case shown in Fig. 1 (a), the decision variables and the performance index are formulated in Subsection III.A. In Subsection III.B, an auxiliary objective function is designed for critical scenario searching, NDD is analyzed to provide the exposure frequency, and SM is constructed to measure the maneuver frequency. With the generated library, a typical CAV is tested and evaluated in Subsection III.C.

A. Problem Formulation

Similar to most existing studies [3], [4], the decision variables of the cut-in case are simplified as two dimensions, i.e.,

$$x = [R, \dot{R}]^T,$$

where R and \dot{R} denote the range (the longitudinal distance between the rear bumper of the preceding vehicle and the front bumper of the ego vehicle) and range rate (the longitudinal speed difference) at the cut-in moment. For this simplification, the background vehicle (BV) is assumed to keep constant velocity after the cut-in behavior, and parameters of road environments are pre-determined. All these pre-determined parameters are denoted as θ . The accident rate is utilized to measure the safety performance of CAVs in the cut-in case. The on-road test method is simulated to estimate the accident rate as a baseline.

B. Library Generation

To implement the library generation method, three questions need to be answered specifically, i.e., auxiliary objective function design, NDD analysis, and SM construction.

TABLE II
THE AUXILIARY OBJECTIVE FUNCTION PARAMETERS
IN THE CUT-IN CASE

Parameter	Value	Parameter	Value
m_d	2	U_I	100
$U_{F,1}$	18	$U_{F,2}$	20
w	1.0	-	-

1) *Auxiliary Objective Function Design*: To provide searching directions for critical scenarios, an auxiliary objective function is designed as the combination of estimated maneuver challenge and exposure frequency.

The maneuver challenge is estimated by **minimal normalized positive enhanced time-to-collision** (mnpETTC). As discussed in [14], [15], ETTC is one of the most widely used indices of safety evaluation for varying velocity scenarios, and it is defined as

$$ETTC(t) = \frac{-\dot{R}(t) - \sqrt{\dot{R}^2(t) - 2u_r(t)R(t)}}{u_r(t)}, \quad (9)$$

where $R(t)$ and $\dot{R}(t)$ are the range and range rate at time t , and $u_r(t)$ is the relative acceleration. Values of ETTC for different scenarios can be obtained by simulations. To make the index comparable, a normalization factor is applied, denoted as U_I , which is calibrated by NDD analysis. Negative values of ETTC, which denote safe situations, will be set to one. Then the minimal normalized positive ETTC (mnpETTC) can be calculated as

$$mnpETTC(t) = \min_t npETTC(t), \quad (10)$$

where

$$npETTC(t) = \begin{cases} ETTC(t)/U_I, & ETTC(t) \geq 0 \\ 1, & ETTC(t) < 0 \end{cases} \quad (11)$$

The exposure frequency of a scenario is estimated by the distance between the scenario and a common set (i.e., scenarios with high exposure frequency). The common set is determined by NDD analysis, and the distance is defined as

$$\begin{aligned} d(x, \Omega) &= \min_{y \in \Omega} d(x, y), \\ &= \min_{y \in \Omega} \sqrt{\frac{1}{m_d} \sum_{i=1}^{m_d} \frac{(x_i - y_i)^2}{U_{F,i}^2}}, \end{aligned} \quad (12)$$

where Ω denotes the common set, m_d is the dimension of the decision variables, and $U_{F,i}$ is the normalization factor for the i -th dimension, which is calibrated by NDD analysis.

The auxiliary objective function for safety evaluation in the cut-in case is formulated as

$$\min_x J(x) = \min_x (mnpETTC(x) + w \times d(x, \Omega)), \quad (13)$$

where $w \in (0, 1]$ is a balance weight. Note the goal of the auxiliary objective function is to provide searching directions, so certain roughness (e.g., caused by w) is reasonable and acceptable. The values of the parameters for the auxiliary objective function are listed in Table II.

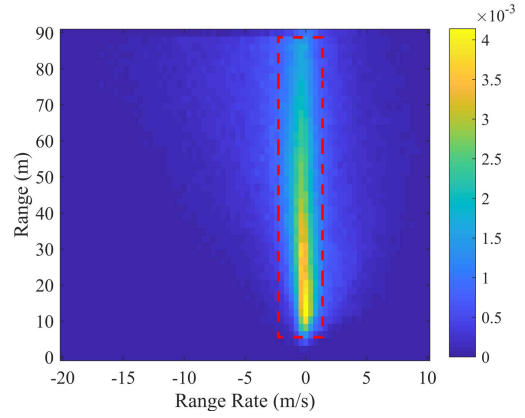


Fig. 3. The exposure frequency (i.e., $P(x|\theta)$) of the cut-in range and range rate. The dashed red rectangle denotes the boundary of the common set (i.e., Ω).

2) *NDD Analysis*: NDD is analyzed to provide exposure frequency measurement, determine parameters of the auxiliary objective function, and calibrate the SM.

The NDD from the Safety Pilot Model Deployment (SPMD) program at University of Michigan [16] is utilized for the cut-in case. The SPMD database is one of the largest databases in the world that recorded naturalistic driving behaviors over 34.9 million travel miles from 2,842 equipped vehicles in Ann Arbor, Michigan. In the database, there are 98 sedans equipped with the data acquisition system and MobilEye, which enables the measurement of both longitudinal and lateral distances between the ego vehicle, preceding vehicles, and lane markings, at a frequency of 10 Hz. By analyzing these lateral distances, cut-in events can be identified. For each cut-in event, the cut-in moment is determined by the time instant when the preceding vehicle crosses the lane marking, and the range and range rate at that moment are recorded for the NDD analysis. In this paper, the following query criteria [4], [17] are designed to extract all cut-in events from the database: (a) the vehicles' speeds at the cut-in time belong to $(2m/s, 40m/s)$; (b) the range at the cut-in time belongs to $(0.1m, 90m)$. As a result, 414,770 qualified cut-in events are successfully obtained. The exposure frequency distribution (i.e., $P(x|\theta)$) is shown in Fig. 3, where brighter color denotes higher exposure frequency. The range and range rate are discretized by $2m$ and $0.4m/s$ respectively. The NDD evaluation method is equivalently sampling testing scenarios from this probability distribution.

We now discuss how to determine the parameters in the auxiliary objective function. First, the common set Ω can be determined by finding a minimal rectangle or hyper-rectangle of scenarios with high probabilities (i.e., $P(x|\theta) > 10^{-3}$). As shown in Fig. 3, the dashed red rectangle denotes the boundaries of the common set in the cut-in case (i.e., $[6, 88]$ for range and $[-2.4, 1.2]$ for range rate). For more complex scenarios, the common set can be further simplified as the most frequent scenario, e.g., $R = 14, \dot{R} = 0$. Second, the normalization factors are determined by the maximal distance between scenarios and the common set. For example, the maximal range rate between scenarios and the common set

TABLE III
THE SURROGATE MODEL PARAMETERS IN THE CUT-IN CASE

Parameter	Value	Parameter	Value
v_{max}	40 m/s	v_{min}	2 m/s
a_{min}	-4 m ² /s	a_{max}	2 m ² /s
α_{IDM}	2	β_{IDM}	18
c_{IDM}	4	s_0	2
L_{IDM}	4	T	1
b_{IDM}	3	d_{acci}	1 m

is smaller than 18, so the normalization factor of the range rate is set to 18.

3) *Surrogate Model Construction*: SM construction is a very important step in the library generation process. It represents what we know about the generic features of CAVs. The “generic features” capture the basic behavior of a CAV, e.g., keep safe distances with surrounding vehicles. Similar to human drivers, where different drivers have different driving habits, generic features exist among all drivers. An ideal SM should be calibrated from actual CAV driving data similar to human driving model calibration [18]. At the current stage, however, there is very little open CAV data available for public research. Therefore, we propose to calibrate the SM based on the human driving data, i.e., NDD.

In this case study, a calibrated intelligent driving model (IDM) [19] is selected as the SM for the car-following behaviors after the cut-in event:

$$u(k+1) = \alpha_{IDM} \left(1 - \left(\frac{v(k)}{\beta_{IDM}} \right)^{c_{IDM}} - \left(\frac{s_{IDM}(v(k), \dot{R}(k))}{R(k) - L_{IDM}} \right)^2 \right), \quad (14)$$

where k denotes the discrete time step, u denotes the acceleration, α_{IDM} , β_{IDM} , c_{IDM} , L_{IDM} are constant parameters, and

$$s_{IDM}(v(k), \dot{R}(k)) = s_0 + v(k)T + \frac{v(k)\dot{R}(k)}{2\sqrt{\alpha_{IDM}b_{IDM}}}, \quad (15)$$

where s_0 , b_{IDM} , and T are constant parameters. Similar to [20], the constraints of acceleration and velocity are added to make the model more practical (i.e., model accident-prone behaviors) as

$$v_{min} \leq v \leq v_{max}, \quad a_{min} \leq u \leq a_{max}. \quad (16)$$

An accident event is defined as the vehicle range is smaller than a threshold, i.e., $R(t) < d_{acci}$. The calibrated values are listed in Table III. Fig. 4 shows the safety performance of the selected SM, where the SM has accidents in scenarios of the yellow region.

4) *Library Generation*: The optimization and seed-fill based method proposed in Part I of this study [8] is applied to search for critical scenarios and construct the library. In this case, 50 points are uniformly sampled as the initial starting points. As discussed in Corollary 2 in Part I, the threshold of critical scenarios is determined as

$$\gamma = \frac{m}{N(\mathbb{X}) - N(\Phi)} \approx \frac{m}{N(\mathbb{X})} = 2.9 \times 10^{-4}, \quad (17)$$

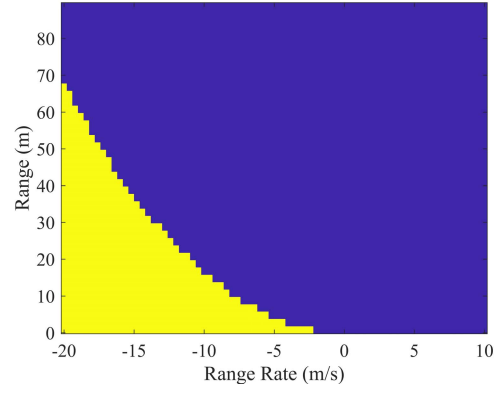


Fig. 4. The safety performance of the SM, where the SM has accidents in scenarios of the yellow region.

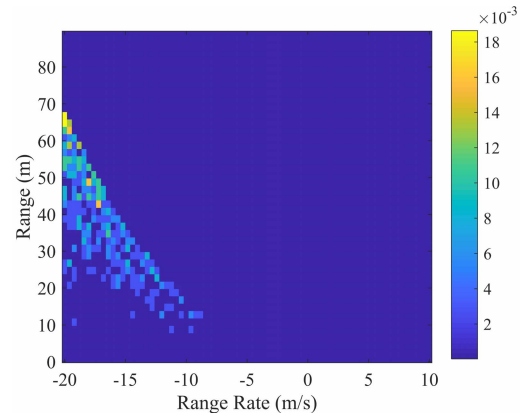


Fig. 5. The generated library of the cut-in case for safety evaluation. The color denotes the testing probability of scenarios.

where $m = 1$, and $N(\mathbb{X}) = 47 \times 76 = 3,420$. The discretization intervals of the range and range rate are $2m$ and $0.4m/s$, and the boundaries of the range and range rate are $(0, 90]$ and $[-20, 10]$ respectively.

Fig. 5 shows the obtained probability distribution after the library generation process. The color denotes the probability of a scenario, which is calculated by the normalized criticality. Compared with Fig. 3, where only exposure frequency is considered, the new distribution considers both the maneuver challenge and exposure frequency of scenarios. All scenarios with criticality values exceeding the threshold γ will be included in the library. In this case, the generated library contains a total number of 184 scenarios, which is about 5.38% of all scenarios.

C. CAV Evaluation

In this step, a specific CAV is evaluated with the generated library. For field implementations, a real CAV should be tested. In this paper, a simulated CAV model is used as a proof of concept to validate the proposed method.

The simulation model used in [4], which combines adaptive cruise control and autonomous emergency braking functions, is adopted in this case. The NDD evaluation method is applied as the baseline, where testing scenarios are sampled from the

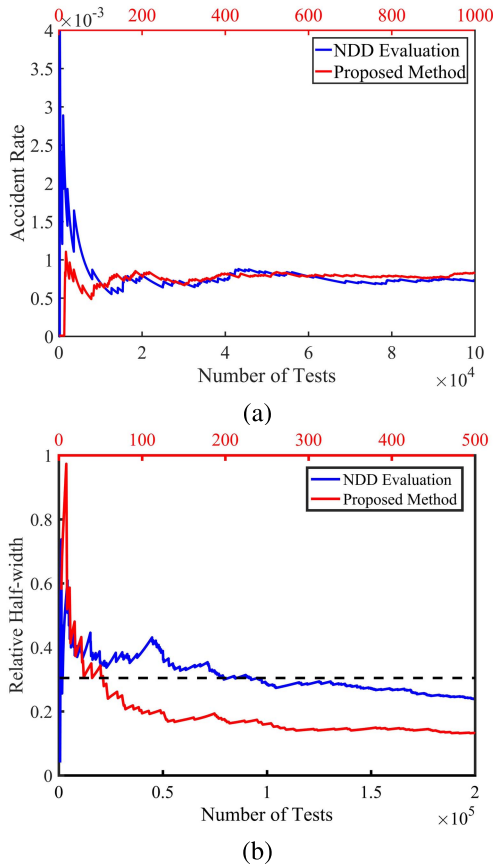


Fig. 6. Results of the cut-in case: (a) estimation results of the accident rate; (b) relative half-width of the estimation results.

NDD distribution in Fig. 3. For the proposed method, testing scenarios are sampled from the generated library in Fig. 5. The ϵ -greedy sampling policy is applied with $\epsilon = 0.05$, which is determined according to Corollary 1 in Part I paper [8]. The chosen CAV model is tested in the sampled scenarios, and the accident event is recorded.

Fig. 6 shows the comparison of the two evaluation methods. The blue line represents the results of the NDD evaluation method, and the bottom x -axis represents its number of tests. The red line represents the results of the proposed method, and the top x -axis represents its number of tests. As shown in Fig. 6, both methods can obtain the accurate estimation of the accident rate for a pre-determined relative half-width. In this paper, we set $\alpha = 0.95$ and $\beta = 0.3$. Fig. 6 (b) shows that the proposed method achieves this confidence level after 51 tests, while the NDD evaluation method needs 9.63×10^4 tests. The proposed method is about 1,888 times faster than the NDD evaluation method.

IV. HIGHWAY EXIT CASE STUDY

As shown in Fig. 1 (b), for the highway exit case, the test CAV needs to make a lane change to the right and exits the highway within a certain distance. Compared with safety evaluation, the major difference of functionality evaluation lies in the design of auxiliary objective function for critical scenario searching. To this end, several new concepts are

TABLE IV
THE INITIAL PROBLEM SETTINGS FOR THE HIGHWAY EXIT CASE

Parameter	Value	Parameter	Value
p_0	0 m	v_0	30 m/s
d_{cf}	2 m	$p_{0,i}$	$[-100, 200]$
$v_{0,i}$	$[20, 40]$	Δt	0.1 s
Δp	5 m	-	-

proposed, i.e., task, task solution, task solution difficulty, and task difficulty. Based on these concepts, an auxiliary objective function is designed.

A. Problem Formulation

The decision variables of the highway exit scenario include initial states of the CAV, number of BVs, and trajectories of each BV, which are high-dimensional. To simplify the problem and focus on the functionality evaluation, the initial position and velocity of the CAV are pre-determined as p_0 and v_0 , and only two BVs are considered. The two BVs will keep their initial velocity unless the distance between them is less than a threshold d_{cf} . In such case, the following BV will change its speed to be the same as the leading BV. As a result, the decision variables are formulated as

$$x = [p_{0,1}, v_{0,1}, p_{0,2}, v_{0,2}]^T, \quad (18)$$

where $p_{0,i}$, $v_{0,i}$ denote the initial position and velocity of the i -th BV. The discrete interval of time and position is chosen as Δt and Δp respectively. In this case study, the initial problem settings are summarized in Table IV.

B. Library Generation

The library generation methods are the same as those presented in the cut-in case. To make the paper concise, only the auxiliary objective function design for the functionality evaluation will be elaborated.

1) *Auxiliary Objective Function Design*: Similar to the cut-in case, the auxiliary objective function is composed of estimated exposure frequency and maneuver challenge. To evaluate the maneuver challenge for generic functionality, four new concepts are proposed, i.e., task, task solution, task solution difficulty, and task difficulty. The “task” is defined based on the functionality, e.g., exit from the highway. The “task solution” f denotes a feasible CAV trajectory to complete the task, i.e., $f \in \mathbb{F}$. \mathbb{F} represents the feasible set of CAV trajectories. The “task solution difficulty” denotes the difficulty in completing the task solution, i.e., $W(f)$, where $W(f)$ is negative and larger $W(f)$ denotes higher difficulty. Finally, the “task difficulty” denotes the difficulty of the task, which can be evaluated by the summation of all task solution difficulties as

$$M_f(x) = \sum_{f \in \mathbb{F}} W(f). \quad (19)$$

This definition can represent both the difficulty in finding a feasible solution to the task and completing that solution.

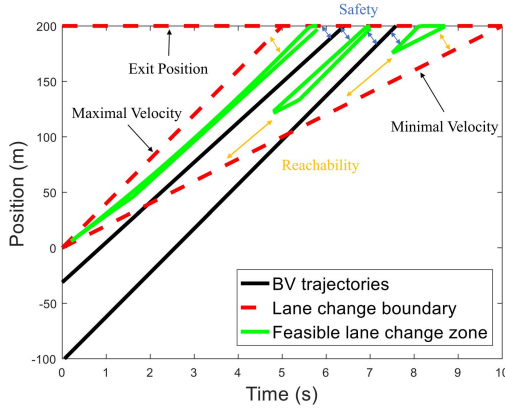


Fig. 7. Illustration of the task difficulty evaluation of the highway-exit case.

For the specified highway exit case, the maneuver challenge is evaluated based on the proposed concepts. The task is to make a lane change to the right before reaching the off-ramp location. The task solution is defined as a feasible lane-change point $f = (t, p)$, where t is the lane-change time and p is the lane-change position. The feasible lane-change zone \mathbb{F} is determined by maximal/minimal velocity (v_{max}, v_{min}), highway exit location (L), safe time-to-collision gaps (t_{min}), and maximal/minimal acceleration (a_{max}, a_{min}). Fig. 7 illustrates an example of the feasible lane change zone for a specific scenario, i.e., $x = [-25, 34.5, -100, 40]^T$. The initial position of the CAV is set zero. The lane change boundary (shown as the red dashed line) is determined by the maximal/minimal velocity and the off-ramp location. The feasible lane change zone (shown as the green lines), i.e., \mathbb{F} , consists of three isolated zones, which are separated by the trajectories of BVs (shown as the black lines).

For simplicity, we assume all task solutions of this case have the same task solution difficulty. Then, the task difficulty can be estimated as

$$M_f(x) = \sum_{f \in \mathbb{F}} W(f) = -S(\mathbb{F}), \quad (20)$$

where $S(\mathbb{F})$ denotes the area of the feasible lane-changing zone. To make the index comparable with exposure frequency, a normalization factor is applied, denoted as U_S , which can be obtained by the area enclosed by the lane change boundary. Finally, the auxiliary objective function of the highway exit case is designed as

$$\min_x J(x) = \min_x (S(\mathbb{F})/U_S + w \times d(x, \Omega)), \quad (21)$$

where w is the weight, and $d(x, \Omega)$ can be obtained similarly as in the cut-in case (Eq. (12)). The common set (Ω) in this case can be constructed by the most frequent scenarios. The parameter values of the auxiliary objective function are listed in Table V.

2) *NDD Analysis*: The NDD from the Integrated Vehicle-Based Safety System (IVBSS) project is used to provide exposure frequency information [21]. In the IVBSS project, 108 randomly sampled drivers from different ages used sixteen Honda Accord vehicles in an unsupervised manner for over

TABLE V
THE PARAMETERS FOR THE HIGHWAY EXIT CASE

Parameter	Value	Parameter	Value
a_{max}	$2 \text{ m}^2/\text{s}$	a_{min}	$-4 \text{ m}^2/\text{s}$
L	200 m	w	1
t_{min}	0.5 s	t	$[0, 10]$
p	$[0, 200]$	U_S	500

40 days. In this paper, the exposure frequency of highway exit scenarios is determined by the car-following events of the two BVs at the rightmost lane. Query criteria are designed to extract car-following events from the database as: (1) vehicle was traveling on a highway; (2) vehicle was traveling at a speed of at least 20 m/s ($\approx 45 \text{ mph}$); (3) cruise control function was not activated; (3) dry surface condition; (4) day light condition. The resulting dataset represents a total of 5×10^4 car-following events and 1.47×10^6 points of car-following trajectories. The exposure frequency of a highway exit scenario can be estimated as

$$P(x|\theta) = P(p_{0,1}|\theta)P(v_{0,1}, R, v_{0,2}|\theta), \quad (22)$$

where $R = p_{0,1} - p_{0,2}$, $P(p_{0,1}|\theta)$ denotes the initial position probability of the leading BV, which can be estimated by uniform distribution, and $P(v_{0,1}, R, v_{0,2}|\theta)$ is obtained from the distribution of car-following trajectories in the NDD.

3) *SM Construction*: The MOBIL ('minimizing overall braking induces by lane changes') model was proposed by [22] to derive human lane-changing rules for discretionary and mandatory lane changes. It provides the utility measurement method for deciding which gap has a desirable lane change position as

$$U_{LG} = \tilde{u} - u + p_{LG}(\tilde{u}_{new} - u_{new} + \tilde{u}_{old} - u_{old}), \quad (23)$$

where \tilde{u} denotes the new acceleration of the CAV after the lane change, p_{LG} is the politeness factor, and u_{new}, u_{old} denote the acceleration of the new follower and old follower respectively. As it is desirable to complete the lane change, the politeness factor is set close to zero, e.g., $p_{LG} = 0.1$. To predict the CAV's trajectories before the lane-change, the Model Predictive Control (MPC) [23] is applied, and the trajectory with higher predictive utility of lane change, i.e., U_{LG} , will be chosen as the solution to the task.

4) *Library Generation*: Similar to the cut-in case, a hundred points are uniformly sampled as the initial starting points for the optimization method, and the threshold of critical scenarios is determined as

$$\gamma = \frac{1}{N(\mathbb{X})} = 6.1 \times 10^{-7}, \quad (24)$$

similar to Eq. (17). The size of total scenarios is $N(\mathbb{X}) = n_p^2 \times n_v^2 = 1.64 \times 10^6$, where $n_p = 61$ and $n_v = 21$ denote the number of the feasible value of variables $p_{0,i}$ and $v_{0,i}$ respectively. After applying the critical scenario searching method, the testing scenario library of the highway exit case is generated. The total number of critical scenarios in the library is 1,895, which is about 0.12% of all scenarios.

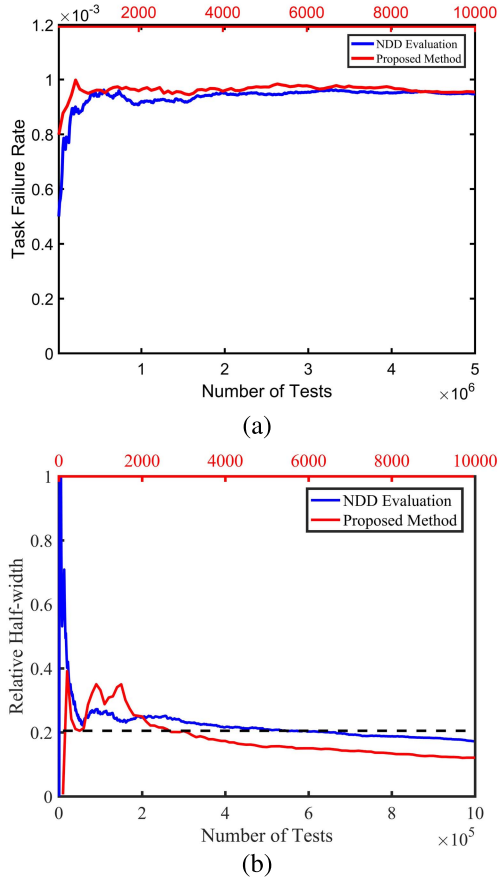


Fig. 8. Results of the highway exit case: (a) estimation results of the task failure rate; (b) relative half-width of the estimation results.

C. CAV Evaluation

The CAV lane-change model developed in [24] is used for evaluation in this case study. Similarly, the NDD evaluation method is used as the benchmark. In the proposed method, testing scenarios are sampled from the generated highway exit library, and events of task failures (i.e., cannot exit from the highway) are recorded. Similar to the cut-in case, the ϵ -greedy sampling policy is applied with $\epsilon = 0.10$. The task failure rate is estimated to measure the functionality performance of the CAV model in the highway exit case. After the estimated task failure rate converges to a certain estimation precision, the estimated task failure rate is obtained, and the evaluation process is completed, as shown in Eq. (8).

Fig. 8 shows the comparison of the two evaluation methods. The legends and axis are the same as those in the cut-in case. Similar to the previous case study, both methods can obtain the unbiased estimation of the failure rate with the relative half-width ($\beta = 0.2$). Fig. 8 (b) shows that the proposed method achieves this estimation precision after 2.6×10^3 tests, while the NDD evaluation method takes 6.6×10^5 tests. The proposed method is about 255 times faster than the NDD evaluation method.

V. CAR-FOLLOWING CASE STUDY

The car-following case is designed to show the ability of the proposed method in solving the TSLG problem with

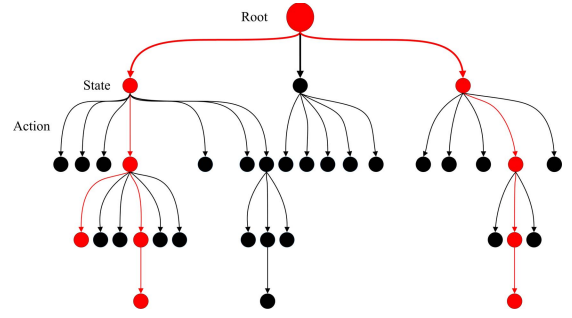


Fig. 9. Illustration of the generated library (i.e., red circles and red arrows) in the scenario space.

high-dimensionality. As shown in Fig. 1 (c), the test CAV follows a BV for a certain period. The decision variables include the initial condition and acceleration profile of the leading BV:

$$x = [v_0, R_0, \dot{R}_0, u_1, u_2, \dots, u_m]^T, \quad x \in \mathbb{X} \quad (25)$$

where v_0 denotes the initial velocity of the leading BV, R_0 and \dot{R}_0 denote the initial range and range rate between the BV and CAV, m denotes the total time steps, and u_1, u_2, \dots, u_m denote the acceleration sequences of the BV. If the BV is controlled every 1s, for a 30s car-following scenario, the dimension of the scenario is 33. Since the computation complexity grows exponentially with the dimension, the problem faces “curse of dimensionality”. To the best of our knowledge, none of the existing methods can be applied to resolve the TSLG problem for high-dimensional cases.

The key to handle high-dimensionality is to formulate the TSLG problem as a Markov Decision Process (MDP) problem. Let $s = (v_{BV}, R, \dot{R}) \in \mathcal{X}$ denote the state, where v_{BV} denotes the speed of the BV, R is the range, \dot{R} is the range rate, and \mathcal{X} is the feasible set of states. Let $u \in \mathbb{U}$ denote the action, where \mathbb{U} is the feasible acceleration set of the leading BV. It is assumed that the Markovian property holds considering the next action is dependent only on the current state, i.e., the acceleration of the BV is dependent only on its current speed. Then a testing scenario x in Eq. (25) can be described as a series of states and actions (i.e., $s_1 \xrightarrow{u_1} s_2 \xrightarrow{u_2} \dots$), and the set of feasible scenarios \mathbb{X} can be represented by the decision tree, as shown in Fig. 9. Every branch from the initial state to the terminal state (i.e., leaf node) specifies a testing scenario. The library generation problem now turns into the problem of finding the critical branches for CAV evaluation.

In order to compute the criticality of each scenario, here we define the value of each state-action pair $Q(s, u)$ as

$$Q(s_k, u_k) = P(S|u_k, s_k)P(u_k|s_k), \quad k = 1, \dots, m. \quad (26)$$

The definition of Q is consistent with the proposed definition of criticality. As shown in Eq. (26), the left term $P(S|u_k, s_k)$ represents “the probability of the event S if the scenario is currently at the state s_k and take the action u_k ”, which measures the maneuver challenge. The right term $P(u_k|s_k)$ represents “the probability of taking action u_k if the scenario is currently at the state s_k ”, which is the exposure frequency.

With such definition, we can prove that the scenario criticality can be computed as

$$V(x) = C(x) \prod_{k=1}^m Q(s_k, u_k),$$

where $C(x)$ is a normalization factor of the scenario x , and θ is omitted to simplify notations. The proof of this equation can be found in Theorem 1.

To obtain $Q(s, u)$, the temporal-difference (TD) reinforcement learning (RL) technique is applied. BVs are the “agent” of the RL scheme, and the state of test CAV is the “environment”, which is represented by a deterministic SM. The state transition is influenced by both BVs and the deterministic SM. The TD-RL method updates the $Q(s_k, u_k)$ based on the estimation of the next state value $Q(s_{k+1}, u_{k+1})$ (i.e., the TD(0) method in [25]). The iterative update is based on the TD error, which measures the difference between the current estimation of $Q(s_k, u_k)$ and the new estimation. Let δ_k denote the TD error at the time step k , then an iteration equation can be obtained as

$$Q(s_{k+1}, u_{k+1}) \leftarrow Q(s_k, u_k) + \alpha \delta_k,$$

where α is the learning rate, e.g., 0.1. In Theorem 2, we prove that, after the training process of TD-RL, $Q(s, u)$ can converge to the values defined in Eq. (26) if the TD error is defined as

$$\delta_k = \left(\sum_{u_{k+1} \in \mathcal{U}} Q(s_{k+1}, u_{k+1}) \right) P(u_k | s_k) - Q(s_k, u_k). \quad (27)$$

By pruning the uncritical state-action pairs of the decision tree, the critical scenario library contains all branches with $V(x|\theta) > 0$. As shown in Fig. 9, the branches of consecutive red nodes and arrows represent the critical scenarios.

A. Problem Formulation for the car-Following Case

In the car-following case, safety is selected as the performance metric and accident rate is used to represent safety. The scenario state contains three variables, i.e., speed of the leading BV (v_{BV}), range (R) between the leading BV and the test CAV, and the range rate (\dot{R}):

$$s = (v_{BV}, R, \dot{R}) \in \mathcal{X}. \quad (28)$$

The leading BV's acceleration (u) is defined as the action. We discretise the range ($R \in (0, 115]$), range rate ($\dot{R} \in [-10, 8]$), velocity ($v \in [20, 40]$), and acceleration ($u \in [-4, 2]$) by $1m$, $1m/s$, $1m/s$, and $0.2m^2/s$ respectively. The leading BV is controlled every $1s$. For a $30s$ car-following case, the size of the entire scenario space is $N(\mathbb{X}) = 21 \times 115 \times 19 \times 31^{30}$. The size of the entire state space is $N(\mathcal{X}) = 21 \times 115 \times 19 = 45,885$, and the size of the entire state-action space is $N(\mathcal{X}) \times N(\mathcal{U}) = 45,885 \times 31 \approx 1.4 \times 10^5$, both of which are much smaller than the entire scenario space $N(\mathbb{X})$.

B. Library Generation

The same NDD of the highway exit case is used in the car-following case, where both the car-following and free-driving events are extracted. The car-following events are utilized to

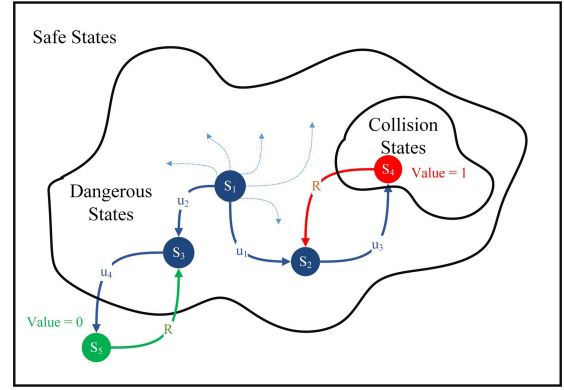


Fig. 10. Illustration of the state transitions for car-following scenarios. States are classified into three zones, i.e., collision zone, dangerous zone, and safe zone.

calculate the exposure frequency of states $P(s)$, while the free-driving events are utilized to estimate the exposure frequency of actions $P(u|s)$.

To improve the searching efficiency of critical scenarios, the state space is classified into three zones, i.e., collision zone, dangerous zone, and safe zone. The collision zone is defined by the states $\mathcal{X}_c = \{s \in \mathcal{X} | R \leq d_{acci}\}$, where d_{acci} is a distance threshold for an accident, e.g., $1m$. The safe zone \mathcal{X}_s is defined by the states which cannot lead to an accident with even the most extreme actions of the leading BV, i.e., BV decelerates with maximal deceleration. The dangerous zone \mathcal{X}_d contains the states which have probabilities leading to an accident. Then values of $P(S|u_k, s_k)$ for states in different zones can be obtained as

$$P(S|u_k, s_k) = \begin{cases} 0, & s_k \in \mathcal{X}_s \\ 1, & s_k \in \mathcal{X}_c \end{cases}$$

As shown in Fig. 10, a non-trivial car-following testing scenario should start from a dangerous state (i.e., root state) and stop at a collision state or a safe state (i.e., terminal state). The critical scenario should contain the state in the collision zone. The same car-following SM is applied as in Eq. (14). By simulations of the SM, the dangerous zone is obtained, which consists of about 5,000 states (10% of all the states).

The initial Q values are obtained from the NDD. To improve the training efficiency, a uniform distribution is applied as the training policy, which guarantees that all state-action pairs can be visited for unlimited number of times if the training has not stopped [25]. The absolute value of the TD error is defined as the stop criteria as $|\delta_t| < \delta_0$, where δ_0 is a pre-determined threshold, e.g., 10^{-10} .

The training is conducted with Matlab 2018, in a workstation equipped with Intel i7-7700 CPU and 16G RAM. It takes about 20 minutes to reach convergence. Fig. 11 (a) shows the convergence of the absolute TD error with learning iterations. The values of state-action pairs converge after about 3×10^6 steps of iterations. Fig. 11 (b) shows an example of the probability distribution of the actions for a dangerous state, i.e., $s = (38, 6, -2)$. The distribution from NDD is represented as the blue line (i.e., $P(u|s)$), while the generated

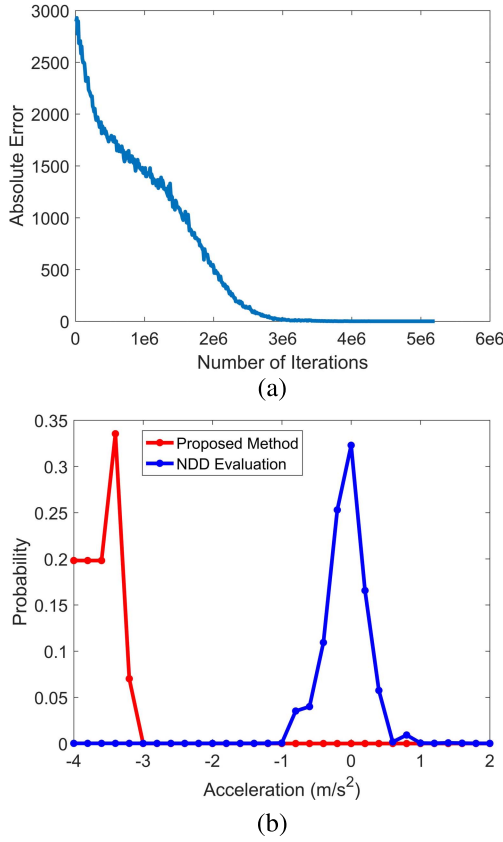


Fig. 11. (a) The training results of the absolute TD error and (b) probability distribution of the actions at the state $s = (38, 6, -2)$.

distribution by the RL-enhanced method is represented as the red line (i.e., $P(u|s, S)$). It shows that the generated distribution behaves more aggressively than NDD with higher probabilities at extreme decelerations. The highest probability lies in $u = -3.4 \text{ m/s}^2$, instead of $u = -4 \text{ m/s}^2$, which is consistent with the proposed definition of criticality combined of both maneuver challenge and exposure frequency.

C. CAV Evaluation

After the above steps, the testing scenario library of the car-following case is generated. Testing scenarios can be sampled from the scenario library. The initial state is generated by Eq. (32), and accelerations of the BV are generated by Eq. (34). Similar to the previous cases, the ϵ -greedy sampling policy is applied in the sampling process with $\epsilon = 0.1$. As shown by the red line in Fig. 11 (b), the probability of acceleration greater than -3 is zero, i.e., out of the library. By adopting the ϵ -greedy policy, however, these acceleration values can be sampled with a small probability. Similarly, the initial state has a small probability to be sampled from the safe states as well. The same CAV car-following model used in the cut-in case study [4] is evaluated with the generated library. The NDD evaluation method is used as the baseline.

Fig. 12 shows comparison of the two evaluation methods. The blue line represents results of the NDD evaluation method, and the red line represents results of the proposed method. As shown in Fig. 12, both methods can obtain the accurate

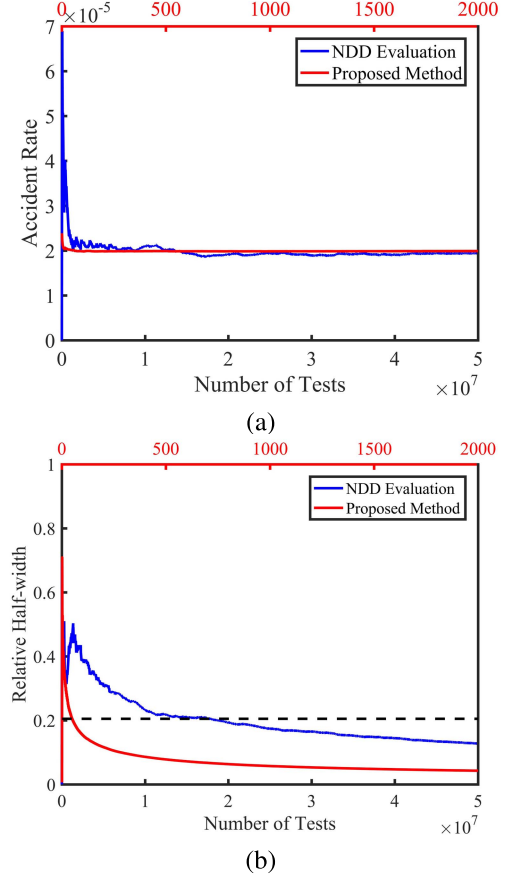


Fig. 12. Results of the car-following case: (a) estimation results of the accident rate; (b) relative half-width of the estimation results.

estimation of accident rate with the same estimation precision ($\beta = 0.2$). Fig. 12 (b) shows that the proposed method achieves this estimation precision after 50 tests, while the NDD evaluation method takes 1.875×10^7 tests. The proposed method is about 3.75×10^5 times faster than the NDD evaluation method.

VI. DISCUSSIONS

In this section, based on the results of the three case studies, the advantages and limitations of the proposed method are discussed.

A. Advantages of the Proposed Method

As demonstrated by the case studies, the proposed method is generic as it can be applied for evaluation with different performance metrics (e.g., safety and functionality) and varying CAV models under different ODDs. It can accelerate the CAV testing for both low and high dimensional scenarios.

As demonstrated by all three cases, to reach the required evaluation accuracy, the proposed method can significantly reduce the number of tests, comparing with the on-road test method. Because the most time-consuming and expensive step in the CAV evaluation process is expected to be vehicle testing, the proposed method can significantly save cost. Similarly, the method can also be applied in simulation platforms with

the same advantage, because testing CAVs in high-fidelity simulations is also the most time-consuming step.

Statistical coverage of scenarios is guaranteed by the proposed method, as all scenarios have possibilities to be tested. It utilizes more domain knowledge (e.g., scenario criticality) and outperforms the enumerative coverage (e.g., enumeration of all possible scenarios with a certain resolution), which suffers from the “curse of dimensionality”.

In addition, the performance index, $P(A|\theta)$, can quantitatively and interpretively measure the performance of CAVs in naturalistic driving environment. Taking safety evaluation as an example, the accident rate is the most natural index to evaluate the safety performance.

B. Limitations of the Proposed Method

As indicated in Theorem 2 in the Part I paper, the efficiency of the proposed method is affected by the “dissimilarity” between the SM and the CAV under test, i.e., $P(A|x, \theta) - P(S|x, \theta)$. This is the major reason why the acceleration effects of the three cases are different, from 255 to 3.75×10^5 times. In this paper, the commonly-used human driving models (such as the IDM in the cut-in case and car-following case) were adopted as SMs, which perform reasonably well and can be served as a starting point for testing library generation. To address the dissimilarity issue, adaptive modification of the SM for different CAVs deserves more investigation [26]. For CAV developers, this problem can be solved naturally as the CAV model can be used directly for scenario generation.

Large-scale NDD is required to calculate the exposure frequency of scenarios, which could be a limitation to apply the proposed method. With the deployment of on-board and infrastructure-based sensors, however, large-scale NDD can be collected with lower cost and become more accessible. For example, both research institutes [27] and companies (e.g., Waymo, Tesla, Mobileye, *etc.*) are collecting such data.

VII. CONCLUSION

This paper complements the general TSLG methodology developed in the Part I paper by providing three example case studies, i.e., cut-in, highway exit, and car-following. More importantly, the proposed method in Part I was enhanced by a temporal-difference reinforcement learning (TD-RL) method to generate high-dimensional scenarios efficiently. For all three cases, our results show that the proposed method can effectively and efficiently generate the testing scenario library, which can accelerate the evaluation process by 255 to 3.75×10^5 times compared with the NDD evaluation method, but with the same accuracy.

Combining Part I and Part II papers, to the best of our knowledge, this is the first study that provides a systematic framework and implementation guidelines for both low and high dimensional scenarios, different performance metrics, and varying CAV models.

There are many interesting topics that can be further investigated. For the proposed TSLG method, since the dissimilarity between the SM and the test CAV is the major cause of evaluation inefficiency, SM as well as the generated library

can be updated adaptively using the data collected from the testing process. Another important topic is to investigate the scenario generation method for extremely high dimensions, for example, highway driving environment with hundreds of background vehicles. As both the spatial and temporal complexity will increase greatly, RL-enhanced method presented in this paper needs further improvement. These topics are left for future studies.

APPENDIX PROOF OF THEOREMS

Theorem 1: If $Q(s, u)$ is defined as Eq. (26) and x is defined as Eq. (25), the scenario criticality can be computed as

$$V(x) = C(x) \prod_{k=1}^m Q(s_k, u_k),$$

where

$$C(x) = \frac{P(s_1)}{\prod_{k=2}^m \left(\sum_{u_k \in \mathbb{U}} Q(s_k, u_k) \right)}.$$

Proof: After scenarios are represented by the decision tree, the exposure frequency of a testing scenario can be denoted as

$$P(x) = P(s_1)P(u_1|s_1) \dots P(u_m|s_m). \quad (29)$$

As shown in Eq. (4), the criticality of a scenario is defined as

$$V(x) = P(S|x)P(x) = P(S)P(x|S), \quad (30)$$

where $P(S)$ is a constant, which can be obtained by Monte Carlo simulation. Similar to the Eq. (29), $P(x|S)$ is denoted as

$$P(x|S) = P(s_1|S)P(u_1|s_1, S) \dots P(u_m|s_m, S). \quad (31)$$

By applying Bayesian equation and Law of total probability, we have

$$P(s_1|S) = \frac{P(S|s_1)P(s_1)}{\sum_{s_1 \in \mathcal{X}} P(S|s_1)P(s_1)}, \quad (32)$$

$$P(S|s_1) = \sum_{u_1 \in \mathbb{U}} P(S|u_1, s_1)P(u_1|s_1), \quad (33)$$

$$P(u_k|s_k, S) = \frac{P(S|u_k, s_k)P(u_k|s_k)}{\sum_{u_k \in \mathbb{U}} P(S|u_k, s_k)P(u_k|s_k)}, \quad (34)$$

where $k = 1, \dots, m$. Substituting Eq. (31), (32), (33), and (34) into Eq. (30), the theorem is concluded. \square

Theorem 2: After the training process of TD-RL, $Q(u, s)$ can converge to the values defined in Eq. (26), if the TD error is defined as

$$\delta_k = \left(\sum_{u_{k+1} \in \mathbb{U}} Q(s_{k+1}, u_{k+1}) \right) P(u_k|s_k) - Q(s_k, u_k).$$

Proof: The TD error is the difference between the estimated value of $Q(s_k, u_k)$ and its estimation from the next state. Therefore, if the first term on the right in Eq. (27) is the

estimation of $Q(s_k, u_k)$ based on the next state, the theorem can be proved [25]. Consider that

$$\begin{aligned} P(S|u_k, s_k) &= \sum_{s_{k+1} \in \mathcal{X}} P(s_{k+1}|u_k, s_k) P(S|s_{k+1}), \\ &= P(S|s_{k+1}), \\ &= \sum_{u_{k+1} \in \mathbb{U}} P(S|u_{k+1}, s_{k+1}) P(u_{k+1}|s_{k+1}), \\ &= \sum_{u_{k+1} \in \mathbb{U}} Q(s_{k+1}, u_{k+1}), \end{aligned}$$

where the second equivalence is derived considering the SM is deterministic. Plugging the equation into Eq. (26), we obtain

$$Q(s_k, u_k) = \left(\sum_{u_{k+1} \in \mathbb{U}} Q(s_{k+1}, u_{k+1}) \right) P(u_k|s_k),$$

which concludes the theorem. \square

ACKNOWLEDGMENT

The authors would like to thank Dr. D. Zhao of Carnegie Mellon University for providing data in the cut-in case study.

REFERENCES

- [1] L. Li, W.-L. Huang, Y. Liu, N.-N. Zheng, and F.-Y. Wang, "Intelligence testing for autonomous vehicles: A new approach," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 2, pp. 158–166, Jun. 2016.
- [2] L. Li *et al.*, "Artificial intelligence test: A case study of intelligent vehicles," *Artif. Intell. Rev.*, vol. 50, no. 3, pp. 441–465, Oct. 2018.
- [3] H. Hunger, "Test specifications for highly automated driving functions: Highway pilot," Inst. Transp., German Aerosp. Center, Cologne, Germany, Tech. Rep., 2017. [Online]. Available: <https://www.pegasusprojekt.de>
- [4] D. Zhao *et al.*, "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 595–607, Mar. 2017.
- [5] D. Jung, D. Jung, C. Jeong, Y. Kou, and H. Peng, "Worst case scenarios generation and its application on driving," SAE Tech. Paper 2007-01-3585, 2007.
- [6] S. Zhang, H. Peng, D. Zhao, and H. E. Tseng, "Accelerated evaluation of autonomous vehicles in the lane change scenario based on subset simulation technique," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 3935–3940.
- [7] L. Li *et al.*, "Parallel testing of vehicle intelligence via virtual-real interaction," *Sci. Robot.*, vol. 4, no. 28, Mar. 2019, Art. no. eaaw4106.
- [8] S. Feng, Y. Feng, C. Yu, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, part I: Methodology," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 3, 2020, doi: 10.1109/TITS.2020.2972211.
- [9] D. Zhao, X. Huang, H. Peng, H. Lam, and D. J. LeBlanc, "Accelerated evaluation of automated vehicles in car-following maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 733–744, Mar. 2018.
- [10] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" *Transp. Res. A, Policy Pract.*, vol. 94, pp. 182–193, Dec. 2016.
- [11] A. B. Owen, *Monte Carlo Theory, Methods and Examples*. 2013. [Online]. Available: <https://statweb.stanford.edu/~owen/mc/>
- [12] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*. New York, NY, USA: Springer, 2013.
- [13] S. M. Ross, *Introductory Statistics*. New York, NY, USA: Academic, 2017.
- [14] K. Vogel, "A comparison of headway and time to collision as safety indicators," *Accident Anal. Prevention*, vol. 35, no. 3, pp. 427–433, May 2003.
- [15] R. Chen, R. Sherony, and H. C. Gabler, "Comparison of time to collision and enhanced time to collision at brake application during normal driving," SAE Tech. Paper 2016-01-1448, 2016.
- [16] D. Bezzina and J. Sayer, "Safety pilot model deployment: Test conductor team report," U.S. Dept. Transp., Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS, 2014, vol. 812, p. 171.
- [17] X. Gong, Y. Guo, Y. Feng, J. Sun, and D. Zhao, "Evaluation of the energy efficiency in a mixed traffic with automated vehicles and human controlled vehicles," 2018, *arXiv:1806.00377*. [Online]. Available: <http://arxiv.org/abs/1806.00377>
- [18] T. A. Ranney, "Models of driving behavior: A review of their evolution," *Accident Anal. Prevention*, vol. 26, no. 6, pp. 733–750, Dec. 1994.
- [19] J. W. Ro, P. S. Roop, A. Malik, and P. Ranjitkar, "A formal approach for modeling and simulation of human car-following behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 639–648, Feb. 2018.
- [20] S. H. Hamdar and H. S. Mahmassani, "From existing accident-free car-following models to colliding vehicles: Exploration and assessment," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2088, no. 1, pp. 45–56, Jan. 2008.
- [21] J. R. Sayer *et al.*, "Integrated vehicle-based safety systems light-vehicle field operational test key findings report," U.S. Dept. Transp., Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS 811 482, 2011.
- [22] A. Kesting, M. Treiber, and D. Helbing, "General lane-changing model MOBIL for car-following models," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1999, no. 1, pp. 86–94, Jan. 2007.
- [23] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI, USA: Nob Hill Publishing, 2009.
- [24] J. Nilsson, J. Silvlin, M. Brannstrom, E. Coelingh, and J. Fredriksson, "If, when, and how to perform lane change maneuvers on highways," *IEEE Intell. Transp. Syst. Mag.*, vol. 8, no. 4, pp. 68–78, Oct. 2016.
- [25] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.
- [26] S. Feng, Y. Feng, H. Sun, Y. Zhang, and H. X. Liu, "Adaptive testing scenario library generation for connected and automated vehicles," 2020, *arXiv:2003.03712*. [Online]. Available: <http://arxiv.org/abs/2003.03712>
- [27] W. Wang, C. Liu, and D. Zhao, "How much data is enough? A statistical approach with case study on longitudinal driving behavior," *IEEE Trans. Intell. Veh.*, vol. 2, no. 2, pp. 85–98, Jun. 2017.



Shuo Feng received the bachelor's and Ph.D. degrees from the Department of Automation, Tsinghua University, China, in 2014 and 2019, respectively. He was a joint Ph.D. Student in civil and environmental engineering with the University of Michigan, Ann Arbor, MI, USA, from 2017 to 2019, where he is currently a Post-Doctoral Researcher. His current research interests include testing, evaluation, and optimization of connected and automated vehicles.



Yiheng Feng received the B.S. and M.E. degrees from the Department of Control Science and Engineering, Zhejiang University, Hangzhou, China, in 2005 and 2007, respectively, the master's degree from the Civil Engineering Department, University of Minnesota, Twin Cities, in 2011, and the Ph.D. degree in systems and industrial engineering from the University of Arizona in 2015. He is currently an Assistant Research Scientist with the University of Michigan Transportation Research Institute. His research interests include traffic signal systems control and security, and connected and automated vehicles testing and evaluation.



Haowei Sun received the bachelor's degree from the Department of Automation, Tsinghua University, China, in 2019. He is currently a Graduate Student with the Department of Civil and Environmental Engineering, University of Michigan. He visited the University of Michigan for a summer Research Internship in 2018. His research interests include intelligent transportation, optimization method, and deep reinforcement learning.



Shan Bao received the Ph.D. degree in mechanical and industrial engineering from the University of Iowa in 2009. She is currently an Associate Professor with the University of Michigan–Dearborn and an Associate Research Scientist with the University of Michigan Transportation Research Institute, where she has been conducting research on human factors issues related to transportation systems. Her current research interests focus on human factors issues related to connected and automated vehicle technologies.



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

and urban traffic control and management. His research fields also cover the advanced control theory and applications, advanced detection and measurement, and systems engineering.



Henry X. Liu (Member, IEEE) received the bachelor's degree in automotive engineering from Tsinghua University in 1993 and the Ph.D. degree in civil and environmental engineering from the University of Wisconsin at Madison in 2000. He is currently a Professor of civil and environmental engineering with the University of Michigan, Ann Arbor, and a Research Professor with the University of Michigan Transportation Research Institute. He also directs the USDOT Region 5 University Transportation Center: Center for Connected and Automated Transportation. His research interests focus on transportation network monitoring, modeling, and control, as well as mobility and safety applications with connected and automated vehicles. On these topics, he has published more than 100 refereed journal articles. He is the Managing Editor of *Journal of Intelligent Transportation Systems* and an Associate Editor of *Transportation Research Part C*.