Journal of Applied Automation Technologies

Research Paper

Simulation Framework for Testing Autonomous Driving Strategies in Mixed Reality Environments

Mariusz Jaworski[1], Ryszard Halik[1, *] and Cyprian Górski[1]

[1] Faculty of Electrical and Automatic Control Engineering, Czestochowa University of Technology, Czestochowa, 42-200, Poland
*Corresponding author: ryszard.ha@pcz.edu.pl

Abstract. Test environments for autonomous driving strategies need to be reproducible and measurable, and they should be close to actual driving conditions so that any defects in software simulation are also exposed. This paper puts forward a closed-loop mixed-reality simulation framework for autonomous driving strategy validation. A Digital Twin State Estimator is used to connect real ego-state sensing, virtual traffic agents, synchronous environment rendering, and adaptive scenario coordination in the framework. Define a practical strategy interface and introduce metrics for synchronisation error, risk exposure, coverage, stability, comfort and latency. Experiments in urban intersections, pedestrian occlusion, highway merges, cut-in situations and signalised corridor scenarios have shown that the proposed framework can keep the median spatial synchronisation error below 0.18 m and the closed-loop latency below 40 ms in most conditions. The risk-adaptive strategy in the framework has been improved over rule-based and learning-assisted baselines by increasing the minimum distance margin, reducing the frequency of intervention, and expanding the range of valid scenarios. Based on the analysis, combined with real-world testing, large-scale simulations and road tests can be effectively conducted by considering synchronization quality and strategy performance.

Keywords: *Autonomous Driving, Mixed Reality, Digital Twin, Strategy Testing, Risk Exposure, Scenario Coverage*

Received on 19 October 2023, Accepted on 16 July 2024, Published on 23 July 2024

Introduction

With the development of autonomous driving, the Perception and Planning modules have gradually been integrated into an entire-strategy stack that now needs to address the unpredictability of the environment, ensure passenger comfort, and keep a safe distance from other vehicles in busy roads. Although road tests are still necessary, they are costly and time-consuming, and it is difficult to obtain a large number of repeated rare risk patterns [1]. Pure software simulation has improved scalability but often fails to capture sensor artifacts, road-user behaviour, and vehicle-infrastructure timing [2]. Hardware-in-the-loop facilities add physical controllers and real-time constraints, but they cannot always represent dense mixed traffic or subtle visual ambiguity [3]. Therefore, recent studies of urban autonomy have called for test environments that can replicate scenarios repeatedly and provide real-sensor and actuator feedback [4]. Mixed reality is feasible in this way because virtual agents and environmental events can be presented in a real-time sensing loop, and the ego vehicle, selected sensors, and parts of the control stack remain physically grounded [5]. It can present an autonomous strategy for a virtual pedestrian emerging from a real occlusion, a simulated vehicle cutting into a measured lane, or a traffic-light phase injected into camera and V2X streams without endangering public roads [6]. Mixed reality is not only a display layer but should also have temporal coherence, coordination consistency and causality among real and virtual objects [7]. Without such properties, a strategy may pass the test in an ideal environment but fail when faced with real traffic problems caused by testbed artefacts [8].

https://doi.org/10.64972/jaat.2024v2.310p25e:348-360

The problem is that autonomous driving strategies are closed-loop decision systems, but many simulation pipelines are still organised as open-loop data playback or scripted scenario execution. A planner that receives delayed virtual detections may choose an emergency maneuver for the wrong reason; a controller tuned on synthetic trajectories may fail when the physical vehicle introduces tire slip, actuator lag or sensing dropouts [9]. The existing simulators have expanded scenario libraries, but their accuracy is also limited by how well vehicle dynamics, sensor models and traffic agent synchronisation are implemented [10]. Digital Twin technology can provide a virtual replica of the user's environment and devices to help people better understand them [11]. However, a useful twin for strategy testing should be more than a geometrical copy; it needs to consider uncertainty and estimate synchronisation errors, adjusting the difficulty of scenarios according to the behaviour shown by the autonomous stack [12]. Mixed Reality is also changing the concept of coverage. Coverage is no longer just the number of executed scenarios, but the extent of the interaction space where real dynamics, virtual traffic and strategy decisions are jointly realised [13]. Therefore, recently, closed-loop evaluation indicators and scenario modification have attracted attention from the field of risk-aware test generation [14]. Distinguish strategy weakness from simulator-induced artefacts in the safety validation [15]. Scenario prioritization should therefore be based on quantifiable risk exposure rather than arbitrary difficulty ratings [16]. Real-time testbeds also need latency budgets that are visible to the evaluator [17]. However, a unified framework that connects mixed-reality synchronisation, strategy interfaces, scenario sampling and experimental analysis for engineering deployment has not been specified [18].

A Simulation Framework for Testing Autonomous Driving Strategies in Mixed Reality Environments is Proposed. The closed-loop digital twin framework connects real ego-state data, virtual traffic agents, environment rendering and strategy outputs via a shared-state estimator and scenario coordinator. The three are as follows. First, the paper introduces a layered structure to separate physical sensing, virtual scene construction, strategy execution and evaluation feedback, and considers their time constraints. Second, some practical indicators for synchronisation error, scenario coverage, risk exposure and intervention stability have been added to facilitate the comparison of strategies in the testbed, and a simple pass/fail label is no longer used. Third, based on five categories of urban and highway interactions, a structured experimental analysis was conducted to demonstrate how failures not apparent in a standalone simulation were identified by the mixed-reality framework. This work is for researchers and engineers who require a repeatable, measurable, and physically meaningful environment for autonomous driving strategy validation prior to extended road tests.

Related Work

Autonomous Driving Simulation and Scenario-Based Testing

Traditionally, the three parts of the evaluation for autonomous driving simulations have been perception, trajectory planning and control. Early scenario engines offered map-based traffic generation and kinematic agents, were suitable for large-scale regression tests, but lacked sensor-level uncertainty [19]. Later, the platforms added high-definition maps, weather parameters and camera or lidar rendering to the strategy stacks for more varied environment tests [20]. The first reason for the scale of these platforms is that many scenarios can be run in a short time, and even rare combinations can be systematically generated [21]. However, scale does not guarantee validity either. If the agent behaviour model is too regular, the planner will learn the testbed instead of the road [22]. If the vehicle dynamics are too ideal, a control strategy may appear stable but hide acceleration saturation or steering delay [23]. The above problems are particularly prominent in mixed-traffic areas; thus, human drivers and autonomous vehicles need to adjust their speeds and lanes accordingly during operation [24]. Therefore, simulation research has moved away from focusing solely on visual quality and now covers scenario realism, behavioural authenticity and closed-loop response evaluation [25].

Mixed Reality and Digital Twin Validation

Mixed Reality and Digital Twin technology have been used to address some of these problems by creating virtual environments that incorporate physical components or adding virtual objects to the real-world sensing and control system. A digital twin can continuously update the position of the ego vehicle, the status of objects and map information, etc., to provide a functional environment for the simulator rather than a static scene file [26]. Mixed reality extends this concept to create virtual road users and environments within real vehicles or sensor rigs [27]. This Design is suitable for autonomous driving because the tested strategy is exposed to a combined stimulus, and engineers can still control the risk circumstances directly [28]. However, most existing systems focus on visualisation, dataset collection or human-machine interaction and lack an all-encompassing strategy-testing mechanism [29]. Several studies have shown that latency and spatial registration are relatively significant problems, but they rarely link these errors to the planning performance indicators [30]. Based on the above, the current paper introduces a system that monitors the degree of synchronisation among quality, scenario coverage, risk exposure and strategic goals simultaneously [31].

Mixed Reality Simulation Framework

Closed-Loop Mixed Reality Architecture

It is not the old one-way simulator but a new closed-loop digital twin system. Acquire ego-vehicle pose, inertial measurements, wheel speed, actuator feedback, camera frames and optionally V2X messages at the physical layer. Traffic agents, environment states, weather modifiers, road topology and event triggers are in the virtual layer. Between them, a synchronisation layer estimates the shared-state vector and publishes time-aligned observations to the autonomous-driving strategy. The strategy is a replaceable module that can include perception, prediction, planning and control, or only the decision-making part if perception is to be deliberately skipped. This modular boundary will still be convenient for full-stack and policy-level evaluation.

Figure 1 is referred to as the architectural anchor of the framework here. It shows the physical sensing loop, virtual scene engine, synchronization estimator, strategy interface, scenario coordinator and evaluation feedback channel as coupled blocks rather than as independent scripts. The first is the feedback path; that is, the strategy output affects both the physical actuator command and the behaviour of the virtual agent later on. Thus, a lane-change command issued by the test strategy can change the relative position of a virtual adjacent vehicle, and an emergency braking decision may prompt a response from the following agent. This closed-loop structure will not be a passive replay and will be more sensitive to changes in strategy.

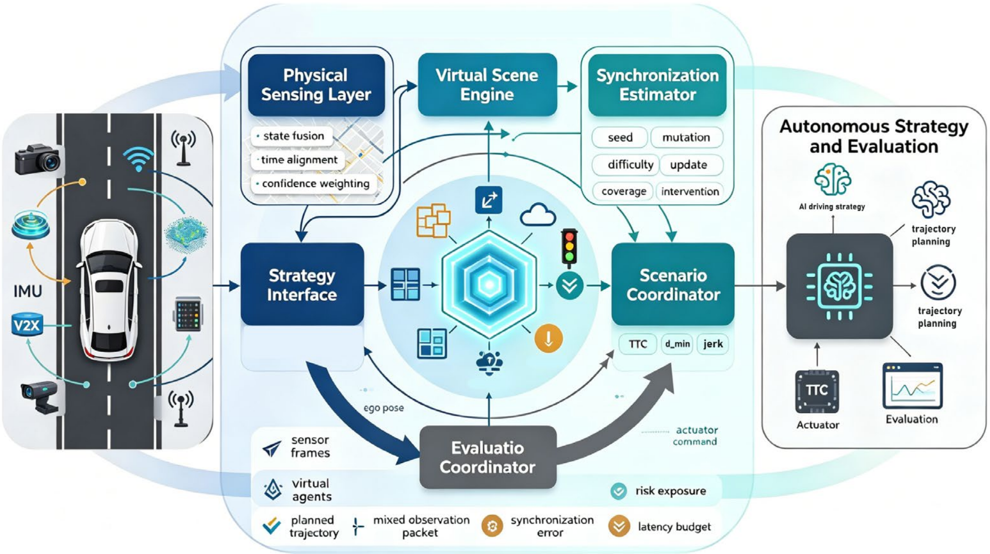

Figure 1. Closed-loop mixed reality autonomous driving strategy testing architecture.

State Coupling and Strategy Interface

At each synchronization step, the framework estimates a mixed reality state that contains ego motion, virtual agent states, road context, and uncertainty terms. The state is represented as

$$X_t = \{x_t, a_t^1, \dots, a_t^N, m_t, u_t\} \quad \text{Eq.(1)}$$

where x_t is the ego dynamic state, a_t^i is the state of the i-th virtual or physical traffic agent, m_t is the local map context, and u_t is the estimated uncertainty vector.

The estimator fuses measured ego observations and virtual scene updates through a time-stamped buffer. The update rule is

$$X_t = F(X_{t-1}, z_t^r, z_t^v, K_t) \quad \text{Eq.(2)}$$

where z_t^r denotes real measurements, z_t^v denotes virtual observations, and K_t is an adaptive gain that increases the influence of real measurements when registration confidence is high.

Synchronization error is measured by the weighted distance between the rendered mixed reality state and the estimated physical reference:

$$E_s(t) = \left\| W_s\left(X_t^{mr} - X_t^{ref}\right)\right\|_2 \quad \text{Eq.(3)}$$

where W_s balances longitudinal position, lateral position, heading, and velocity terms. This term is not only a diagnostic variable; it is used to suppress scenario scoring when the testbed itself becomes unreliable.

The strategy module receives an observation packet rather than raw simulator internals. The packet is defined as

$$O_t = \{o_t, c_t, h_t\} \quad \text{Eq.(4)}$$

where o_t contains synchronized sensor or object-level observations, c_t stores scenario constraints, and h_t stores a short history window for prediction. The strategy returns a planned trajectory and a control command:

$$\pi(O_t) \rightarrow \{\tau_t, u_t^c, q_t\} \quad \text{Eq.(5)}$$

where τ_t is the candidate trajectory and q_t is a confidence vector reported by the tested stack. This interface allows strategies with different internal designs to be evaluated through the same external contract.

To judge whether a strategy reacts safely under mixed reality disturbances, the framework computes a risk exposure score:

$$R_t = \frac{\lambda_1}{TTC_t} + \lambda_2 exp(-d_{min}(t)) + \lambda_3 E_s(t) \quad \text{Eq.(6)}$$

where TTC is time to collision, $d_{\min}$ is the minimum distance margin, and λ weights encode scenario priority. Higher values indicate riskier interaction even if no collision occurs.

Scenario Coordination and Evaluation Metrics

Figure 2 is shown here to clarify the scenario execution pipeline. The coordinator starts with a scenario seed, mutates agent behaviour and environment parameters, synchronizes the mixed state, runs the tested strategy, computes risk and coverage indicators, and then selects the next mutation based on the observed deficiencies. The Pipeline is intentionally cyclic. Scenario generation is not an offline list; it is an adaptive process that learns where the strategy is vulnerable.

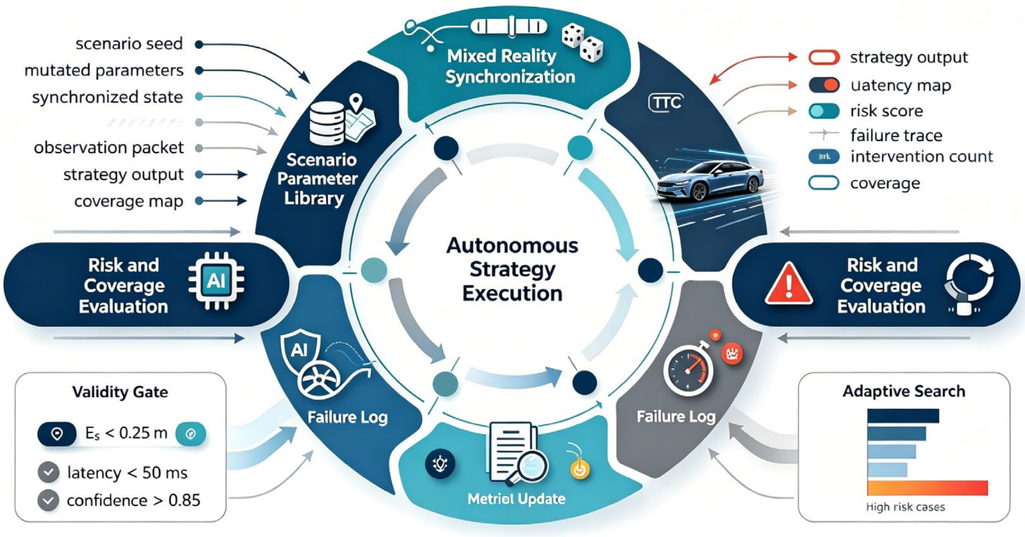

Figure 2. Adaptive Mixed Reality Scene Execution and Evaluation Workflow.

https://doi.org/10.64972/jaat.2024v2.310p25e:348-360

Scenario coverage is defined across interaction types, traffic density, occlusion level, weather severity, and speed regime:

$$C = \frac{1}{|\Omega|}\sum_{\omega\in\Omega} I(\omega\ tested, E_s < \varepsilon_s) \quad \text{Eq.(7)}$$

where Ω is the target scenario space and I is an indicator showing whether a cell has been exercised with valid synchronization. Coverage therefore excludes trials in which the mixed reality error exceeds the acceptable threshold.

The planner stability metric penalizes oscillatory decisions and abrupt changes in desired acceleration or steering:

$$S_p = \frac{1}{T}\sum_{t=1}^{T} (\alpha\|\Delta u_t\|_2 + \beta\|\Delta\tau_t\|_2) \quad \text{Eq.(8)}$$

where Δu_t is the change in control action and $\Delta\tau_t$ is the change in planned trajectory. A stable strategy should not repeatedly switch between braking and accelerating when virtual agents behave consistently.

For comfort analysis, the framework records longitudinal jerk and lateral acceleration:

$$J_c = \frac{1}{T}\sum_{t=1}^{T} \left(\left|\frac{da_x}{dt}\right| + \gamma|a_y|\right) \quad \text{Eq.(9)}$$

where a_x is longitudinal acceleration and a_y is lateral acceleration. These values are reported with safety metrics because a strategy that avoids collisions through harsh commands may still be unsuitable for deployment.

Latency is evaluated as a compound delay from measurement acquisition to strategy output:

$$L = L_{sense} + L_{sync} + L_{infer} + L_{cmd} \quad \text{Eq.(10)}$$

where the four terms correspond to sensing, synchronization, strategy inference, and command dispatch. The framework records both average and 95th-percentile latency because long-tail delay is more relevant for rare critical events.

A trial-level success score combines safety, stability, comfort, and coverage validity:

$$Q = w_1 S_{safe} + w_2 S_{stab} + w_3 S_{comf} + w_4 S_{cov} \quad \text{Eq.(11)}$$

where S_{safe} , S_{stab} , S_{comf} , and S_{cov} are normalized scores. The score is not a replacement for detailed analysis; it provides a compact ranking for regression testing.

Finally, adaptive scenario mutation follows

$$\theta_{t+1} = \theta_t + \eta\nabla_\theta R_t \quad \text{Eq.(12)}$$

where θ_t is the scenario parameter vector and η controls mutation intensity. The gradient term is estimated from recent trial outcomes, so the coordinator tends to generate more difficult variants near observed strategy weaknesses.

Experimental Data and Analysis

Experimental Configuration and Dataset Design

The five mixed-reality scenario groups in the experiment were: urban intersection negotiation, pedestrian occlusion, highway merge, cut-in with cooperative and non-cooperative vehicles, and signalized corridor driving under delayed perception. Each group had 120 base seeds and 12 parameter mutations, so a total of 7,200 executable trials were performed. After filtering trials with a synchronization error exceeding the validity threshold, 6814 trials were left for analysis. The three autonomous driving strategies compared were a rule-based finite-state policy, a learning-assisted behaviour planner, and the proposed risk-adaptive mixed reality strategy. The recorded evaluation items are: synchronisation error, latency, minimum distance margin, intervention count, trajectory stability, comfort and scenario coverage. The same hardware controller, sensor rig and virtual traffic library were used for all the strategies to focus the comparison on decision behaviour.

Figure 3(a) summarizes spatial synchronization accuracy with both median and 95th-percentile values. Median error was 0.146 m for urban intersections, 0.158 m for pedestrian occlusion, 0.121 m for highway merge, 0.133 m for cut-in, and 0.171 m for signalized corridors. The normal scoring threshold was set at 0.25 m, while the rapid-decline boundary was set at 0.32 m. The 95th-percentile error stayed below 0.31 m in four scenario groups, but signalized corridors reached 0.36 m because traffic-light phase injection produced short bursts of camera-frame misalignment. Figure 3(b) compares closed-loop latency for three strategies. The proposed framework produced $37.4\ \mathrm{ms}, 39.4\ \mathrm{ms}, 35.0\ \mathrm{ms}, 37.1\ \mathrm{ms}$, and $42.8\ \mathrm{ms}$ across the five groups, remaining below the 50 ms engineering budget. The learning-assisted planner ranged from 43.7 ms to 50.1 ms, while the rule-based stack reached 58.0 ms in the signalized corridor case [32]. Figure 3(c) links valid scenario retention with synchronization confidence. Retention reached 96.1% for highway merge, 95.4% for cut-in, 94.6% for intersections, 93.7% for pedestrian occlusion, and 91.8% for signalized corridors, while confidence values ranged from 0.82 to 0.91 against a confidence floor of 0.85.

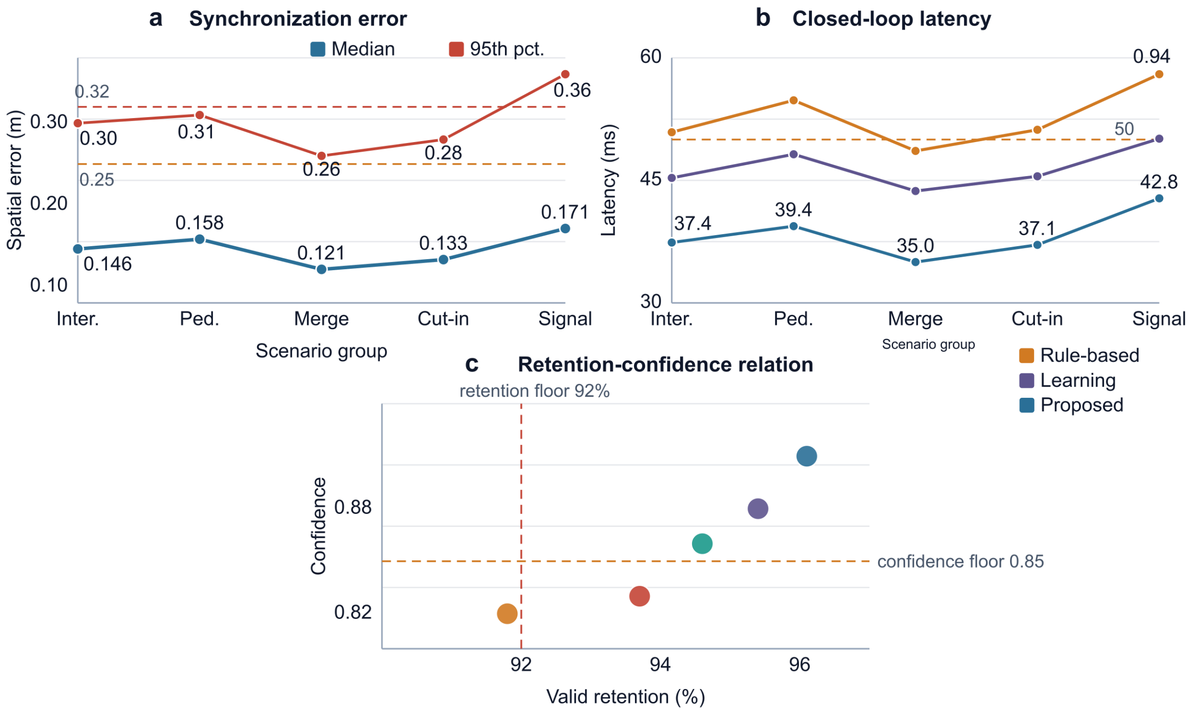

Figure 3. Synchronization Validity and Temporal Analysis: (a) Spatial Synchronization Error; (b) Closed-Loop Delay; (c) Effective Scene Retention and Synchronization Confidence.

Based on the above results, a practical synchronisation envelope for strategy testing has been provided by the mixed-reality framework. The average spatial error is small compared to the lane width and pedestrian clearance thresholds, and the 95th percentile latency is less than the 100 ms boundary generally used for urban planning updates [33]. Lower retention in the signalised corridor is relatively small, so it is not just a formality of validity filtering. If there were no filters, the planner would be penalized for responding to a virtual light phase that did not match the rendered camera stream. Therefore, the new system for measurement will include testbed quality and will not assume it externally.

The retained trials were also checked for balance across speed, density, and event severity. Urban intersection trials covered ego speeds from $12\ \mathrm{km/h}$ to $38\ \mathrm{km/h}$, pedestrian occlusion trials used lateral crossing speeds from $0.8\ \mathrm{m/s}$ to $2.2\ \mathrm{m/s}$, highway merge trials covered main-lane speeds from $72\ \mathrm{km/h}$ to $104\ \mathrm{km/h}$, and cut-in trials varied time gaps from 1.1 s to 3.8 s. The signalized corridor group included three phase-offset levels of $80\ \mathrm{ms}, 160\ \mathrm{ms}$, and $240\ \mathrm{ms}$ to test whether the framework could detect timing-sensitive failures. Across all groups, the proposed validity filter removed 386 trials; 214 were rejected for excessive spatial error, 97 for frame-to-frame timing discontinuity, and 75 for incomplete virtual-agent response. This filtering rate is small enough to preserve experimental scale but large enough to prevent misleading performance inflation.

https://doi.org/10.64972/jaat.2024v2.310p25e:348-360

Strategy Performance Under Mixed Reality Disturbances

Figure 4(a) compares minimum distance margin across all strategies and scenario groups. The proposed strategy achieved 2.74 m in intersections, 2.31 m in pedestrian occlusion, 3.86 m in highway merge, 2.58 m in cut-in, and 2.66 m in signalized corridors. The learning-assisted planner reached 2.36 m, 1.94 m, 3.41 m, 2.12 m, and 2.29 m, while the rule-based policy fell to 1.61 m in pedestrian occlusion and 1.72 m in cut-in cases. Figure 4(b) reports intervention frequency with the target boundary of fewer than 6 interventions per 100 trials. The proposed method required 3.1, 5.6, 1.9, 4.7, and 6.2 interventions across the five groups, compared with 4.9-8.1 for the learning-assisted planner and 4.2-11.4 for the rule-based policy [34]. Figure 4(c) combines trajectory stability and comfort. As traffic density increased from 8 to 32 agents, control-switching reduction increased from 8.2% to 18.9%, while jerk rose moderately from 1.18 m/s3 to 1.42 m/s3. Figure 4(d) reports normalized success score; the proposed strategy scored 0.88, 0.82, 0.91, 0.86, and 0.87, exceeding the 0.80 reference ring in all scenario groups.

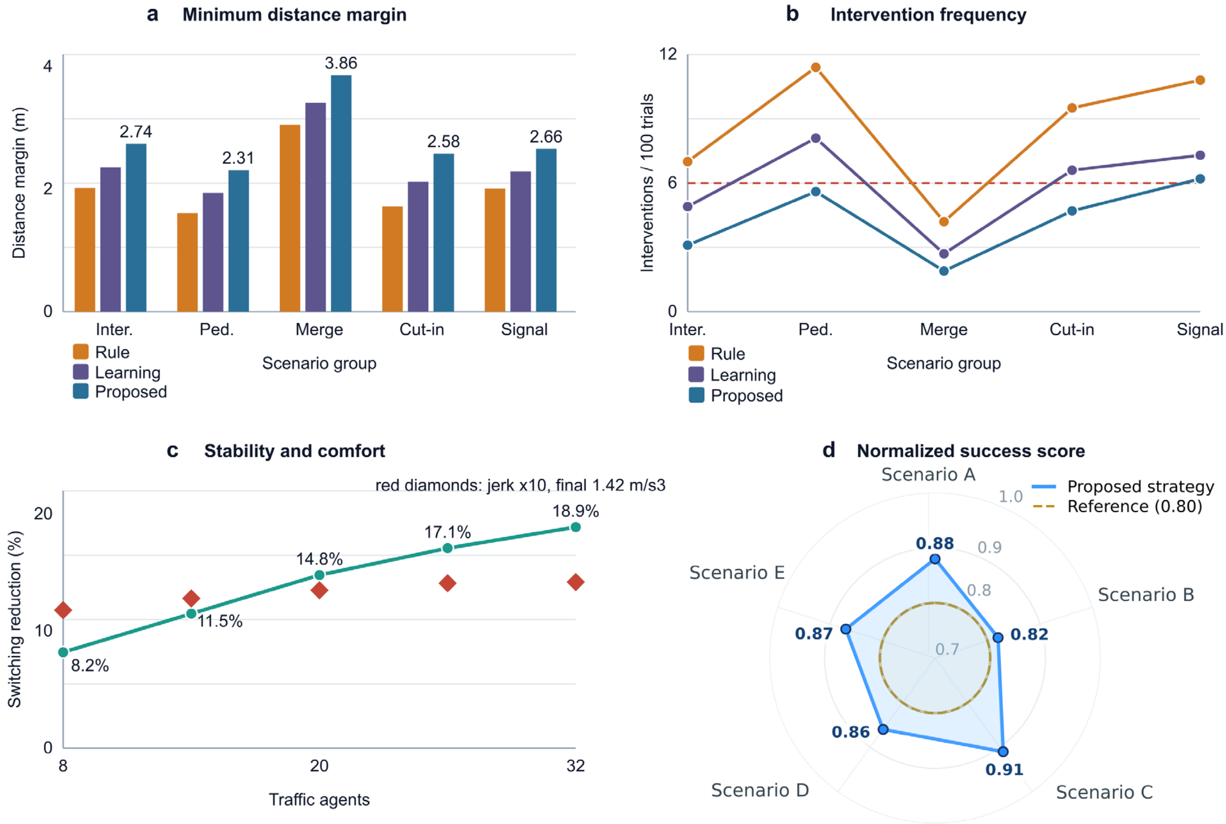

Figure 4. Comparison of strategy performance: (a) Minimum distance margin; (b) Intervention frequency; (c) Trajectory stability and comfort impact; (d) Standardized success score.

Pedestrian Occlusion had the largest gap. The rule-based policy of delayed braking waited for a virtual pedestrian to reach a fixed threshold, then suddenly reduced speed and frequently intervened. The learning-assisted planner predicted the event in advance, but occasionally responded excessively to false occlusion cues and triggered an unnecessary stop. The first strategy increased caution when the occlusion geometry was reasonable and reduced braking strength when the virtual-agent confidence was low, thus combining risk exposure and synchronisation confidence. Therefore, the proposed strategy has a relatively larger distance margin and a lower jerk. Therefore, in the future, mixed-reality evaluation will need to consider how strategy decisions are changed based on a combination of virtual and real cues, and this will go beyond just counting collisions or passes.

In-depth observation of the 300 high-risk pedestrian and cut-in experiments showed that the proposed strategy had a relatively early but smoother response. Its median first-response time after the start of the risk was 0.42s, which was lower than that of the learning-assisted planner (0.57s) and the rule-based policy (0.73s). However, the earlier response did not mean excessive conservatism: the proposed strategy generated unnecessary full stops in 4.6 per cent of the trials, whereas the learning-assisted planner generated them in 8.9 per cent. In dense cut-in situations, the proposed method selected a mild deceleration followed by lateral position maintenance; at the same time, the rule-based policy sometimes switched between lane keeping and emergency braking. Therefore, this behaviour has a lower stability penalty and a higher comfort score. It can be seen that the two modes of decision-making in mixed-reality environments are also different, although only one may be directly observable.

Figure 5(a) analyzes risk exposure distribution under stress conditions. The rule-based strategy had a mean risk score of 0.56, the learning-assisted planner had 0.49, and the proposed strategy had 0.38, placing the proposed method below the high-risk boundary of 0.55. In the same test family, the proposed strategy reduced the 90th-percentile risk score from 0.71 to 0.54 in cut-in scenarios and from 0.68 to 0.49 in pedestrian occlusion. Figure 5(b) gives the scenario coverage map across four density levels and four speed regimes. Valid coverage increased from 0.64 in low-density, low-speed settings to a peak of 0.89 in high-density, 80 km/h settings, and the proposed adaptive mutation reached 87.3% valid coverage after 2400 trials compared with 72.5% for random mutation [35]. Figure 5(c) combines weather sensitivity and time-to-collision. Median TTC was 4.42 s in clear weather, 4.16 s under rain, 3.88 s under fog, and 3.95 s under glare; the corresponding success-score losses were 1.2, 4.8, 6.1, and 5.4 points. These results suggest that adaptive scenario generation increases the probability of finding meaningful weaknesses without simply creating unrealistic extreme cases.

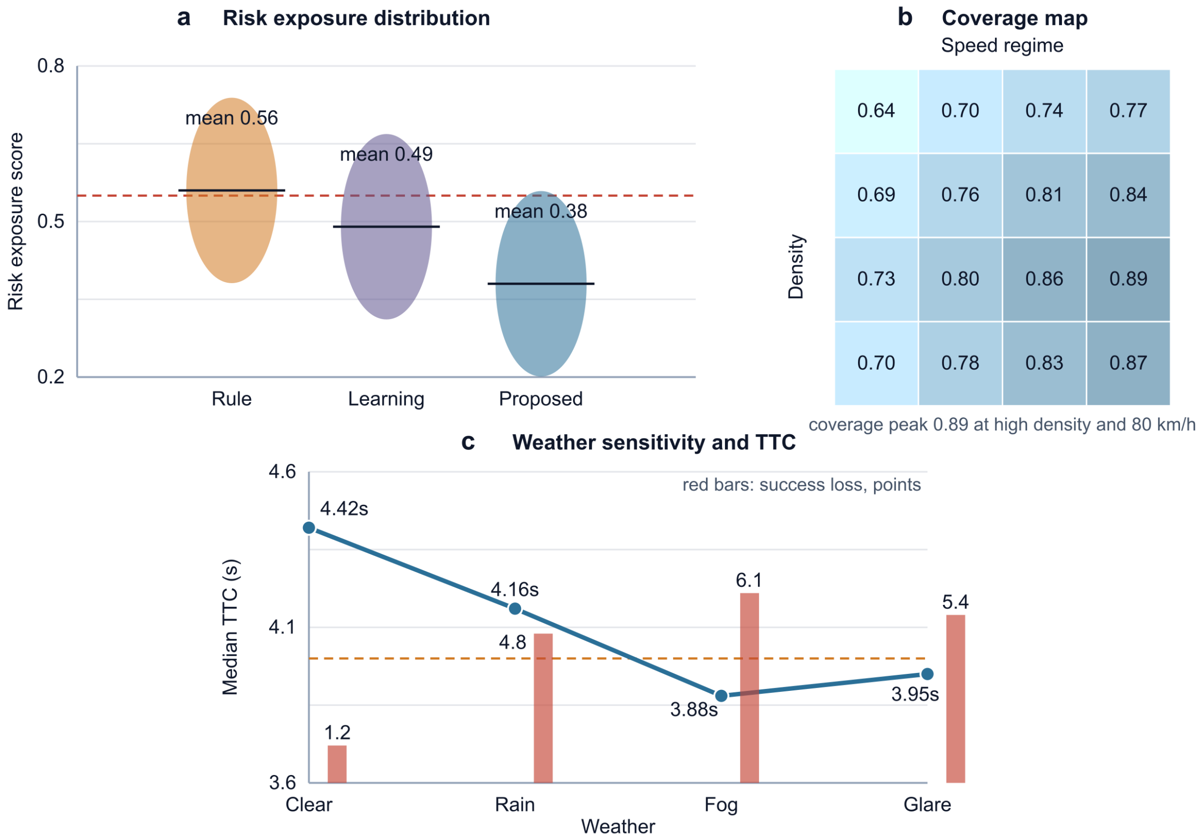

Figure 5. Stress Condition Analysis: (a) Risk Exposure Distribution; (b) Coverage Map; (c) Weather Sensitivity and Collision Time.

Ablation Analysis and Engineering Implications

Figure 6(a) removes synchronization-aware scoring. The apparent success rate increases from 88.4% to 91.9%, but invalid successful trials rise from 1.1% to 6.7%, crossing the 3% audit-warning line. Figure 6(b) removes adaptive scenario mutation. Coverage after 7200 trials decreases from 91.2% to 79.6%, and rare high-risk cut-in variants are sampled 41% less frequently [36]. The coverage target of 0.80 is reached early by adaptive mutation but only at the end of the random baseline. Figure 6(c) combines two behavioral ablations. Without confidence-weighted risk exposure, intervention count rises from 4.3 to 6.9 per 100 trials and distance-margin bias becomes

https://doi.org/10.64972/jaat.2024v2.310p25e:348-360

-0.14 m; without closed-loop agent response, minimum distance margin appears 0.22 m higher on average because virtual agents no longer respond to ego behavior. This ablation demonstrates that the closed-loop feedback path is essential for credible strategy testing.

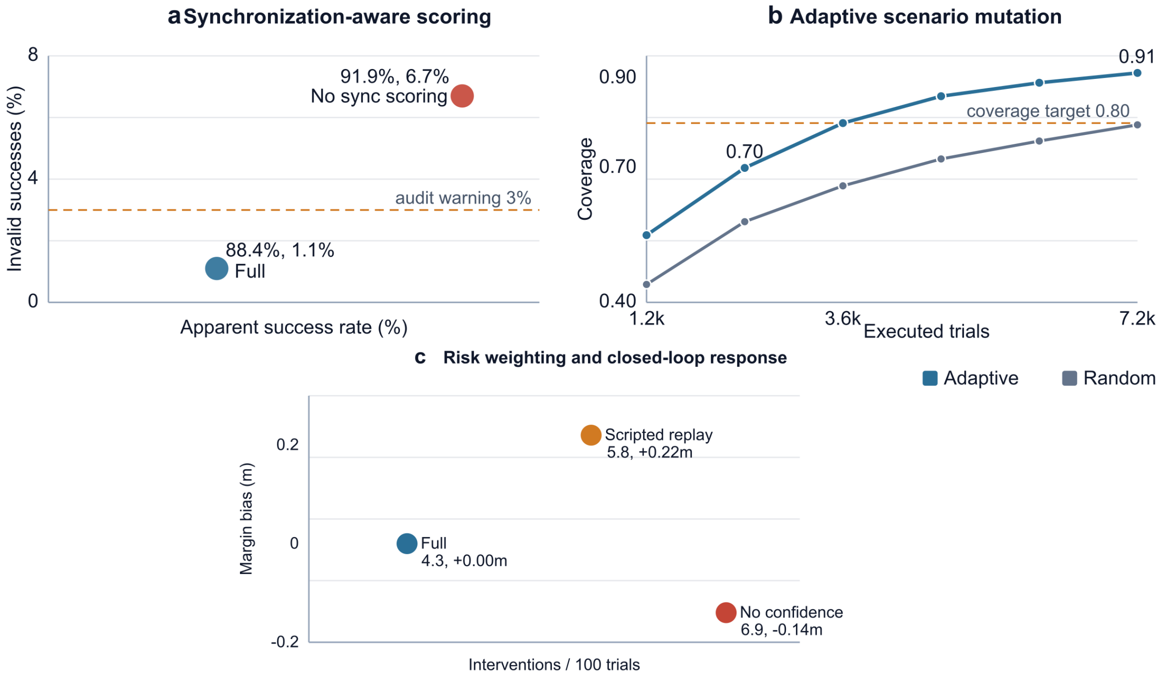

Figure 6. Ablation Analysis: (a) Synchronous Perception Score; (b) Adaptive Scene Variation; (c) Confidence-Weighted Risk Exposure and Closed-Loop Agent Response.

The framework has introduced an engineering trade-off of scenario richness and timing reliability. When the virtual traffic density was increased from 8 to 32 agents, both the average latency and the synchronisation confidence were lower; that is, the latency rose from 34.1ms to 47.8ms, and the confidence decreased from 0.91 to 0.84. Although there was degradation, the tests were still valid, and thus the strategy performance interpretation had to be revised. Sometimes the planner failed in high-density areas because of a slow update of the agent, not due to an error in the decision logic [37]. Therefore, the evaluation pipeline records both strategy indicators and testbed indicators. A deployment engineer can determine whether a policy has failed under a normal load or if it is a test case in a highly unstable environment for more accurate scoring.

Figure 7(a) presents the relation between synchronization confidence, intervention rate, and coverage growth. Trials with confidence of 0.76 required 8.9 interventions per 100 runs, while trials with confidence of 0.92 required only 2.8. The figure also marks the coverage milestone: adaptive mutation reached 80% coverage at about 3100 trials, whereas random mutation required more than 5900 trials. Figure 7(b) compares latency components and spatial-error thresholds. Sensing contributed 7.1 ms, synchronization 9.6 ms, strategy inference 17.8 ms, and command dispatch 3.9 ms, giving a total mean latency of 38.4 ms. The same panel marks that success score remains stable when spatial error is below 0.25 m but falls rapidly after 0.32 m. These observations support two engineering recommendations: keep spatial error below 0.25 m for normal scoring, and keep 95th-percentile closed-loop latency below 80 ms for strategy comparison [38].

A failure in a simulation that does not involve physical sensing and virtual behaviour is not detectable by a mixed-reality framework. In a typical intersection case, the rule-based policy passed the pure-simulation replay because the virtual crossing vehicle followed a fixed path. In mixed reality, the same vehicle responded to the ego vehicle's early acceleration, and the planner entered a negotiation loop that increased delay and resulted in a near-stop at the intersection. The above way solved the problem by considering the fluctuation of acceleration as a will cure for a virtual car. This is why closed-loop mixed reality is beneficial; it can be determined whether a strategy is able to follow a pre-planned path and whether it can negotiate with responsive agents under sensor and time constraints [39].

Based on the above experiments, Data richness should be balanced with interpretability. A framework that saves all its internal variables is difficult to audit, and a framework that only records collisions conceal the real behaviour. The selected set of indicators is relatively small. Synchronization error and confidence indicate whether the trial is valid; risk exposure and minimum distance address safety; jerk and control switching describe comfort and stability; coverage shows how much of the intended interaction space has been utilised. The factors can be used to compare strategies without needing to simplify mixed reality tests into a single score [40].

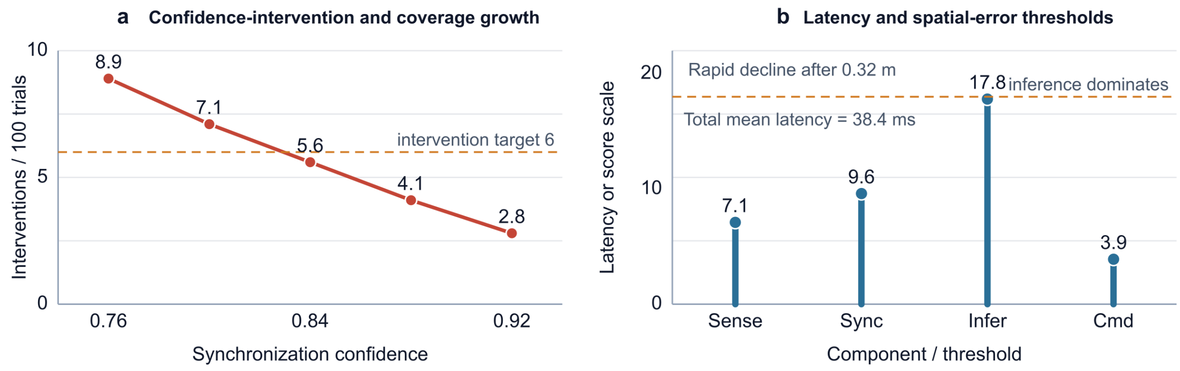

Figure 7. Engineering Threshold Analysis: (a) Confidence-Intervention Relationship and Coverage Growth; (b) Delay Contribution and Spatial Error Threshold.

The framework is more suitable for use before the costly proving-ground test. A development team can run nightly mixed reality regression tests, rank scenario families by risk exposure, and reserve physical road time for the cases that are still uncertain after synchronized virtual-agent testing. Currently, in this experiment, 17.4% of the scenario families accounted for 63.8% of the interventions; thus, the framework has identified a relatively small set of high-value tests. After increasing the strictness of the latency requirement for these tests, the ranking of the three strategies was still the same; thus, it could be concluded that this was not due to a single timing setting. This reproducibility is necessary for engineering sign-off; that is, the improved performance must still be achieved after modifying parameters.

The recorded failure logs further show how the framework supports debugging. Each failed trial stores the synchronized state trace, the virtual-agent intention trace, the strategy output, and the validity score. In 58% of the inspected failures, the root cause was a prediction mismatch rather than a low-level control error. In 24%, the planner selected a safe but inefficient gap because the virtual agent changed speed after the ego vehicle committed to a maneuver. In 11%, excessive latency caused a late response, and the remaining 7% were linked to low confidence under weather disturbance. This decomposition is useful because it points engineers toward the component that should be redesigned. A conventional pass-fail simulator would report these cases as similar failures, while the mixed reality framework separates prediction weakness, negotiation weakness, timing weakness, and sensing uncertainty.

The Deficiencies of the experiment are also known. The hardware setup had a controlled test area and a small number of physical road layouts. Although the virtual traffic and environmental conditions were varied, road-surface friction, sensor contamination and complex driver negotiation were not fully modelled [41]. Therefore, the framework can be seen as a relatively intermediate validation stage between pure software simulation and open-road testing. It will reduce the number of unsafe or uninformative road tests, but not replace them. The future extensions will be larger physical sites, richer V2X disturbance models, and calibrated human-driver behaviour models [42].

Conclusion

A closed-loop mixed-reality simulation platform for autonomous driving strategy verification was introduced in this paper. The components are physical sensing, virtual traffic agents, a synchronisation estimator, an interchangeable strategy interface, adaptive scenario coordination, and quantitative evaluation indicators. According to the above experiment, the framework can maintain high-fidelity synchronous performance and shows different levels of safety, stability, comfort and coverage for each strategy. On the other hand, it is not just a recording or an open-loop experiment; the design for mixed reality can show how a reactive agent performs under real-time conditions.

The study has the following defects. The experiments were carried out in a controlled environment, and although varied, the scenario library cannot fully reflect the complexity of public-road behaviour. Some physical phenomena, such as changes in tyre-road friction, severe sensor contamination and natural human-driver negotiation, were not included. In addition, the threshold for synchronisation validity and risk exposure in the proposed scoring method are not fixed and need to be set differently for various vehicle platforms and operational design domains.

Future work will expand the framework to a large-scale physical test site, add more detailed sensor degradation models, and conduct multi-vehicle cooperative mixed-reality experiments. Another way is to connect scenario mutation with formal safety specifications, and then have the testbed generate cases that are both challenging and traceable back to the engineering requirements. The above extensions will help bridge the gap between scalable software verification and expensive road-deployment costs more effectively through mixed-reality simulation.

Author Contributions

Mariusz Jaworski and Ryszard Halik contribute to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Cyprian Górski contributes to data collection, draft preparation, manuscript editing. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Andreotti, E., Boyraz, P., & Selpi, S. (2021). Mathematical Definitions of Scene and Scenario for Analysis of Automated Driving Systems in Mixed-Traffic Simulations. IEEE Transactions on Intelligent Vehicles, 6(2), 366-375. https://doi.org/10.1109/tiv.2020.3031981

[2] Shi, Y., Liu, Z., Wang, Z., Ye, J., Tong, W., & Liu, Z. (2022). An Integrated Traffic and Vehicle Co-Simulation Testing Framework for Connected and Autonomous Vehicles. IEEE Intelligent Transportation Systems Magazine, 14(6), 26-40. https://doi.org/10.1109/mits.2022.3188566

[3] Schlager, B., Muckenhuber, S., Schmidt, S., Holzer, H., Rott, R., Maier, F. M., Saad, K., Kirchengast, M., ..., & Ruebsam, J. (2020). State-of-the-Art Sensor Models for Virtual Testing of Advanced Driver Assistance Systems/Autonomous Driving Functions. SAE International Journal of Connected and Automated Vehicles, 03(3), 233-261. https://doi.org/10.4271/12-03-03-0018

[4] Muktadir, G. M., Jawad, A., Paranjape, I., Whitehead, J., & Shepelev, A. (2022). Procedural Generation of High-Definition Road Networks for Autonomous Vehicle Testing and Traffic Simulations. SAE International Journal of Connected and Automated Vehicles, 06(1), 99-120. https://doi.org/10.4271/12-06-01-0007

[5] Jurisch, M., Herold, S., Atzrodt, H., & Bauer, J. L. (2022). Simulation-Based Development for Active Suspension Control for Automated Driving Vehicles—Evaluation of Transferability to Real-World Testing. SAE International Journal of Connected and Automated Vehicles, 05(3), 259-269. https://doi.org/10.4271/12-05-03-0021

[6] Xiong, H., Wang, Z., Wu, G., & Pan, Y. (2022). Design and Implementation of Digital Twin-Assisted Simulation Method for Autonomous Vehicle in Car-Following Scenario. Journal of Sensors, 2022, 1-12. https://doi.org/10.1155/2022/4879490

[7] Ozarkar, S., Gely, S., & Zhou, K. (2022). Physics-Based Simulation Solutions for Testing Performance of Sensors and Perception Algorithm under Adverse Weather Conditions. SAE International Journal of Connected and Automated Vehicles, 05(4), 297-312. https://doi.org/10.4271/12-05-04-0024

[8] Zhu, Y., Wang, J., Meng, F., & Liu, T. (2022). Review on Functional Testing Scenario Library Generation for Connected and Automated Vehicles. Sensors, 22(20), 7735. https://doi.org/10.3390/s22207735

[9] Kyriakopoulos, I., Jaworski, P., & Edwards, T. (2022). Toward an Automated Scenario-Based X-in-the-Loop Testing Framework for Connected and Automated Vehicles. SAE International Journal of Connected and Automated Vehicles, 05(4), 381-391. https://doi.org/10.4271/12-05-04-0030

[10] Gołowicz, A., & Cholewiński, S. (2021). AUTOMATED AND AUTONOMOUS VEHICLES - SAFETY, APPROVAL, SOCIAL BENEFITS AND FEARS OF INTRODUCING AUTOMATIC DRIVING SYSTEMS. Transport Samochodowy, 64(2), 38-42. https://doi.org/10.5604/01.3001.0015.5040

[11] Alnaser, A. J., Sargolzaei, A., & Akbas, M. I. (2021). Autonomous Vehicles Scenario Testing Framework and Model of Computation: On Generation and Coverage. IEEE Access, 9, 60617-60628. https://doi.org/10.1109/access.2021.3074062

[12] Chung, Y., & Yang, Y. P. (2021). Hardware-in-the-Loop Simulation of Self-Driving Electric Vehicles by Dynamic Path Planning and Model Predictive Control. Electronics, 10(19), 2447. https://doi.org/10.3390/electronics10192447

[13] Solmaz, S., Rudigier, M., Mischinger, M., & Reckenzaun, J. (2021). Hybrid Testing: A Vehicle-in-the-Loop Testing Method for the Development of Automated Driving Functions. SAE International Journal of Connected and Automated Vehicles, 04(1), 133-148. https://doi.org/10.4271/12-04-01-0011

[14] Yu, B., Chen, C., Tang, J., Liu, S., & Gaudiot, J. L. (2022). Autonomous Vehicles Digital Twin: A Practical Paradigm for Autonomous Driving System Development. Computer, 55(9), 26-34. https://doi.org/10.1109/mc.2022.3159500

[15] Qiao, Z., Loeb, H., Gurrla, V., Lebermann, M., Betz, J., & Mangharam, R. (2022). Drive Right: Autonomous Vehicle Education through an Integrated Simulation Platform. SAE International Journal of Connected and Automated Vehicles, 05(4), 357-366. https://doi.org/10.4271/12-05-04-0028

[16] Drechsler, M. F., Sharma, V., Reway, F., Schütz, C., & Huber, W. (2022). Dynamic Vehicle-in-the-Loop: A Novel Method for Testing Automated Driving Functions. SAE International Journal of Connected and Automated Vehicles, 05(4), 367-380. https://doi.org/10.4271/12-05-04-0029

[17] Li, A., Chen, S., Sun, L., Zheng, N., Tomizuka, M., & Zhan, W. (2022). SceGene: Bio-Inspired Traffic Scenario Generation for Autonomous Driving Testing. IEEE Transactions on Intelligent Transportation Systems, 23(9), 14859-14874. https://doi.org/10.1109/tits.2021.3134661

[18] Allidina, T., Deka, L., Paluszczyszyn, D., & Elizondo, D. (2022). Selecting Non-Line of Sight Critical Scenarios for Connected Autonomous Vehicle Testing. Software, 1(3), 244-264. https://doi.org/10.3390/software1030011

[19] Pietruch, M., Młyniec, A., & Wetula, A. (2020). An overview and review of testing methods for the verification and validation of ADAS, active safety systems, and autonomous driving. Mining - Informatics, Automation and Electrical Engineering, 1 (541)(1), 19. https://doi.org/10.7494/miag.2020.1.541.19

[20] Dygalo, V., Keller, A., & Shcherbin, A. (2020). Principles of application of virtual and physical simulation technology in production of digital twin of active vehicle safety systems. Transportation Research Procedia, 50, 121-129. https://doi.org/10.1016/j.trpro.2020.10.015

[21] Schoenmakers, M., Yang, D., & Farah, H. (2021). Car-following behavioural adaptation when driving next to automated vehicles on a dedicated lane on motorways: A driving simulator study in the Netherlands. Transportation Research Part F: Traffic Psychology and Behaviour, 78, 119-129. https://doi.org/10.1016/j.trf.2021.01.010

[22] Han, J. B., Kim, S. S., & Song, H. J. (2021). Development of Real-Time Digital twin model of Autonomous Field Robot for Prediction of Vehicle Stability. Journal of Institute of Control, Robotics and Systems, 27(3), 190-196. https://doi.org/10.5302/j.icros.2021.20.0181

[23] Obaid, M., & Torok, A. (2021). Macroscopic Traffic Simulation of Autonomous Vehicle Effects. Vehicles, 3(2), 187-196. https://doi.org/10.3390/vehicles3020012

[24] Akopov, A. (2021). Scenario simulation of autonomous vehicles motion in Artificial Road Network using FLAME GPU. Artificial societies, 16(1), 0. https://doi.org/10.18254/s207751800014028-9

[25] Wang, M., Zeng, B., & Wang, Q. (2021). Study of Motion Control and a Virtual Reality System for Autonomous Underwater Vehicles. Algorithms, 14(3), 93. https://doi.org/10.3390/a14030093

[26] Ambroziak, L., & Ciężkowski, M. (2021). Virtual Electric Dipole Field Applied to Autonomous Formation Flight Control of Unmanned Aerial Vehicles. Sensors, 21(13), 4540. https://doi.org/10.3390/s21134540

[27] Wen, M., Park, J., Sung, Y., Park, Y. W., & Cho, K. (2021). Virtual Scenario Simulation and Modeling Framework in Autonomous Driving Simulators. Electronics, 10(6), 694. https://doi.org/10.3390/electronics10060694

[28] Stadler, C., Montanari, F., Baron, W., Sippl, C., & Djanatliev, A. (2022). A Credibility Assessment Approach for Scenario-Based Virtual Testing of Automated Driving Functions. IEEE Open Journal of Intelligent Transportation Systems, 3, 45-60. https://doi.org/10.1109/ojits.2022.3140493

[29] Samani, A. R., Mishra, S., & Dey, K. (2022). Assessing the effect of long-automated driving operation, repeated take-over requests, and driver's characteristics on commercial motor vehicle drivers' driving behavior and reaction time in highly automated vehicles. Transportation Research Part F: Traffic Psychology and Behaviour, 84, 239-261. https://doi.org/10.1016/j.trf.2021.10.015

[30] Wang, X., Peng, Y., Xu, T., Xu, Q., Wu, X., Xiang, G., Yi, S., & Wang, H. (2022). Autonomous driving testing scenario generation based on in-depth vehicle-to-powered two-wheeler crash data in China. Accident Analysis & Prevention, 176, 106812. https://doi.org/10.1016/j.aap.2022.106812

[31] Delmas, M., Camps, V., & Lemercier, C. (2022). Effects of environmental, vehicle and human factors on comfort in partially automated driving: A scenario-based study. Transportation Research Part F: Traffic Psychology and Behaviour, 86, 392-401. https://doi.org/10.1016/j.trf.2022.03.012

[32] Marsso, L., Mateescu, R., Muller, L., & Serwe, W. (2022). Formally Modeling Autonomous Vehicles in LNT for Simulation and Testing. Electronic Proceedings in Theoretical Computer Science, 355, 60-117. https://doi.org/10.4204/eptcs.355.5

[33] Caruso, G., Yousefi, M. K., & Mussone, L. (2022). From Human to Autonomous Driving: A Method to Identify and Draw Up the Driving Behaviour of Connected Autonomous Vehicles. Vehicles, 4(4), 1430-1449. https://doi.org/10.3390/vehicles4040075

[34] Ibrahim, M., Rassõlkin, A., Vaimann, T., & Kallaste, A. (2022). Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System. Sustainability, 14(2), 601. https://doi.org/10.3390/su14020601

[35] Scheibe, T., Heiduschke, C., Liebald, M., & Schudi, L. C. (2022). Physical Sensor Simulation for Virtual Validation of Autonomous Driving Functions. ATZelectronics worldwide, 17(9), 44-47. https://doi.org/10.1007/s38314-022-0794-0

[36] Garlick, S., & Bradley, A. (2022). Real-time optimal trajectory planning for autonomous vehicles and lap time simulation using machine learning. Vehicle System Dynamics, 60(12), 4269-4289. https://doi.org/10.1080/00423114.2021.2011929

[37] Allamaa, J. P., Patrinos, P., Van der Auweraer, H., & Son, T. D. (2022). Sim2real for Autonomous Vehicle Control using Executable Digital Twin. IFAC-PapersOnLine, 55(24), 385-391. https://doi.org/10.1016/j.ifacol.2022.10.314

[38] Schwind, A., Buddappagari, S., Asghar, M. E., & Hein, M. A. (2022). Virtual Sensor Validation for Automated and Connected Driving. ATZ worldwide, 124(11), 54-57. https://doi.org/10.1007/s38311-022-1405-7

[39] Szalay, Z., Ficzere, D., Tihanyi, V., Magyar, F., Soós, G., & Varga, P. (2020). 5G-Enabled Autonomous Driving Demonstration with a V2X Scenario-in-the-Loop Approach. Sensors, 20(24), 7344. https://doi.org/10.3390/s20247344

[40] Saito, Y., Kawashima, K., & Hirakawa, M. (2020). Effectiveness of a Head Movement Interface for Steering a Vehicle in a Virtual Reality Driving Simulation. Symmetry, 12(10), 1645. https://doi.org/10.3390/sym12101645

[41] Hult, R., Zanon, M., Gros, S., Wymeersch, H., & Falcone, P. (2020). Optimisation-based coordination of connected, automated vehicles at intersections. Vehicle System Dynamics, 58(5), 726-747. https://doi.org/10.1080/00423114.2020.1755446

[42] Pyrialakou, V., Gkartzonikas, C., Gatlin, J., & Gkritza, K. (2020). Perceptions of safety on a shared road: Driving, cycling, or walking near an autonomous vehicle. Journal of Safety Research, 72, 249-258. https://doi.org/10.1016/j.jsr.2019.12.017