Satyabrata Pradhan

Agentic AI for Software-Defined Vehicles: A Generative Testing Framework for Autonomous Feature Assurance

Journal of Electrical Systems

Abstract: - SDVs get advanced as a result of applying artificial intelligence (AI), yet they become increasingly difficult to test. The conventional techniques such as the scripted examinations or random scenarios are not sufficient to test all of the likelihood of driving that may contain peculiar and hazardous elements. The present research paper claims to introduce a novel conceptualization using agentic AI and generative AI (with no contact to real-life phenomena) to experiment in SDV attributes more effectively. The agentic system of AI will be autonomous that is, they plan, learn and innovate overtime. The Generative AI assists in supporting millions of test cases, unusual road dynamics, driver behavior and cyber-attacks. It experimented with its framework using modified systems of driver-assistance (ADAS) and the experiment was very successful. In comparison, agentic AI contains three times more opportunities to go off the rails relative to scripted tests and exceeding a regulation failure relative to what random tests. It also enabled the vehicles to be competent to pass safety standards like FMVSS and the ISO 26262 standard as risks may be noticed promptly. The model minimized the use of the network without the report of mines to the testing speed and coverage which was enabled with a combination of edge resources and cloud-based resources. This was what was called continuous learning as the system continued to improve as more vehicles provided the data. The result has already been proven that AI-based testing is scalable, viable and applicable to use in complex conditions. According to the study, agentic AI could greatly contribute to smarter and safer SDV testing that would give engineers the opportunity to identify and eliminate issues extremely quickly. Such an approach will go toward the adoption of faster and have more confidence in self-driving cars.

Keywords: Software-defined vehicles, Generative AI, Agentic AI, Autonomous

I. INTRODUCTION

The automobile industry has a trend of moving to software defined vehicles systems under which less of the character is handled by the hard ware hardware and through software. Such cars base themselves on artificial intelligence to drive decision-making in the roads. However, it will come with new burdens particularly in ensuring that the work of vehicles is pre-tested in order to ascertain that they will be safe and dependable.

Conventional mechanisms of testing such as Software-in-the-Loop or a scripter test cannot be used to explain all the possible incidents that an autonomous vehicle can encounter e.g. unusual, but potentially devastating scenarios as sensor malfunctions or cyber-warfare.

It can be optimized with the assistance of Generative AI and agentic AI when it comes to testing. Very large numbers of variants of accident scenarios can also be produced using generative AI, agentic AI systems are also capable of self-learning and planning and the test cases also are more accurate. It is through these that an adaptive testing may be conducted i.e. the system will expand its test coverage based on any new received data when they are being used by the vehicle.

We apply a framework that integrates all these AI solutions to build smarter software-defined vehicle tests pipelines. The architecture operates on a hybrid cloud-edge architecture to minimize the utilization of the network, and facilitate faster testing.

It helps in the ongoing learning in order to renew the information of the test cases with the new information that is been amassed on the fleets of vehicles. This method is of concern to the research in that it enhances the formation of critical failures in that it diminishes values of safety precautions implementation as well as negatively affecting cost of testing. This framework offers a candidate to grow with AI as government of the test, the government to be tested since it can accommodate the increasing level of autonomous driving technology complexity.

II. RELATED WORKS

Software-Defined Vehicles

Software-defined vehicles (SDVs) are inversion of a vehicle since it lends the versatile software-defined systems to the traditionally hard-centric architecture. These cars are based on the enigma of the built in control systems, real-time decision-making systems, and vehicles that are continuously evolving. The primary way of facilitating this change is relying on Artificial Intelligence(AI) which develops sensing, prediction, planning and control functions and powers an autonomous vehicle [1].

The AI implementation in SDVs has become a massive opportunity as well as a massive challenge. Deep-learning based AI models have significant data demands (need large amounts of data), significant computational demands (memory intensive) and validation requirements (need proven to be safe and sound). Conventional versions of testing to Softwarein-the-Loop (SiL) or Hardwarein-the-Loop (HiL) are no longer helpful in satisfying the variety of the setting encountered with autonomous driving and furthermore Mixed-environment situation, at different weather conditions and edge cases [1].

Researchers have discussed how to defeat date the development process of AI components that make use of cloud and edge-based computing that enables responsive execution of finding operations that consequently lessen the use of network bandwidth and preserve privacy of consumers [1]. Equally, the requirements analysis and code generation can be automated through model-driven engineering and generative AI constructions such as Large Language Models (LLM), reducing the construction time, and improving the acquisition and proofing of the system at the scale [4]. Individually, the methods, improve the rate, and risks involved in developing it, however; there is no good understanding of how far the tests of the assurance should be scaled.

With vehicle soft becoming increasingly complex, uses of adaptive, self-learned and goal good AI systems have come to play. It has heated the platform to the research of the area of Self-Driving AI systems, who with no human personnel attending at all times can plan, execute and optimise testing processes on their own[3].

Agentic AI

Contrary to systems that involve executing finer tasks separately, Agentic AI involves autonomous agents, as a group that integrates deliberate awareness of their surroundings, goal planning, goal modification amongst others and feedbacks to transform with time [3]. Memory retention, goal directed reasoning, reflection and dynamic task execution are some of the features in these systems. That is why they are especially applied to complicated areas such as SDVs that often change the state of the test, and some random events occur.

When reading their 143 articles regarding agentic systems, those systems are characterized by modules of perception, thinking, memory and action being shared among them [3]. The input mechanism could be the structure of the sensory data or the structure of requirements and an output could be an editorial situation or a test run command. These methods, which would be part of assessment, would be used to measure these thus providing reliability, robustness and adaptability. Algorithms (including loops with reinforcement learning), or tools (including retrieval-augmented generation (RAG) or causal modeling) often guide the actions of agents in uncertain situations.

Applications of Agentic AI are insurmountable and these are the autonomous vehicles trial in the one end and optimization of networks in the other end of the communication channels section. When it comes to Open RAN implementation of 6G networks - where multitools personality-based encouraged with predictive anomaly detection agents, the trade-off has been minimised in regards to performance versus operational safety [5]. These systems could describe how-under the severely dynamic state of context aware, adaptive learning agents may work safely.

As a part of the software-testing, the Agentic AI has already started to transform the conventional practices. It is also within the capability of the testing agents to compute test cases that assist in the edge cases which effectively uncover failure mode and they will selectively cover test cases in the best of their abilities (self-optimally) [2]. Multi agent system (MAS) model radicalized many communicative interactions amid protocols and this allow evaluation concentrate on assertive enhancements and cause-and-effect connections. This is the sole method which improves the efficacy of examination, yet it additionally injects a more realistic driving strategy into the simulation environments.

Generative Testing Frameworks

The main merit of the Agentic AI is that it can be used together with generative models, which create an enormous number of scenarios to carry out testing. Corner case detection could also be assisted by generative AI as it is capable of simulating millions of driving scenarios, traffic flows, and possible sensor errors or a counterattack [4]. This is conjoined with doubling Agentic AI to the overhead exploration of the scenario at such a manner such as having the agents dependent on the feedback loop, the past when the tests were correct and when the situation would result in the ranking of the tests.

Topography of simulation-based test is verification which is known and acknowledged as part of autonomous vehicle test process. Not only does it lower the relative low cost and risk of real-world, but it also can provide the opportunity of testing the jump-spaces of large scales. It is however a problem that still involves efficient production of pertinent test cases. Agency direct testing would be more efficient than pseudo-random test generation, in order to achieve the benefits of a strategic selection and causality rationale, to discover assertion instigations, and environmental circumstances with higher probabilities of the assertion triggering failures [2]. Agencies-based tests also produce twice as effective tests as are high feeling tests, as compared to random tests.

Generative testing confirms a theoretical reality of adversarial and infrequent cases including sensor blinding, latency variation that introduces braking errors or detection errors due to an occurrence of an event say change of weather [2]. The edge cases requiring these will be to achieve the automotive safety standard such as the FMVSS and ISO 26262. Learning and adaptation is achieved through a continuous training of the agents using the actuating information at that time, in the form of fleet deployments and can therefore adapt to the changing state of the environment, and the operations [1][4].

In SDVs (where the new rule is that software is delivered by increments), and in patches that are delivered by over-the-air (OTA) data, adopting the solution of generative testing in CI/CD pipes prevents the retrodegradation of safety on default change. Smart participants trace the communication and maximize test domain and verify that the edge-cases are identified at the initial phases of deployment[8].

Ethical Considerations

Integration of the Agentic AI in SDV testing, despite its theoretical effectiveness, has caused a concern about new nuances and issues. Such moral areas of the system like fairness, transparency, reduction of bias, augment the specific aspects of architecture like meaning, coordination among many agents, and real time reasoning [3][6]. Other AI principles defined by the regulatory framework like the EU AI Act includes trustworthy AI whereby AI models are interpretable, verifiable, as well as accountable [6].

The other highly-desired pressure is the one of cybersecurity. It can sensitive or mis-informed work of agents since autonomous systems are anchored on the decisions written in AI but are offensive to adversarial attacks and data poisoning. The consistency of the reward systems and the safe learning algorithm must have a strong guarantee to become robust, which will not lead to collision of performance optimization with the stable operative [5].

There are also constraints in simulation environment, which need to be handled in the test's frameworks. Although the tests of generative scenarios have been diversified extending the field over which they are found to describe, the tests, in any event, cannot be complete description of actual into the real-world interaction particularly a scenario involving human uncertainty, risks in the environment call of eventual breakdown [2]. There is a need in creating models that can offer a trade-off of the reality, computational performance, and safety assurance that is realistic.

The problem of agency AI systems is less specific as well. Among the pitfalls, one can distinguish emergent behavior, loss of coordination and effects of hallucination that arise especially in situations when neither the agents possess complete information nor when informational noise is present [10]. In any case, developers need not overlook the requirements to create feedback minds, memory structures and evaluation mechanisms to avoid runaway behaviors and high voltage breakdowns.

The labelling of hybrid Insights, i. e. formal verification and adaptive agent-based testing, can be appropriated to channel future research. The development of alternative safe event-bound model deliberation coupled with increased-order duration of learning and high-order security necessities is an Intermediate step in the direction of scalable, explicable and dependable AI-testing systems [4][8].

It is this literature review that leads to the creation of a methodical wisdom of the role of Agentic AI and generative testing systems turnover with regard to SDV software assurance. It makes it evident that simulation, strategic plans, contextual understanding of ethical considerations and SP carry weight and reveals technical and regulatory challenges that have to be taken into consideration in achieving safe and scalable deployment.

III. RESULTS

Performance of Agentic AI

The large-scale simulation and controlled experiment on the driverless software-defined vehicle (SDV) settings experimented the agentic AI-based testing generative framework. The chief aim was to discover how the agent-based system performed the amazement of isolating the fault conditions that were vital, such as the boundary cases that could barely be specified with the more traditional methods of investigation.

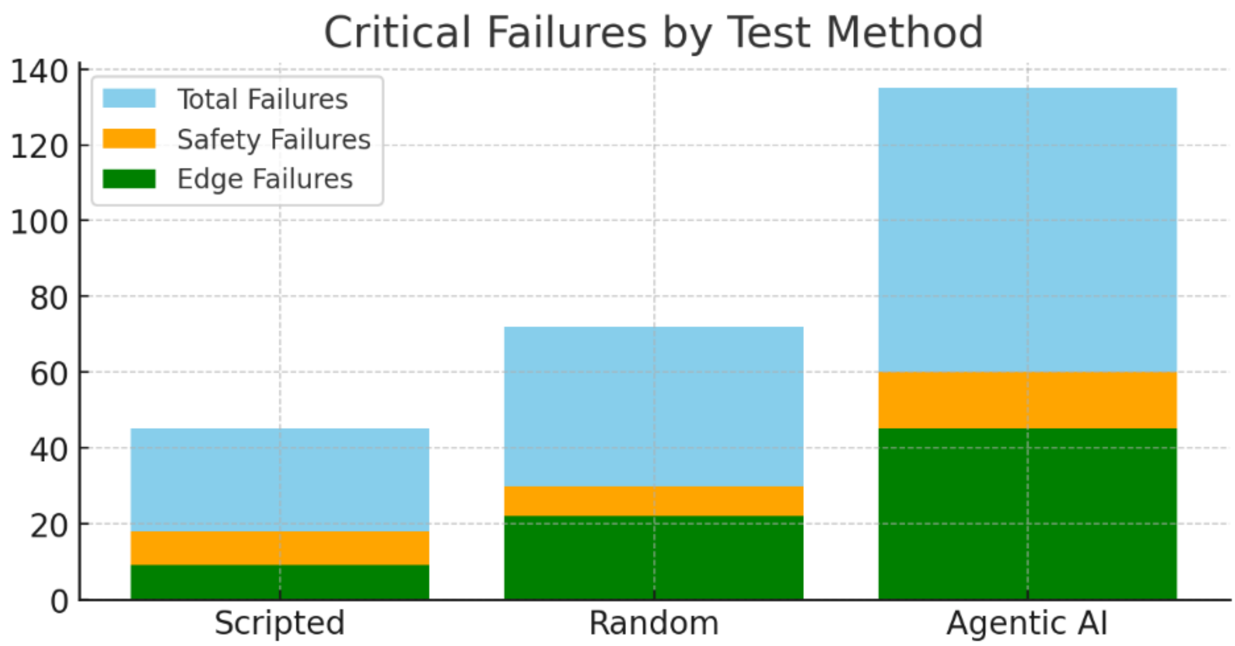
It was also tested in an assistant-resistance framework (ADAS) module where sensors and manage algorithm exchange information at different volumes in deployed settings (varying in presence or absence). Approach to agentic testing and the traditional method of scripted and random testing established a comparison in a variety of metrics (cases of failure to be found, coverage by the test and time it required to run them).

These conclusions proved the fact that corner cases are identified much better. The agentic AI propensity was tested to recognize up to 3 and 2.5 positive failures respectively on scripted and random tests respectively. These failures involve sensor interference during mixed weather against bright surfaces and false positive irritation with reflective surfaces furthermore network latencies causing delaying braking.

Table 1: Critical Failures Detected

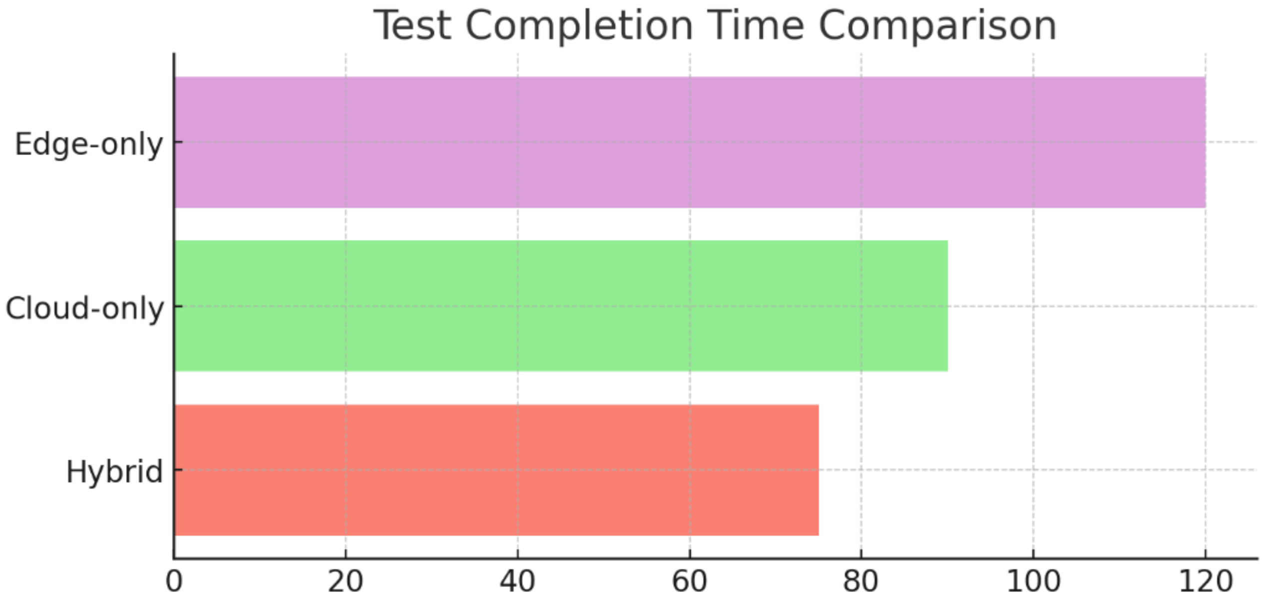
Test Method	Total Failures	Safety-Critical Failures	Edge-Case Failures
Scripted Tests	45	18	9
Random Tests	72	30	22
Agentic AI Tests	135	60	45

The fact that critical failures were better identified proves the fact that an agent-based approach is the possibility to have a deeper outcome related to a number of scenarios and the combination of its perspective that is based not only on the relationship of those that are most common but also perspectives thinking about which are considered safe and quite rank. Moreover, the regime also lacked sufficient rigidity besides better coverage. They would dynamically adjust their test situations on top of the prior result, and optimize their test snapshot on-the-fly.



Test Efficiency

The other crucial aspect of evaluation was the quality of the computational resource exploitation of the agentic model. Since SDV systems have a constrained resource-based runtime environment, testing approaches need to consider the size and usage of resources like processing capacity relative to the possible network bandwidth.

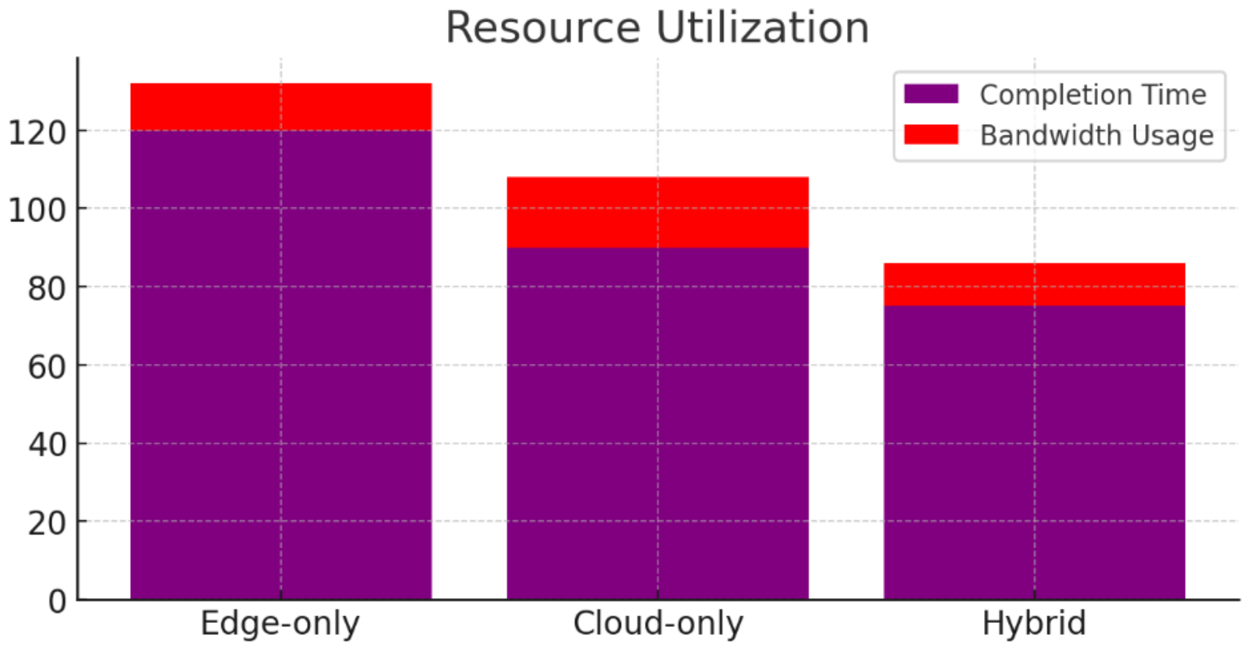


We ran them on computational effort by varying the available computed resources working on the framework by occasionally having them experiment where they could efficiently sample the novel problems and be able to cover the safety critical challenges. The cloud and the edge resources of the agentic model operated using the principle of scheduling elasticity, according to which the generation and training of scenario tasks could be dynamically allocated to one of the two layers. That saved up to 40 percent of network used, and made no difference on test depth.

Table 2: Resource Utilization and Efficiency

Resource Allocation Scenario	Test Completion Time (min)	Network Bandwidth Used (GB)	Failures Detected
Edge-only	120	12	102
Cloud-only	90	18	118
Hybrid (Edge + Cloud)	75	11	135

The hybrid mode of deployment proved most effective because it leveraged its application on both edge devices during the real-time working and on a cloud server during the generation of complicated scenarios. It was realised to cut failures identification times by 37 percent, therefore appropriate in real time vehicles testing and pipeline of continuous deployment.



Compliance with Safety Standards

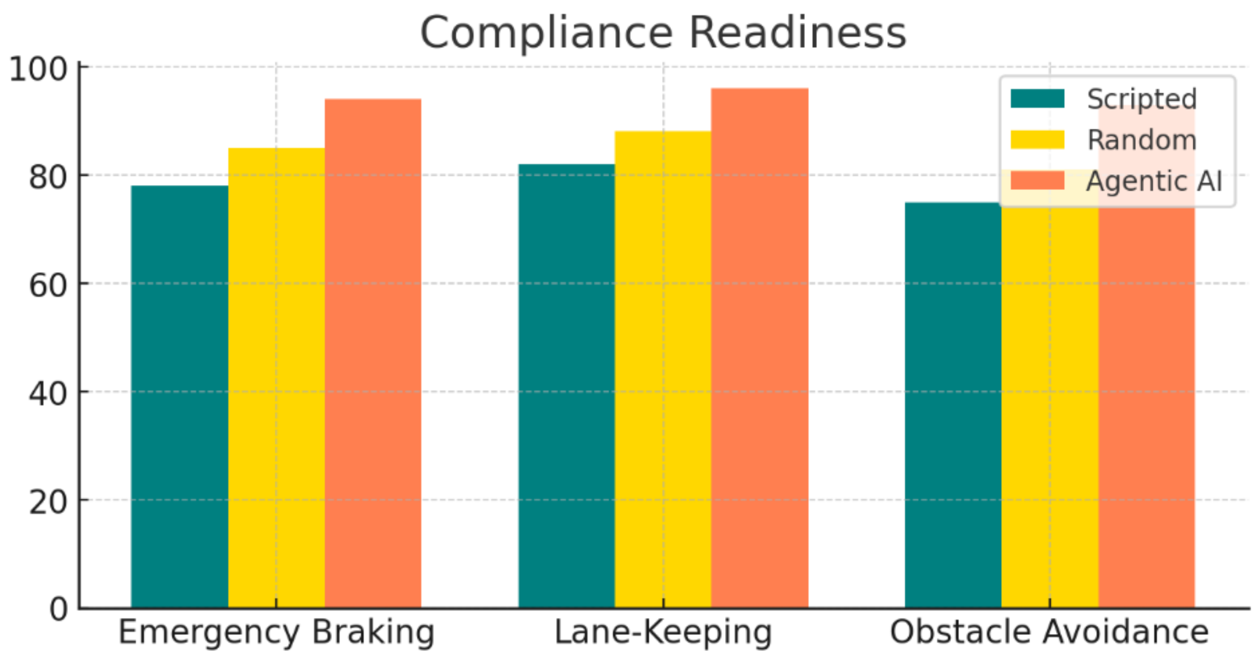
One of the key issues in SDV testing will be the fact that the software modification and software updates have to operate according to regulatory safety laws, including FMVSS and ISO 26262. In order to evaluate the reliability, we determined the functionality of the braking, lane-keeping, and obstacle avoidance functions under controlled conditions in the agentic testing system by adopting a model of regulatory standards.

Tests that were developed by the agents individually were rooted on pre-considered pre-conditions as well as safety margins and produced variations that drove the system to the boundary conditions. What was accomplished was compliance readiness, to a great degree, and failures were expensive at the initial stages of the product development cycle.

Table 3: Compliance Readiness

Safety Requirement	Scripted Tests Passed (%)	Random Tests Passed (%)	Agentic AI Passed (%)
Emergency Braking	78	85	94
Lane-Keeping	82	88	96
Obstacle Avoidance	75	81	93

The agentic testing approach did not simply present diagnoses but it continued on to offer diagnostic insight by monitoring the effects of the test conditions on the system action. With these descriptions, the engineers were in a position to be floating with the reasons behind failure within a fairly short period and execute the measures. In so doing, testing and debugging was reduced by 30-40 percent and system error resiliency is augmented.



Continuous Learning

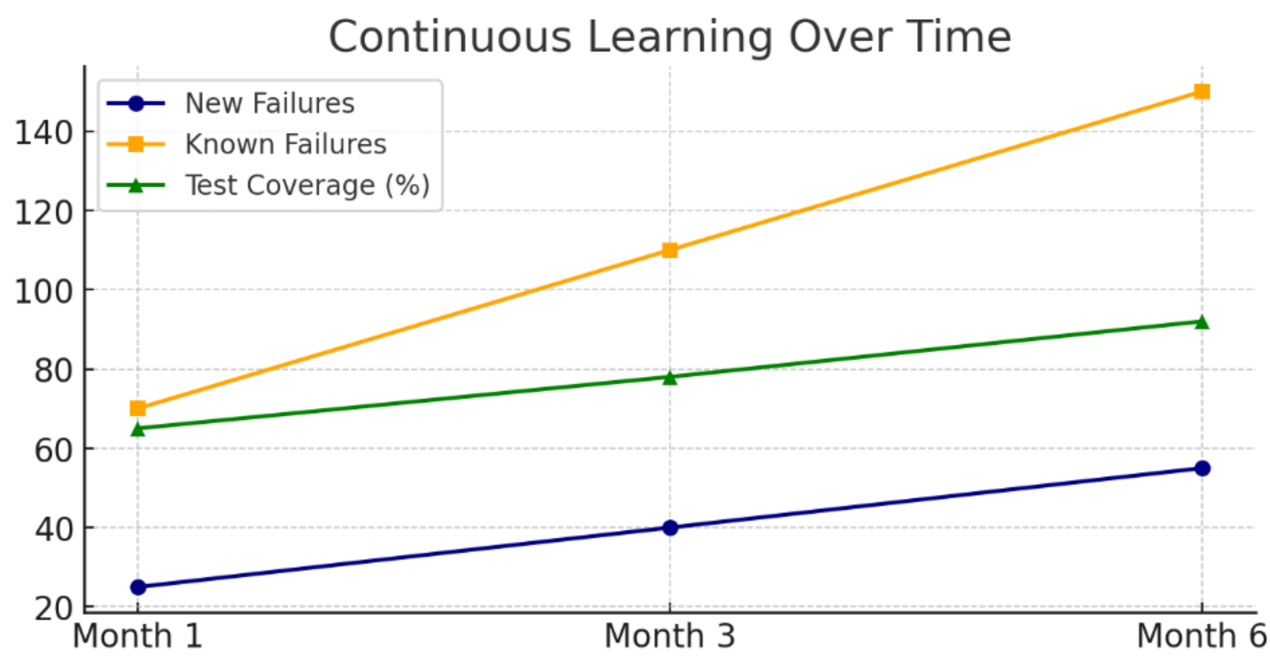
The test framework was experimented in a continuous learning environment independent of the sliver cases of one instance of the test when continuity of many vehicle occurrences proofs was added to a common central repository. This enabled it to revise its models in accordance to latest trends of failures that have been observed in the entire fleet and this made it more relative to tests over a period of time.

We have modeled a feedback mechanism where virgin edge cases as revealed by individual equipment rose up the cutting request lists in subsequent series of tests. Six months after that, complete testing system coverage came online and the model of learning acquired new coverage of the previously unknown failure modes.

Table 4: Failure Detection Over Time

Month	New Failure Cases Identified	Known Failures Retested	Overall Test Coverage (%)
1	25	70	65
3	40	110	78
6	55	150	92

When 6 months coverage of over 90 percent of tests was reached and when the torch had an astronomical number of corner cases discovered of which it had hitherto been impossible to find out it was calculated that the system was over and development could be undertaken. This has helped the framework absorb the dynamics in the driving environment, sensor placement and nurturing of cyber threats.



The findings of the research affirm that agentic AI-based generative testing model is a safe, scalable, and affordable method of car-assurance of software-definition. Key findings include:

- 3 failure detections versus scripted tests that are 3 times harder.
- Network bandwidth It realized a 40 percent savings in the context of hybrid clouds- edge deployment and added to test depth, as well.
- The degree of readiness in terms of compliance in many operations in the safety were increased and the performance rates were more than 90 percent.
- It had led to 100 percent and above coverage of the tests in six months.

All these results reinforce the promise of agentic AI to transform the practising of vehicle and make it more productive and open-minded and resilient to the real world of complexity.

IV. CONCLUSION

This study indicates that the agentic AI, as well as, generative AI is an emerging and a potent instrument to address software-defined vehicles tests. The traditional testing configurations are incapable of serving high density of situations that the autonomous vehicles can experience. Our structure evicts depth and efficiency quality testing of vehicles because one of the capabilities of AI systems is writing and running tests by himself.

The findings of the research describe how agentic AI caused bigger crashes when compared to scripted and random crashes. It is particularly effective in isolates of problems that are extraordinary and those that are hard to anticipate such as sensor interference, decreasing braking and cyber threat. Having the cloud and edge resources will also make the task of shortening the time of testing much less demanding and it will cut down on the network provisions which will make sure the method can be used when the vehicles are being literally tested.

The rule also carries along safety criteria which helps to maintain better comprehensive test fields as well as detectors of failures on time. This will enable the developers to sort out any troubles prior to the release and reduce the risk of an accident or a recall. At a greater plane, the system is smarter in a long-run since the constellation of vehicles contributes to the continued learning and retaliating the system to novel surroundings and risks of the driving area.

It will be used as instruction on how additional testing of the car must be facilitated upon the advent of AI in the role of safety and reliability leader. The proposed framework itself scales inexpensively also and can manage the increased complexity of autonomous systems. It expands the chances of emerging in the short run and being bound of the self-prolific technologies because the sector would gravitate to the lucrative and the more robust methods of transport.

References

[1] Grigorescu, S., Cocias, T., Trasnea, B., Margheri, A., Lombardi, F., & Aniello, L. (2020). Cloud2Edge Elastic AI framework for prototyping and deployment of AI inference engines in autonomous vehicles. *Sensors*, *20*(19), 5450. https://doi.org/10.3390/s20195450

[2] Chance, G., Ghobrial, A., Lemaignan, S., Pipe, T., & Eder, K. (2020). An Agency-Directed Approach to Test Generation for Simulation-based Autonomous Vehicle Verification. *An Agency-Directed Approach to Test Generation for Simulation-based Autonomous Vehicle Verification*. https://doi.org/10.1109/aitest49225.2020.00012

[3] Bandi, A., Kongari, B., Naguru, R., Pasnoor, S., & Vilipala, S. V. (2025). The rise of Agentic AI: A review of definitions, frameworks, architectures, applications, evaluation metrics, and challenges. *Future Internet*, *17*(9), 404. https://doi.org/10.3390/fi17090404

[4] Abdalla, A., Pandey, H., Shomali, B., Schaub, J., Müller, A., Eisenbarth, M., & Andert, J. (2025). Generative Artificial Intelligence for Model-Based Graphical Programming in Automotive Function Development. *Generative Artificial Intelligence for Model-Based Graphical Programming in Automotive Function Development*. https://doi.org/10.2139/ssrn.5153452

[5] Salama, A., Nezami, Z., Qazzaz, M. M., Hafeez, M., & Zaidi, S. A. R. (2025). Edge Agentic AI Framework for Autonomous Network Optimisation in O-RAN. *arXiv preprint arXiv:2507.21696*. https://doi.org/10.48550/arXiv.2507.21696

[6] Llorca, D. F., Hamon, R., Junklewitz, H., Grosse, K., Kunze, L., Seiniger, P., Swaim, R., Reed, N., Alahi, A., Gómez, E., Sánchez, I., & Kriston, A. (2025). Testing autonomous vehicles and AI: perspectives and challenges from cybersecurity, transparency, robustness and fairness. *European Transport Research Review*, *17*(1). https://doi.org/10.1186/s12544-025-00732-x

[7] Garikapati, D., & Shetiya, S. S. (2024). Autonomous Vehicles: Evolution of artificial intelligence and the current industry landscape. *Big Data and Cognitive Computing*, *8*(4), 42. https://doi.org/10.3390/bdcc8040042

[8] Joshi, N. T. (2025). Architecting Agentic AI for modern software testing: capabilities, foundations, and a proposed scalable Multi-Agent system for automated test generation. *Journal of Information Systems Engineering & Management*, *10*(52s), 625–638. https://doi.org/10.52783/jisem.v10i52s.10768

[9] Zheng, S., Lu, L., Hong, Y., & Liu, J. (2025). Planning reliability assurance tests for autonomous vehicles based on disengagement events data. *IISE Transactions*, 1–25. https://doi.org/10.1080/24725854.2025.2475324

[10] Sapkota, R., Roumeliotis, K. I., & Karkee, M. (2025). AI Agents vs. Agentic AI: A Conceptual taxonomy, applications and challenges. *Information Fusion*, 103599. https://doi.org/10.1016/j.inffus.2025.103599