A Metamorphic Testing Framework and Toolkit for Modular Automated Driving Systems

Riley Underwood, Quang-Hung Luu and Huai Liu✉

***Abstract*—Autonomous vehicles (AV), at their highest potentials, will provide a greater mobility, an increased traffic efficiency and, more importantly, safer trips for millions of people every day. While assuring their safety and reliability is, thus, of great importance, it is also a huge challenge. Metamorphic testing (MT) has been shown to be one of the most successful testing techniques to validate automated driving systems (ADS) underpinning the AV. Having said that, there are still lots of rooms for further improving the ADS testing with MT. On one hand, the non-determinism in ADS' behaviors poses great challenges for precisely judging their correctness. On the other hand, the testing scenarios used in the existing studies are still not very much complex for mimicking various realistic traffic conditions. In this study, we propose a new framework which takes into account the hypothesis testing to provide a more solid way for judging the non-deterministic behaviors of test outcomes. On top of that, we develop a new toolkit to implement more complex and realistic ADS testing scenarios. To demonstrate its practicability, we design complex traffic scenarios and pay attention to examine the ADS' behaviors in non-collision cases which are often unable to be detected by conventional testing methods. It is then applied to test Autoware, a state-of-the-art modular ADS using the Carla simulator. An analysis of results with the Mann-Whitney-Wilcoxon test and Cohen's d values reveals a large number of consistencies and reliability issues of Autoware. The findings highlight the flexibility and capability of our MT-based framework in validating the AV using a non-deterministic measure and realistic scenarios that can work in the absence of ground truth datasets.**

***Index Terms*—Metamorphic testing, self-driving cars, autonomous vehicles, diversity of traffic scenarios**

I. Introduction

Autonomous vehicles (AV) are desirably an indispensable component of our future intelligent transportation systems, providing more convenient trips, a less congested traffic and much safer transportation. The development of AV underpinned by the automated driving system (ADS) has principally been emerging and accelerating within the last 20 years, especially after the success of DARPA Grand Challenge [1]. Being set to serve and interact with millions of road users every day, the AVs are required to meet a very high standard of safety in order to be allowed for a mass adoption on public roads [2], [3]. In a word, assuring the safety of AVs is always of great importance.

Having said that, this task is also a huge challenge. It is because, first and foremost, there is currently a lack of a mechanism to determine the correctness of decisions for each scenario made by the AV, which is referred to as the test oracle problem in software engineering [4], [5], [6], [7]. For example, while the AV arrives an intersection and turns right, how can we know that the decision (turning right at a certain steering angle, or speed) is correct? Manual inspection of every scenario will be a tedious task if not almost impossible, considering a large number of possible traffic scenarios.

Metamorphic testing (MT) is one of the most effective methods in addressing the test oracle problem [8], [9]. MT makes use of the relations among multiple inputs and corresponding executions of the specified program, being referred to as *metamorphic relations* (MRs), to determine the faultiness of its implementation. If an MR is violated, the program is known to be faulty. This technique has been used to reveal real-life faults in various software systems in the areas of biomedicine [10], statistics [11], machine learning [12], [13], web services [14], [15], map and navigation [16], to name just a few. In testing self-driving cars, it has been applied to validate both autonomous driving models as well as end-to-end and modular autonomous driving systems [17], [4], [18], [19].

Despite its notable success, there is currently a challenge in testing the modular ADS. For exactly the same scenario and test conditions, the ADS under test may make two completely different driving decisions. This issue, being referred to as the reproducibility problem, arises from the nature of both hardware and software components used by the ADS, including deep learning algorithms that are commonly responsible for perceptions and tactical planning tasks [20], [1]. We have observed this issue while testing Apollo[1] [19] and recently Autoware[2], two state-of-the-art open-source autonomous driving systems. The non-deterministic behaviors make the judgement of violation in testing with MT indecisive.

In testing the modular autonomous driving systems, which is more complex than end-to-end autonomous driving models, previous studies have successfully applied MT to validate Apollo [4], [21], [19]. In particular, Zhou *et al.* [4] had discovered and reported real-life faults with this system before the first fatal accident of the ADS happened. Having said that, existing scenarios used in those studies are not very much complex and realistic as compared to a wide range of traffic scenarios that the AV must handle every day. In other words, more large-scale empirical studies are still required for confirming whether or not MT is applicable to test real-world ADS using complex scenarios.

On the other hand, simulation-based testing has been becoming crucial for the assessment and development of safe ADS [22], [23]. It allows for developing diverse collections of scenarios, and examining their responses with hundreds of

[1]https://developer.apollo.auto/

[2]https://www.autoware.org/

scenarios safely. On top of that, testing with the simulation may reduce the exposure of dangerous risks to the public with on-road testing. Carla[3] and similar simulators have been used frequently to examine the performance of ADS before it was integrated into the AV [23], [19].

In this study, we develop a new framework that is mainly based on MT to validate the modular ADS. To this end, we take advantage of hypothesis testing to resolve the non-deterministic issue with the system outputs in validating the violation of MR. It facilitates a comparison between completely different driving scenarios in the absence of reproducibility in the experiments. Note that our study uses hypothesis tests to verify non-deterministic systems, which differs from the work of Guderlei and Mayer [24] who developed a new type of hypothesis tests based on MT. On top of that, we show that it is practical and feasible to test the state-of-the-art ADS with more complex and realistic traffic scenarios. In doing so, we choose source scenarios from a report by the Department of Transportation's National Highway Traffic Safety Administrations on the most frequently happened crashes in the United States [25] and design new follow-up scenarios with a focus on the factors that will affect driving decisions. The experiments are supported by a new assessment toolkit for testing the ADS, allowing us to design new scenarios and test the system flexibly and effectively.

The organization of this paper is summarized as follows. In Section II, we will briefly introduce MT concepts, and then describe our framework and its toolkit. In Section III, we describe our experimental setup including scenarios and evaluations. Section IV, we highlight our findings from the tested scenarios. Section V briefly discusses the insights of our results and highlights main remarks of our study.

II. New Metamorphic Testing Framework and Toolkit for Modular ADS

A. The framework

1) Methodology: Our framework is mainly based on MT method. This technique makes use of the relations between multiple inputs and their outputs, which are referred to as *metamorphic relations* (MRs) for testing software systems. The adoption of MRs allows MT to alleviate the missing of test oracle, a mechanism to determine whether the test passes or fails. With the MR, we are able to generate new test cases, namely *follow-up test case*, from original test case, being referred to as *source test cases* in MT.

For example, in testing the program P to find the shortest path between two nodes in an undirected graph G, it is very expensive to determine the correctness of P, because a manual process to test a graph of n nodes may involve validating against $O(n^2)$ possible paths. Let $|P(G, x, y)|$ be the distance computed by the program P from the node x to the node y in the graph G. We can verify the program by employing the MR $|P(G, x, y)| \leq |P(G, x, z)| + |P(G, z, y)|$ where x, y and z are three different nodes in the graph. That is, we can generate the follow-up test cases (G, x, z) and (G, z, y) from its source test case (G, x, y), and then validate the satisfaction of their outputs against the above-mentioned MR. If the MR is violated, we know that P is faulty.

The main focus of our framework is the improvement of practicability of MT in testing ADS. As discussed in the previous section, one of the main challenges in testing the ADS is the non-deterministic behaviors of the system under test. That is, for the same given input, the system execution results will return two different outputs. This is due to the nature of randomness existing in both DLM algorithm as well as the hardware used in the complete ADS [27]. To resolve this issue, we have conducted a large number of tests for the same scenario, so that the statistical analysis is significant. We consider two sets of outputs, one for the source scenario and another for the follow-up scenario as two random variables, and the hypothesis test is applied to reject the null-hypothesis of probabilities. In this study, we apply it to the stopping distance, and suggest that it could be used for other safety measures such as speeds, positions and other behaviors of the ego vehicle driven by the ADS in the environment.

On the other hand, our framework is applicable to test a complete modular ADS with a wider range of traffic scenarios, which have not been adopted earlier to test Autoware or Apollo (as summarized in Table I). In our framework, traffic scenarios are designed in by first considering the factor of changes that may make the ADS decisions consistent. We then transform the source scenarios to the follow-up ones using this factor. For example, we may replace the leading car in the source scenario by a truck in the follow-up scenario, and require that the ADS should make a consistent decision.

2) Testing process: The fundamental steps in conducting the testing in our framework is summarized as follows:

1) Given a (source) driving scenario S which consists of the environment and objects, our ego vehicle AV_E is set to drive automatically in the driving environment by the ADS under test).
2) The ADS is required to avoid any collision with *n* traffic objects O_k $(k = 1, 2, \ldots, n)$ (e.g., cars, pedestrians, cyclists) and, in the high level, makes consistent dynamic driving tasks when obeying the traffic rule. In doing so, we have to evaluate the situations that may influence the decisions, and gather it for designing the follow-up scenarios.
3) We construct a follow-up scenario S', that is similar to S, except that the objects in the environment may be different, that is, we have m objects O'_k $(k = 1, 2, \ldots, m)$ (which can be the same or different from the ones in the source scenario). For the sake of simplicity, the type and number of objects, as well as their speed are modified in this study.
4) We execute both source and follow-up scenarios for the same number of times and obtain their results for statistical tests.
5) Given the traffic rule, the ego vehicle AV_E should respond to objects in new (follow-up) scenario S' con-

[3]https://carla.org/

TABLE I
SUMMARY OF KEY MT-BASED STUDIES IN TESTING AUTONOMOUS DRIVING

Study	Systems			Sensor data		Categories of scenarios		MR satisfaction
	DLM	ADS	Systems under test	Camera	LiDAR	Total	Description	
Tian *et al.* [17]	x	-	Udacity models	x	-	1	Environment	No change in steering after environment changes
Zhang *et al.* [26]	x	-	Udacity models	-	x	2	Environment	No change in steering after environment changes
Zhou *et al.* [4]	-	x	Apollo ADS	-	x	1	Object	No change in number of objects after adding LiDAR points
Han and Zhou [21]	-	x	Apollo ADS	-	x	1	Object	No change in minimum stopping distance
Seymor *et al.* [19]	-	x	Apollo ADS	-	x	1	Environment	No change in number of objects when the environment changes
Luu *et al.* [7]	x	-	Udacity models	x	-	Various	Environment and object	Different changes in steering and objects consistency
Deng *et al.* [5]	x	-	Udacity, ResNet101, VGG16 models	x	-	Various	Environment and object	Different changes in steering and object consistency
This study	-	x	Autoware ADS	x	x	5	Object	No change in stopping distance for different objects

sistently with the corresponding objects in the scenario S. For this purpose, statistical tests are then applied to measure the difference between the outputs of scenarios, and hence determine the violation of MR.

We adopt the MRs that were used in previous studies [19], [7], [5], that is, the ego vehicle AV_E should make consistent decisions for the similar driving scenario. We leverage the notion of Metamorphic Relation Input Patterns (MRIP) [16] to group scenarios together.

B. The toolkit

1) Overview: The toolkit is developed to test the ADS against different driving safety scenarios semi-automatically with a potential for the scalability. In our current version, the framework supports the following components

- Compatible systems under test: Autoware, which is an open-source modular ADS for autonomous vehicles based on the Robot Operating System (ROS)[4]. Taking the input from sensors (LiDAR, camera, radar, GPS, IMU, etc.), it combines hardware, software components and other peripherals to perceive the driving environment and make driving decisions.
- Test environment: Carla, an open-source platform designed to simulate the operation of autonomous vehicles in a realistic urban environment.
- Bridging software systems, which mainly consist of Robot Operating System (ROS), Carla-Autoware Bridge and ROS bridge. Here, ROS bridge allows for an integration between Autoware and ROS operating system as well as its software libraries for the communication with ROS nodes and topics.

2) Architecture: The architecture of the framework is shown in Fig. 1. The system flow is briefly described as follows. The safety evaluation process for a scenario is activated by executing the Python scripts. These scripts directly communicate with the driving simulator (Carla) through the Python API through a two-way communication system that connects Carla and toolkit. Within Carla, the scenario's specific simulated environment is generated using graphical engine, that is, Epic Game's Unreal Engine 4. ROS Bridge Agent is one of the main components for signal exchanges between Carla and Autoware. Signals from both camera and LiDAR sensors in the ego vehicle in Carla are transmitted to Autoware through these complex systems. Autoware will use this data to perceive the environment, evaluate the scenario, and make driving decisions.

3) Interfaces: Testing scenarios are controlled by the user from a graphic user interface (GUI), through the whole process of selecting and running the scenarios. For each scenario, datasets representing the input data for source and follow-up test cases are created and stored in JSON format. The output of recording environment data while a simulation test run is exported to a .txt file with the following information

- Time of event
- X, Y, Z coordinates of AV_E
- Velocity and acceleration of AV_E
- Collision status
- Lane invasion status
- Distance to objects related to AV_E

Our framework is open source and accessible at the GitHub repository: https://github.com/rjunderwood/avtestkit

III. EXPERIMENTS

A. Scenarios

The framework is able to assist users to generate a wide range of scenarios. To demonstrate its effectiveness, we adopt five categories of scenarios as illustrated in Fig. 2. They are motivated by the pre-crash scenarios reported by the U.S. Department of Transportation's National Highway Traffic Safety Administrations [25].

For each set of scenarios, we start with the source (as shown in Fig. 3), and define the factors that may affect the driving decisions. From there, we derive the follow-up scenarios and

[4]https://www.ros.org/

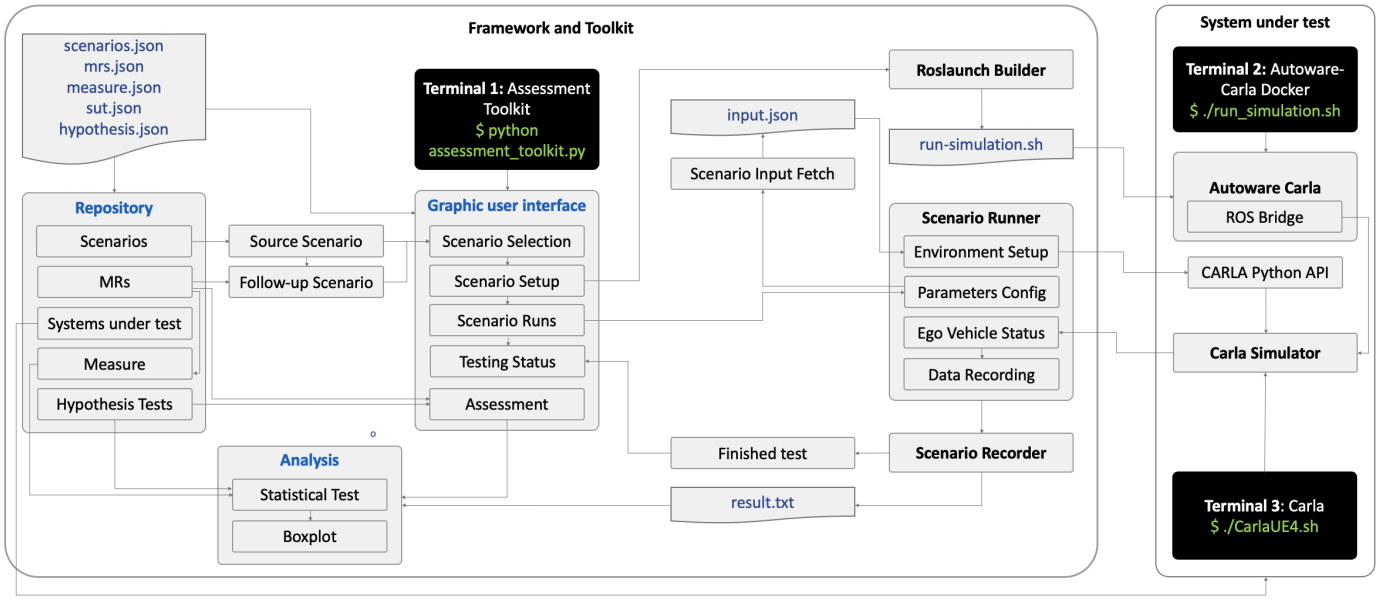

Fig. 1. The architecture of our framework.

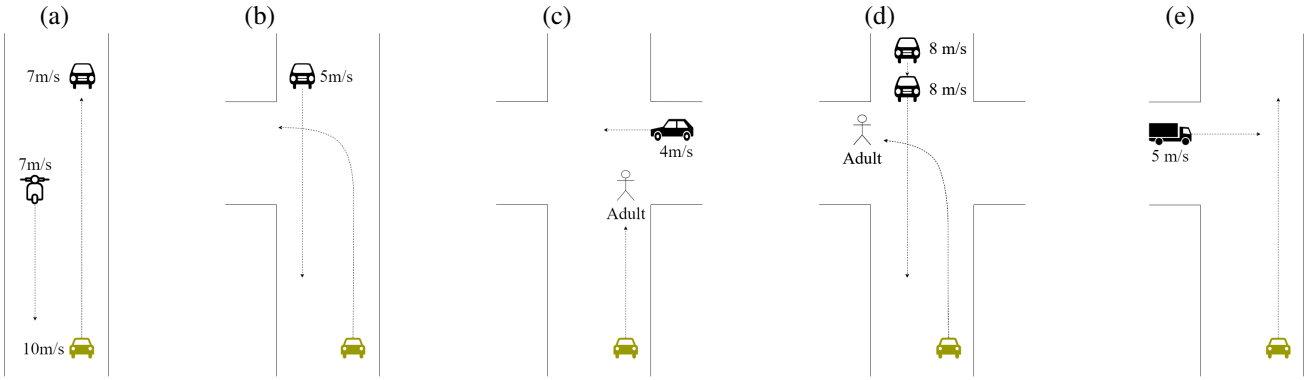

Fig. 2. Illustration of selected sets of scenarios in the framework: **a)** scenarios A (car-following); **b)** scenarios B (left-turning at intersection); **c)** scenarios C (pedestrian crossing); **d)** scenarios D (Maneuvering at intersection with pedestrian crossing); and **e)** scenarios E (Red light with complex traffic). A complete description of scenarios is available in our GitHub at https://github.com/rjunderwood/avtestkit/tree/main/scenario_figures

consider the metric for evaluation of MRs. The detailed setup of scenarios are shown in Table II and illustrated in Fig. 2. They are elaborated as follows.

1) Scenarios A (car following): In this set of scenarios, the AV follows leading vehicles in the mixed traffic (with other vehicles moving in the opposite lane as well). The AV needs to safely adjust its speed, distance and maneuvers in responding to the lead and surrounding vehicles. This is a very common scenario on roads that requires the actions of acceleration, braking, and lane changes if necessary. Conventionally, it is safer for the following vehicle to keep a greater distance from the lead, which allows more time to process and come up with safe decisions such as stopping or slowing when the lead vehicle reduces the speed urgently. From a source scenario S with an ego car moving behinds a leading car, we generated six follow-up scenarios (Table II).

2) Scenarios B (left-turning at intersection): The AV should turn left at an intersection while there are oncoming vehicles. This scenario requires the AV to consider the oncoming vehicles so that it needs to do a complete stop before making a turn if needed. The left-turning at intersection is an important scenario to test the AV because it requires the AVs to make a decision based on multiple variables and the ability to safely navigate through an intersection. From a source scenario S with an ego car turning left at an intersection, we had four follow-up scenarios (Table II).

3) Scenarios C (pedestrian crossing): In this scenario, the AV drives on the straight road while one or multiple pedestrians are crossing directly in the vehicle's oncoming path. The AV must adjust its speed, either slowing down or transitioning to a complete stop if necessary so that the pedestrians are able to cross safely. From a source scenario S

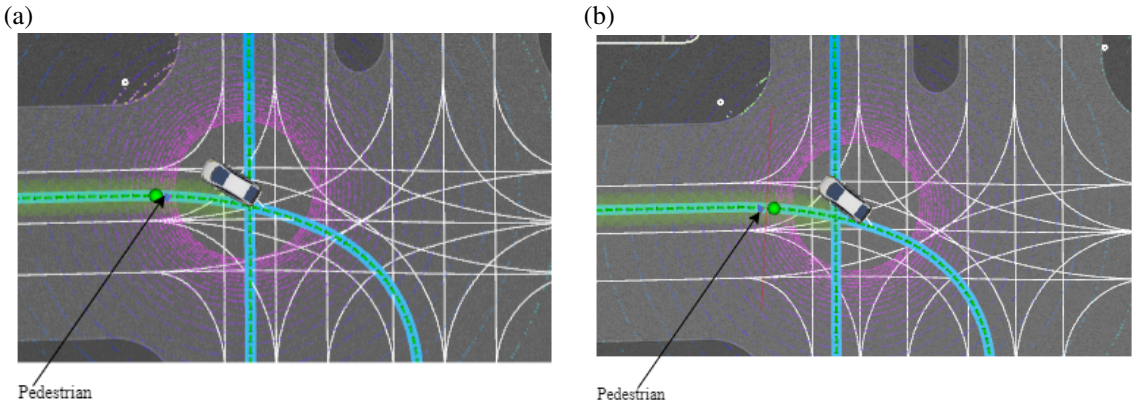

Fig. 3. Horizontal view in Autoware ADS with LiDAR point clouds, map and objects of the scenario associated with a pedestrian crossing prior vehicle maneuver for different stop positions related to (a) Source test case (b) Follow-up test case.

with an ego car going straight at an intersection, we derived three follow-up scenarios (Table II).

4) Scenarios D (maneuvering at intersection with pedestrian crossing): This set of scenarios is more complex than set B and set C. Here, pedestrians cross the road on the path of a AV which is making a turning maneuver at an intersection, facing the pedestrians in the vehicle's oncoming path. The AV must adjust its vehicle speed and come to a complete stop if necessary so that the pedestrians are able to cross safely. In this scenario, the AV should also be aware of other oncoming vehicles. This set of scenarios demonstrates the AV's ability to recognize and respond to pedestrians while also managing main driving tasks, and hence it is particularly challenging. From a source scenario S with an ego car maneuvering at an intersection, we generated follow-up scenarios (Table II).

5) Scenarios E (red light with complex traffic): This set of scenarios involves an AV approaching a traffic-signaled intersection. When there is a green light, it should move safely, or the AV should stop when there is a red light. The challenging part is that the AV should be able to handle other vehicles coming in from the side of the intersection or heading to cross in front of the AV's path, causing a potentially fatal collision. This requires the AV to be aware of the vehicles running the red light, testing its ability to react to unexpected situations and then make quick and safe decisions. From a source scenario S with an ego car approaching an intersection with a traffic light, we had six follow-up scenarios (Table II).

B. Experimental setup and evaluation

For each source scenario, we have an AV_E, a set of n objects $O_k(k = 1, 2, \ldots, n)$ whose basic properties are the type, starting and ending positions, velocity profile. A human object can either be an adult or a child, while other objects can be a car, a truck, a motorbike with different colors.

Since our focus is to examine the robustness of the system, test cases resulting in collision are straight-forward and hence will be ignored. We only focus on the non-collision test cases and see if our framework can be leveraged to detect any failure in such situations. To over the randomness in the results caused by the stochastic nature of the system, we run each source scenario and follow-up ones 10 times. The recorded data was managed in the framework so that each test run would start recording at the same X, Y, and Z location in the scenario and finish recording after a set amount of time. This would allow for each, and every test run not to miss the potential collision event.

The violations of MR in each metamorphic group (MG) of scenarios are calculated for each pair of follow-up and source test cases. We use stopping distance as the measure since it is a critical factor in evaluating the safe operation of the AV [28]. We adopt the Mann Whitney-Wilcoxon statistical test, which is a non-parametric test of the null hypothesis from two random values R_1 and R_2 from two populations, assuming that the probability of R_1 being greater than R_2 is equal to the probability of R_2 being greater than R_1. Apart from the hypothesis tests, we have also carried out an analysis on the effect size with Cohen's d value [29], which measures the strength of the relationship between R_1 and R_2.

IV. Result

A. Scenarios A: Car following

In this scenario, the MR is violated when the AV has inconsistent stopping distances across the source and follow-up test cases. It is an important set of scenarios because it requires the AV to maintain a safe following distance, anticipate the actions of the lead vehicles and make correct adjustments to its own speed as well as come to a stop if needed. Fig. 4a and Table III report violations in increasing the initial velocity of the AV or changing the vehicle types to trucks or motorbikes. The violation of follow-up test cases associated with motorbike showed a decrease in the stopping distance which can be viewed as unsafe, whereas the stopping distances for other cases are not significantly different in terms of statistics, and thus can not be regarded as unsafe. Passing cars and passing motorbikes seemed to have little effect on the stopping distance as there were no violations.

TABLE II
SCENARIOS SETUP

Category	Name	Source scenario	Follow-up scenarios
A	Car-following (Fig. 2a)	S: ego car (10m/s), 1 leading car (speed=7m/s)	F1: ego car (10m/s), 1 leading truck (speed=7m/s) F2: ego car (10m/s), 1 leading motorbike (speed=10m/s) F3: ego car (10m/s), 1 leading motorbike (speed=7m/s) F4: ego car (10m/s), 2 leading cars (speeds=7m/s) F5: ego car (10m/s), 1, leading car + 1 leading motorbike (speeds=7m/s)
B	Left-turning at intersection (Fig. 2b)	S: ego car, (speed=5m/s), 1 oncoming car	F1: ego car, 1 oncoming truck (speed=5m/s) F2: ego car, 1 oncoming motorbike (speed=5m/s) F3: ego car, 1 oncoming car (speed=6m/s) F4: ego car, 2 oncoming car (speed=5m/s)
C	Pedestrian crossing (Fig. 2c)	S: ego car, 1 pedestrian	F1: ego car, 2 pedestrian F2: ego car, 1 pedestrian + 1 left crossing car (speed=4m/s) F3: ego car, 1 pedestrian + 1 right crossing car (speed=4m/s)
D	Maneuvering at intersection with pedestrian crossing (Fig. 2d)	S: ego car, 1 pedestrian	F1: ego car, 2 pedestrian F2: ego car, 1 pedestrian + 1 oncoming car (speed=5m/s) F3: ego car, 1 pedestrian + 1 oncoming car (speed=8m/s) F4: ego car, 1 pedestrian + 2 oncoming cars (speed=8m/s)
E	Red light with complex traffic (Fig. 2e)	S: ego car, 1 running car (speed=5m/s)	F1: ego car, 1 truck (speed=5m/s) F2: ego car, 1 running car + 1 oncoming car (speed=5m/s) F3: ego car, 1 running car + 2 oncoming cars (speed=5m/s) F4: ego car, 1 running car + 1 oncoming truck (speed=5m/s) F5: ego car, 1 running car + 1 oncoming motorbike (speed=5m/s) F6: ego car, 1 running car (speed=6m/s)

S is the source scenario; while F1, F2, F3, F4, F5 and F6 are the follow-up scenarios. Each scenario consists of 10 repeated test cases. Detail information about the variants and their experiments are given at https://github.com/rjunderwood/avtestkit

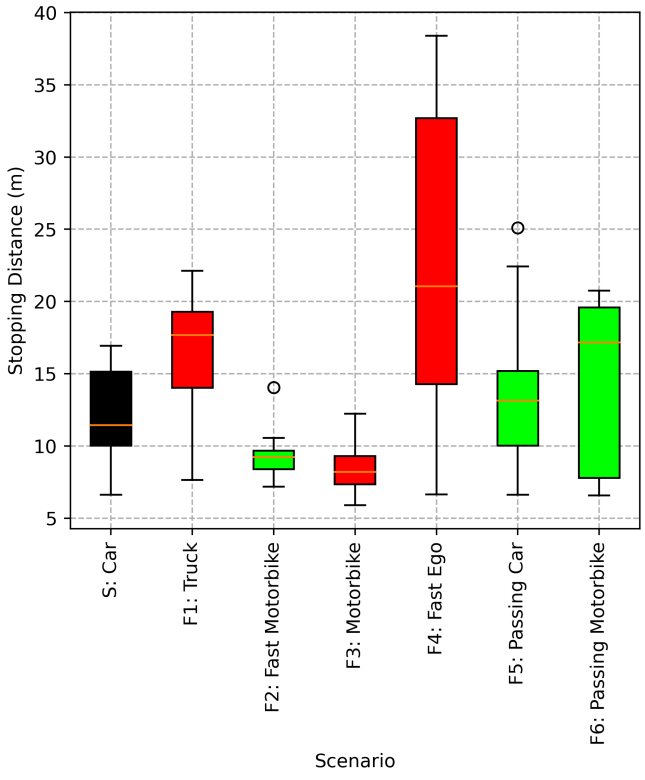

Fig. 4. Stopping distances for a sample set of scenarios (Scenarios A). The black box indicates the statistics of the source test case. Red boxes represent the follow-up test cases where the MR is observed as violated under the Mann-Whitney-Wilcoxon test. Green boxes represent the follow-up test cases where the MR is not violated.

B. Scenarios B: Left-turning at intersection

The MR for the set of scenarios associated with the left-turning at intersection is to assure that the AV should respond safely. Similar to the set of scenarios A, it must be statistically consistent in stopping distances across different metamorphic groups of scenarios. Table III shows that changing the oncoming vehicle to a motorbike or changing the speed of the oncoming car will lead to MR violations. The scenarios associated with motorbike have a shorter stopping distance than the source test case with the car, suggesting that the AV is making less safe decisions. The greater stopping distance observed with the scenario in which the vehicle moves faster may be attributed to the impact of a faster vehicle imposing on the perception of an AV.

C. Scenarios C: Pedestrian crossing

The MR for this set of scenarios guarantees that there will be consistent outputs when the driving scenarios are basically unchanged. The main difference is the involvement of different pedestrians, or other vehicles appearing in the AV's camera view. Table III shows that there were violations for test cases associated with two pedestrians, which produce a greater stopping distance. The larger stopping distance between pedestrians and AV may suggest that the system is better aware of multiple pedestrians as compared to the single pedestrian. Other finding is that having extra vehicles in the scenario further up the road in front of the AV may distract it from noticing the pedestrian ahead as the stopping distances becomes shorter. These observations are arguably interesting and may serve as data for understanding the behaviours of AV and also how to improve AV's reliability.

TABLE III
MANN–WHITNEY–WILCOXON TESTS AND COHEN'S D VALUES FOR THE DIFFERENCES OF FOLLOW-UP TEST CASES AGAINST THE SOURCE TEST CASES BY MEANS OF STOPPING DISTANCE (IN UNITS OF METRES).

Set	Avg	StdDev	MG	p-value	Cohen's d
A	S: 11.99m	S: 3.63m			
	F1: 16.44m	F1: 4.96m	$< F1, S >$	0.0257^{*}	X (1.02)
	F2: 9.41m	F2: 1.92m	$< F2, S >$	0.0639	L (0.89)
	F3: 8.37m	F3: 1.95m	$< F3, S >$	0.0211^{*}	X (1.24)
	F4: 22.79m	F4: 11.32m	$< F4, S >$	0.0257^{*}	X (1.28)
	F5: 13.88m	F5: 5.91m	$< F5, S >$	0.5966	M (0.39)
	F6: 14.17m	F6: 6.08m	$< F6, S >$	0.3417	M (0.43)
B	S: 4.56m	S: 0.95m			
	F1: 5.61m	F1: 3.68m	$< F1, S >$	0.9158	M (0.38)
	F2: 3.49m	F2: 0.65m	$< F2, S >$	0.0030^{**}	X (1.34)
	F3: 7.60m	F3: 2.95m	$< F3, S >$	0.0043^{**}	X (1.36)
	F4: 5.54m	F4: 2.55m	$< F4, S >$	0.6499	M (0.51)
C	S: 4.50m	S: 1.58m			
	F1: 7.52m	F1: 1.30m	$< F1, S >$	0.0004^{***}	H (2.09)
	F2: 3.56m	F2: 0.61m	$< F2, S >$	0.2897	L (0.78)
	F3: 3.64m	F3: 0.99m	$< F3, S >$	0.2566	L (0.65)
D	S: 6.37m	S: 1.26m			
	F1: 10.23m	F1: 0.60m	$< F1, S >$	0.0002^{***}	H (3.91)
	F2: 9.33m	F2: 0.99m	$< F2, S >$	0.0006^{***}	H (2.61)
	F3: 9.73m	F3: 0.96m	$< F3, S >$	0.0002^{***}	H (3.05)
	F4: 10.60m	F4: 0.72m	$< F4, S >$	0.0002^{***}	H (4.12)
E	S: 7.35m	S: 0.69m			
	F1: 8.36m	F1: 0.82m	$< F1, S >$	0.0172^{*}	X (1.33)
	F2: 6.95m	F2: 0.90m	$< F2, S >$	0.3846	M (0.50)
	F3: 6.97m	F3: 0.69m	$< F3, S >$	0.3846	M (0.56)
	F4: 7.66m	F4: 0.58m	$< F4, S >$	0.4961	M (0.49)
	F5: 10.87m	F5: 0.58m	$< F5, S >$	0.0002^{***}	H (5.49)

Avg and StdDev are arithmetic means and standard deviations of test cases, respectively. S is the value of source scenario; while F1, F2, F3, F4, F5 and F6 are the values of follow-up scenarios. Symbols *, ** and *** highlight p-values that are smaller than $0.05, 0.01$ and 0.001, respectively. Notation H, X, L and M denote the huge, very large, large and medium differences, respectively, in the Cohen's d effect size [29], [30] whose concrete values are given inside the brackets.

D. Scenarios D: Maneuvering at intersection with pedestrian crossing

The MR was to check whether there would still be statistical consistency across stopping distances between the AV and pedestrians on its path when there are changes in the number of pedestrians and other vehicles during AV's maneuver. Table III shows that there is an MR violation for all follow-up test cases with very high confidence. In other words, the ADS made significantly faulty decisions in this set of scenarios.

E. Scenarios E: Red light with complex traffic

In this set of scenarios, when the traffic light is changed (e.g., from a green light to a red light) under one scenario, the AV should be able to handle the decision consistently for the similar scenario. Table III shows two distinctive cases of violations; one for trucks and one for a fast car. The truck has a greater stopping distance which suggests that a larger vehicle coming through a red light is more appealing to the AV, making the reaction time for the AV to stop faster. The fast car arrives in front of the AV quicker than others, requiring a longer reaction time to make a complete stop, and thus resulting in a shorter stopping distance.

On top of that, we also looked at the relationships between the source scenarios and the follow-up ones. Table III showed that their differences are significant, demonstrated by the Cohen's d effect size ranging from medium to huge. Notably, the huge values coincide with p-values that are smaller than 0.001. The results on effect size reinforce the statistical significance of our research findings.

V. CONCLUSIONS

To sum up, we have proposed a framework based on MT technique which takes advantage of hypothesis test to resolve the non-deterministic behaviors of the modular ADS. In conjunction with it, we have developed a new toolkit and adopted it to design complex traffic scenarios for testing. We applied it to test Autoware via the Carla simulator-based environment with a focus on the non-collision cases using five categories of traffic scenarios. Our results were evaluated with the Mann-Whitney-Wilcoxon test and Cohen's d sample size, which revealed a large number of consistency and reliability issues of Autoware. The study highlights the benefits of MT-based frameworks in validating the AV which does not require the existence of a ground truth data.

From the practical perspective, the MR violations give us new information about the reliability and safety of Autoware, one of the state-of-the-art open-source ADS. Each set of scenarios presents a unique traffic situation. The violations measured by stopping distances demonstrate the vulnerability of Autoware system. As referred from the large number of violations in the majority of scenarios tested, a wider range of scenarios are expected to validate the Autoware. Autoware-Carla couple system allows for building many complex driving situations that in the real world cannot be iteratively tested to the amount of and the fine-tuning control that the Autoware-Carla system has. The findings suggest that more efforts should be done to improve the perception and intelligence of the state-of-the art automated driving systems.

ACKNOWLEDGMENT

This project is supported by the grant DP210102447 from Australian Research Council. We thank James Sanders, Jason Vlajankov, Billy Croxford, Chirathi Perera and Aidan Cheung for their contributions as well as Thai M. Nguyen for sharing his knowledge on Autoware. We appreciate Prof. Tsong Yueh Chen for his encouragement to start this project and his comments for the paper.

REFERENCES

[1] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. M. Paixão, F. Mutz, L. de Paula Veronese, T. Oliveira-Santos, and A. F. De Souza, "Self-driving cars: A survey," *Expert Systems with Applications*, vol. 165, p. 113816, 2021.

[2] E. Awad, S. Dsouza, R. Kim, J. Schulz, J. Henrich, A. Shariff, J.-F. Bonnefon, and I. Rahwan, "The moral machine experiment," *Nature*, vol. 563, no. 7729, pp. 59–64, Nov 2018.

[3] S. Nyholm, "The ethics of crashes with self-driving cars: A roadmap, I," *Philosophy Compass*, vol. 13, no. 7, pp. e12 507(1–10), 2018.

[4] Z. Q. Zhou and L. Sun, "Metamorphic testing of driverless cars," *Communications of the ACM*, vol. 62, no. 3, pp. 61–67, Feb. 2019.

[5] Y. Deng, X. Zheng, T. Zhang, H. Liu, G. Lou, M. Kim, and T. Y. Chen, "A declarative metamorphic testing framework for autonomous driving," *IEEE Transactions on Software Engineering*, 2022.

[6] T. Y. Chen and T. H. Tse, "New visions on metamorphic testing after a quarter of a century of inception," in *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE 2021. New York, NY, USA: ACM, 2021, p. 1487–1490.

[7] Q.-H. Luu, H. Liu, T. Y. Chen, and H. L. Vu, "A sequential metamorphic testing framework for understanding automated driving systems," *arXiv:2206.03075v1*, pp. 1–21, 2022.

[8] S. Segura, G. Fraser, A. Sanchez, and A. Ruiz-Cortes, "A on metamorphic testing," *IEEE Transactions on Software Engineering*, vol. 42, no. 9, pp. 805–824, Sept 2016.

[9] T. Y. Chen, F. C. Kuo, H. Liu, P. Poon, D. Towey, T. H. Tse, and Z. Q. Zhou, "Metamorphic testing: A review of challenges and opportunities," *ACM Computing Surveys*, vol. 51, no. 1, 2018.

[10] T. Y. Chen, J. W. Ho, H. Liu, and X. Xie, "An innovative approach for testing bioinformatics programs using metamorphic testing," *BMC Bioinformatics*, vol. 10, no. 1, p. 24, Jan 2009.

[11] Q.-H. Luu, M. F. Lau, S. P. Ng, and T. Y. Chen, "Testing multiple linear regression systems with metamorphic testing," *Journal of Systems and Software*, vol. 182, pp. 111 062(1–21), 2021.

[12] X. Xie, J. W. Ho, C. Murphy, G. Kaiser, B. Xu, and T. Y. Chen, "Testing and validating machine learning classifiers by metamorphic testing," *Journal of Systems and Software*, vol. 84, no. 4, pp. 544–558, 2011, the Ninth International Conference on Quality Software.

[13] X. Xie, Z. Zhang, T. Y. Chen, Y. Liu, P.-L. Poon, and B. Xu, "Mettle: A metamorphic testing approach to assessing and validating unsupervised machine learning systems," *IEEE Transactions on Reliability*, vol. 69, no. 4, pp. 1293–1322, 2020.

[14] C.-a. Sun, G. Wang, B. Mu, H. Liu, Z. Wang, and T. Chen, "Metamorphic testing for web services: Framework and a case study," in *2011 IEEE International Conference on Web Services*, 2011, pp. 283–290.

[15] J. Ahlgren, M. Berezin, K. Bojarczuk, E. Dulskyte, I. Dvortsova, J. George, N. Gucevska, M. Harman, M. Lomeli, E. Meijer, S. Sapora, and J. Spahr-Summers, "Testing web enabled simulation at scale using metamorphic testing," in *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, 2021, pp. 140–149.

[16] Z. Q. Zhou, L. Sun, T. Y. Chen, and D. Towey, "Metamorphic relations for enhancing system understanding and use," *IEEE Transactions on Software Engineering*, vol. 46, no. 10, pp. 1120–1154, 2020.

[17] Y. Tian, K. Pei, S. Jana, and B. Ray, "DeepTest: Automated testing of deep-neural-network-driven autonomous cars," in *Proceedings of the 40th International Conference on Software Engineering*, ser. ICSE '18. New York, NY, USA: ACM, 2018, pp. 303–314.

[18] J. Garcia, Y. Feng, J. Shen, S. Almanee, Y. Xia, and Q. A. Chen, "A comprehensive study of autonomous vehicle bugs," in *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*, 2020, pp. 385–396.

[19] J. Seymour, D.-T.-C. Ho, and Q.-H. Luu, "An empirical testing of autonomous vehicle simulator system for urban driving," in *2021 IEEE International Conference on Artificial Intelligence Testing (AITest)*, 2021, pp. 111–117.

[20] B. Haibe-Kains, G. A. Adam, A. Hosny, F. Khodakarami, T. Shraddha, R. Kusko, S.-A. Sansone, W. Tong, R. D. Wolfinger, C. E. Mason, W. Jones, J. Dopazo, C. Furlanello, L. Waldron, B. Wang, C. McIntosh, A. Goldenberg, A. Kundaje, C. S. Greene, T. Broderick, M. M. Hoffman, J. T. Leek, K. Korthauer, W. Huber, A. Brazma, J. Pineau, R. Tibshirani, T. Hastie, J. P. A. Ioannidis, J. Quackenbush, H. J. W. L. Aerts, and M. A. Q. C. M. S. B. of Directors, "Transparency and reproducibility in artificial intelligence," *Nature*, vol. 586, no. 7829, pp. E14–E16, Oct 2020.

[21] J. C. Han and Z. Q. Zhou, "Metamorphic fuzz testing of autonomous vehicles," in *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*, ser. ICSEW'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 380–385. [Online]. Available: https://doi.org/10.1145/3387940.3392252

[22] N. Kalra and S. M. Paddock, *Driving to Safety: How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability?* Santa Monica, CA: RAND Corporation, 2016.

[23] P. Kaur, S. Taghavi, Z. Tian, and W. Shi, "A survey on simulators for testing self-driving cars," *arXiv:Robotics*, 2021.

[24] R. Guderlei and J. Mayer, "Statistical metamorphic testing testing programs with random output by means of statistical hypothesis tests and metamorphic testing," in *Seventh International Conference on Quality Software (QSIC 2007)*, Oct 2007, pp. 404–409.

[25] W. G. Najm, J. D. Smith, and M. Yanagisawa, *Pre-Crash Scenario Typology for Crash Avoidance Research*. U.S. Department of Transportation, National Highway Traffic Safety Administration, 2007.

[26] 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)*, 2018, pp. 132–142.

[27] G. Xiao, J. Liu, Z. Zheng, and Y. Sui, "Nondeterministic impact of cpu multithreading on training deep learning systems," in *2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE)*, 2021, pp. 557–568.

[28] H. Wu, D. Lyu, Y. Zhang, G. Hou, M. Watanabe, J. Wang, and W. Kong, "A verification framework for behavioral safety of self-driving cars," *IET Intelligent Transport Systems*, vol. 16, no. 5, pp. 630–647, 2022.

[29] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. Routledge, 1998.

[30] S. S. Sawilowsky, "New effect size rules of thumb," *Journal of Modern Applied Statistical Methods*, vol. 8, no. 2, 2009.