Delft University of Technology

Safety Performance Boundary Identification of Highly Automated Vehicles
A Surrogate Model-Based Gradient Descent Searching Approach

Wang, Yiyun; Yu, Rongjie; Qiu, Shuhan; Sun, Jian; Farah, Haneen

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Safety Performance Boundary Identification of Highly Automated Vehicles

A Surrogate Model-Based Gradient Descent Searching Approach

Wang, Yiyun; Yu, Rongjie; Qiu, Shuhan; Sun, Jian; Farah, Haneen

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Safety Performance Boundary Identification of Highly Automated Vehicles: A Surrogate Model-Based Gradient Descent Searching Approach

Yiyun Wang, Rongjie Yu, Shuhan Qiu, Jian Sun, and Haneen Farah

Abstract—Highly automated vehicles (HAVs) have been introduced to the transportation system for the purpose of providing safer mobility. Considering the expected long co-existence period of HAVs and human-driven vehicles (HDVs), the safety operation of HAVs interacting with HDVs needs to be verified. To achieve this, HAVs' Operational Design Domain (ODD) needs to be identified under the scenario-based testing framework. In this study, a novel testing framework aiming at identifying the Safety performance boundary (SPB) is proposed, which assures the coverage of safety-critical scenarios and compatible with the black-box feature of HAV control algorithm. A surrogate model was utilized to approximate the safety performance of HAV, and a gradient descent searching algorithm was employed to accelerate the search for SPB. For empirical analyses, a three-vehicle following scenario was adopted and the Intelligent Driver Model (IDM) was tested as a case study. The results show that only 4% of the total scenarios are required to establish a reliable surrogate model. And the gradient descent algorithm was able to establish the SPB by identifying 97.42% of collision scenarios and only false alarming 0.29% of non-collision scenarios. Furthermore, the concept of safety tolerance was proposed to measure the possibilities of boundary scenarios dropping in safety performance. The applications of helping to construct ODD and compare different control algorithms were discussed. It shows that the IDM performs better than the Wiedemann 99 (W99) model with larger ODD.

Index Terms—Highly automated vehicle, safety performance boundary, gradient descent, surrogate model, operational design domain.

I. INTRODUCTION

HIGHLY automated vehicle (HAV) technology (with SAE automation level of 4 or 5) has been introduced to

the transportation system with the promise of reducing crash risk and improving traffic safety [1]. Countries have listed HAV as national level strategic layouts. For instance, the Chinese government expects to achieve mass production of HAV by 2025 [2], while European Union (EU) targets at fully automated mobility by 2030 [3]. Moreover, United States has released their fourth edition of *Automated Vehicles* to ensure American leadership in automated vehicle technologies [4]. However, it is pessimistically evaluated that not until 2040 would the level-5 automated driving be feasible [5], which indicates that the transition from the current traffic system to the traffic flow consisting of fully automated vehicles would take a long period. Human-driven vehicles (HDVs) and HAVs will coexist in the transportation system.

The safety operation issue of HAV in mixed traffic has been attracting a lot of attention. Given its limited operation on public roads, HAVs have been frequently involved in high-risk events and even crashes with HDVs. To be specific, about 233 HAV-involved traffic crashes occurred in California by 2020, and 72% of them occurred under the automatic driving mode [6]. Besides, driving behaviors of HAVs were claimed to be difficult to understand by human drivers, which has led to large proportions of HAVs being rear-ended by HDVs [7]. Therefore, the safety operation of HAVs in mixed traffic needs to be validated before HAVs can be widely deployed into the market [8].

To evaluate the operation safety of HAVs and its Operational Design Domain (ODD), testing methods are the key. ODD is defined as the conditions under which a given automated vehicle can operate safely [9], and is one of the fundamental properties of HAV that is required to be clarified before marketing. The boundaries of ODD can be determined by operational constraints such as speed and acceleration. As for the testing method, traditional distance-based public road-testing approach is impractical due to the estimated long period and resources needed. For instance, a previous study concluded that it needs 100 HAVs to drive 24 hours per day continuously for about 500 years [10]. Besides, public road-testing holds threats to the safety of the public due to frequent incidents [11]. To solve this, scenario-based simulation testing approach has been widely adopted [12]. Following this approach, key parameters, such as, velocity, acceleration, and distance gap, of the HAV in different driving scenarios (for instance, car-following, lane changing, etc.) are first selected

and based on this the parameter space is established. Then scenarios with specific values are extracted from the parameter space, and testing and evaluation are conducted. However, with the increase number of key parameters (or critical parameters that could be utilized to describe a specific driving scenario), the permutations of their values explode exponentially, resulting in the parameter combinatorial explosion problem [13]. Take the cut-in scenario for example, which can be defined by 5 key parameters (HAV velocity, HDV velocity, acceleration, cut-in distance, road width). When each parameter has 10 possible values, the total scenarios under test can be over 9.76 million ($5^{10} = 9765625$). If testing one scenario needs 1 minute, then the exhaustive testing of all scenarios requires 1.85 years ($5^{10} \div$ (60minutes $\times$ 24hours $\times$ 365days $\times$ 10 accelerated ratio)), which is not tractable. To speed up the testing process, accelerated testing methods that aim at identifying the most safety-critical scenarios are needed.

Most existing accelerated testing methods regard the HAV control algorithm as a "white box", whose acceleration control mechanism is known in advance, and then apply modified sampling methods to accelerate the search of safety-critical scenarios. However, simply identifying safety-critical scenarios could not guarantee the coverage of scenarios [14], which is one of the basic requirements for HAV testing [15]. Besides, HAV control algorithms are confidential and are not made public. HAV manufacturers own the propriety knowledge of their algorithms, and modern HAV control algorithms unavoidably employ Artificial Intelligent (AI) techniques that have a black-box feature. Therefore, a testing method that is able to assure the coverage of safety-critical scenarios, and can consider the HAV control algorithm as a black box is required [16].

To fill the gap, this study aims to develop a testing method that accelerates the identification of Safety performance boundary (SPB) for black-box HAV control algorithm. SPB refers to the transition zone that is composed of critical scenarios separating the safe area from the hazardous area within the parameter space [17]. Considering this objective, the following research questions were defined:

- How to quickly identify SPB given a black-box HAV control algorithm?
- Can the SPB be utilized to delineate regions with safe and unsafe scenarios?
- How to determine a preventive area for the boundary to improve the safety tolerance, and how to quantify its safety tolerance?

The main contributions of this study are as follows:

- Utilizing an adaptive sampling method to establish a Multi-layer Perception (MLP)-based surrogate model with only 4% of total scenarios, which could quickly estimate the safety performance of a black-box HAV control algorithm.
- Proposing a new concept of safety tolerance performance boundary (STPB). The safety tolerance of SPB is quantified by the safety performance variations of scenarios.
- Proposing a method that can identify the pros and cons of different algorithms. Intelligent Driver Model (IDM)

and Wiedemann 99 (W99) model control algorithms are compared based on the identified SPBs.

Compared with previous studies, the method developed in this study could accelerate the efficiency of SPB establishment, and meanwhile quantify the safety tolerance of SPB. The results could help HAV manufactures to better test and determine the ODD of their developed control algorithms.

II. LITERATURE REVIEW

In the current literature, Accelerated Evaluation (AE) has been proposed to improve the HAV testing efficiency [18]. Considering the heavy tail distribution characteristic of scenario criticality, Zhang *et al.* [18] applied the Importance Sampling (IS) method to enhance the exposure of rare events, which are regarded as safety-critical scenarios. Although AE has been proved to be capable of identifying the rare occurring risky scenarios, it requires prior knowledge of the HAV control algorithm under testing to shape the IS distribution [19]. However, emerging HAV control algorithms are mostly Artificial Intelligent (AI) driven, which have a black-box nature thus can hardly be handled by the AE framework [16]. Studies focusing on testing methods that are applicable for black-box HAV algorithms are needed.

Other types of HAV testing method, such as the unsafe-scenario-oriented testing method [20] and search-based testing method [21], identify the most safety-critical scenarios within the pre-defined parameter space [22]. Starting from the random selected scenarios, much more critical scenarios would be identified via step-by-step searching. The searching direction is toward the safety reduction and could be optimized by the utilization of genetic algorithm [23], learnable evolutionary algorithm [24] and reinforcement learning method [25]. Since this type of method only requires the testing outcome, it is compatible with black-box HAV control algorithms. However, it holds the issue of converging to local optimum. Thus, the coverage of the identified safety-critical scenarios within the whole parameter space cannot be guaranteed [21].

In order to ensure the scenario coverage, researchers attempted to identify a safety performance boundary (SPB) [26]. The SPB includes safety-critical scenarios differentiates the hazardous area from the safe area [17]. Studies have been conducted to identify SPB, for example, Batsch *et al.* [27] utilized supervised learning method to recognize the boundary based on samples extracted off-line by Latin Hypercube Sampling. Latin Hypercube Sampling method can extract samples uniformly from the parameter space (aiming at exploration), however, not all areas within the parameter space deserve to be equivalently searched. The boundary area, where slight changes of parameters might cause tremendous change in safety performance, requires to be more exploited (aiming at exploitation). To enhance the efficiency and accuracy of SPB establishment, recent studies [13], [16] employ adaptive sampling method. This method considers both exploration and exploitation to search for the boundary scenarios.

In this study, we employ an adaptive sampling method to iteratively establish the surrogate model for HAV safety
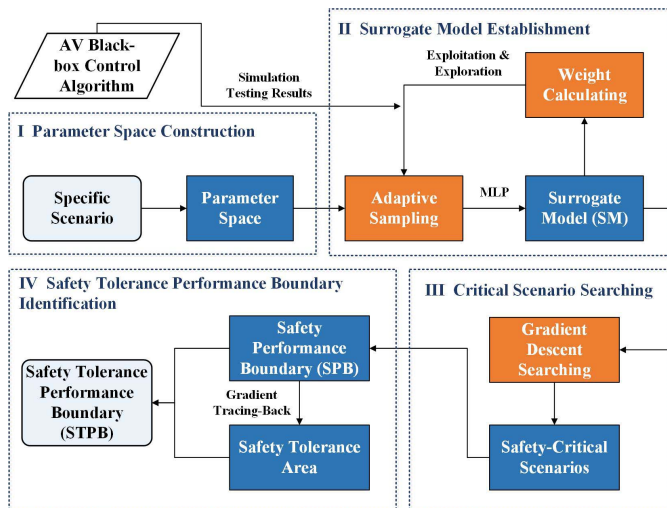
Fig. 1.   Overall framework of the proposed testing method.

Fig. 2.   Structure of MLP.

performance. Based on the surrogate model, gradient descent algorithms are utilized to search for boundary scenarios. And finally, a concept of safety tolerance zone (STZ) is proposed to guarantee the operational safety of HAV.

The remainder of this paper is organized as follows: in the *Methodology* section, the overall framework of the proposed HAV testing method, the surrogate model and gradient descent algorithms, are presented. In the *Simulation Experiment* section, a three-vehicle following scenario was considered as an example to illustrate how the proposed framework can be applied, then the simulation results are exhibited in the *Simulation Results* section. In the *Discussion* section, applications of the proposed method on ODD and the illustration of control algorithm comparison are presented. Finally, the conclusions and limitations as well as future work outlook are provided in the *Conclusion* section.

III. METHODOLOGY

A. Overall Framework

The overall framework of the proposed testing method is shown in Figure 1.

There are four parts of the framework. In Part I, key parameters of the testing scenario are defined and the testing parameter space would be established. In Part II, a surrogate model, which describes the safety performance of the under-tested HAV black-box algorithm, would be established. This process is iterative with the utilization of adaptive sampling. In Part III, critical scenarios would be identified by optimized gradient descent searching algorithms. Finally, in Part IV, the safety performance boundary (SPB) and the safety tolerance performance boundary (STPB) would be built.

B. Surrogate Model

Surrogate models have typically been adopted to test complex systems [28]. To establish the surrogate model, only part of the samples need be tested on the original system, and the testing results would be utilized to approximate the safety performance of the original system.
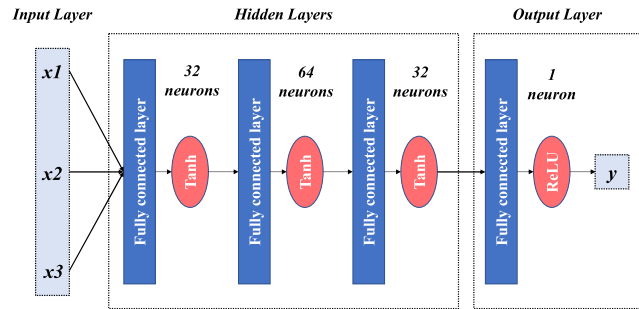
Most of HAV control algorithms are AI driven. The decision outputs of HAV control algorithms are probabilistic and exhibit nonlinear characteristics [29]. Regarding this issue, various surrogate models have been employed [30], [31]. For instance, Kriging models (also known as Gaussian Process Regression model) [13], [32] and Artificial Neural Networks (ANN) [23], [33]. ANN has the advantages of easy construction and utilization. Besides, the established ANN could provide the gradient information of specific scenario conveniently, which has great importance for this study. Thus, in this study, a Multi-layer Perception (MLP) model, which is a simple ANN consisting of multiple neurons, is employed., An overview of various surrogate models can be found in Sun *et al.*'s study [28].

The structure of MLP is presented in Figure 2. For a specific scenario x under testing, the input layer is its key parameters $(x_1, x_2, x_3, \ldots, x_n)$. The output layer is the predicted operational safety performance $\hat{Y} = \hat{y}_{(x_1, x_2, x_3, \ldots, x_n)}$ of scenario x, which could be represented by surrogate safety measures such as Time to collision (TTC) [34].

The mathematical expressions and the loss function of MLP are described in Equation (1) to Equation (3):

$$y_{(x_1, x_2, x_3, \ldots, x_n)} = g\left(h_k\right) \tag{1}$$

$$h_k = \sum_{j=0}^{M} \omega_{jk} x_{jk} \tag{2}$$

$$\text{MSE} = \frac{1}{2} \sum_{k=1}^{N} \left(\hat{y}_{(x_1, x_2, x_3, \ldots, x_n)} - y_{(x_1, x_2, x_3, \ldots, x_n)}\right)^2 \tag{3}$$

where $g()$ is the activation function (Tanh is utilized for the hidden layers and ReLU for the output layer), h_k is the weighted sum of inputs from the last layer of neuron k, ω_{jk} is the jth weight of the inputs from the last layer, M is the number of neurons, $y_{(x_1, x_2, x_3, \ldots, x_n)}$ is the true label of scenario $(x_1, x_2, x_3, \ldots, x_n)$, MSE is the Mean Square Error between the trained model ($\hat{y}_{(x_1, x_2, x_3, \ldots, x_n)}$) and the true model ($y_{(x_1, x_2, x_3, \ldots, x_n)}$). The Back propagation algorithm is utilized to update the weights [35]. True labels for the testing parameters were obtained from simulation experiments, and were calculated by the surrogate safety measure Time to collision (TTC). This will be explained in details in section *IV. Simulation Experiment*.

C. Adaptive Sampling

In adaptive sampling method, the exploration aims at exploring the entire parameter space with the least number of

samples. Exploration method on its own might be inefficient if different regions of the parameter space offer different extents of interest. Exploitation focuses on the regions with more interesting features, while it might cluster in specific regions and only focus on them will make other regions remain unsampled [32]. Therefore, combining the exploration and the exploitation methods would complement each other.

In this study, considering both the exploration and the exploitation, a generative adaptive sampling method was adopted. Exploitation is used to identify the boundaries with nonlinear safety change, and exploration is applied to enhance the coverage of the sampling.

The parameter space X^N is split into N sub-parameter spaces. For the ith sub-parameter space X^i, it contains n_i samples. The weight ω_i of sub-parameter space X^i is calculated by Equation (4) to Equation (6). Exploration is measured by overall fitting error ε_i of the surrogate model compared with the true model, exploitation is measured by range R of the true model, which represents the nonlinear degree of X^i. If we consider that the exploration has the same importance as the exploitation, then α would be equal to 0.5.

$$\omega_i = \alpha \varepsilon_i + (1 - \alpha) R_i \tag{4}$$

$$\varepsilon_i = \frac{\sum \sqrt{(\hat{y}_{[x_1, x_2, \ldots x_n]_i} - y_{[x_1, x_2, \ldots x_n]_i})^2}}{n_i} \tag{5}$$

$$R_i = max\left\{ y_{[x_1, x_2, \ldots x_n]_i} \right\} - min\left\{ y_{[x_1, x_2, \ldots x_n]_i} \right\} \tag{6}$$

D. Gradient Descent Searching Algorithm

Searching for the critical scenarios can be seen as an optimization problem, whose objective is to find the most safety-critical scenarios with minimal searching steps.

Let J represent the safety performance of scenario x, then the objective function is the $min J(x)$. The gradient of objective function regarding the present scenario point $x^{(j)}$ is $g^{(j)} = \nabla J(x^{(j)})$. Within the maximum searching step S_{max}, the scenario parameters are updated by the following formulation:

$$x^{(j+1)} = x^{(j)} - \alpha h(g^{(j)}) \tag{7}$$

where $h\left(g^{(j)}\right)$ is the function of $g^{(j)}$, and has three different updating methods applied to this problem [36], which are basic Steepest Descent (SD) algorithm, Gradient Descent with Momentum (GDM) algorithm [37], and Adaptive Moment Estimation (Adam) algorithm [38]. α is the fixed step size for the updates and its value is $\alpha = (0.0002, 0.02, 0.01)$ in this study.

(1) As for SD algorithm:

$$h\left(g^{(j)}\right) = g^{(j)} \tag{8}$$

(2) As for GDM algorithm, $h\left(g^{(j)}\right) = \sum_{i=0}^{j} \eta g^{(i)}$. η (equals to 0.9 in this study) is a momentum value added for each searching step, to mitigate the possible vibration during the search. Thus, the searching steps could be reduced;

(3) As for Adam algorithm, it can adaptively adjust and update the searching steps according to the importance

of each parameter, and:

$$h\left(g^{(j)}\right) = \frac{\tilde{m}^{(j+1)}}{\sqrt{\tilde{v}^{(j+1)}} + \varepsilon} \tag{9}$$

$$\tilde{m}^{(j+1)} = \frac{m^{(j+1)}}{1 - b_1^{j+1}} \tag{10}$$

$$\tilde{v}^{(j+1)} = \frac{v^{(j+1)}}{1 - b_2^{j+1}} \tag{11}$$

$$m^{(j+1)} = b_1 m^{(j)} + (1 - b_1) g^{(j)} \tag{12}$$

$$v^{(j+1)} = b_2 v^{(j)} + (1 - b_2) g^{(j)^2} \tag{13}$$

where $m^{(j)}$ is the first-order moment estimation for gradient, $v^{(j)}$ is the second-order moment estimation for gradient. $\tilde{m}^{(j)}$ and $\tilde{v}^{(j)}$ are the bias correction of $m^{(j)}$ and $v^{(j)}$, respectively. b_1 and b_2 are the hyperparameters and equal to 0.9 and 0.999, respectively.

E. Polynomial Surface Fitting Based on Least Square (LS) Method

A bivariate cubic polynomial fitting is employed to fit the safety performance boundary [39]. The fitting expression is as in Equation (14) and LS method is as in Equation (15).

$$\begin{aligned} fv\,(dec, dis1) = {} & p00 + p10dec + p01dis1 + p20dec^2 \\ & + p11dec * dis1 + p02dis1^2 + p30dec^3 \\ & + p21dec^2 dis1 + p12dec * dis1^2 \\ & + p03dis1^3 \end{aligned} \tag{14}$$

$$min\,J\,(p) = \sum_i [fv_i - fv\,(dec, dis1)]^2 \tag{15}$$

In this study, the fitting was conducted by the Curve Fitting Tool of MATLAB [40].

IV. SIMULATION EXPERIMENT

A. Testing Scenario and Key Parameters Set-Ups

A three-vehicle following scenario was designed and experimented to illustrate the proposed SPB identification framework. The simulation experiment involves a highly automated vehicle (HAV) following a human-driven vehicle (HDV1), and the HAV itself is followed by another human-driven vehicle (HDV2) on a good-condition roadway. The leading HDV1 performs a sudden deceleration maneuver in reaction to downstream congestion or its leading vehicle's deceleration. The HAV responds and brakes to avoid collision. The following HDV2 also responds accordingly. This scenario is shown in Figure 3.

Three key parameters are selected to describe the above car-following scenario (Figure 3): deceleration of HDV1 (dec), distance gap between HDV1 and HAV (dis1), and velocity of three vehicles (fv). The three vehicles are assumed to have the same initial velocity. The value range of each parameter is shown in Table I. The value range of the velocity and the distance gap were selected based on the parameters' true distributions of Shanghai expressway system. Different deceleration values of HDV1 were set according to the literature [41].
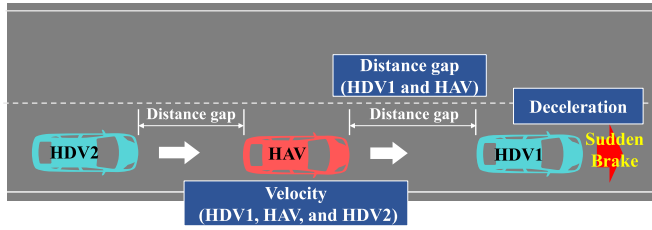
Fig. 3. Testing scenarios and key parameters. (Note: Key parameters of scenario are in blue boxes).

TABLE I

VALUE SCOPE OF KEY PARAMETERS

Key parameters	Description	Value range
dis1	Distance gap between HDV1 and HAV	(25, 65) m, 1m interval
dec	Deceleration of HDV1	(0.35, 0.75) g, 0.01g interval
fv	Velocity of HDVs and HAV	(15, 35) m/s, 0.5m/s interval

B. Vehicle Control Algorithms and Simulation Set-Ups

The classical Intelligent Driver Model (IDM) [42], which was often undertaken as the tested control algorithm [43] was adopted to control the vehicles' following behaviors. It is worth mentioning that the proposed testing framework can be applied on any given HAV control algorithm.

HDV1 does the deceleration under an initial velocity. HAV and HDV2 follow the leading vehicle under the control of IDM algorithm. The IDM model is described in Equations (16) and (17). The initial distance gap between HAV and HDV2 (*dis2*) is decided by the desired following distance ($\tilde{S}$), which is calculated by Equation (17).

IDM is one of the most widely employed models using desired measures, which contains the desired speed and the desired following distance. The acceleration calculation of IDM is defined in Equation (16):

$$a\left(t\right) = a_{max}\left[1 - \left(\frac{v\left(t\right)}{\tilde{v}\left(t\right)}\right)^{\beta} - \left(\frac{\tilde{S}\left(t\right)}{S\left(t\right)}\right)^{2}\right] \quad (16)$$

where a_{max} is the maximum acceleration / deceleration of the following vehicle, v is the speed, $\tilde{v}$ is the desired speed, S is the distance gap between two vehicles (measured from the end of the leading vehicle to the front of the following vehicle), β is the acceleration parameter, $\tilde{S}$ is the desired following distance and its calculation is as in Equation (17):

$$\tilde{S}\left(t\right) = S_{jam} + \rho v\left(t\right) + \max(0, v\left(t\right)\tilde{h}\left(t\right) - \frac{v(t)\Delta v(t)}{2\sqrt{a_{max}a_{comf\_dec}}}) \quad (17)$$

The desired following distance is dependent on speed (v), speed difference between following vehicle and leading vehicle (Δv), maximum acceleration (a_{max}), comfortable deceleration (a_{comf_dec}), safe distance gap in the standstill (S_{jam}), and desired headway ($\tilde{h}$).

TABLE II

SIMULATION PARAMETERS DESCRIPTION

Simulation parameter	Description	Value
a_{max}	maximum acceleration / deceleration (m/s²)	5.0
$\tilde{v}$	desired speed (m/s)	the same with the initial velocity *fv*
S_{jam}	safe distance gap in the standstill (m)	1.0
a_{comf_dec}	comfortable deceleration (m/s²)	2.4
$\tilde{h}$	desired headway (s)	2
β	acceleration parameter	4
HAV control algorithm		
ρ	Response time (s)	0.5
HDV2 control algorithm		
ρ	Response time (s)	1.5

The parameter values of the IDM model are illustrated in Table II. These values were adopted from the literature [44].

The simulation interval is 0.01s, i.e., 100 HZ. The simulation terminates when one of the following conditions are met: a) collision happens; b) three vehicles stopped safely. When the simulation ends, the risk evaluation index is recorded corresponding to the current scenario. In this study, minimum time to collision index TTC_{min} during the specific simulation is employed, the lower the TTC_{min} the higher the probability of a collision will be [45]. The calculation of TTC_{min} is as in Equation (18). During Ts of simulation, the longitudinal distance of the following vehicle x_f and the leading vehicle x_l at time ts are recorded, the longitudinal velocity v_f and v_l accordingly are also recorded. T is the maximum time length of simulation time, which is the time when all three vehicles stop. t is the current time of simulation.

$$TTC_{min} = min\left\{min_t^T\left(\frac{x_f - x_l}{v_f - v_l}\right)_{HDV1\ and\ HAV},\right.$$
$$\left.\times min_t^T\left(\frac{x_f - x_l}{v_f - v_l}\right)_{HAV\ and\ HDV2}\right\} \quad (18)$$

C. Adaptive Sampling and Surrogate Model Establishment

Adaptive sampling approach illustrated in III. C is utilized to build the MLP-based surrogate model. There are three parts: parameter space construction, adaptive sampling, and surrogate model establishment. The latter two parts are conducted iteratively and the surrogate model built by the last sampling is reserved for further procedures.

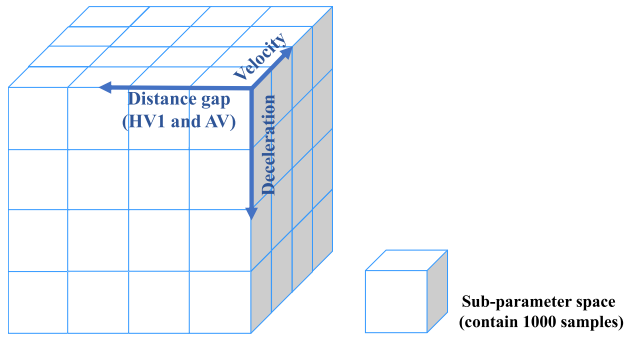*1) Parameter Space Construction:* Parameter space is constructed by the following steps:

Fig. 4.   Parameter space splitting diagram.

(a) Based on the defined driving scenario, the selected key parameters and their value ranges, the testing scenario parameter space is built. In this study, the parameter space contains 64000 (40∗40∗40) scenarios for testing, resulting from the split intervals described in Table I.

(b) The parameter space is split into sub-parameter spaces, which contain the same number of samples. In this study, 64 sub-parameter spaces are obtained by dividing each parameter into 4 slices; thus, each sub-parameter space contains 1000 samples. Figure 4 shows a rough diagram of the splitting. The splitting method employed was inspired from Delaunay-Hybrid Adaptive Sequential Design (DHASD), which was adopted by Ajdari and Mahlooji [32]. When determining the number of sub-parameter spaces, the trade-off between accuracy and efficiency needs to be considered. While it is more based on practice not on theory, and not the focus of this study, additional research could be conducted in future work.

2) The First Sampling and Surrogate Model Establishment: For the first sampling, the weights of sub-parameter spaces are initialized and the original surrogate model is established. The steps are as follows:

(a) Note that ω_i^{1st} is the initial sampling weight for sub-parameter space X^i, and $\sum \omega_i^{1st} = 1$ ($i = 1, \ldots, 64$). Initially, each sub-parameter space has the same weight, i.e., $\omega_i^{1st} = \frac{1}{64}$.

(b) $a\%$ of total samples are sampled in the first iteration (in this study a equals to 1), therefore, 640 (1%∗64000) samples (represented by $X^{1st} = \{(x_1, x_2, x_3)_i\}$) are extracted with the sampling weight ω_i^{1st} from corresponding sub-parameter space X^i.

(c) A total of 640 scenario samples X^{1st} are tested through the simulation platform, the safety performance of X^{1st} is measured by TTC_{min}, and is recorded as $Y^{1st} = \{y_{(x_1,x_2,x_3)_i}\}$.

(d) Train MLP model based on (X^{1st}, Y^{1st}), and the fitted safety performance of MLP model is obtained as ($X^{1st}, \hat{Y}^{1st}$).

(e) Calculate ε_i and R_i for each sub-parameter space by Equations (5) and (6), then update the weights for each sub-parameter space by Equation (4) and mark it as ω_i^{2nd}. Here the exploration is considered to have the same importance as the exploitation, thus, $\alpha = 0.5$.

3) The Iterative Sampling and Surrogate Model Establishment: The weights of sub-parameter spaces were updated iteratively, until the final surrogate model is established.

(a) For each of following iteration, another $a\%$ of total samples are required (again a equals to 1). Therefore, another 640 samples (represented by $X^{jth} = \{(x_1, x_2, x_3)_i\}$) are extracted with the sampling weight ω_i^{jth} for corresponding sub-parameter space X^i.

(b) Repeat the procedure *1)-* (c) to *1)-* (e): conduct the simulation with X^{jth} and obtain Y^{jth}, establish the MLP model with ($X^{1st}, \ldots, X^{jth}, Y^{1st}, \ldots, Y^{jth}$), and update the sampling weight as $\omega_i^{(j+1)th}$.

(c) Save the MLP model at the last iteration (4th iteration is employed in this study), which would be utilized as the surrogate model for the following parts.

D. Gradient Descent Search and Safety Performance Boundary (SPB) Identification

Gradient descent algorithms are employed to accelerate the process of searching the most safety-critical scenarios within the parameter space, and SPB is identified during this process. It follows the procedures as below:

(a) To ensure the covering rate of the search, 1% of total samples (640 in this study) are randomly selected as the initial points $X^{init} = \{x_1, x_2, \ldots x_m, m = 640\}$.

(b) From each initial point, based on the gradient descent algorithm, search along the direction of the gradient which is provided by MLP model, and update the scenario parameters. Three kinds of gradient descent algorithms were tested, which are SGD, Momentum, and Adam. The parameter settings of the three algorithms have been illustrated in the *Methodology* section. The search stops when the gradient of safety performance is below the error precision of ε, or the maximum searching step S_{max} has been reached. In this study, $\varepsilon = 0.001$ and $S_{max} = 1000$.

(c) During the process, the searching trajectory of each initial point is recorded in the step-unit, for example, for the mth initial point, it triggers the termination condition of maximum searching step S_{max}, meaning that it searches for S_{max} steps. The scenario points are recorded as $X^m = \{x_1, x_2, \ldots x_{S_{max}}\}$, the safety performance of scenarios are recorded as $Y^m = \{y_1, y_2, \ldots y_{S_{max}}\}$, and the gradients provided by MLP surrogate model for scenario points are marked as $G^m = \{g_{x_1}, g_{x_2}, \ldots, g_{x_{S_{max}}}\}$.

(d) The SPB scenarios are identified based on the trajectory. In this study, SPB is defined as the boundary that classifies the collision and non-collision scenarios. Therefore, the SPB scenarios occur on the trajectory where the safety performance (i.e., TTC_{min}) of adjacent points transfer from over 0 to 0.

(e) Based on the SPB scenarios, the bivariate cubic polynomial is employed to fit SPB interface.

V. SIMULATION RESULTS

A. Surrogate Model Development

For each iteration of surrogate model development, the model was trained for 500 epochs, with 32 batch size and

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

WANG *et al.*: SAFETY PERFORMANCE BOUNDARY IDENTIFICATION OF HIGHLY AUTOMATED VEHICLES 7

TABLE III
MODELING RESULTS OF ADAPTIVE SAMPLING

Iteration	Sample size	MSE
1	640	0.2269
2	1280	0.1062
3	1920	0.0535
4	2560	0.0097
5	3200	0.0067
6	3840	0.0078
7	4480	0.0086
8	5120	0.0046

Fig. 5. MSE of each iteration.

Fig. 6. (a) Diagrams of the true model; (b) diagrams of the established surrogate model.

TABLE IV
SEARCHING PERFORMANCE OF DIFFERENT
GRADIENT DESCENT ALGORITHMS

GD algorithm	Searching time (s)	Average searching steps
SGD	402.136	740.706
Momentum	388.294	697.961
Adam	247.904	552.600
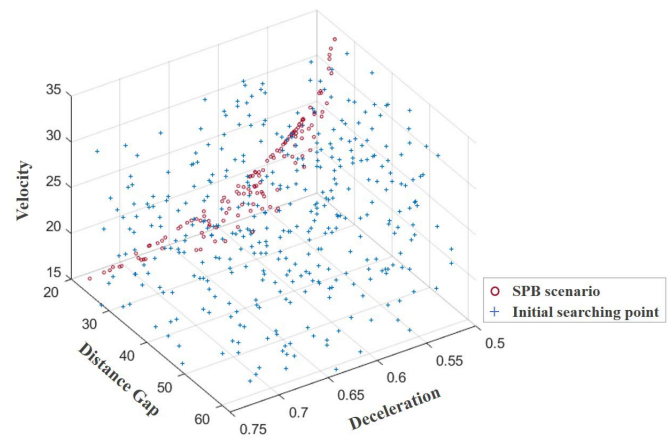
Fig. 7. Initial searching points and SPB scenarios.
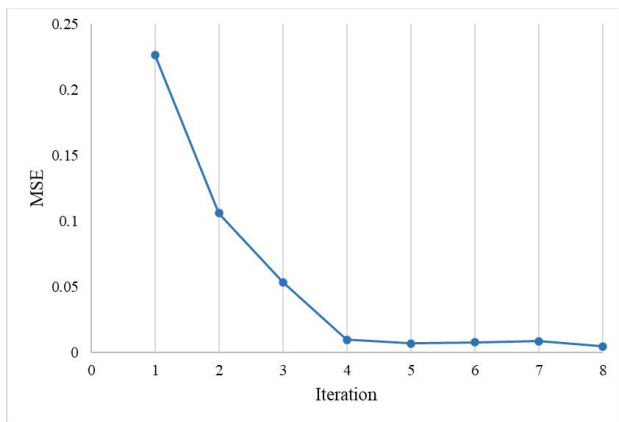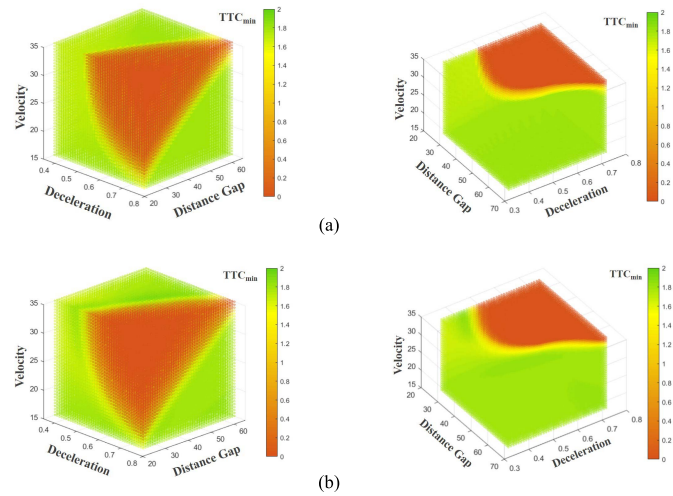
0.0001 learning rate. There is no testing dataset, since as many as possible samples are desired to build the surrogate model.

4% of total samples which equals to 2560 in this study were utilized to establish the final surrogate model. Table III and Figure 5 exhibit the modeling results of the surrogate model with eight iterations, and the fit errors (calculated by MSE) compared with the true model.

It is shown in Figure 5 that MSE drops rapidly from 1 to 4 iteration, and shows no obvious change after 4th iteration. Therefore, 4% of total samples are considered enough to build the surrogate model. Figure 6 visualized the true model and the established MLP model from different perspectives.

B. Gradient Descent Search

Table IV exhibits the searching performance of three gradient descent (GD) algorithms. Searching time represents the termination time of the gradient descent searching. Average searching steps is the average steps of each searching trajectory when it terminates. The results indicate that Adam GD algorithm performed best, which had the least searching time and least average searching steps. It reduced the searching time by about 40% compared with the basic algorithm.

According to the mechanism of the gradient descent algorithm, the searching direction is the fastest way to reduce TTC_{min}. The SPB scenarios were identified on the trajectory where the TTC_{min} of adjacent points transfer from over

0 to 0, meaning that from non-collision to collision. The initial searching points and the identified SPB scenarios are shown in Figure 7.

C. Safety Performance Boundary (SPB)

Based on the SPB scenarios, the cubic polynomial fitting is utilized to fit the boundary. The fitting results can be seen in Table V.

To evaluate the effects of the constructed performance boundary, HAV's real performance (simulation results) under all the scenarios of parameter space are classified by the

TABLE V

THE CUBIC POLYNOMIAL FITTING RESULTS OF SAFETY PERFORMANCE BOUNDARY

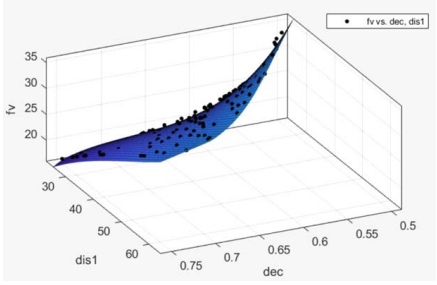|	Results or descriptions
Function	Classify the hazardous and safe area
Fit expression	$fv = 591.4 - 2929dec + 7.868dis1 + 4759dec^2 - 19.45dec*dis1 - 0.0066dis1^2 - 2550dec^3 + 13.87dec^2dis1 - 0.0083dec*dis1^2 + 0.0001dis1^3$ (Four significant digits are retained in MATLAB)
Adjusted R-square	0.9942
RMSE	0.3572
Diagram	

TABLE VI

SAFETY PERFORMANCE BOUNDARY CLASSIFICATION RESULTS

Boundary classification results / Simulation results	Collision	No collision	Sum
Collision	9448	250	9698
No collision	157	54145	54302
Sum	9605	54395	64000

constructed SPB. For the true label, it is processed from true TTC_{min}. When TTC_{min} is less than 0.01s, it is marked as a collision and labelled as 1. When TTC_{min} is larger than 0.01s, it is marked as a non-collision and labelled as 0. It is worth mentioning that the threshold setting of TTC_{min} can be considered not only from the view of safety, but also comfort. However, in this study our focus is only on safety.

Table VI presents the confusion matrix of classification, the sensitivity and the false alarm rate are 97.42% and 0.29% respectively, indicating that 97.42% of collision scenarios were successfully identified, and only 0.29% of non-collision scenarios were falsely classified as collisions.

D. Safety Tolerance Performance Boundary (STPB)

The size of gradient value is utilized to quantify the tolerance of the scenario. The scenario with a higher gradient indicates that the same alteration of parameters disturbances would lead to a larger change of safety performance, thus, its tolerance is low. Figure 8 exhibits the gradient distributions of three types of scenarios (no collision, collision and SPB scenarios) within the parameter space.

It can be seen that scenarios on the SPB and near to the SPB (some of collision scenarios are distributed near to the
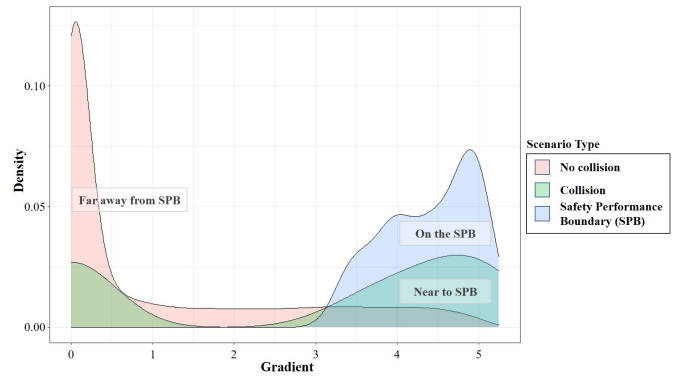

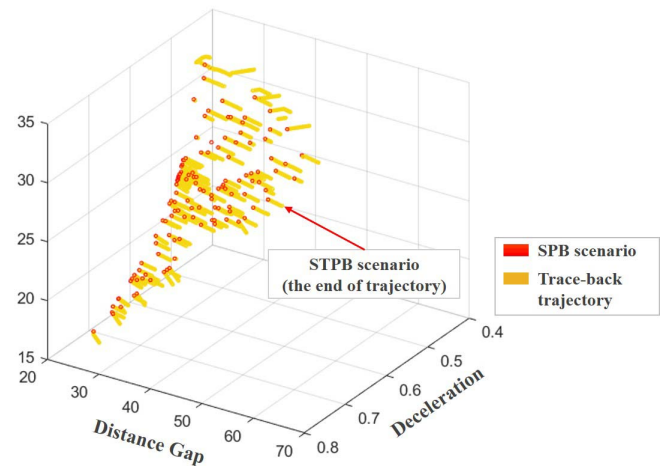Fig. 8. The gradient distributions of three types of scenarios.

Fig. 9. Trace-back for STPB scenarios identification.

boundary) have evidently higher gradients than those scenarios that are far away from the SPB. Although the distribution of no collision scenarios tends to have small gradients, there are still a lot of them holding the same high gradients as those of SPB. Therefore, additional attention needs to be paid for them and a safety tolerance performance boundary (STPB) need to be built.

To establish STPB, the gradient trace-back approach was applied. Based on the SPB scenarios in Figure 7, tracing one step back along the opposite direction of the gradient with the size of gradient value. The terminal points constitute the STPB which is presented in Figure 9.

The scenario points colored with yellow form the safety performance zone (STZ), whose upper interface is the SPB and its lower interface is STPB. The scenarios located at SPB and STPB are presented in Figure 10. Similar to the approach employed in *Section 3.5*, the scenario points located at the STPB were utilized to fit STPB by the bivariate cubic polynomial. The fitting results can be seen in Table VII.

E. Simulation Results of Wiedemann-99 (W99) Model

To illustrate the application of proposed method on comparing different HAV control algorithms, Wiedemann-99 (W99) model [46] was also tested. W99 is a psychophysical car-following model, using four different 'perceptual thresholds'
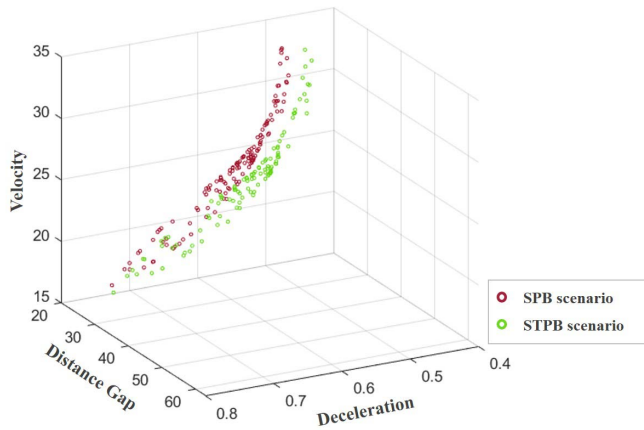
This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

WANG *et al.*: SAFETY PERFORMANCE BOUNDARY IDENTIFICATION OF HIGHLY AUTOMATED VEHICLES
9

Fig. 10.   Scenarios on the SPB and STPB.

TABLE VII
THE CUBIC POLYNOMIAL FITTING RESULTS OF SAFETY
TOLERANCE PERFORMANCE BOUNDARY

	Results or descriptions
Function	Classify the safety tolerance zone and the safe area
Fit expression	$fv = 367.1 - 1889dec + 6.157dis1 + 3190dec^2 - 15.61dec*dis1 - 0.0012dis1^2 - 1768dec^3 + 11.39dec^2dis1 - 0.0086dec*dis1^2 + 0.0001dis1^3$ (Four significant digits are retained in MATLAB)
Adjusted R-square	0.9955
RMSE	0.3146
Diagram	

for four driving regimes including free-flowing, approaching, following and emergency. The driver reacts to different regimes when the corresponding threshold values are reached. Besides, W99 model is also applied in many simulation platforms like VISSIM to model HAV's car-following behavior.

The same testing scenario and key parameters were set for W99. The calculation of acceleration of W99 model was referred from Zhu *et al.* [44], and the default simulation parameters settings were adopted [47], which are described in Table VIII.

Following the similar procedures in *IV. Simulation experiment* section, the main experimental results were presented. Figure 11 presents the diagrams of the true model and the established surrogate model.

VI. DISCUSSION

A. Application to Construct ODD

One of the important applications of the SPB is to help the manufacturers determine the Operational Design

TABLE VIII
SIMULATION PARAMETERS DESCRIPTION FOR W99 MODEL

| Simulation parameter	Description	Value
$CC0$	standstill distance (m)	1.50
$CC1$	Headway time (s)	0.90
$CC2$	Following variation (m)	4.00
$CC3$	Threshold for entering "following" state (s)	-8.00
$CC4$	Negative "following" threshold (m/s)	-0.35
$CC5$	Positive "following" threshold (m/s)	0.35
$CC6$	Speed dependency of oscillation	11.44
$CC7$	Oscillation acceleration (m/s^2)	0.25
$CC8$	Standstill acceleration (m/s^2)	3.50
$CC9$	Acceleration at 80 km/h (m/s^2)	1.20
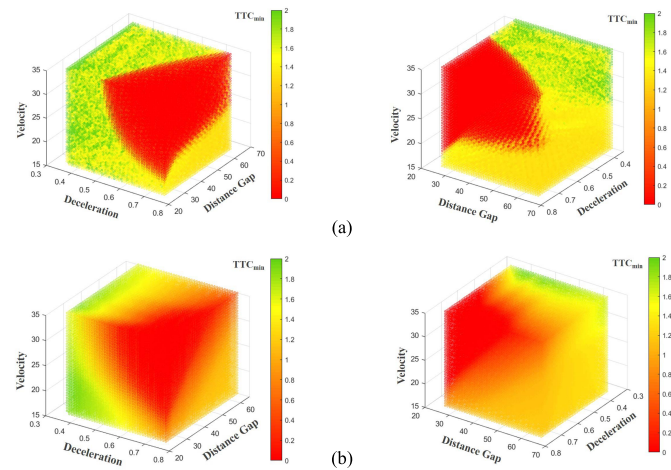
Fig. 11.   (a) Diagrams of the true model; (b) diagrams of the established surrogate model.

Domain (ODD) for HAVs. Based on the identified SPB and STPB, the parameter space can be divided into three parts, which are hazardous area, STZ, and safe area (shown in Figure 12). The boundaries of ODD can be then designed by SPB, and the STZ could be regarded as the subsidiary attribute of ODD corresponding to the magnitude of safety tolerance. When HAV drives into STZ and has the trend to fall into hazardous situations, proactive measures such as early warning, trajectory planning and control could be implemented to avoid potential risks.

Moreover, different HAV manufacturers may have different safety tolerance propensity, they can determine how large the safe tolerance of boundary need to be, according to their different requirements. For instance, for the need to guarantee HAV not to exit ODD [48], the safety tolerance degree should be set higher.

Figure 13 presents an example of the constructed ODD of the tested control algorithm under a deceleration of 0.50g.

B. Application to Compare Different Control Algorithms

Besides ODD construction, different HAV control algorithms could be compared according to their identified ODD. The results could help HAV manufacturers to improve the control algorithms to enhance safety performance.
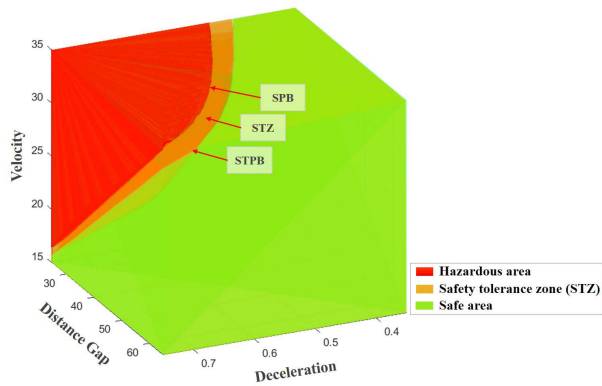
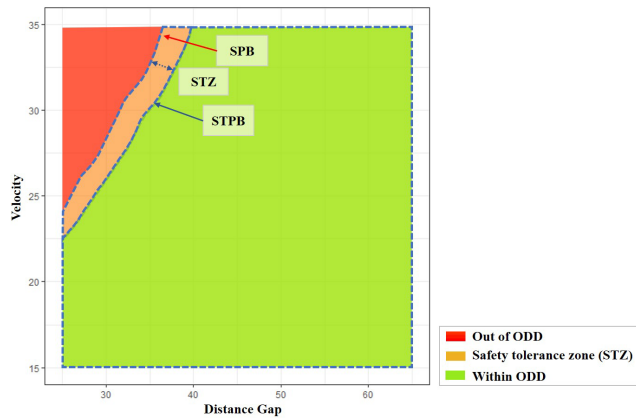Fig. 12.   The safety tolerance zone of the performance boundary.

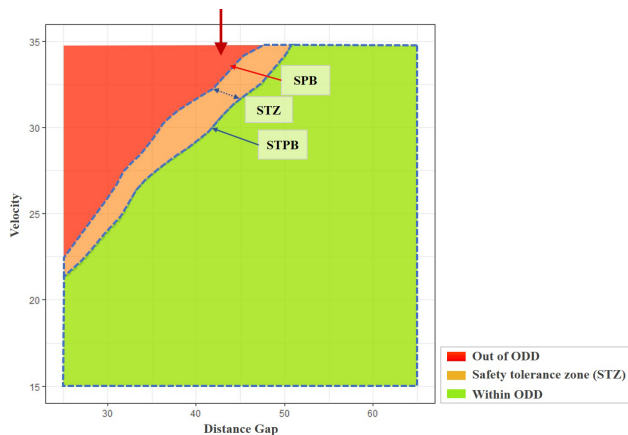Fig. 13.   The ODD for IDM (deceleration = 0.50g).

Fig. 14.   The ODD for W99 (deceleration = 0.50g).

The cross section of W99 under deceleration of 0.50g is presented in Figure 14. Compared with IDM (Figure 13), the ODD of W99 is smaller than that of IDM, indicating that under the same pre-defined emergency driving scenario, W99 cannot handle as many scenarios as IDM, especially in the decreasing distance gap. It might due to the transitions between different driving regimes of W99 need to be activated by several thresholds at the same time. Investigating into areas indicated by the red arrow in Figure 14, it is found that HAV

remains the 'following' or 'approaching' regime for a while, and not switches to the 'emergency' regime until the velocity gap and distance gap are both met at the threshold conditions. While IDM is 'gap sensitive' and will respond immediately when the distance gap is smaller than the safe one.

VII. CONCLUSION

Highly automated vehicle (HAV) has become the unavoidable trend in the transportation system, which is expected to reduce the traffic crash risk. To ensure the operational safety of HAVs, the Operational Design Domain (ODD) of HAVs where HAVs could operate safely is desired to be quickly identified through sufficient testing. Regarding the testing demand for the black-box HAV control algorithm, the coverage of safety-critical scenarios within the parameter space, and the searching efficiency, an accelerated testing framework by identifying the Safety performance boundary (SPB) was proposed.

There are four procedures of the framework: First, the testing parameter space is constructed according to the specific driving scenario and the key parameters; second, the surrogate model is established by MLP model; third, the safety-critical scenarios are searched through optimized gradient descent algorithms; and finally, the SPB and the STPB are identified. A three-vehicle following scenario where HAV is following a HDV and is followed by another HDV was taken as the case study to validate the proposed framework.

Compared with existing techniques, this study considered the black-box characteristic of HAV control algorithm, and employed the surrogate model to approximate the safety performance of HAV. Moreover, since establishing the surrogate model requires a large amount of scenario samples, an adaptive sampling method considering both exploration and exploitation was adopted. The proposed adaptive sampling method requires only 4% of scenarios in the parameter space, therefore, greatly accelerate the HAV testing process.

Besides, to identify SPB quickly and accurately, the optimized gradient descent searching algorithm was utilized. Based on the identified SPB, 97.42% of collision scenarios could be properly identified, and only 0.29% of non-collision scenarios were falsely identified. Furthermore, the concept of safety tolerance performance boundary (STPB) was proposed in this study, which has not been considered in previous work. It quantifies the safety tolerance of SPB, for situations where AV could drive too close to the SPB and has a sudden drop in its performance. HAV manufacturers could determine the safety tolerance according to their different requirements.

Finally, the proposed method could support the development of HAV decision-making control system for HAV manufactories. The Operational Design Domain (ODD) of Intelligent Driver Model (IDM) and Wiedemann 99 (W99) model control algorithms were constructed accordingly based on the established SPB. They were compared to shed light on the pros and cons of different algorithms.

Although this study provides insight for a novel SPB identification method, there are also several issues that we intend to investigate in future work. First, scenarios in the real world usually have higher dimensions, although the proposed framework is theoretically not limited to the number of

key parameters, it will inevitably increase the computational complexity as the dimension raises. Several approaches could be applied to solve the problem, for instance, investigating surrogate models that could adapt well to high dimensional problem, dividing the complex driving scenario into sub-scenarios, and applying dimensionality reduction method such as Principal Component Analysis (PCA), to reduce the number of parameters necessarily needed to build the surrogate model. Second, although the sampling method was not the focus of this study, it could be improved to reduce the sample size needed to construct surrogate model. Third, the shortcoming of the GD is that it can fall into local optimum, as a result some convergent points are not safety-critical scenarios and thus reduce the searching efficiency. The gradient descent searching algorithms could be improved in case of local optimization. Other searching algorithms such as the heuristic algorithms could be investigated to further enhance the searching efficiency especially for higher dimensional space. Finally, future studies can consider the indicator comprehensively such as both safety and comfort, to provide a multi-dimensional view for HAV performance evaluation.

References

[1] *Automated Driving Systems 2.0: A Vision for Safety*, NHTSA, Washington, DC, USA, 2017.

[2] *Smart Car Innovation Development Strategy*, Ministry Ind. Inf. Technol., Beijing, China, 2020.

[3] *On the Road to Automated Mobility: An EU Strategy for Mobility of the Future*, Eur. Commission, Brussels, Belgium, 2018.

[4] *Ensuring American Leadership in Automated Vehicle Technologies: Automated Vehicles 4.0*, NHTSA, Washington, DC, USA, 2020.

[5] S. E. Shladover, "The truth about 'self-driving' cars," *Sci. Amer.*, vol. 314, no. 6, pp. 52–57, May 2016.

[6] Y. Song, M. V. Chitturi, and D. A. Noyce, "Automated vehicle crash sequences: Patterns and potential uses in safety testing," *Accident Anal. Prevention*, vol. 153, Apr. 2021, Art. no. 106017.

[7] C. Xu, Z. Ding, C. Wang, and Z. Li, "Statistical analysis of the patterns and characteristics of connected and autonomous vehicle involved crashes," *J. Saf. Res.*, vol. 71, pp. 41–47, Dec. 2019.

[8] C. Luetge, "The German ethics code for automated and connected driving," *Philos. Technol.*, vol. 30, pp. 547–558, Sep. 2017.

[9] E. Thorn, S. C. Kimmel, M. Chaka, and B. A. Hamilton, "A framework for automated driving system testable cases and scenarios," United States Dept. Transp., Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. DOT HS 812 623, 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] R. Magizov, E. Mukhametdinov, and V. Mavrin, "Responsibility for causing harm as a result of a road accident involving a highly automated vehicle," in *Proc. 6th Int. Conf. Vehicle Technol. Intell. Transp. Syst.*, 2020, pp. 606–613.

[12] P. Junietz, W. Wachenfeld, K. Klonecki, and H. Winner, "Evaluation of different approaches to address safety validation of automated driving," in *Proc. IEEE 21st Int. ITSC*, Nov. 2018, pp. 491–496.

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

[14] I. Majzik *et al.*, "Towards system-level testing with coverage guarantees for autonomous vehicles," in *Proc. ACM/IEEE 22nd Int. Conf. Model Driven Eng. Lang. Syst. (MODELS)*, Sep. 2019, pp. 89–94.

[15] Z. Tahir and R. Alexander, "Coverage based testing for V&V and safety assurance of self-driving autonomous vehicles: A systematic literature review," in *Proc. Int. Conf. Artif. Intell. Test.*, 2020, pp. 23–30.

[16] D. Zhao *et al.*, "Accelerated evaluation of automated vehicles in lane change scenarios," in *Proc. ASME Dyn. Syst. Control Conf. (DSCC)*, vol. 1, 2016, p. 9.

[17] B. Zhu, P. Zhang, J. Zhao, and W. Deng, "Hazardous scenario enhanced generation for automated vehicle testing based on optimization searching method," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7321–7331, Jul. 2022, doi: 10.1109/TITS.2021.3068784.

[18] 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. IEEE 21st Int. ITSC*, Nov. 2018, pp. 3935–3940.

[19] M. Arief *et al.*, "Deep probabilistic accelerated evaluation: A robust certifiable rare-event simulation methodology for black-box safety-critical systems," in *Proc. Int. Conf. Artif. Intell. Statist.*, vol. 130, Mar. 2021, pp. 595–603.

[20] J. Sun, H. Zhang, H. Zhou, R. Yu, and Y. Tian, "Scenario-based test automation for highly automated vehicles: A review and paving the way for systematic safety assurance," *IEEE Trans. Intell. Transp. Syst.*, early access, Dec. 28, 2021, doi: 10.1109/TITS.2021.3136353.

[21] F. Batsch, S. Kanarachos, M. Cheah, R. Ponticelli, and M. Blundell, "A taxonomy of validation strategies to ensure the safe operation of highly automated vehicles," *J. Intell. Transp. Syst.*, vol. 26, no. 1, pp. 14–33, Jan. 2022.

[22] S. Khastgir, G. Dhadyalla, S. Birrell, S. Redmond, R. Addinall, and P. Jennings, "Test scenario generation for driving simulators using constrained randomization technique," SAE Tech. Paper 2017-01-1672, 2017.

[23] R. Ben Abdessalem, S. Nejati, L. C. Briand, and T. Stifter, "Testing advanced driver assistance systems using multi-objective search and neural networks," in *Proc. 31st IEEE/ACM 31st Int. Conf. Automated Softw. Eng.*, Aug. 2016, pp. 63–74.

[24] R. B. Abdessalem, S. Nejati, L. C. Briand, and T. Stifter, "Testing vision-based control systems using learnable evolutionary algorithms," in *Proc. IEEE/ACM 40th Int. Conf. Softw. Eng.*, May 2018, pp. 1016–1026.

[25] W. Ding, B. Chen, B. Li, K. J. Eun, and D. Zhao, "Multimodal safety-critical scenarios generation for decision-making algorithms evaluation," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 1551–1558, Apr. 2021.

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

[27] F. Batsch, A. Daneshkhah, M. Cheah, S. Kanarachos, and A. Baxendale, "Performance boundary identification for the evaluation of automated vehicles using Gaussian process classification," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 419–424.

[28] J. Sun, H. Zhou, H. Zhang, Y. Tian, and Q. Ji, "Adaptive design of experiments for accelerated safety evaluation of automated vehicles," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 1–7.

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

[30] Z. Huang, H. Lam, and D. Zhao, "Towards affordable on-track testing for autonomous vehicle—A kriging-based statistical approach," in *Proc. IEEE 20th Int. ITSC*, Oct. 2017, pp. 1–6.

[31] H. Beglerovic, M. Stolz, and M. Horn, "Testing of autonomous vehicles using surrogate models and stochastic optimization," in *Proc. IEEE 20th Int. ITSC*, Oct. 2017, pp. 1–6.

[32] A. Ajdari and H. Mahlooji, "An adaptive exploration-exploitation algorithm for constructing metamodels in random simulation using a novel sequential experimental design," *Commun. Statist., Simul. Comput.*, vol. 43, no. 5, pp. 947–968, Jan. 2014.

[33] J. Eason and S. Cremaschi, "Adaptive sequential sampling for surrogate model generation with artificial neural networks," *Comput. Chem. Eng.*, vol. 68, pp. 220–232, Sep. 2014.

[34] C. Hydén, "Traffic conflicts technique: State of the art," in *Traffic Safety Work With Video Processing*. Kaiserslautern, Germany: Univ. Kaiserslautern, Transportation Department, 1996, pp. 3–14.

[35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[36] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.

[37] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural Netw.*, vol. 12, no. 1, pp. 145–151, Jan. 1999.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[39] C. K. Chui, *Multivariate Splines*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1988.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

12   IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS

[40] MathWorks. (2021). *Curve Fitting Toolbox User's Guide, Version 3.5.13*. [Online]. Available: https://www.mathworks.com/help/pdf_doc/curvefit/curvefit.pdf

[41] X. Wang, M. Zhu, M. Chen, and P. Tremont, "Drivers' rear end collision avoidance behaviors under different levels of situational urgency," *Transp. Res. C, Emerg. Technol.*, vol. 71, pp. 419–433, Oct. 2016.

[42] M. Treiber, A. Kesting, and D. Helbing, "Delays, inaccuracies and anticipation in microscopic traffic models," *Phys. A, Stat. Mech. Appl.*, vol. 360, no. 1, pp. 71–88, Jan. 2006.

[43] W. Do, O. M. Rouhani, and L. Miranda-Moreno, "Simulation-based connected and automated vehicle models on highway sections: A literature review," *J. Adv. Transp.*, vol. 2019, pp. 1–14, Jun. 2019.

[44] M. Zhu, X. Wang, A. Tarko, and S. Fang, "Modeling car-following behavior on urban expressways in Shanghai: A naturalistic driving study," *Transp. Res. C, Emerg. Technol.*, vol. 93, pp. 425–445, Aug. 2018.

[45] A. R. A. Van der Horst, "A time-based analysis of road user behaviour in normal and critical encounters," Ph.D. dissertation, Dept. Electr., Math. Comput. Sci., Delft Univ. Technol., Delft, The Netherlands, 1999.

[46] R. Wiedemann, "Simulation des straßenverkehrsflusses," Inst. Transp. Sci., Univ. Karlsruh, Karlsruhe, Germany, Schriftenreihe Heft 8, 1994.

[47] A. Anil Chaudhari, K. K. Srinivasan, B. Rama Chilukuri, M. Treiber, and O. Okhrin, "Calibrating Wiedemann-99 model parameters to trajectory data of mixed vehicular traffic," *Transp. Res. Rec.*, vol. 2676, no. 1, pp. 718–735, 2022, doi: 10.1177/03611981211037543.

[48] M. Gyllenhammar *et al.*, "Towards an operational design domain that supports the safety argumentation of an automated driving system," in *Proc. 10th Eur. Congr. ERTS*, Jan. 2020, pp. 1–10.

Rongjie Yu received the B.Sc. degree from Tongji University in 2010 and the M.Sc. and Ph.D. degrees in traffic engineering from the University of Central Florida in 2012 and 2013, respectively. He is currently an Associate Professor with the College of Transportation Engineering, Tongji University. His research interests include traffic safety, human behavior, and safety evaluation of connected and autonomous vehicles.

Shuhan Qiu received the B.Sc. degree in transportation engineering from the College of Transportation Engineering, Tongji University, in 2020. He is currently pursuing the Ph.D. degree with the College of Transportation Engineering, Tongji University. His main research interests include driving behavior analysis, path flow reconstruction, and connected and autonomous vehicles.

Jian Sun received the Ph.D. degree from Tongji University in 2006. He was with Tongji University as a Lecturer and then promoted to a Full Professor in 2011. He is currently a Professor with the College of Transportation Engineering; and the Chair of the Department of Traffic Engineering, Tongji University. His main research interests include traffic flow theory, traffic simulation, and connected and autonomous vehicles.

Yiyun Wang received the B.Sc. degree in transportation engineering from the College of Transportation Engineering, Tongji University, in 2017, where she is currently pursuing the Ph.D. degree with the College of Transportation Engineering. Her main research interests include traffic safety analysis and safety evaluation of connected and autonomous vehicles.

Haneen Farah received the Ph.D. degree in transportation engineering from the Technion—Israel Institute of Technology in 2009. She is currently an Associate Professor with the Department of Transport and Planning and also the Co-Director of the Traffic and Transportation Safety Laboratory, Delft University of Technology. Her main research interests include traffic safety, road design, road user behavior, and connected and automated vehicles.