Verification and Validation Methods for Decision-Making and Planning of Automated Vehicles: A Review

Yining Ma, Chen Sun, *Graduate Student Member, IEEE,* Junyi Chen, Dongpu Cao, *Member, IEEE,* and Lu Xiong

***Abstract*—Verification and validation (V&V) hold a significant position in the research and development of automated vehicles (AVs). Current literature indicates that different V&V techniques have been implemented in the decision-making and planning (DMP) system to improve AVs' safety, comfort, and energy optimization. This paper aims to review a range of different V&V approaches for the DMP system of AVs and divides these approaches into three distinct categories: scenario-based testing, fault injection testing, and formal verification. Further, scenario-based testing is categorized into fundamental and advanced approaches based on the interaction between road users in generated scenarios. In this paper, six criteria are proposed to compare and evaluate the characteristics of V&V approaches, which could help researchers gain insight into the benefits and limitations of the reviewed approaches and assist with approach choices. Next, the DMP system is broken down into a hierarchy of modules, and the functional requirements of each module are deduced. The suitable approaches are matched to verify and validate each module aiming at their different functional requirements. Finally, the current challenges and future research directions are concluded.**

***Index Terms*—Verification and Validation, Decision Making, Planning, Automated Vehicles, Survey.**

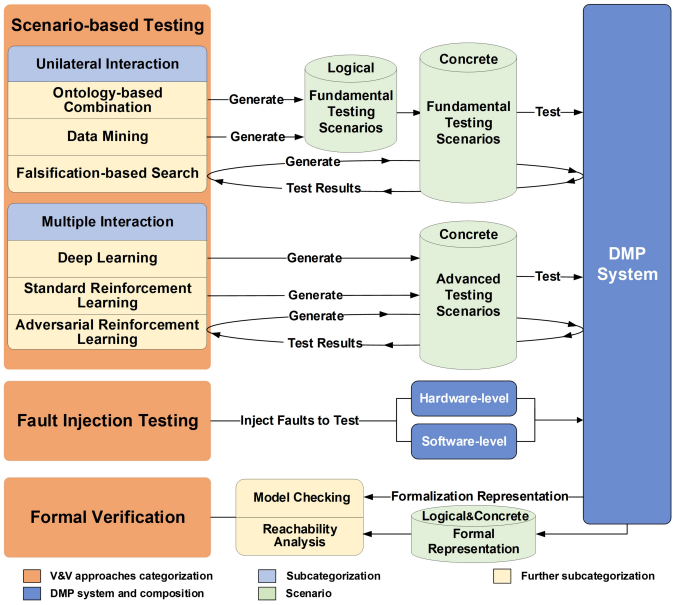


Fig. 1. The state-of-the-art V&V approaches for the decision-making and planning (DMP) system are categorized into three parts, including Scenario-based Testing, Fault Injection Testing, and Formal Verification.

I. Introduction

AUTOMATED vehicles (AVs) are a promising evolution for a better prospect of reducing accidents, pollution, and congestion while increasing transport accessibility and saving fuel [1], [2]. However, there are still many substantial challenges in achieving the industrialization of AVs [3], [4]. The National Highway Traffic Safety Administration (NHTSA) has identified 11 crashes since 2018 involving various Tesla models that were using autopilot systems according to documents filed in Aug. 2021 [5]. And NHTSA estimates that thousands of crashes involving Level 2 AVs will occur each year in addition to 200 crashes involving Level 3-5 AVs in the future [6]. To prevent the occurrence of the above accidents, verification and validation (V&V) play a significant role [7]–[9], and may be a decisive factor for the introduction of AVs into the market [10], [11]. The mere distinction between V&V is minute yet substantive, acknowledging that validation ensures that the product caters to the needs, whereas, verification ensures the correctness and aptness of the product. In most cases of system development, validation is left till nearly the end of the project when the user meets the system to give her or his final approval. Within this context the word testing would be the tool to examine these two aspects [12].

Before the mass production of AVs, safety along with module-wise performances need to be guaranteed by completing the system level V&V, including perception, decision-making and planning (DMP), and control systems [13]. Among them, the DMP system is the crucial driving factor underpinning autonomy [14], [15]. The DMP system is at the core of the entire information transmission chain inside an AV [16], [17]. It needs to process and interact with the dynamic, uncontrollable, and diverse environment information obtained from the perception system continuously and sends the correct messages to the control system [18]. Moreover, as a sophisticated system, the DMP is composed of several modules. The logical relationships between each module are especially

This work was supported in part by the National Key Research and Development Program of China under Grant 2021YFB2501205, and the Perspective Study Funding of Nanchang Automotive Institute of Intelligence and New Energy under Grant TPD-TC202110-05. *(Corresponding author: Junyi Chen)*

Y. Ma, J. Chen, and L. Xiong are with the School of Automotive Studies, and also with the Institute of Intelligent Vehicles, Tongji University, Shanghai 201804, China. (e-mail:myn,chenjunyi,xiong_lu@tongji.edu.cn).

C. Sun is with the Department of Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada. (e-mail: chen.sun@uwaterloo.ca).

D. Cao is with the School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China.(e-mail:dp_cao2016@163.com).

complex due to the internal influences and interactions, which results in functional problems and brings great difficulties in implementing decision-making algorithms. Hence, much V&V work for the DMP system is needed to ensure real-time autopilot performances.

Recently, many published research studies proposed the V&V methods mainly used in the DMP system. However, the existing surveys concentrate on vehicle-level testing or the testing of perception systems for AVs' safety [15], [19]–[22]. Compared with the mentioned surveys, the main contributions of this paper are listed as follows. (a) The V&V approaches for the DMP system are reviewed, and they are classified into three categories, namely, scenario-based testing, fault injection testing, and formal verification. The schematic overview of this categorization is depicted in Fig. 1, which could provide guidance in selecting appropriate V&V strategies based on the specific use case. In particular, a novel classification approach based on the interaction relation between road users in scenario-based testing is proposed. (b) Six criteria are proposed to evaluate the characteristics of the V&V approaches, allowing the understanding of the strengths and weaknesses of both the scenario generation methods and the application-wise metrics. (c) Module-wise analyses are made in detail of the DMP system, and the functional requirement for each part is deduced. The performances of each V&V approach are discussed, aiming at every functional requirement. Finally, the research gaps and challenges are listed with the necessary future research trends and potential directions.

The remainder of the paper is structured as follows: In Section II, III, and IV, the in-depth reviews of the three categorized V&V approaches, scenario-based testing, fault injection, and formal verification are presented, followed by the discussions of their specific characteristics. In Section V, we propose six evaluation criteria to compare the V&V approaches. Moreover, the workflow and functional requirements of four modules in the DMP system are clarified. Aiming at each module's functional requirement, we match the suitable V&V approaches for it. Finally, Section VI presents the remaining challenges and future research directions in V&V developments based on the discussions and comparisons above.

II. Scenario-Based Testing

Combined with the characteristics of AVs and the traffic environment they are in, scenarios-based testing is advantageous and has been used extensively in the current state of the art [23]. Scenario-based testing has the advantages of flexible configuration, high efficiency, safe guarantee, and low cost. Automatic and accelerated tests can be realized through technologies such as mirror worlds, shadow systems, digital twins, and parallel computing, which can reduce the labor costs [24], [25]. Therefore, it has become an indispensable part of AVs' validation and evaluation [26]. In general, AV testers or scenario designers prefer to generate challenging or complex scenarios where the core task of the DMP system under test (DMP-SUT) is to interact with kinds of dynamic entities. From the perspective of the interaction relationship between road users in testing scenarios, two types of scenarios can be generated. One type of scenario is "fundamental" with predefined and unilateral interaction, where only the SUT can react to other traffic participants. The other is more "advanced" associated with multiple intelligent components. The traffic participants in the advanced scenario are implemented with intelligent driver models or controlled by other humans in a multi-player game fashion [27].

A. Fundamental Scenario for Testing

1) Ontology-based Combination The ontology-based combination is an essential approach to generate testing scenarios, which combines scenario entities based on ontology theory for the primary goal of coverage. Ontology is known as a formal and explicit conceptualization of entities, interfaces, behaviors, and relationships [28], [29]. In the test scenario for the DMP system, the concept of ontologies includes the knowledge of environment entities about traffic infrastructure design and the traffic participants' behaviors, and interactions [30].

Previous studies that applied ontology to AV are generally used to develop the DMP system [31], [32]. Geyer et al. first proposed ontology for testing scenario generation, which is a positive contribution for simplifying communication and the exchange of findings [33]. Then Bagschik et al. proposed a process for an ontology-based scene creation method consisting of three parts: the knowledge acquisition, the knowledge modeling, and the combination process [30]. These created scenes in natural language can be used to investigate and define an AV's safe behavior decision in the concept phase or to derive testing scenarios for simulation environments with a wide range of varieties.

Research [34] focuses on acclimatizing the operational design domain (ODD) in the development and test cycle of an automated driving system (ADS). Their innovation lies in the adoption of a unified ODD ontology and attribute formal, quantifiable modeling of the driving environment. Another ontology-based approach is proposed in [35] to automatically generate scenarios for testing the inside decision algorithms of the Advanced Driver-Assistance Systems (ADAS) in the highway context. The testing scenario contains three layers: basic layer, interaction layer, and generation layer. The hierarchy exploits static and mobile concepts they have defined in the context of three ontologies entities: highway, weather, and vehicle. They put forward three types of relationships between the entities of the highway ontology: inheritance relationship (unary), composition relationship (binary), and position relationship (binary) and combined them to infer more complex relationships and scenarios.

The ontology-based approach can transform scenario description from semantic expression to coded computer language [36]. Li et al. discussed how to automatically convert ontologies to the corresponding combinatorial testing input models and presented two conversion algorithms to generate testing scenarios [28], [37]. A crash scenario shows an example where pedestrians are involved crossing the street, and the attributes of pedestrians and ego-vehicle are constructed with well-defined constraints. Two algorithms are used to convert the attributes to parameters and the parameters are inputted into a combinatorial testing model for the Autonomous

Emergency Braking (AEB) function. Because the combination model is in a virtual simulation environment without complex scenario entities for perception and focuses on validating the strategies and behaviors of ego-vehicle, the test of the AEB function in this article is essentially a test of the DMP system. In addition to the academia, Christian et al. [38] from Mercedes-Benz proposed an ontology-based methodology for identifying driving scenarios in abstracted field data. The abstraction is achieved by using universal elements of an ontology represented by a domain model. By examining a set of test data, they showed that this proposed method is appropriate for identifying further driving scenarios.

In terms of ontology-based combination, the construction of the input model is essential. The ontology model provides a common and shared understanding of knowledge (i.e., entities and their attributes) that is easy to organize, maintain and update in a machine-processable format [39]. If the ontology model is comprehensive and contains various entities with different attributes, the entities and attributes can be combined to derive many testing scenarios with high coverage. Also, this method is biased towards theoretical analysis, which relies heavily on expert experience. Expert knowledge modeled for computer-aided processing helps to provide interpretable scenarios. Besides, due to the constructed input model being more macroscopic and abstract, this method is applied chiefly to generate logical scenarios [23]. The generated logical scenarios can be further detailed to more concrete scenario combined with other processes [23], [40]. However, the number of scenarios grows exponentially with the increase of environment entities, resulting in a large number of less efficient scenarios even with those physically unreasonable and meaningless test cases. Moreover, these scenarios are within the knowledge framework of experts, and it is not easy to automatically explore the edge cases.

2) Data Mining Data mining extracts the critical data from naturalistic driving data sets to generate challenging scenarios [41]. PEGASUS project developed a recording and integrating relevant traffic scenarios from real-world measurements into a database and implemented them to test AVs [42]–[44]. After individually converting the raw data from environment perception into harmonized signal names, data structure, and coordinate systems as the scene's snippets, time-continuous snippets' likelihoods are calculated and added to each data set. These scene snippets are clustered and could be used to generate logical scenarios. For evaluating the appropriateness of the DMP system, it is required to consider the interaction between human drivers and AVs, so the generated logical scenario must include the relationship between human drivers, which means the interaction likelihoods of the continuous scenes should be contained in the data set [42].

Data mining also could be used to generate concrete scenarios directly. Based on the assumption that sensors and controls work correctly [45], Zhao et al. used the Importance Sampling (IS) technique to realize much faster stochastic simulations, which can be used to generate motions of the other primary background vehicles to accelerate the verification of AVs in simulations [18], [46]. The core idea of IS is to replace the original joint density function with a new density function that has a higher likelihood for rare events to happen. Therefore, the background vehicles' behaviors are intense, which is applied for verifying the DMP system in the simulation environment. The IS approach is also actively used in industry to extract critical scenarios efficiently to accelerate testing. However, when applying this approach, the initial scenes could not be optimized into regions of higher criticality. Jesenski et al. [47] from Robert Bosch used sum-product networks combined with the IS approach to avoid this problem so that the combined approach can find a better distribution.

Compared with other scenario generation methods for unilateral and non-intelligent interaction, scenarios generated through data mining have high fidelity because they are extracted from plenty of naturalistic driving data [48]. Generally, testers prefer to extract data from rare events or safety-critical events in the real world [49]. This method concentrates these critical data in the generated testing scenarios, thus significantly improving test efficiency over road validation. Since these events have already occurred, they have better traceability. However, the critical data are rare because massive data are from safe situations and without test significance, and the collection process is consuming. Another limitation of this method is that the data is fixed after being used to generate the scenario, which means the scenario generation process cannot automatically explore the edge case. The test scenario will not evolve according to the behavior of the DMP-SUT. Hence this method can only be used to generate static scenarios. Also, there is no feedback loop for the evaluation, making it less efficient than the following closed-loop method methods. The process of scenario generation from data mining is similar to screening out important fragments and then accelerating playback, so the mined data can only be used for fixed roads and topologies.

3) Falsification-based Search The DMP system can be searched for fails in the scenario generation by utilizing the idea of falsification in software engineering for reference. This approach aims to search counterexamples violating the functional requirements under test [50], [51]. The testing results are evaluated and then taken as the optimization goal to generate more critical scenarios iteratively in the loop. In particular, Kapinski et al. pointed out in [52] that one of the primary purposes of testing is to find critical scenarios that can cause system failure effectively. If the testing process is based on evolutionary algorithms, it improves the effectiveness and efficiency for finding critical scenarios by transforming testing processes into optimized search problems, and the evolutionary computations are applied to solve these problems. Most optimization algorithms can directly generate concrete scenarios without any assignment to predefined functional or logical scenarios. These algorithms can typically search novelties in existing parameter distribution and generate critical concrete scenarios.

The DMP system is a complex system with multiple modules and features combinations. One challenge is that features tend to interact or impact each other's behaviors in unknown ways. For detecting and managing the feature interactions that violate system requirements and lead to failures, a many-objective search method based on the combination of tra-

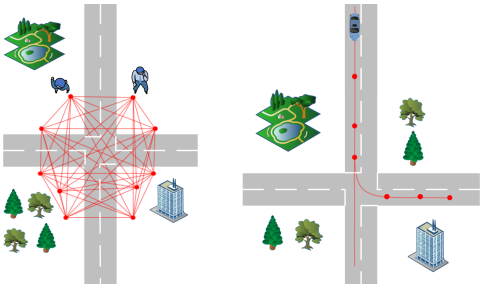


Fig. 2. Left: the nodes a pedestrian can visit, and their edges allow both normal and jaywalking behavior. Right: the graph determines the background vehicles' decision points [53].

ditional coverage heuristics with new heuristics is proposed in [54]. This paper converts the task of detecting undesired feature interactions into a search-based testing work. Through testing on two versions of industrial ADS, the proposed method can significantly improve feature interaction failure identification compared with the baseline search method.

Mullins et al. [55], [56] designed a method to search key parameters oriented to the DMP system. Because the most informative scenario configurations for testing path searching algorithms occur in the transition regions between performance modes, they used adaptive sampling to search the state space for testing scenarios on the boundary between distinct performance modes. This method takes a scenario configuration and mission parameters from the search results as an input and returns a set of score metrics of its mission performance as output. In later work, through unsupervised learning with clustering techniques, they improved the generation speed, and efficiency of critical scenarios [57].

Authors in [58] and [59] presented an approach that automatically creates critical testing scenarios with a small solution space to verify the path search for successful collision avoidance. The approach combines reachability analysis for determining the size of the solution space with optimization techniques to shrink it. The solution space is reduced by shifting the initial states of traffic participants via binary search, meanwhile demands an immediate and correct action of the vehicle under testing.

In order to automate the generation of scenarios in high-fidelity simulators, authors in [53] use Bayesian Optimization to search adversarial testing scenarios that increase collision risk with simulated traffic participants and expose poorly-engineered decision-making policies. To make their optimization process efficient and avoid wasteful calculation, they design a hierarchy on the pedestrian and vehicle behavior that uses a simulator map. In such case, the pedestrians and background vehicles have to make decisions on some high-level parametrization points, as illustrated in Fig. 2.

Since the search process takes the fail performance of DMP-SUT as a purpose, the scenario has high efficiency in falsification after being generated. This method also improves the availability of developing parametric scenarios and mitigates the risks that the number of useless scenarios explodes with the simulation environment's scenario parameters [60]. However, as a typical falsification-based method, the apparent limitation of the search method is that it can only prove there are issues but not vice versa. In addition, this method is black-box testing, whereas the DMP system is a complex system that comprises a few different integrated modules. Developers of the system may have expertise in the individual module. Still, the complex interplay of these modules resulting in the final emergent behavior cannot be easily characterized or predicted. Therefore, black-box testing could not reveal the exact problem spots for developers. Besides, existing research still has some limitations in analyzing search space, determining the parameter space's initial source.

In addition to search methods, another representative method that belongs to falsification is adversarial reinforcement learning, which will be further discussed in the following advanced scenario generation part.

B. Advanced Scenario for Testing

Machine learning (ML) has been used in the development of AVs widely [61]–[63]. Several mainstreams of ML, such as deep learning (DL) and reinforcement learning (RL) approaches, are increasingly used in scenario generation to test the DMP system [64]. The following ML techniques generate scenarios used to test high-level AVs primarily due to the multiple and intelligent interactions among road users. These kinds of scenarios are called advanced in this review.

1) Deep Learning Classical ML methods (such as support vector machine, linear regression, K-means clustering) are restricted to their ability to process massive naturalistic raw data [65]. In contrast, DL technologies discover intricate structures well in high-dimensional data and learn the idea of correct representation of data [66], [67]. Nowadays, the ability of deep models to generalize well from limited data sets has many applications in AV testing scenario generation [68], [69].

The neural networks in a recurrent (RNN) frame of accounting the history, such as long short-term memory (LSTM) [70] and gated recurrent units (GRU), can tackle sequences ordered by time, like the trajectories formed by the interaction behaviors among road users [71], [72]. Jenkins et al. generate potential testing scenarios based on modeling accident data with RNNs and LSTM cells without manual specification in [73]. The prototype of scenario generation is shown in Fig. 3. The input and output data of RNN are represented by orange dashed lines. The black line shows the flow of information within RNN. Because the agents in the scenario are trained based on accident data, they may exhibit dangerous behaviors when interacting with the tested vehicle. However, this strategy lacks interpretability because RNN is an end-to-end training method in substance. Besides, compared with the data from naturalistic driving, the data used for training in this research comes from a simulator which lacks fidelity.

Krajewski et al. reasoned that only failure scenarios are inadequate for testing high-level AVs, as other scenarios (e.g., non-collision scenarios) have to be tested as well [74]. Therefore, they use unsupervised DL to train a neural network (NN) to model lane change trajectories in a scenario automatically. More than 5600 measured lane change trajectories extracted

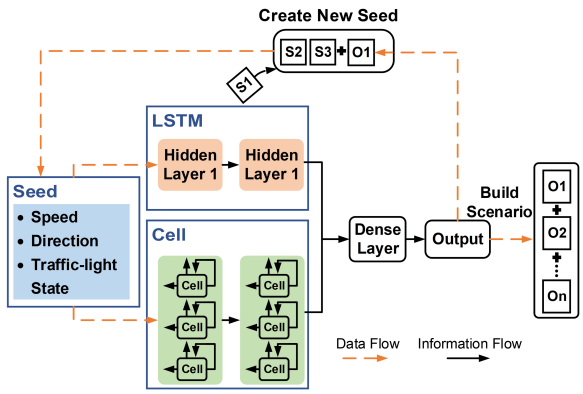


Fig. 3. The prototype design of RNN with LSTM [73].

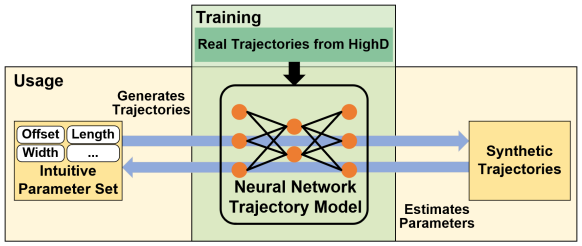


Fig. 4. The unsupervised deep learning (DL) uses highD data set to train a neural network (NN) to generate trajectories [74].

from the highD data set are used to train. The NNs can map the existing trajectories to the learned parameters to generalize numerous trajectories and perform the inverse mapping from the synthetic trajectories to corresponding parameter values under the framework shown in Fig. 4.

As DL heavily relies on existing data resulting in low flexibility, the generated behaviors of traffic participants (Non-Player Characters, or NPCs for short) are often difficult to modify according to testing intentions. Authors in [71] introduced a multi-vehicle trajectory generator (MTG) integrated with the GRU module to encode multi-vehicle interaction scenarios into a better interpretable representation. The scheme of MTG consisting of three parts: encoder (orange), sampling process (blue), and decoder (green) are illustrated as Fig. 5. New driving interaction scenarios are generated by sampling using publicly available data. MTG mainly contributes to generating variable trajectories for interaction in complex traffic scenarios, which could provide more high-fidelity data for engineers and researchers to develop and validate the interaction strategy of the DMP system and develop evaluation scenarios for AVs.

DL-generated trajectories can imitate and generalize the interaction behavior in the real world [67], and we think the NPCs generated by this method are intelligent in the interaction process. If the DL model is well trained, more scenarios can be derived from the original scenarios. Although a strength of DL is the ability to generalize from training data, the challenge for analysis is that testers cannot completely understand how they generalize for all possible cases due to the internal hidden network. Owing to the parameters learned based on training data, it is hard to characterize

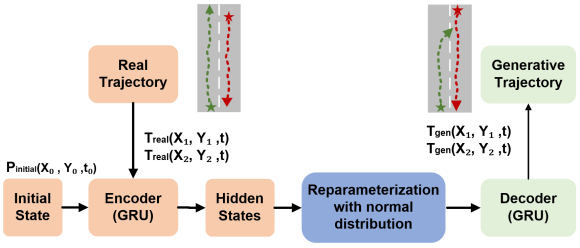


Fig. 5. An illustration of the scheme of multi-vehicle trajectory generator (MTG) [71].

the NPCs' behaviors. In other words, characterizing NPCs' behaviors is as difficult as characterizing the training data [75]. Therefore, the testing scenarios generated from this method are relatively weak in interpretability. Moreover, many released databases [76] do not offer sufficient information on multi-vehicle interaction scenarios because of technical limitations and the required costs of collecting data [77]. Besides, if the data set is replaced, the effect may not be satisfactory. The training data set can only be continuously increased to try to cover all possible states. Still, it is not practical because the cost of collecting data is high in some dangerous states.

2) Standard Reinforcement Learning RL is an agent learning via interaction with its environment driven by a feedback signal (reward) [78]. The environment reinforces the agent to determine better actions to promote the learning process [79]. RL involves many branching algorithms, and the standard RL mainly focuses on the NPCs, whereas the adversarial RL (will be described in the following sub-section) concentrates on the DMP-SUT during the training process. In terms of DMP-SUT is mainly used in the final application and validation stage, whereas the adversarial RL approaches are used in both the failure scenario training and validation process.

Li et al. combine a specific game-theoretic formalism—hierarchical reasoning (also referred to as "level-k") game theory to evolving player interactions for generating a time-extended (multi-move) scenario [80]. The logic of the traffic model is abstract from the "level-k" game theory, and the multi-agents are trained as three levels of decision-making logical rules and these rules are adjusted though changing the weight in the reword functions [81]. Two kinds of decision-making algorithms (Stackelberg and Decision Tree) are verified and validated in the generated testing scenarios [82]. In the following work, the authors used deep RL (DRL) to realize the hierarchical game in simulation, which reflects the real world because of reducing the crash rate. This method can model a dramatically larger class of scenarios than previous research benefit from the exploitation of deep Q-learning (DQL) [83]. To increase the variety of interactions between the DMP-SUT and the surrounding vehicles, authors in [84] characterized the interactive behavior using level-k game theory and social value orientation and trained a diverse set of surrounding vehicles using Double Deep-Q network (DDQN) algorithm. The results show that the method can simulate a wide range of interaction behaviors and also effectively identify some safety-critical test scenarios.

RL is a trial-and-error process and is usually combined

with specific game-theoretic models to generate a scalable-to-multiple-vehicles traffic model in a simulation environment. It can be used to explore more unforeseen scenarios for validation. Besides, many different decision-making algorithms could also be easily integrated and tested in the scenario generated by standard RL. However, this scenario is not well-targeted and may not reveal the weaknesses of DMP-SUT precisely compared with the scenario generated by accident data. Moreover, tabular RL strategies that explore randomly and learn have low efficiency, and become hard to calculate in high-dimensional state space and complex environments. Although RL combines with NN, such as DRL or DQL, can deal with high-dimensional data input issue [85], it is less stable due to the sensitivity to hyper-parameter settings and architecture choices [86].

3) Adversarial Reinforcement Learning Intuitively, the evaluation would be much more efficient if an agent can find the weakness of the DMP-SUT based on its behaviors and guide the change of testing scenario to be more challenging, adaptively. Specifically, the adversarial agents generated by RL or DRL can evaluate closed-loop properties and find an optimal policy to achieve the highest discounted cumulative rewards, making the testing and evaluation process a fully competitive zero-sum game [75]. Fig. 6 shows the adversarial agents (NPCs) training structure that the DMP-SUT (same as Player) is in the loop. This technology focuses on determining perturbations in the configuration of a testing scenario where contains the DMP-SUT of ego-vehicle, meaning that seeking to find scenarios that lead to unexpected behaviors, such as unsafe behaviors and potential collisions both in the training and testing periods.

To efficiently finding failure scenarios, authors in [87] trained the adversarial agents using multi-agent RL where the tested rule-based agent makes collisions. The authors put forward personal and adversarial rewards to balance the relationship between testability and authenticity. They present a contributor identification to identify the NPCs that should be provided with an adversarial reward and use prioritized sampling by replay buffer partition to resolve the problem that NPCs cannot acquire an acceptable policy.

A novel framework based on DRL to measure the reliability of motion planning and collision avoidance mechanisms in AVs under the worst-case scenario of dealing with an optimal adversarial agent and trained to drive the system into unsafe states is proposed recently in [88]. They argue that both the training and test-time procedures of adversarial policies provide quantitative measures of reliability, which can be used for benchmarking behaviors in worst-case scenarios. Except for generation the adversarial testing scenario using DRL, authors in [89] utilized a nonparametric Bayesian approach to cluster the adversarial policies of the interactive vehicles. Significantly, they care more about diversity rather than optimality when generating adversarial scenarios. This paper is achieved by performing ensemble RL with random initializations and no exploration, aiming to collect adversarial policies' local optimums. Similarly, to increase the diversity of background vehicle trajectories in the test scenario,Du et al. propose an extension of the reward function used by the Monte Carlo Tree Search (MCTS) solver to encourage exploration in an efficient way by adding a trajectory dissimilarity component [90].

Cho and Behl integrated Baidu's Apollo into the LG Silicon Valley Lab (LGSVL) simulator (a photo-realistic AV simulator that accepts scenario parameters through a Python API) to generate failure and unexpected traffic scenarios for the AV software implementation [91]. For an input scenario, the API provides a way to vary the simulator's static and dynamic parameters. The critical insight is that the simulator's parameter space is the RL agent's action space. Their first test case aims at the perception system, and the second is at the DMP system. The second test only changed the NPC velocity, leading to the Apollo crash. Finally, they point out that the main computational bottleneck in training the RL model came from the LGSVL simulator because of the rendering problem.

The standard RL and other scenario generation methods lack weakness-aiming and evolving abilities, leading to insufficiency. Adversarial RL solves this problem and improves test efficiency. However, the disadvantage of adversarial RL is that it may lead to over-fitting for one type of DMP-SUT and the generated scenario is not highly real and universal compared with standard RL. It is not easy to obtain the tested DMP system algorithm due to the security measures. Like standard RL, adversarial RL also has better exploitation and exploration capabilities than DL in scenarios' evolvability. How to balance the trade-off between exploitation and exploration remains a future challenge.

C. Discussion

Fundamental scenarios generated by the previous mentioned methods are like predefined scripts. When the traffic participants interact with the DMP-SUT, their initial locations, approximate trajectories, and output behaviors are specified by the scenario designers or testers. Their speeds are also automatically selected from a wide range of options, such as the falsification-based search algorithm traverses as far as possible within the fixed-parameter space. As a result, the scope of validation for scripted scenarios is also limited. In general, fundamental testing scenarios are primarily used to verify AVs' basic functions, such as the ADAS test [92], or are more meaningful for specific performance tests.

It is vital to understand the interaction with dynamic entities in the driving environment to develop L3+ AVs efficiently [93], [94]. Hence, the testing scenarios for high-level AVs should contain some intelligent interaction tests. However, the traffic participants generated in fundamental scenarios cannot interact with ego-vehicle (be tested vehicle) or other traffic participants adaptively, which may avoid the possible conflict or risk behaviors. Therefore, more effective and fidelity interaction and game process should be applied for testing the behavior decision with the best result of the high-level AVs.

Advanced testing scenarios are the inevitable goal of future development. Road users interact intelligently with each other in advanced scenarios. Moreover, interactions between multiple intelligent entities mean that numerous possibilities could occur, which increases the uncertainties of driving behaviors

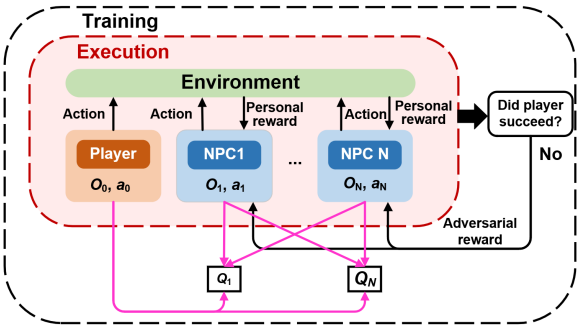


Fig. 6. A training process of a player and N NPCs [87].

[95]–[97]. Hence, it is difficult for AVs to predict the motion of traffic participants accurately. Therefore, these kinds of scenarios are more complex and suitable for testing high-level intelligent AVs.

III. Fault Injection Testing

Fault injection (FI) is a well-established approach for testing the resilience and error-handling capabilities of computing, and cyber-physical systems under faults [98], which has already been used in AVs' V&V broadly [99]. It can be used to verify system-level robustness by deliberately introducing faults in situations that might otherwise be rarely tested [100]. We reviewed the injection methods and faults types in terms of injection objects divided into software and hardware.

A. Fault Injection in Hardware

Heavy-ion radiation, high temperature and high pressure (HTHP), electromagnetic interference, and other FI techniques [100], [101] are used to test the capacity of resisting disturbance of the hardware processors. For example, FI can cover/restrain the obtainment of sensor hardware or break a classification or identification from the recognition system to the DMP system, or intercept and corrupt a behavior command from the DMP system and then forward them to the controller [102]. Saurabh Jha et al. conducted a Bayesian FI framework named DriveFI, which can find faults through causal and counterfactual reasoning about the behavior under a fault [103]. This framework includes the fault propagation and masking in GPUs hardware-level. The results show that 1.9% of injected faults (2400 experiments) led to the corruption of actuation outputs, which are the module's final outputs.

B. Fault Injection in Software

Software FI is more inclined to implementation details, which can be applied to program state and communication [104]. Specifically, altering logic and arithmetic expressions in the program by modifying constants, operators, variables, and the source code are used to test the software level. For example, FI confuses the sensor data from the recognition algorithm, modifies the state of dynamic participants, or generates disturbing data according to a sensor-specific fault model, and then forwards it to the DMP system to test the output results of the DMP system [102], [105]. Moreover, when the prediction module is located in or before the DMP system, timing faults include delays in the flow of data, loss of data, or out-of-order delivery of the data packets [102]. These delays can be injected into the communication paths from prediction to modules, examining the robustness and stability of the prediction module and performance in decision-making. Besides, Juez et al. presented a simulation-based FI approach and applied it at an essential sub-function of the DMP system: behavioral models for hazard analysis and risk assessment [106].

C. Discussion

A white-box or gray-box analysis of the SUT is performed to find a suitable time and location for FI. Therefore, the method needs to be interpretable to some extent. Besides, this method can help engineers have the destination to find the critical testing issues [107] and is not constrained by spatial structures such as road topology, so it is fast and easy to implement compared with other methods. However, finding rare hazardous events is time-consuming and uncertain, as faults might manifest only under specific conditions (e.g., a particular software state) [108]. Moreover, if fault testing lacks interaction between functions under the premise of prior knowledge, it is challenging to determine the wrong parameters or the initial state [109]. Also, a complete simulation system requires expensive time cost. Considering the complex testing space handled by the ADS, it is impossible to test all combinations of initial conditions and fault sequence [110], [111].

IV. Formal Verification

Formal verification (FV) is an integrated approach that accompanies the whole development and mature application phase of AVs. The term *formal* is reserved for the methods that use mathematics and especially logics as their foundation [112]. FV enables developers to deal with complexity using well-proven logic and mathematics tools, providing strong assurance on compliance with requirements or properties [113]. When FV is used in the DMP system, the state of the DMP-SUT and the parameters of the traffic environment (mainly roads and interaction relations) are represented and modeled in mathematical logic language, and the correctness of the DMP system is proved after calculation.

A. Model Checking

Model checking is a widely used technique for verifying whether an abstract representation of a system is correct relative to a formal specification describing the desired system behaviour. It has been used in a distributed DMP system composed of multiple modules. Zita et al. checked whether the extended finite state machines could reach any bad states in the lane change module of the decision and control system [114]. As it turned out, the Lateral State Manager system has over 3 million reachable states with the lane change request specification, but there are 10 million blocking states that violate the specification. Although model checking can provide

helpful information about system correctness and reveal subtle design errors, its main disadvantage is the state explosion problem. In contrast, formal modeling named timed automata is used to verify decision policies of communicating AVs for safety and traffic fluidity and to limit the state space explosion [115].

Authors in [116] illustrated how to decompose demanding system-level requirements into a set of component-level requirements through the canonical software architecture for verification. This technique is applied to verify the state consistency between two software modules: the path following block and the motion control. The method consists of two key elements: 1) a specification language for describing the system and its requirements and 2) an analysis to verify the correctness of the system specification relative to the requirements. In a later study, they utilized linear temporal logic (LTL) specifications to represent a large-scale of properties such as safety, stability, response, and guarantee for the automatic synthesis of the trajectory planner and continuous controller [117].

The safe decision-making capabilities of an AV do not have a nominal safety standard. Therefore, Shalev et al. focus on nominal safety and introduce an interpretable, mathematical, formal model for safety assurance called Responsibility-Sensitive Safety (RSS), which combines safety and scalability through semantics' language providing a complete methodology for AVs [118]. This approach based on model checking targets the planning modules (the original called planning module, the same concept with this paper's DMP system) within AV's typical architecture. They used many axioms and lemmas and formally validated the planning algorithm to satisfy the worst-case assumptions and mathematical generalizations. The concept of RSS was introduced by researchers and engineers from Mobileye and later expanded by researchers and engineers from Intel [119], focusing on vulnerable road users, such as cyclists and pedestrians performed, to analyze RSS-based computation of unstructured instances and their application in structured environments.

B. Reachability Analysis

The crucial challenge of the DMP-SUT is the uncertainty of interaction with other traffic participants. In order to calculate and reduce this uncertainty as much as possible, Althoff et al. proposed reachability analysis to consider every possible behavior of a mathematical model considering uncertain input and partially unknown initial state [120], [121]. Safety is guaranteed concerning the modeled uncertainties and behaviors if the AV occupancy does not overlap with other traffic participants at all times. Compared with model checking and other testing and validation methods, reachability analysis is prominent that can be used for online verification. The safety of the vehicle can be monitored and calculated during the operation of the vehicle [122]. The authors consider the uniqueness of each traffic environment and use a coarse model abstractions method to give expeditious and conservative verification results in real-time. Furthermore, the results could be refined adaptively, corresponding to the available computational constraints. Traffic rules are an aspect that cannot be ignored in the process of FV. The BMW Group [123] has validated the safety of DMP in lane-change maneuvers by fully considering the liability of traffic participants in the event of a collision based on the reachability analysis.

C. Discussion

FV is the most reliable and complete method but usually only applicable to the DMP system [20], [118]. FV is based on white-box analysis, but the internal structure of the DMP system limits it. The sub-modules or sub-functions are more and more abundant, so the module or function boundaries are fuzziness. Moreover, with ML methods such as end-to-end learning in the DMP system, it becomes difficult to apply FV to it. Besides, due to commercial sensitivity, car manufacturers have made scientific advances that do not always publicly disclose the details on their approaches or algorithms. Therefore, it is difficult to know the actual behavior decision and path planning details of other traffic participants or default them along the road during the verification process.

Besides, most research papers address traffic rules' formalization as one essential requirement, which implicates that traffic is guaranteed to be safe if all traffic participants comply with these contracts [124]. However, many uncertainties come from the weak regular region like an intersection where vehicles make the unprotected turn. It is difficult for some non-strict traffic rules to be abstracted to the real world's formal language or model, increasing formal modeling complexity. Therefore, some FV methods lack scalability to complex systems and needs expensive computation. Whereas, research [125] did an incremental step towards further formalization for scenario representation and build complex scenarios composed of dynamic objects with their subjective point of views. Finally, the problem of time synchronization between ego-vehicle and other traffic participants is a key to overcome, which increases the difficulty and scalability of the calculation. For now, it is unrealistic to verify AV safety solely based on FV, and this method must be accompanied by other validation approaches, which can support industrialization.

It should be noted that the above V&V methods are not always employed independently, and some of them usually are used in combination. For example, the ontology-based combinational method is used to generate logical scenarios first, then the data mining and search-based methods are used to regenerate concrete scenarios [126]–[129]. The falsification-based search method combines with ML to generate critical scenarios automatically are brought up in [55], [57], [130], [131]. In particular, Feng et al. combined three methods of data mining, search and RL to generate a scenario library for testing the intelligence of the DMP system. Their work considered the unbiasedness between the generated scenario and the real traffic environment based on data mining, and further trained the adversarial agents based on RL, and improves the test efficiency based on search and IS methods [132]–[136].

In summary, based on the review and analysis of each classification of methods in Section II, III, and IV, we describe the characteristics of 8 methods and compare the advantages and disadvantages between them. In addition, besides methods,

some popular simulation platforms have more applications in V&V at this stage, which are also crucial for testing the DMP system. Therefore, the simulation platforms used in the reviewed papers are also summarized in Table I. In addition to this, the industry has also done much contributory work in the V&V methods of the DMP system. While searching and reading literature, we found that universities and automotive-related companies co-authored many valuable papers. To facilitate readers to obtain information about the research methods related to enterprises quickly, we have summarized the work involved by companies with the V&V methods in Table II.

V. Strategy Analysis

This section presents an analysis of V&V approaches from two aspects. The first part is the parallel comparison among different kinds of approaches, and the second part is the analysis of the DMP system's internal module structure for matching the appropriate V&V approaches to each module.

Six criteria derived by the expert judgment are used to evaluate the different V&V methods. Three are used to evaluate the methods in scenario generation, and the others are used to evaluate the above methods' application performance. Then, based on the evaluation results, we compare the above V&V methods in parallel. It should be noted that the main object of our evaluation and comparison is this category of methods, not an individual paper.

A. Evaluation Criteria in Scenario Generation

1) Coverage Testing scenarios with wide coverage refer to a large variety and quantity of entities with different attributes, fulfilling the fundamental requirements of different tested functions comprehensively. For the scenario where the tested object is a DMP system, the main test entities are dynamic traffic participants and their interactions. Therefore, the coverage is mainly reflected in the wide variety and a large number of traffic participants.

2) Evolvability The methods with exploitation and even exploration capabilities can generate some edge scenarios outside the existing scope, which is used to discover more potential or unforeseen problems of DMP-SUT. This kind of scenario must be a necessary experience before the industrialization of high-level AVs in the future.

3) Fidelity Testing scenarios with accurate fidelity have a high degree of similarity to real-world traffic environments. This fidelity specifically refers to the realism of the causality of the traffic participant interactions in the testing scenario, and the imitated interactions should be as close to human cognition as possible and can be understood.

B. Evaluation Criteria in Application Metrics

1) Efficiency High testing efficiency means that the V&V methods are used to check whether the tested requirements are satisfied with expectation within a certain period as faster as possible and whether the found critical functional problems of the DMP-SUT are accurate. We do not consider the computer power or hardware performance but purely talk about the V&V methods.

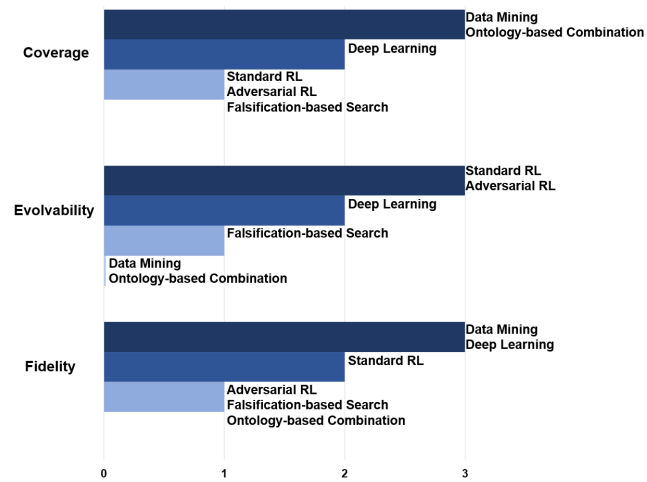


Fig. 7. Evaluation and comparison of the approaches in scenario generation.

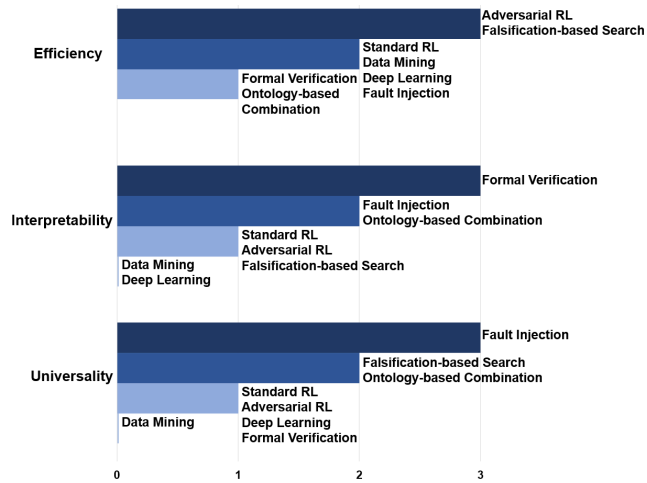


Fig. 8. Evaluation and comparison of the approaches in application performance.

2) Interpretability Interpretability means that the simulation models of V&V methods can be analyzed using a white-box or grey-box manner. Therefore, the testing process and results can be interpreted from configurations in the internal model combined with the tested system's characteristics.

3) Universality Except for dynamic entities, when V&V methods are applied in the DMP system, the static entities are also of great importance, such as road topology. People in different countries have diverse driving styles, and their driving behaviors are also unlike under different road structures or topologies. Suppose a specific test method requires mass data to train agents or needs to be applied in a particular region or a fixed traffic geometry. In that case, it is difficult to deploy this method when it is transferred to other countries or road structures, and it requires much effort to modify and adapt.

C. Methods Evaluation

Combined with the above evaluation criteria and the discussion contents presented in previous sections, the following results (Fig. 7 & 8) are delivered.

TABLE I
SUMMARY OF THE ADVANTAGES/DISADVANTAGES OF V&V METHODS AND THE APPLICATION OF SIMULATION PLATFORMS IN REFERENCES

V&V Methods	Descriptions	Advantages	Disadvantages	Simulation Platforms	References
① **Ontology-based Combination**	- Ontology is a formal and explicit conceptualization of entities, interfaces, behaviors, and relationships. - Combines scenario entities and their attributes based on ontology theory.	- High coverage. - Provides high interpretable scenarios.	- Bias toward theoretical analysis and heavy reliance on expert experience. - Inefficient test cases may be generated that are physically implausible or meaningless. - It is difficult to explore edge cases.	VTD	[28]
				Java-based encoding	[30]
				CyberCars-2	[31]
				Carla	[34]
				Python-based encoding	[36]
				\	[32], [33], [35] [37], [39], [40]
② **Data Mining**	- Extracts the critical data from naturalistic driving data sets to generate challenging scenarios.	- Generates testing scenarios with high fidelity and traceability. - Extracts safety-critical data to generate more efficient testing scenarios than road tests.	- Safety-critical data are rare, and the collection process is consuming. - As the data is fixed, the generated scenarios have no real-time interactivity and evolution.	VTD	[41], [137]
				MATLAB	[45], [46]
				Carla	[48]
				\	[42]–[44]
③ **Falsification-based Search**	- Aims to search counterexamples violating the functional requirements under test.	- Generates highly efficient testing scenarios. - Improves the availability of developing parametric scenarios - Reduces the number of useless scenarios.	- Only proves the DMP-SUT executes the wrong decision, not the right one. - It is difficult to analyze search space and determine the initial source of parameter space. - Treating the DMP-SUT as a black box is harder to reveal the exact problem points for developers.	VTD	[50]
				Carla	[53], [138]
				PreScan and Simulink	[54]
				BeamNG.drive	[60]
				\	[55]–[59]
④ **DL**	- Discovers complex structures well in high-dimensional naturalistic driving data. - Learns the concept of representing data correctly for generating testing scenarios.	- Good generalization to the interaction behaviors. - More fidelity scenarios can be derived from the original scenarios. - NPCs can interact with the SUT in real-time.	- Lacks interpretability of the NPCs' interaction behaviors. - High costs of collecting safety-critical data.	Carla	[68]
				Python-based encoding	[70]
				Python with TensorFlow	[71]
				Python with TensorFlow, and Road Traffic Simulator	[73]
				Sim-ATAV	[75]
				\	[72], [74], [76]
⑤ **Standard RL**	- Trains NPCs (agents) via interaction with the traffic environment driven by a reward function. - The training process does not include the DMP-SUT.	- Generates scalable-to-multiple-vehicles traffic models in the testing scenario. - Generates evolutionary interactions to explore more unforeseen scenarios.	- Not well-targeted and may not reveal the weaknesses of DMP-SUT precisely. - Tabular RL strategies are hard to calculate in high-dimensional state space and complex environments. - Less stable due to the sensitivity to hyper-parameter settings and architecture choices.	Java-based encoding	[80]–[82]
⑥ **Adversarial RL**	- Trains adversarial NPCs (agents) to obtain the optimal strategy to expose the weaknesses of the DMP-SUT, driven by a reward function. - The training process includes the DMP-SUT.	- Generates highly efficient testing scenarios. - Generates evolutionary interactions to explore more safety-critical testing scenarios and edge cases.	- May lead to over-fitting for one type of DMP-SUT. - The generated testing scenarios are not universal for different DMP-SUTs.	AirSIM	[87]
				TORCS and TensorFlow	[88]
				Carla	[89]
				LGSVL	[91]
				CarMaker	[92]
				\	[83], [85], [90], [91]

V&V Methods	Descriptions	Advantages	Disadvantages	Simulation Platforms	References
⑦ FI	- Tests the resilience, robustness, and error handling of the cyber-physical system in the event of a failure.	- Treating DMP-SUT as a white or grey box to inject faults is more interpretable. - Helps engineers have the destination to find the critical testing issues.	- Finding rare hazardous events is time-consuming and uncertain. - It is challenging to determine the wrong parameters and the initial state.	Dynacar by Tecnalia	[99]
				TORCS	[100]
				Nvidia DriveWorks	[101]
				Carla	[102]
				NVIDIA DriveSim, and Carla	[103]
				Dynacar	[106], [107]
⑧ FV	- Uses mathematical and logic-based tools to verify the system, providing strong assurance of compliance with design requirements.	- Reliable and complete. - White-box analysis with high interpretability.	- Lacks scalability to complex systems and needs expensive computation. - Due to commercial sensitivity, it is difficult for FV to analyze the algorithmic details of DMP-SUT.	Simulink	[113], [114]
				CommonRoad	[120]
				Simulink	[121]
				CommonRoad	[122]
				SUMO	[125]
				\	[115]–[118], [124]
Combination of Different Methods	①+②+③			Sim-ATAV	[127]
	①+③			MATLAB, PreScan and Carsim	[128]
				Prescan	[129]
	③+④			MATLAB PeVi	[130]
				Carla	[131]
	①+③+⑤			Augmented Reality (AR) platform designed for Mcity	[132]–[134]
				MATLAB toolbox	[135]
				Carla	[136]
	①+②			\	[126]

In the early testing stage, the fundamental scenario needs to be as broad coverage as possible to meet its basic requirements. The ontology-based combination method combines the types and attributes of entities to generate scenarios. Data mining methods can select data to generate scenarios from databases containing different types of entities and attributes. Therefore, both are more befitting for fundamental testing. By comparison, because the standard RL, adversarial RL, and falsification-based search methods need to train or calculate mass data from different types of entities, which is likely to cause dimensional explosion and complexity in computation, the scenarios generated by these methods have low coverage, as shown in Fig. 7. By contrast, both RL methods are better suited for high-level AVs' testing. They can generate advanced scenarios with evolutionary potential and explore more edge scenarios to expose more problems. Usually, the authenticity of generated scenario from data extraction is closer to real events than knowledge.

As depicted in Fig. 8, RL and data-derived methods (data mining and DL) usually have higher test efficiency, but their interpretability is relatively weak. FV has the best interpretability, whereas it needs to be applied based on many constraints and rules, so it is not widely applied. As for universality, sampled data is strongly dependent on the topology in the acquisition location to generate scenarios, and if testers want to change the scenario topology, resampling is often needed. The topologies in RL and DL's training process are predefined, so the universality is weak and costly. Although FI with high universality, it is usually suitable for special test requirements.

D. Applicable V&V Approaches for DMP System Modules

The DMP system is coupled with several functional modules, and the internal logic among them is complex. These modules are not uniformly defined, and their partitioning is rather blurred with diverse variations of schemes existing

TABLE II
SUMMARY OF METHODS USED BY DIFFERENT COMPANIES IN THE V&V OF THE DMP SYSTEM

Companies	Cooperation Types	V&V Methods	References
AUDI	○	DL Falsification-based Search Data Mining	[139] [140] [137]
AVL	○ ●	Ontology-based FI Data Mining	[28], [37] [109] [41]
Baidu	●	DL	[141]
BeamNG GmbH	○	Falsification-based Search	[60]
Boeing Phantom Works	○	Ontology-based	[32]
BMW Group	○	FV Data Mining	[123] [142], [143]
Daimler AG	○	Ontology-based & Data Mining	[126]
Ford Motor Company & Toyota Research Institute	○	Standard RL	[84]
fka (Forschungsgesellschaft Kraftfahrwesen GmbH)	○	Data Mining DL Ontology-based	[42], [43] [74] [40]
Jianghuai Automobile Company	○	Adversarial RL	[89]
IBM Research AI	●	Adversarial RL	[87]
IEE S.A.	○	Falsification-based Search Falsification-based Search & DL	[54] [130]
Intel	●	FV	[119]
IPG Automotive GmbH	○	Ontology-based	[33]
Leidos	○	FI	[110]
Mercedes-Benz AG	○	Ontology-based	[38]
Mobileye	●	FV	[118]
Motional (Hyundai Corporation & APTIV)	●	Falsification-based Search	[138]
NXP & TNO	○	FI	[100]
NVIDIA	○	FI	[101], [103]
Robert Bosch GmbH	○	Data Mining	[47]
Tecnalia	●	FI	[99], [106], [107]
TNO & TomTom Symphony Co. & 2getthere	○	Ontology-based	[36]
Toyota Group	○ ●	DL Falsification-based Search Ontology-based & Data Mining & Falsification-based Search FV	[72], [75] [52] [127] [124]
Valeo	●	Standard RL	[144]
Veoneer Research & Zenuity	○	DL	[73]
Volvo Car Corporation	○	FV	[114]

○ represents collaborative research with universities.
● represents independent research.

in various works of literature [145]. Based on the related literature review, we find several alternative structures of modules and synthesize them so that a common understanding of the DMP system can be built [14]. After analyzing the relationships and workflow among these modules, we deduce the functional requirement of each module. Aiming at every functional requirement, we select the suitable approaches to test and verify the corresponding module.

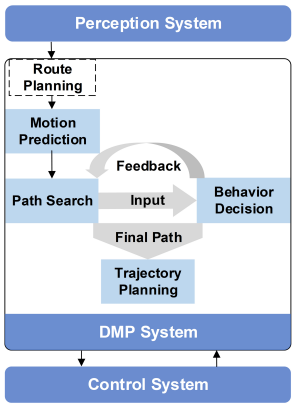


Fig. 9. A flow chart of the hierarchy of DMP system (adapted from [14]).

1) DMP System Workflow In terms of AV's internal workflow, a typical structure of a DMP system is hierarchically decomposed into four main modules (light blue sign), as shown in Fig. 9. The environment information obtained from the perception system is transformed from physical space into a digital representation, then the route will be planned and transmitted into the motion prediction. The prediction module needs to consider messages over the future motion of other agents (the prediction module is regarded as part of the DMP system as [146], [147]). Depending on the prediction results and how they influence the path search, different categories of behavior decisions and planning methods can be decided [148], [149]. The primary purpose of path search is to provide the vehicle with collision-free and safe paths to its destination, while the constraints of the vehicle's motion model, way-points that the vehicle should follow, and the traffic environment, including traffic rules, static and dynamic obstacles are taken into account [150], [151]. A searched path strongly correlates with the behavior decision, which acts as input to the decision module for the "best" behavior in terms of safety or efficiency. Then the behavior decision module will feedback the feasibility of the path search module. Critical path search and behavior decision are both vital to autonomy and are implemented through trajectory planning algorithms [152]. The final path may change based on the best behavior, as shown with a feedback loop between these two modules in Fig. 9 (grey sign). Once the path is finalized, the final trajectory planning is generated. According to the given constraints, the last level concentrates on finding the best trajectory to follow through the optimization of a geometric line. Finally, the control system reactively sends feedbacks to the trajectory planning execution.

2) Functional Requirements and Applicable Approaches to the Modules The DMP system is a sophisticated system with several modules, but part of the research reviewed in this paper are not module-target and testing the system as a whole. After describing the four modules of the DMP system, we propose the functional requirement for each module. Aiming at the functional requirements, we think the reviewed V&V approaches above could focus on different modules. Therefore, the appropriate approaches are summarized in Fig. 10.

a) Motion Prediction

The prediction module must have the ability to predict the path of motion of traffic participants accurately [153]. Some AVs' prediction modules have advanced prediction capabilities obtained by training traffic participants through ML approaches [154]–[157]. The complexity of NPCs (challengers) in testing scenarios can be represented as the various path of motion, which increases the difficulty of prediction [158]. If the prediction module is unfamiliar with challengers' motions, which means the prediction module had not learned from these kinds of trajectories when it was in the training process. Among the reviewed scenario generation methods, the prediction module can be tested better in the scenario generated by ML. According to the skillful design of the reward function, the challengers can learn variable trajectory policies by exploring different traffic environment states. As the environment states increase, it is more likely for the challengers to learn the various path of motion to test prediction performance.

b) Path Search

The essential requirement for path searching is to avoid collisions. During the path search calculation, the dynamic and static obstacles are all considered in the testing scenario. For the static obstacles, the type, shape, position, and sequence in space; on the other hand, for the dynamic obstacles, the threshold range of its motion state, all above can challenge the path search module. Testing scenarios generated based on search for falsification reduce AV's space reachability, and the ontology-based combination method increases the variety of static obstacle types and distribution. Therefore, both ways are suitable for the testing of the path search module.

c) Behavior Decision

For the behavior decision module, safe interaction with the traffic participants is the primary principle. The behavior diversities and state uncertainties of challengers may lead to accidents during the interaction with the tested vehicle in testing scenarios [159]. Moreover, it is necessary to test the behavior decision module with real-time interaction, which increases the computing capability for the testing process. Among the reviewed scenario generation methods, if different strategies are used to train the challengers by ML, the behavior diversities and state uncertainties are increased. Besides, the extracted scenarios from data mining in naturalistic driving condense the safe-critical interactions among road users, especially some corner and dangerous behaviors. So, it also can be used to test the safety interaction ability of the behavior decision module.

d) Trajectory Planning

For the trajectory planning module, calculating the optimal trajectory is the priority target [160]. The trajectory's optimality can be reflected in some indicators, such as economy, executability (the deviation degree of the actuator tracking the planned trajectory), path smoothness, passenger comfort [161], [162]. The scenarios for examining the above indicators include dynamic and static irregular obstacles, irregular road topologies, rough and bumpy road properties [163], [164]. It also provides traffic conditions (such as traffic jams) that can break the test module's operating continuity, which can pose challenges to both economy and comfort. The ontological

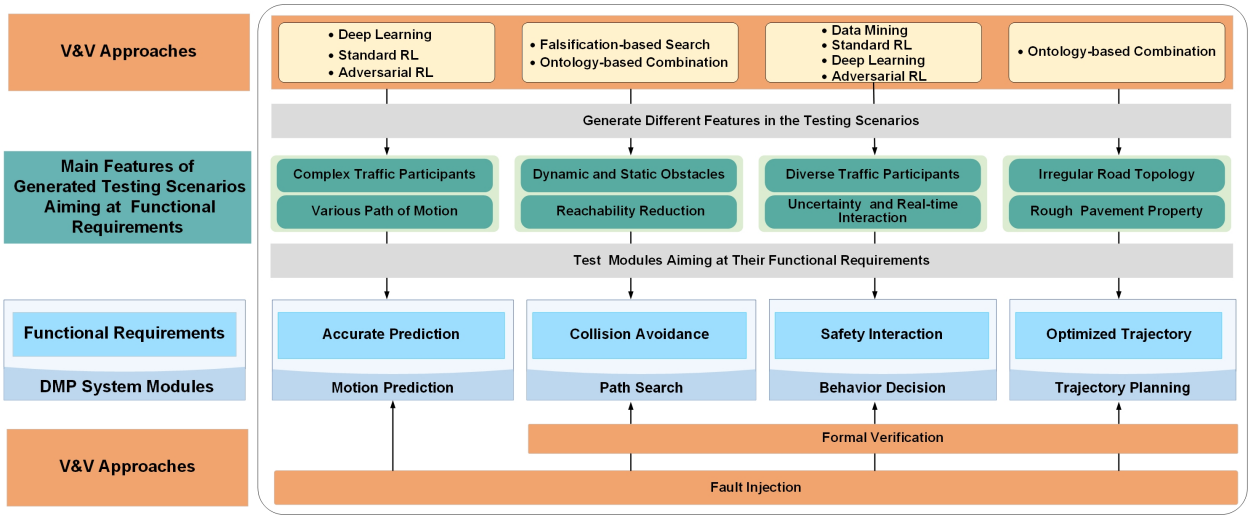


Fig. 10. Modules of DMP system adapted to the V&V methods.

framework contains road topology structure entities and road properties, combined with other entities and properties to generate various scenarios to test this module.

In general, FI can test every module based on white-box analysis or test the DMP system end-to-end as a black-box. By contrast, FV can verify vehicle trajectories and behaviors that have occurred or are occurring. Hence, in addition to the prediction module, the validation of the other three modules is also appropriate. It should be noted that all the V&V methods summarized in this paper are applicable to DMP systems in different frameworks (consisting of several modules coupled or end-to-end systems based on black boxes). However, each method has its own characteristics and pertinence, and readers can select the appropriate method based on different test requirements or evaluation criteria.

VI. Challenges and Future Trends

Based on the comparison results in Section V, some challenges among test methods in V&V for the DMP system are presented. Also, a summary of future directions is discussed.

A. Challenges

1) Methods are Difficult to Cover Multiple Metrics Currently, there is no one-size-fits-all V&V solution, and each strategy has its own pros and cons. The method based on theory and analysis is more interpretable but short of fidelity. The data-driven based strategies, on the contrary, lack the interpretable ability. Therefore, the future mainstream methods should better combine data and knowledge to generate scenarios for the effectiveness of V&V [165], such as DRL and inverse RL, which are increasingly significant in the current research [166]–[169]. Also, the GAN-based algorithm can be explored more deeply in generating multimodal human driver models. There is a greater potential for generating test scenarios with both complexity and realism, which may shed some light on DMP-oriented V&V [170]–[173].

As depicted in Fig. 7, the scenario generation methods cannot be satisfied to both coverage and evolvability. The future development of scenario generation needs to find a balance between these metrics. This balance point should be oriented to test requirements and satisfy the transition from elementary to advanced testing.

A highly efficient method may generate scenarios that are difficult to guarantee in terms of fidelity. For example, pure adversarial RL may train the NPCs to behave unnaturally and lead to deliberate failure. Suppose NPCs intend to collide with the tested vehicles purposely, in that case, the DMP-SUT will make meaningless failures as these cases are either physically non-applicable or rarely occur in naturalistic driving [87]. Such failure scenarios might seem like a highly efficient test, but it is not true at all and unnecessary for testing the DMP-SUT. Therefore, these methods should combine some reasonable and hard constraints to enhance fidelity while ensuring test efficiency.

2) DMP System is Difficult to Decouple Completely Also, it is not easy to implement the targeted V&V to modules. The trend of the black-box DMP system is emerging gradually due to it is difficult to decouple between modules [174], [175]. However, when a negative issue occurs, the end-to-end development or testing program may appear unconvincing. Therefore, the module development and testing of the DMP system is promising. In addition, the problem of coupling and interference between modules is also the focus of research.

B. Gaps to be Filled in the Future

At the same time, there are still gaps in the field of V&V for the DMP system, which are in urgent need of future investigation by both academic research and industrial efforts.

1) Gaps in Methodology There is currently no method that can well consider all criteria proposed in Section V. Before the large-scale deployment of AVs, a method that covers multiple criteria or a combination of multiple methods can cover multiple criteria is needed in the future V&V for the DMP system.

Section V.D considered that the DMP system are composed of four modules. Although there have been undesired

interaction tests between different assisted driving functions, such as AEB and Adaptive Cruise Control (ACC) functions [54], few studies on interference tests exist among several modules within the decision system, or even almost a blank state. In [114], the author found a failure was terrible interactions between the reactive obstacle avoidance module and the reacting path planner. This failure only happened in a particular situation, and it was doubtful for the bug to be found. Therefore, adverse interactions testing between modules or subsystems is crucial.

2) Gaps in Database The future V&V of AVs could rely much more on the data-driven approaches. NHTSA launched the "Autonomous Driving Transparency and Safety Test Participation" (AV TEST) initiative in June 2020, which aims to provide an online platform for the public to share the data on the on-road testing and safety performance of ADS. Data from other countries besides the United States is also important because each country has its characteristics of the traffic environment, which are crucial for scenario generation. Therefore, it is necessary to integrate data types and forms from all parts of the world as much as possible to build a universal standard database.

Interaction data is the focus of testing scenario generation for decision system testing. However, there is no benchmark interaction database for DMP system tests, so standardized comparisons between the systems under test cannot be guaranteed.

It is worth emphasizing that NHTSA collects data about the testing of ADS equipped vehicles. According to the experimental results of [176], AVs' actions have effects on what human drivers do, which means that the interaction data between humans is different from the interaction data between humans and AVs. Therefore, in the following research, the interaction data between human drivers and AVs, which can be collected from the simulation (e.g., driver-in-loop), should be taken into account in the test scenarios generation.

3) Gaps in Standardization To facilitate communication between industries, achieve fair horizontal comparisons between different research institutions or enterprises, and jointly promote development of DMP system, the standards research of V&V is vital. Relevant standards (such as ISO 21448 and ISO 34501) have been issued for the V&V of the AV level. However, there are currently no testing and safety standards specifically for the DMP system. In the future, standards research can be developed from the following aspects, such as information input standards from perception to decision-making, semantic-level decision-making behavior and strategy standards, functional requirements standards for various modules, safety standards for DMP system and control-oriented information output standards. All the above standards are used to ensure different DMP systems to be bench-marked on a fair manner.

VII. Conclusion

The DMP system plays a vital role in AVs' autonomy, and the V&V for this system is essential. However, since the system is located in the middle of the information transmission chain and its own internal is complex, the V&V for this system is full of challenges. The relevant and representative articles about the V&V for the DMP system in recent years are reviewed in this paper. The testing methods are divided into scenario-based for safety and faults injection for robustness from the evaluated performance perspective. As a correctness-proof method, formal verification is also summarized for completeness. Subsequently, we compare the advantages and disadvantages of these approaches. Based on the analysis of the various modules' functional requirements in the DMP system, we propose the more suitable V&V methods matching the modules. Finally, the challenges and future directions in this field are indicated, and these problems should be solved before the vastly landing of AVs on the market.

References

[1] F.-Y. Wang, N. Zheng, L. Li, J. Xin, X. Wang, L. Xu, B. Tian, G. Wu, Z. Zhang, C. Wang, and L. Chen, "China's 12-year quest of autonomous vehicular intelligence: The intelligent vehicles future challenge program," *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 2, pp. 6–19, Summer 2021.

[2] L. Li, Y.-L. Lin, N.-N. Zheng, F.-Y. Wang, Y. Liu, D. Cao, K. Wang, and W.-L. Huang, "Artificial intelligence test: A case study of intelligent vehicles," *Artificial Intelligence Review*, vol. 50, no. 3, p. 441–465, Oct. 2018.

[3] C. Zhao, L. Li, X. Pei, Z. Li, F.-Y. Wang, and X. Wu, "A comparative study of state-of-the-art driving strategies for autonomous vehicles," *Accident Analysis & Prevention*, vol. 150, p. 105937, Feb. 2021.

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

[5] P. Steven, "Autopilot & first responder scenes," National Highway Traffic Safety Administration, 2021, Accessed: Jun. 7, 2022. [Online]. Available: https://static.nhtsa.gov/odi/inv/2021/INOA-PE21020-1893.PDF

[6] J. V. Steenburg, "Incident reporting for automated driving systems and level 2 advanced driver assistance systems," National Highway Traffic Safety Administration, 2021, Accessed: Jun. 7, 2022. [Online]. Available: https://www.regulations.gov/document/NHTSA-2021-0070-001

[7] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 4, pp. 6–22, Winter 2014.

[8] H. Wang, A. Khajepour, D. Cao, and T. Liu, "Ethical decision making in autonomous vehicles: Challenges and research progress," *IEEE Intelligent Transportation Systems Magazine*, vol. 14, no. 1, pp. 6–17, Jan.-Feb. 2022.

[9] L. Li, N. Zheng, and F.-Y. Wang, "A theoretical foundation of intelligence testing and its application for intelligent vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 10, pp. 6297–6306, Oct. 2021.

[10] J. E. Stellet, M. R. Zofka, J. Schumacher, T. Schamm, F. Niewels, and J. M. Zöllner, "Testing of advanced driver assistance towards automated driving: A survey and taxonomy on existing approaches and open questions," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 1455–1462.

[11] L. Li, X. Wang, K. Wang, Y. Lin, J. Xin, L. Chen, L. Xu, B. Tian, Y. Ai, J. Wang, D. Cao, Y. Liu, C. Wang, N. Zheng, and F.-Y. Wang, "Parallel testing of vehicle intelligence via virtual-real interaction," *Science Robotics*, vol. 4, no. 28, p. eaaw4106, Mar. 2019.

[12] A. Dasso, *Verification, validation and testing in software engineering*. IGI Global, 2006.

[13] S. Chen, Y. Chen, S. Zhang, and N. Zheng, "A novel integrated simulation and testing platform for self-driving cars with hardware in the loop," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 3, pp. 425–436, Sep. 2019.

[14] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, Nov. 2015.

[15] H. Wang, Y. Huang, A. Khajepour, Y. Zhang, Y. Rasekhipour, and D. Cao, "Crash mitigation in motion planning for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3313–3323, Sept. 2019.

[16] P. Hang, C. Lv, Y. Xing, C. Huang, and Z. Hu, "Human-like decision making for autonomous driving: A noncooperative game theoretic approach," *arXiv preprint arXiv:2005.11064*, 2020.

[17] W. Wang, X. Na, D. Cao, J. Gong, J. Xi, Y. Xing, and F.-Y. Wang, "Decision-making in driver-automation shared control: A review and perspectives," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 5, pp. 1289–1307, 2020.

[18] D. Zhao, H. Lam, H. Peng, S. Bao, D. J. LeBlanc, K. Nobukawa, and C. S. Pan, "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 595–607, Mar. 2017.

[19] 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 Transactions on Intelligent Transportation Systems*, pp. 1–16, Dec. 2021.

[20] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 8, pp. 87 456–87 477, 2020.

[21] Y. Xing, C. Lv, L. Chen, H. Wang, H. Wang, D. Cao, E. Velenis, and F.-Y. Wang, "Advances in vision-based lane detection: Algorithms, integration, assessment, and perspectives on acp-based parallel vision," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 3, pp. 645–661, 2018.

[22] P. Koopman, U. Ferrell, F. Fratrik, and M. Wagner, "A safety standard approach for fully autonomous vehicles," *Computer Safety, Reliability, and Security*, pp. 326–332, 2019.

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

[24] F.-Y. Wang, "Metavehicles in the metaverse: Moving to a new phase for intelligent vehicles and smart mobility," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 1, pp. 1–5, 2022.

[25] D. Cao, X. Wang, L. Li, C. Lv, X. Na, Y. Xing, X. Li, Y. Li, Y. Chen, and F.-Y. Wang, "Future directions of intelligent vehicles: Potentials, possibilities, and perspectives," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 1, pp. 7–10, 2022.

[26] D. Nalic, T. Mihalj, M. Baeumler, M. Lehmann, A. Eichberger, and S. Bernsteiner, "Scenario based testing of automated driving systems: A literature survey," in *FISITA Web Congress*, Oct. 2020.

[27] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[28] Y. Li, J. Tao, and F. Wotawa, "Ontology-based test generation for automated and autonomous driving functions," *Information and Software Technology*, vol. 117, p. 106200, Jan. 2020.

[29] C. Sun, C. Wang, Z. Deng, and D. Cao, "Dimensionless model-based system tracking via augmented kalman filter for multiscale unmanned ground vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 2, pp. 600–610, Apr. 2021.

[30] G. Bagschik, T. Menzel, and M. Maurer, "Ontology based scene creation for the development of automated vehicles," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1813–1820.

[31] R. Regele, "Using ontology-based traffic models for more efficient decision making of autonomous vehicles," in *Fourth International Conference on Autonomic and Autonomous Systems (ICAS'08)*, 2008, pp. 94–99.

[32] R. Provine, C. Schlenoff, S. Balakirsky, S. Smith, and M. Uschold, "Ontology-based methods for enhancing autonomous vehicle path planning," *Robotics and Autonomous Systems*, vol. 49, no. 1, pp. 123–133, Nov. 2004.

[33] S. Geyer, M. Baltzer, B. Franz, S. Hakuli, M. Kauer, M. Kienle, S. Meier, T. Weißgerber, K. Bengler, R. Bruder, F. Flemisch, and H. Winner, "Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance," *IET Intelligent Transport Systems*, vol. 8, no. 3, pp. 183–189, May. 2014.

[34] C. Sun, Z. Deng, W. Chu, S. Li, and D. Cao, "Acclimatizing the operational design domain for autonomous driving systems," *IEEE Intelligent Transportation Systems Magazine*, vol. 14, no. 2, pp. 10–24, Mar.-Apr. 2022.

[35] C. Wei and K. Leïla, "An ontology-based approach to generate the advanced driver assistance use cases of highway traffic," in *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, 2018, pp. 75–83.

[36] E. De Gelder, J.-P. Paardekooper, A. Khabbaz Saberi, H. Elrofai, O. o. d. Camp, S. Kraines, J. Ploeg, and B. De Schutter, "Towards an ontology for scenario definition for the assessment of automated vehicles: An object-oriented framework," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2022.

[37] F. Klueck, Y. Li, M. Nica, J. Tao, and F. Wotawa, "Using ontologies for test suites generation for automated and autonomous driving functions," in *2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, 2018, pp. 118–123.

[38] C. Reichenbächer, M. Rasch, Z. Kayatas, F. Wirthmüller, J. Hipp, T. Dang, and O. Bringmann, "Identifying scenarios in field data to enable validation of highly automated driving systems," *arXiv preprint arXiv:2203.03515*, 2022.

[39] H. Naseer and A. Rauf, "Validation of ontology based test case generation for graphical user interface," in *2012 15th International Multitopic Conference (INMIC)*, 2012, pp. 465–469.

[40] H. Weber, J. Bock, J. Klimke, C. Roesener, J. Hiller, R. Krajewski, A. Zlocki, and L. Eckstein, "A framework for definition of logical scenarios for safety assurance of automated driving," *Traffic Injury Prevention*, vol. 20, no. sup1, pp. S65–S70, Jun. 2019.

[41] A. Erdogan, E. Kaplan, A. Leitner, and M. Nager, "Parametrized end-to-end scenario generation architecture for autonomous vehicles," in *2018 6th International Conference on Control Engineering & Information Technology (CEIT)*, 2018, pp. 1–6.

[42] A. Pütz, A. Zlocki, J. Küfen, J. Bock, and L. Eckstein, "Database approach for the sign-off process of highly automated vehicles," in *Proceedings of the 25th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2017.

[43] A. Pütz, A. Zlocki, J. Bock, and L. Eckstein, "System validation of highly automated vehicles with a database of relevant traffic scenarios," in *12th ITS European Congress*, 2017.

[44] C. Sun, J. M. U. Vianney, Y. Li, L. Chen, L. Li, F.-Y. Wang, A. Khajepour, and D. Cao, "Proximity based automatic data annotation for autonomous driving," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 395–404, Mar. 2020.

[45] D. Zhao, "Accelerated evaluation of automated vehicles." Ph.D. dissertation, University of Michigan, 2016.

[46] D. Zhao, X. Huang, H. Peng, H. Lam, and D. J. LeBlanc, "Accelerated evaluation of automated vehicles in car-following maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 733–744, Mar. 2018.

[47] S. Jesenski, N. Tiemann, W. Branz, and J. M. Zöllner, "Creation of critical traffic scenes for usage with importance sampling," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 3162–3169.

[48] X. Yan, S. Feng, H. Sun, and H. Liu, "Distributionally consistent simulation of naturalistic driving environment for autonomous vehicle testing," *arXiv preprint arXiv:2101.02828*, Jan. 2021.

[49] Y. Kang, H. Yin, and C. Berger, "Test your self-driving algorithm: An overview of publicly available driving datasets and virtual testing environments," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 171–185, Jun. 2019.

[50] S. Wu, H. Wang, W. Yu, K. Yang, D. Cao, and F. Wang, "A new sotif scenario hierarchy and its critical test case generation based on potential risk assessment," in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, 2021, pp. 399–409.

[51] A. Saeed, S. H. Ab Hamid, and M. B. Mustafa, "The experimental applications of search-based techniques for model-based testing: Taxonomy and systematic literature review," *Applied Soft Computing*, vol. 49, pp. 1094–1117, Dec. 2016.

[52] J. Kapinski, J. V. Deshmukh, X. Jin, H. Ito, and K. Butts, "Simulation-based approaches for verification of embedded control systems: An overview of traditional and advanced modeling, testing, and verification techniques," *IEEE Control Systems Magazine*, vol. 36, no. 6, pp. 45–64, Dec. 2016.

[53] Y. Abeysirigoonawardena, F. Shkurti, and G. Dudek, "Generating adversarial driving scenarios in high-fidelity simulators," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8271–8277.

[54] R. B. Abdessalem, A. Panichella, S. Nejati, L. C. Briand, and T. Stifter, "Testing autonomous cars for feature interaction failures using many-objective search," in *2018 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 2018, pp. 143–154.

[55] G. E. Mullins, A. G. Dress, P. G. Stankiewicz, J. D. Appler, and S. K. Gupta, "Accelerated testing and evaluation of autonomous vehicles via imitation learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 5636–5642.

[56] 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," *Journal of Systems and Software*, vol. 137, pp. 197–215, Mar. 2018.

[57] 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 *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1443–1450.

[58] M. Klischat and M. Althoff, "Generating critical test scenarios for automated vehicles with evolutionary algorithms," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 2352–2358.

[59] M. Althoff and S. Lutz, "Automatic generation of safety-critical test scenarios for collision avoidance of road vehicles," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1326–1333.

[60] A. Gambi, M. Mueller, and G. Fraser, "Automatically testing self-driving cars with search-based procedural content generation," in *Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis*. Association for Computing Machinery, 2019, p. 318–328.

[61] L. Li, Y. L. Lin, D. P. Cao, N. N. Zheng, and F. Y. Wang, "Parallel learning-a new framework for machine learning," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 43, pp. 1–8, Jan. 2017.

[62] A. O. Ly and M. Akhloufi, "Learning to drive by imitation: An overview of deep behavior cloning methods," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 195–209, 2021.

[63] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, Jun. 2022.

[64] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: A survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 315–329, 2020.

[65] N. K. Chauhan and K. Singh, "A review on conventional machine learning vs deep learning," in *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*. IEEE, 2018, pp. 347–352.

[66] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May. 2015.

[67] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, 2016.

[68] C. Sun, L. Su, S. Gu, J. M. Uwabeza Vianney, K. Qin, and D. Cao, "Cross validation for cnn based affordance learning and control for autonomous driving," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 1519–1524.

[69] M. Al-Sharman, D. Murdoch, D. Cao, C. Lv, Y. Zweiri, D. Rayside, and W. Melek, "A sensorless state estimation for a safety-oriented cyber-physical system in urban driving: Deep learning approach," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 1, pp. 169–178, 2021.

[70] L. Liu, S. Feng, Y. Feng, X. Zhu, and H. X. Liu, "Learning-based stochastic driving model for autonomous vehicle testing," *Transportation Research Record*, vol. 2676, no. 1, pp. 54–64, Aug. 2022.

[71] W. Ding, W. Wang, and D. Zhao, "A multi-vehicle trajectories generator to simulate vehicle-to-vehicle encountering scenarios," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4255–4261.

[72] W. Wang, A. Ramesh, J. Zhu, J. Li, and D. Zhao, "Clustering of driving encounter scenarios using connected vehicle trajectories," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 3, pp. 485–496, Sep. 2020.

[73] I. R. Jenkins, L. O. Gee, A. Knauss, H. Yin, and J. Schroeder, "Accident scenario generation with recurrent neural networks," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 3340–3345.

[74] R. Krajewski, T. Moers, D. Nerger, and L. Eckstein, "Data-driven maneuver modeling using generative adversarial networks and variational autoencoders for safety validation of highly automated vehicles," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2383–2390.

[75] C. E. Tuncali, G. Fainekos, H. Ito, and J. Kapinski, "Simulation-based adversarial test generation for autonomous vehicles with machine learning components," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1555–1562.

[76] W. Wang and D. Zhao, "Extracting traffic primitives directly from naturalistically logged data for self-driving applications," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1223–1229, Apr. 2018.

[77] W. Wang, C. Liu, and D. Zhao, "How much data are enough? a statistical approach with case study on longitudinal driving behavior," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 2, pp. 85–98, Jun. 2017.

[78] M. Littman, "Reinforcement learning improves behaviour from evaluative feedback," *Nature*, vol. 521, pp. 445–51, May. 2015.

[79] A. Haydari and Y. Yilmaz, "Deep reinforcement learning for intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 11–32, Jan. 2022.

[80] N. Li, D. W. Oyler, M. Zhang, Y. Yildiz, I. Kolmanovsky, and A. R. Girard, "Game theoretic modeling of driver and vehicle interactions for verification and validation of autonomous vehicle control systems," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 5, pp. 1782–1797, Sept. 2018.

[81] D. W. Oyler, Y. Yildiz, A. R. Girard, N. I. Li, and I. V. Kolmanovsky, "A game theoretical model of traffic with multiple interacting drivers for use in autonomous vehicle development," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 1705–1710.

[82] N. Li, D. Oyler, M. Zhang, Y. Yildiz, A. Girard, and I. Kolmanovsky, "Hierarchical reasoning game theory based approach for evaluation and testing of autonomous vehicle control systems," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 727–733.

[83] B. M. Albaba and Y. Yildiz, "Driver modeling through deep reinforcement learning and behavioral game theory," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 2, pp. 885–892, Mar. 2022.

[84] X. Wang, H. Peng, S. Zhang, and K.-H. Lee, "An interaction-aware evaluation method for highly automated vehicles," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 394–401.

[85] S. Ghiassian, B. Rafiee, Y. L. Lo, and A. White, "Improving performance in reinforcement learning by breaking generalization in neural networks," *arXiv preprint arXiv:2003.07417*, 2020.

[86] Y. Li, "Deep reinforcement learning: An overview," *arXiv preprint arXiv:1701.07274*, 2017.

[87] A. Wachi, "Failure-scenario maker for rule-based agent using multi-agent adversarial reinforcement learning and its application to autonomous driving," *arXiv preprint arXiv:1903.10654*, 2019.

[88] V. Behzadan and A. Munir, "Adversarial reinforcement learning framework for benchmarking collision avoidance mechanisms in autonomous vehicles," *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 2, pp. 236–241, 2021.

[89] B. Chen, X. Chen, Q. Wu, and L. Li, "Adversarial evaluation of autonomous vehicles in lane-change scenarios," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, Jun. 2021.

[90] P. Du and K. Driggs-Campbell, "Finding diverse failure scenarios in autonomous systems using adaptive stress testing," *SAE International Journal of Connected and Automated Vehicles*, vol. 2, no. 12-02-04-0018, pp. 241–251, 2019.

[91] H. J. Cho and M. Behl, "Towards automated safety coverage and testing for autonomous vehicles with reinforcement learning," *arXiv preprint arXiv:2005.13976*, 2020.

[92] M. R. Zofka, F. Kuhnt, R. Kohlhaas, C. Rist, T. Schamm, and J. M. Zöllner, "Data-driven simulation and parametrization of traffic scenarios for the development of advanced driver assistance systems," in *2015 18th International Conference on Information Fusion (Fusion)*, 2015, pp. 1422–1428.

[93] F.-Y. Wang, N.-N. Zheng, D. Cao, C. M. Martinez, L. Li, and T. Liu, "Parallel driving in cpss: A unified approach for transport automation and vehicle intelligence," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 577–587, Sep. 2017.

[94] P. Koopman and M. Wagner, "Toward a framework for highly automated vehicle safety validation," in *2018 SAE WCX World Congress Experience*, 2018.

[95] L. Peng, H. Wang, and J. Li, "Uncertainty evaluation of object detection algorithms for autonomous vehicles," *Automotive Innovation*, vol. 4, no. 3, pp. 241–252, 2021.

[96] C. Wang, F. Xiong, and H. Winner, "Reduction of uncertainties for safety assessment of automated driving under parallel simulations," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 110–120, 2021.

[97] W. Wang, W. Han, X. Na, J. Gong, and J. Xi, "A probabilistic approach to measuring driving behavior similarity with driving primitives," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 1, pp. 127–138, 2020.

[98] M.-C. Hsueh, T. K. Tsai, and R. K. Iyer, "Fault injection techniques and tools," *Computer*, vol. 30, no. 4, pp. 75–82, Apr. 1997.

[99] G. Juez, E. Amparan, R. Lattarulo, A. Ruíz, J. Pérez, and H. Espinoza, "Early safety assessment of automotive systems using sabotage simulation-based fault injection framework," in *International Conference on Computer Safety, Reliability, and Security*. Springer, 2017, pp. 255–269.

[100] Y. Fu, A. Terechko, T. Bijlsma, P. J. Cuijpers, J. Redegeld, and A. O. Örs, "A retargetable fault injection framework for safety validation of autonomous vehicles," in *2019 IEEE International Conference on Software Architecture Companion (ICSA-C)*. IEEE, 2019, pp. 69–76.

[101] S. Jha, T. Tsai, S. Hari, M. Sullivan, Z. Kalbarczyk, S. W. Keckler, and R. K. Iyer, "Kayotee: A fault injection-based system to assess the safety and reliability of autonomous vehicles to faults and errors," *arXiv preprint arXiv:1907.01024*, 2019.

[102] S. Jha, S. S. Banerjee, J. Cyriac, Z. T. Kalbarczyk, and R. K. Iyer, "Avfi: Fault injection for autonomous vehicles," in *2018 48th annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W)*. IEEE, 2018, pp. 55–56.

[103] S. Jha, S. Banerjee, T. Tsai, S. K. Hari, M. B. Sullivan, Z. T. Kalbarczyk, S. W. Keckler, and R. K. Iyer, "Ml-based fault injection for autonomous vehicles: A case for bayesian fault injection," in *2019 49th annual IEEE/IFIP international conference on dependable systems and networks (DSN)*. IEEE, 2019, pp. 112–124.

[104] H. Ziade, R. A. Ayoubi, R. Velazco *et al.*, "A survey on fault injection techniques," *The International Arab Journal of Information Technology*, vol. 1, no. 2, pp. 171–186, Jan. 2004.

[105] Y. Tian, K. Pei, S. Jana, and B. Ray, "Deeptest: Automated testing of deep-neural-network-driven autonomous cars," in *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*, 2018, pp. 303–314.

[106] G. Juez, E. Amparan, R. Lattarulo, J. P. Rastelli, A. Ruiz, and H. Espinoza, "Safety assessment of automated vehicle functions by simulation-based fault injection," in *2017 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, 2017, pp. 214–219.

[107] G. J. Uriagereka, R. Lattarulo, J. P. Rastelli, E. A. Calonge, A. R. Lopez, and H. E. Ortiz, "Fault injection method for safety and controllability evaluation of automated driving," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 1867–1872.

[108] T. Stolte, S. Ackermann, R. Graubohm, I. Jatzkowski, B. Klamann, H. Winner, and M. Maurer, "A taxonomy to unify fault tolerance regimes for automotive systems: Defining fail-operational, fail-degraded, and fail-safe," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2021.

[109] H. Beglerovic, M. Stolz, and M. Horn, "Testing of autonomous vehicles using surrogate models and stochastic optimization," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–6.

[110] K. M. Betts and M. D. Petty, "Automated search-based robustness testing for autonomous vehicle software," *Modelling and Simulation in Engineering*, vol. 2016, Aug. 2016.

[111] H. Beglerovic, A. Ravi, N. Wikström, H.-M. Koegeler, A. Leitner, and J. Holzinger, "Model-based safety validation of the automated driving functio highway pilot," in *8th International Munich Chassis Symposium 2017*. Springer, 2017, pp. 309–329.

[112] T. Zhao, E. Yurtsever, J. Paulson, and G. Rizzoni, "Formal certification methods for automated vehicle safety assessment," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2022.

[113] A. Domenici, A. Fagiolini, and M. Palmieri, "Integrated simulation and formal verification of a simple autonomous vehicle," in *International Conference on Software Engineering and Formal Methods*. Springer, Feb. 2018, pp. 300–314.

[114] A. Zita, S. Mohajerani, and M. Fabian, "Application of formal verification to the lane change module of an autonomous vehicle," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*. IEEE, 2017, pp. 932–937.

[115] J. Arcile, R. Devillers, and H. Klaudel, "Verifcar: A framework for modeling and model checking communicating autonomous vehicles," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 3, p. 353–381, May. 2019.

[116] T. Wongpiromsarn and R. M. Murray, "Formal verification of an autonomous vehicle system," in *Conference on Decision and Control*, 2008.

[117] T. Wongpiromsarn, U. Topcu, and R. M. Murray, "Receding horizon temporal logic planning," *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2817–2830, Nov. 2012.

[118] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," *arXiv preprint arXiv:1708.06374*, 2017.

[119] F. Pasch, F. Oboril, B. Gassmann, and K.-U. Scholl, "Vulnerable road users in structured environments with responsibility-sensitive safety," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 270–277.

[120] S. Manzinger, C. Pek, and M. Althoff, "Using reachable sets for trajectory planning of automated vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 232–248, Aug. 2021.

[121] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, Aug. 2014.

[122] F. Gruber and M. Althoff, "Anytime safety verification of autonomous vehicles," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1708–1714.

[123] C. Pek, P. Zahn, and M. Althoff, "Verifying the safety of lane change maneuvers of self-driving vehicles based on formalized traffic rules," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 1477–1483.

[124] N. Arechiga, "Specifying safety of autonomous vehicles in signal temporal logic," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 58–63.

[125] E. Andreotti, P. Boyraz, and S. Selpi, "Mathematical definitions of scene and scenario for analysis of automated driving systems in mixed-traffic simulations," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 366–375, Jun. 2021.

[126] M. Elgharbawy, A. Schwarzhaupt, M. Frey, and F. Gauterin, "Ontology-based adaptive testing for automated driving functions using data mining techniques," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 66, pp. 234–251, Oct. 2019.

[127] C. E. Tuncali, G. Fainekos, D. Prokhorov, H. Ito, and J. Kapinski, "Requirements-driven test generation for autonomous vehicles with machine learning components," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 2, pp. 265–280, Jun. 2019.

[128] F. Gao, J. Duan, Z. Han, and Y. He, "Automatic virtual test technology for intelligent driving systems considering both coverage and efficiency," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 365–14 376, Dec. 2020.

[129] J. Duan, F. Gao, and Y. He, "Test scenario generation and optimization technology for intelligent driving systems," *IEEE Intelligent Transportation Systems Magazine*, vol. 14, no. 1, pp. 115–127, Feb. 2022.

[130] R. Ben Abdessalem, S. Nejati, L. C. Briand, and T. Stifter, "Testing advanced driver assistance systems using multi-objective search and neural networks," in *2016 31st IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2016, pp. 63–74.

[131] D. Karunakaran, S. Worrall, and E. M. Nebot, "Efficient statistical validation with edge cases to evaluate highly automated vehicles," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 2020, pp. 1–8.

[132] 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 Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1573–1582, Mar. 2021.

[133] S. Feng, Y. Feng, H. Sun, S. Bao, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, part ii: Case studies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5635–5647, Sep. 2021.

[134] S. Feng, Y. Feng, X. Yan, S. Shen, S. Xu, and H. X. Liu, "Safety assessment of highly automated driving systems in test tracks: A new framework," *Accident Analysis & Prevention*, vol. 144, p. 105664, Sep. 2020.

[135] S. Feng, Y. Feng, H. Sun, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles: An adaptive framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1213–1222, 2022.

[136] S. Feng, X. Yan, H. Sun, Y. Feng, and H. X. Liu, "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment," *Nature Communications*, vol. 12, no. 1, pp. 1–14, Feb. 2021.

[137] C. Stadler, F. Montanari, W. Baron, C. Sippl, and A. Djanatliev, "A credibility assessment approach for scenario-based virtual testing of automated driving functions," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 45–60, 2022.

[138] A. Collin, A. Y. Bin-Nun, and R. D. Tebbens, "Plane and sample: Maximizing information about autonomous vehicle performance using submodular optimization," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 530–537.

[139] M. Harth, M. S. Ali, R. Kates, and K. Bogenberger, "Data-driven modelling of car-following behavior in the approach of signalized urban intersections," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 1721–1728.

[140] M. Langer, M. Harth, L. Preitschaft, R. Kates, and K. Bogenberger, "Calibration and assessment of urban microscopic traffic simulation as an environment for testing of automated driving," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 3210–3216.

[141] S. Jiang, Y. Wang, L. Lin, W. Lin, Y. Cao, J. Miao, and Q. Luo, "Drf: A framework for high-accuracy autonomous driving vehicle modeling," *arXiv preprint arXiv:2011.00646*, 2020.

[142] S. Wagner, K. Groh, T. Kühbeck, and A. Knoll, "Towards cross-verification and use of simulation in the assessment of automated driving," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 1589–1596.

[143] K. Groh, S. Wagner, T. Kuehbeck, and A. Knoll, "Simulation and its contribution to evaluate highly automated driving functions," *SAE International Journal of Advances and Current Practices in Mobility*, vol. 1, no. 2019-01-0140, pp. 539–549, 2019.

[144] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "Deep reinforcement learning framework for autonomous driving," *Electronic Imaging*, vol. 2017, no. 19, pp. 70–76, 2017.

[145] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, Mar. 2016.

[146] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang, and H. Chen, "A survey on trajectory-prediction methods for autonomous driving," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2022.

[147] C. Hubmann, J. Schulz, M. Becker, D. Althoff, and C. Stiller, "Automated driving in uncertain environments: Planning with interaction and uncertain maneuver prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 1, pp. 5–17, Mar. 2018.

[148] T. Zhang, W. Song, M. Fu, Y. Yang, and M. Wang, "Vehicle motion prediction at intersections based on the turning intention and prior trajectories model," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 10, pp. 1657–1666, 2021.

[149] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Attention based vehicle trajectory prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 175–185, 2021.

[150] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 187–210, 2018.

[151] S. Zhang, W. Deng, Q. Zhao, H. Sun, and B. Litkouhi, "Dynamic trajectory planning for vehicle autonomous driving," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, 2013, pp. 4161–4166.

[152] C. Zu, C. Yang, J. Wang, W. Gao, D. Cao, and F.-Y. Wang, "Simulation and field testing of multiple vehicles collision avoidance algorithms," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 4, pp. 1045–1063, 2020.

[153] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, "Trafficpredict: Trajectory prediction for heterogeneous traffic-agents," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6120–6127.

[154] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *Robomech Journal*, vol. 1, no. 1, p. 1, Jul. 2014.

[155] J. Kim and D. Kum, "Collision risk assessment algorithm via lane-based probabilistic motion prediction of surrounding vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2965–2976, Sep. 2018.

[156] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1179–1184.

[157] Y. Jeong, S. Kim, and K. Yi, "Surround vehicle motion prediction using lstm-rnn for motion planning of autonomous vehicles at multi-lane turn intersections," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 1, pp. 2–14, Jan. 2020.

[158] C. Sun, R. Tan, J. Deng, R. Zhou, L. Chen, F.-Y. Wang, and D. Cao, "Accident prediction in mesoscopic view: A cpss-based social transportation approach," in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, 2021, pp. 306–311.

[159] L. Wang, C. Fernandez, and C. Stiller, "High-level decision making for automated highway driving via behavior cloning," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2022.

[160] K. Bergman, O. Ljungqvist, and D. Axehill, "Improved path planning by tightly combining lattice-based path planning and optimal control," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 57–66, 2021.

[161] Y. Wang, D. Pan, H. Deng, Y. Jiang, and Z. Liu, "Dynamic trajectory planning of autonomous lane change at medium and low speeds based on elastic soft constraint of the safety domain," *Automotive Innovation*, vol. 3, no. 1, pp. 73–87, 2020.

[162] J. Tan, C. Xu, L. Li, F.-Y. Wang, D. Cao, and L. Li, "Guidance control for parallel parking tasks," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 301–306, 2020.

[163] M. Waqas and P. Ioannou, "Automatic vehicle following under safety, comfort, and road geometry constraints," *IEEE Transactions on Intelligent Vehicles*, pp. 1–1, 2022.

[164] F. Tian, R. Zhou, Z. Li, L. Li, Y. Gao, D. Cao, and L. Chen, "Trajectory planning for autonomous mining trucks considering terrain constraints," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 4, pp. 772–786, 2021.

[165] L. Li, Y. Lin, N. Zheng, and F.-Y. Wang, "Parallel learning: A perspective and a framework," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 3, pp. 389–395, Jul. 2017.

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

[167] Z. Wu, L. Sun, W. Zhan, C. Yang, and M. Tomizuka, "Efficient sampling-based maximum entropy inverse reinforcement learning with application to autonomous driving," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5355–5362, Oct. 2020.

[168] M. Menner, K. Berntorp, M. N. Zeilinger, and S. D. Cairano, "Inverse learning for data-driven calibration of model-based statistical path planning," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 131–145, 2021.

[169] Z. Huang, J. Wu, and C. Lv, "Driving behavior modeling using naturalistic human driving data with inverse reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, Oct. 2021.

[170] M. Zhang, Y. Zhang, L. Zhang, C. Liu, and S. Khurshid, "Deeproad: Gan-based metamorphic testing and input validation framework for autonomous driving systems," in *2018 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 2018, pp. 132–142.

[171] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.

[172] K. Liu, Z. Ye, H. Guo, D. Cao, L. Chen, and F.-Y. Wang, "Fiss gan: A generative adversarial network for foggy image semantic segmentation," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 8, pp. 1428–1439, 2021.

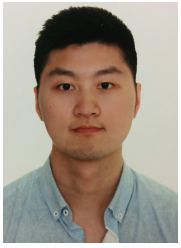
[173] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.

[174] M. Bojarski, D. W. del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[175] M. Bojarski, P. Yeres, A. Choromanska, K. Choromanski, B. Firner, L. Jackel, and U. Muller, "Explaining how a deep neural network trained with end-to-end learning steers a car," *arXiv preprint arXiv:1704.07911*, 2017.

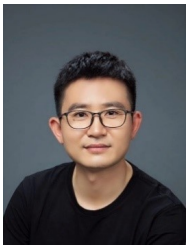
[176] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions," in *Robotics: Science and Systems*, vol. 2, Jun. 2016, pp. 1–9.

Yining Ma received the M.A.Sc degree in Mechanical Engineering from Pusan National University, Pusan, Korea, in 2016 and B.Eng. degree in Vehicle Engineering from the University of Shanghai for Science and Technology, Shanghai, China, in 2014. She is currently working toward the Ph.D. degree in Vehicle Engineering at School of Automotive Studies, Tongji University, Shanghai, China. Her research interests include testing scenario generation and evaluation of autonomous vehicles.

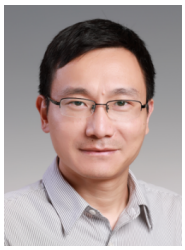


Chen Sun received the M.A.Sc degree in Electrical & Computer Engineering from University of Toronto, ON, Canada in 2017 and B.Eng. degree in automation from the University of Electronic Science and Technology of China, Chengdu, China, in 2014. He is currently working toward the Ph.D. degree in Mechanical & Mechatronics Engineering at University of Waterloo. He is a member of Cognitive Autonomous Driving Lab (CogDrive) and supervised by Prof. Dongpu Cao and Prof. Amir Khajepour. His research interests include end-to-end autonomous driving, safety validation for cyber-physical systems, planning and control for robots.

Junyi Chen received the B.S., and Ph.D. degrees from the School of Automotive Studies, Tongji University, Shanghai, China, in 2004, and 2009, respectively. She also received the joint Ph.D. degree from the Institut für Mathematische Stochastik, TU Braunschweig, Germany, in 2008. In 2010, she joined the School of Automotive Studies, Tongji University, as a lecturer, till now. Her research topics include safety of autonomous vehicles, driving assistance systems, test and evaluation of autonomous vehicles.



Dongpu Cao received the Ph.D. degree from Concordia University, Canada, in 2008. He is a Professor at Tsinghua University. His current research focuses on driver cognition, automated driving and cognitive autonomous driving. He has contributed more than 200 papers and 3 books. He received the SAE Arch T. Colwell Merit Award in 2012, IEEE VTS 2020 Best Vehicular Electronics Paper Award and 6 Best Paper Awards from international conferences. Prof. Cao has served as Deputy Editor-in-Chief for IET INTELLIGENT TRANSPORT SYSTEMS JOURNAL, and an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE/ASME TRANSACTIONS ON MECHATRONICS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE/CAA JOURNAL OF AUTOMATICA SINICA, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, and ASME JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT AND CONTROL. Prof. Cao is an IEEE VTS Distinguished Lecturer.



Lu Xiong received the B.E., M.E., and the Ph.D. degrees in Vehicle Engineering from the School of Automotive Studies, Tongji University, Shanghai, China, in 1999, 2002, and 2005, respectively. From November 2008 to 2009, he was a Postdoctoral Fellow with the Institute of Automobile Engineering and Vehicle Engines, University of Stuttgart, Germany, with Dr. Jochen Wiedemann. He is currently a Professor with Tongji University. He is also an Executive Director of the Institute of Intelligent Vehicles and an Associate Director of the Clean Energy Automotive Engineering Center, Tongji University. His research interests include perception, decision and planning, dynamics control and state estimation and testing and evaluation of autonomous vehicles.