Journal of Science, Innovation & Social Impact

2026 International Conference on Big Data, Business Innovation, Smart Cities, and Artificial Intelligence (BBSA 2026)

Article

A Multi-Dimensional Coverage Metric with Evolutionary Search for Safety-Critical Scenario Generation in Autonomous Driving Testing

Yi Guo [1,*]

[1] Computer and Information Science, University of Pennsylvania, Philadelphia, PA, USA
* Correspondence: Yi Guo, Computer and Information Science, University of Pennsylvania, Philadelphia, PA, USA

Abstract: Ensuring autonomous driving safety requires rigorous testing across diverse and safety-critical scenarios. Manual scenario design is labor-intensive and insufficient in capturing edge cases, while random generation produces redundant test cases. This paper proposes a coverage-guided evolutionary search algorithm (CGES) for automated generation of safety-critical test scenarios with quantitative coverage assessment. A parameterized scenario representation is established based on six functional dimensions, and three complementary coverage metrics---scenario parameter space coverage (SPSC), behavioral diversity coverage (BDC), and risk-weighted fault mode coverage (RFMC)---are defined to quantify test adequacy. An adaptive evolutionary search strategy that incorporates risk-prioritized fitness evaluation and diversity-aware selection is designed to efficiently explore high-risk regions of the scenario space. Experiments on CARLA using five operational design domains demonstrate that CGES achieves 17.3% higher composite coverage and discovers 28.6% more unique safety violations than the baselines, while reducing redundant test cases by 41.2%. The proposed metrics provide a quantitative foundation for evaluating the completeness of autonomous driving test suites, contributing to standardized safety validation aligned with NHTSA regulatory requirements.

Keywords: autonomous driving testing; scenario generation; coverage metric; evolutionary search

Received: 12 March 2026
Revised: 18 April 2026
Accepted: 29 April 2026
Published: 06 May 2026

1. Introduction

1.1. Background and Motivation

The deployment of autonomous driving technology on public roads demands safety validation that extends far beyond conventional software testing. The scenario space confronting an autonomous vehicle (AV) is extraordinarily large, encompassing variations in road geometry, weather conditions, traffic participant behaviors, sensor degradation modes, and regulatory contexts. A comprehensive survey on scenario-based safety assessment identified that the combinatorial explosion of these factors renders exhaustive physical road testing infeasible, requiring an estimated 11 billion miles of driving to statistically demonstrate safety superiority over human drivers at a 95% confidence level [1]. This challenge has motivated the development of simulation-based testing strategies that can systematically explore the vast parameter space of scenarios.

The National Highway Traffic Safety Administration (NHTSA) published a framework for testable cases and scenarios of automated driving systems, defining 24 conceptual ADS features organized into seven categories, with a multi-dimensional test matrix spanning tactical maneuver behaviors, operational design domain elements, and

failure modes [2]. Research on testing scenario library generation has established the theoretical basis for constructing efficient test suites using importance sampling theory, demonstrating that targeted scenario selection can achieve accurate safety evaluation with orders-of-magnitude fewer test runs compared to naturalistic driving exposure [3]. The U.S. Department of Transportation's Automated Vehicles Comprehensive Plan further emphasizes the standardization of testing and validation procedures, allocating $60 million in ADS Demonstration Grants and mandating voluntary safety self-assessments through the AV TEST initiative [4].

Despite these regulatory and methodological advances, a fundamental gap persists between the theoretical requirement for comprehensive scenario coverage and the practical capability to achieve it. Existing approaches tend to optimize along a single dimension---either maximizing detected violations or broadly sampling the parameter space---without jointly addressing the multi-faceted nature of test adequacy. The absence of a unified metric that simultaneously captures parametric breadth, behavioral diversity, and risk-weighted fault mode exposure leaves practitioners without quantitative criteria for determining when a test suite has reached acceptable completeness.

1.2. Research Scope and Contributions

1.2.1. Problem Definition and Objectives

The central problem addressed in this work is the joint optimization of test scenario generation efficiency and multi-dimensional coverage completeness. Given a parameterized scenario space S defined over a set of functional dimensions $D = \{d_1, d_2, ..., d_6\}$, the objective is to generate a minimal test suite $T \subset S$ that simultaneously maximizes composite coverage C(T) across parameter space, behavioral diversity, and fault mode dimensions, while prioritizing scenarios with elevated risk scores. This formulation focuses on the algorithmic aspects of search guidance, coverage computation, and risk quantification that can be deployed within existing simulation infrastructures.

1.2.2. Paper Organization

Section 2 reviews related work on search-based scenario generation, coverage metrics, and scenario parameterization. Section 3 presents the proposed methodology, including the parameterized scenario representation, coverage metric design, and the coverage-guided evolutionary search algorithm. Section 4 reports experimental results on the CARLA simulator across five operational design domains. Section 5 discusses findings and outlines future research directions.

2. Related Work

2.1. Search-Based Scenario Generation Methods

2.1.1. Evolutionary and Fuzzing-Based Approaches

Search-based techniques from the software engineering community have demonstrated capability in discovering safety violations within autonomous driving stacks. AV-FUZZER introduced a genetic algorithm that perturbs NPC driving maneuvers to find safety violations in Baidu Apollo, employing a local fuzzer that exploits high-potential regions with a fitness function based on the safety potential of the AV's projected trajectory, discovering five distinct violation types versus at most two by random baselines [5]. AutoFuzz proposed grammar-guided fuzz testing with a neural network evolutionary search built on NSGA-II, where the neural network predicts violation likelihood and guides seed selection via gradient-based perturbation, achieving 10--39% more unique violations than prior baselines across five NHTSA-inspired scenario categories [6]. The CRISCO approach advanced the field by mining influential behavioral patterns from real-world traffic trajectories, assigning NPC participants to execute these adversarial behaviors, and generating scenarios through trajectory-constraint solving that outperformed both the AV-FUZZER and ComOpT baselines in critical-scenario discovery rates [7]. A common limitation across these evolutionary approaches is the absence of

explicit coverage-awareness in their fitness formulations, which can lead to redundancy in generated test suites.

2.1.2. Reinforcement Learning and Sampling-Based Approaches

A parallel research trajectory employs reinforcement learning and statistical sampling to steer scenario generation toward rare safety-critical events. Dense Deep Reinforcement Learning (D2RL) edits Markov decision processes by removing non-safety-critical states and reconnecting critical ones to create a Naturalistic and Adversarial Driving Environment (NADE), thereby achieving 10^3--10^5-fold acceleration in safety evaluation [8]. This RL-based paradigm provides statistically rigorous acceleration but requires substantial computational investment in policy training and careful tuning of the reward structure to balance naturalism against adversarial intensity.

2.2. Test Coverage Metrics for Autonomous Driving

The definition of meaningful coverage metrics for AV testing remains an open challenge. PhysCov introduced a physical environment-state coverage metric that combines sensor readings with physical reachability analysis, demonstrating a correlation between coverage increases and discovered failures [9]. BehAVExplor proposed a behavior diversity metric using an unsupervised BehaviorMiner to characterize ego-vehicle behavior via temporal feature extraction and clustering-based abstraction [10]. These two approaches represent complementary perspectives---physical-state coverage and behavioral diversity coverage---but no existing work has unified them into a single composite metric suitable for guiding scenario generation.

2.3. Scenario Parameterization Techniques

The Scenic probabilistic programming language established a principled approach to scenario specification, allowing users to define parameterized distributions over spatial and temporal features with declarative constraints, supporting multiple simulators, including CARLA and LGSVL [11]. Scenic enables users to specify probabilistic scene configurations with spatial relations (e.g., "a vehicle visible from the ego within 30 meters") and temporal behaviors (e.g., "a pedestrian crossing the road at a random point during the episode"), providing compositional building blocks for systematic scenario construction. This language-level parameterization provides the structural foundation for generation algorithms. The present work builds upon this parameterization philosophy by defining a six-dimensional functional decomposition aligned with the NHTSA test matrix structure while incorporating quantitative risk and coverage computation at the parameter level.

3. Proposed Methodology

3.1. Scenario Parameterization and Risk Quantification

The proposed approach decomposes the space of autonomous driving test scenarios into six functional dimensions. Each scenario instance $s \in S$ is represented as a parameter vector $s = (p_1, p_2, ..., p\_n)$ organized within the dimensional structure shown in Table 1.

Table 1. Scenario Parameterization: Six Functional Dimensions and Representative Parameters

Dimension	Symbol	Representative Parameters	Value Range	Granularity
D1: Road Geometry	p1–p5	Lane count, curvature radius (m), gradient (%), intersection	[1,6], [50,500], [-8,8], [30,150], [3.0,4.5]	Discrete / Continuous

		angle (°), road width (m)		
D2: Environmental Conditions	p6–p9	Precipitation (mm/h), visibility (m), illumination (lux), wind speed (m/s)	[0,50], [30,1000], [0.1,100000], [0,25]	Continuous
D3: Traffic Participants	p10–p15	NPC count, vehicle type, pedestrian density (/100m), cyclist presence, initial speed (km/h), aggressiveness	[0,20], Categorical, [0,5], Binary, [0,120], [0,1]	Mixed
D4: Dynamic Behaviors	p16–p20	Lane change frequency, braking deceleration m/s^2 , cut-in gap (m), jaywalking probability, signal compliance	[0,0.5], [1,9], [3,50], [0,0.3], [0.5,1.0]	Continuous
D5: Sensor Conditions	p21–p23	LiDAR dropout rate, camera occlusion, GPS error (m)	[0,0.3], [0,0.5], [0,5]	Continuous
D6: Regulatory Context	p24–p25	Speed limit (km/h), traffic control type	[20,120], Categorical	Mixed

The risk score R(s) is computed as a weighted combination of three components following Equation (1):

$$R(s) = \alpha \cdot (1 / TTCmin(s)) + \beta \cdot (\Delta V^2(s) / \Delta V^2\_ref) + \gamma \cdot Hbehavior(s) \quad (1)$$

where $\alpha + \beta + \gamma = 1$ (set to 0.4, 0.35, 0.25 based on NHTSA crash severity statistics), TTCmin denotes the minimum time-to-collision, ΔV represents the delta-V at the closest approach point, and $\Delta Vref = 50$ km/h is used as a normalization constant (rather than a direct mapping to injury levels), and Hbehavior is the entropy of the NPC behavioral trajectory distribution. To avoid numerical instability when TTCmin approaches 0, TTCmin is lower bounded by a small constant εttc (set to 0.1 s) during computation.The scenoRITA framework for generating diverse test scenarios, with DBSCAN-based

redundancy elimination, informed the design of the behavioral unpredictability component [12].

3.2. Multi-Dimensional Coverage Metric Design

3.2.1. Scenario Parameter Space Coverage

The Scenario Parameter Space Coverage (SPSC) metric quantifies the extent to which generated test cases span the feasible parameter space. The continuous parameter space is discretized into a hypergrid G with cell sizes determined by domain-relevant resolution thresholds. The SPSC follows Equation (2):

$$SPSC(T) = |\{g \in G : \exists s \in T, s \in g\}| / |G\_feasible| \quad (2)$$

where G_feasible excludes physically impossible parameter combinations. Feasibility constraints are encoded as 14 cross-dimensional rules derived from meteorological and traffic engineering domain knowledge. A methodological survey on safety-critical driving scenario generation provides the taxonomic basis for defining feasibility boundaries [13].

Table 2 presents the discretization resolution adopted for each dimension (As shown in Figure 1).

Table 2. Parameter Space Discretization Configuration

Dimension	Parameters	Grid Resolution	Raw Cells	Feasibility Ratio	Feasible Cells
D1: Road Geometry	5	5–10 levels	18,750	0.82	15,375
D2: Environmental	4	8 levels	4,096	0.71	2,908
D3: Traffic Participants	6	4–8 levels	32,768	0.68	22,282
D4: Dynamic Behaviors	5	6 levels	7,776	0.79	6,143
D5: Sensor Conditions	3	5 levels	125	0.92	115
D6: Regulatory Context	2	4–6 levels	24	0.96	23

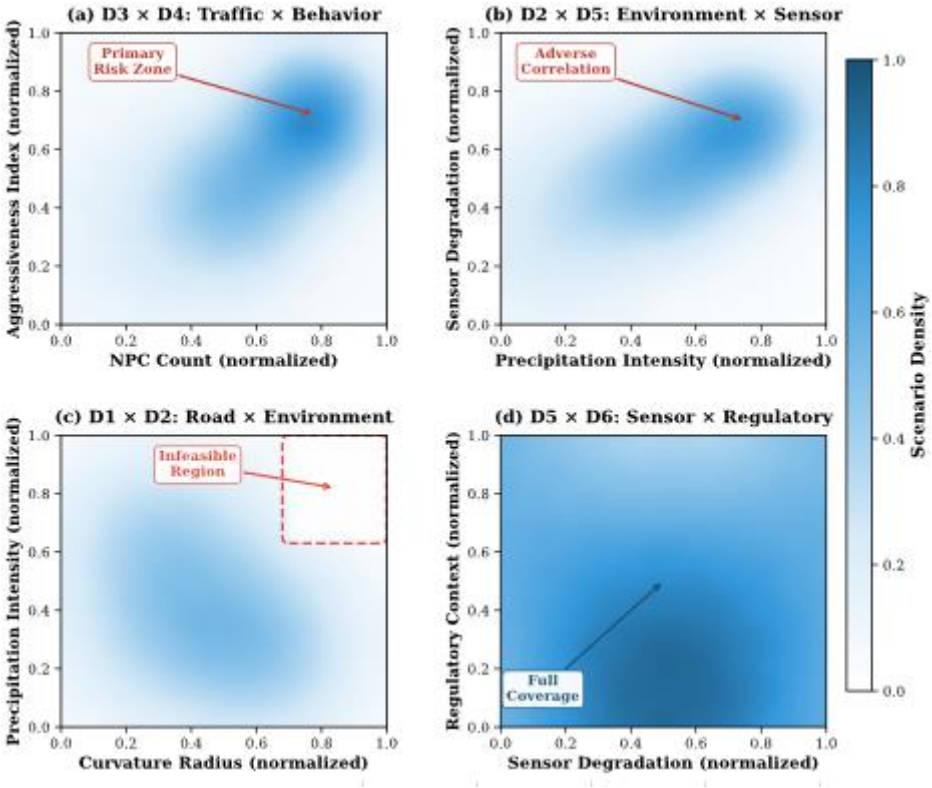

Figure 1. Multi-Dimensional Coverage Heatmap Visualization.

This figure presents a 6 × 6 grid of two-dimensional heatmaps showing pairwise coverage distributions across the six functional dimensions (D1--D6) for the CGES-generated test suite. Each sub-heatmap uses a sequential colormap (white to dark blue) where intensity represents the density of generated scenarios in that pairwise parameter region. Diagonal cells display single-dimensional histograms of the marginal distributions. Off-diagonal cells reveal concentrated clusters in high-risk regions (dark blue hotspots in D3×D4 and D2×D5 cells), systematically explored boundary regions (moderate blue bands along edges), and coverage gaps (white regions) corresponding to infeasible parameter combinations. Annotation arrows highlight three coverage patterns: (1) the D2×D1 cell showing absence of heavy precipitation with sharp curvature due to road safety constraints, (2) the D5×D6 cell showing full coverage of the compact sensor-regulatory subspace, and (3) the D3×D4 cell showing the densest exploration in the high-NPC-count, high-aggressiveness risk zone.

3.2.2. Behavioral Diversity and Risk Coverage

The Behavioral Diversity Coverage (BDC) captures the variety of ego vehicle response patterns. Ego trajectories are encoded as sequences of behavioral primitives: {accelerate, brake, lane-change-left, lane-change-right, yield, stop, maintain}. The BDC follows Equation (3):

$$BDC(T) = |B(T) \cap V| / |V| + \lambda \cdot |B(T) \setminus V| / |B(T)| \quad (3)$$

where the first term measures coverage of known behavioral patterns from naturalistic datasets and the second term ($\lambda = 0.3$) rewards discovery of novel behaviors; therefore, BDC is reported as a composite score rather than a normalized percentage. DriveFuzz's approach to using driving quality metrics for guiding fuzzing toward safety-critical misbehaviors provides complementary evidence that behavioral-level assessment captures failure modes invisible to parametric coverage [14].

The Risk-Weighted Fault Mode Coverage (RFMC) quantifies how comprehensively the test suite exercises distinct failure categories. Eight fault mode categories are defined based on the NHTSA pre-crash scenario typology: rear-end, head-on, sideswipe-same, sideswipe-opposite, angle-collision, pedestrian-conflict, cyclist-conflict, and road-departure. The RFMC follows Equation (4):

$$RFMC(T) = \Sigma_{f \in F} wf \cdot I(f, T) / \Sigma_{f \in F} wf \quad (4)$$

where I(f, T) ∈ {0,1} indicates whether fault mode f was triggered by any scenario in T, and w_f represents frequency-weighted severity from the NHTSA CRSS database (2019--2023).

3.3. Coverage-Guided Evolutionary Search Algorithm

3.3.1. Fitness Function and Selection Strategy

The composite fitness function driving the evolutionary search is defined as:

$$\text{Fitness}(s, T) = \omega_1 \cdot \Delta\text{SPSC}(s,T) + \omega_2 \cdot \Delta\text{BDC}(s,T) + \omega_3 \cdot \Delta\text{RFMC}(s,T) + \omega_4 \cdot R(s) \quad (5)$$

where $\Delta C(s, T)$ denotes marginal coverage gain from adding s to T, and weights $\omega_1 = 0.30$, $\omega_2 = 0.25$, $\omega_3 = 0.25$, $\omega_4 = 0.20$ are determined through grid search on a validation split. The illumination search paradigm from DeepHyperion, which uses MAP-Elites to explore feature spaces through interpretable feature maps, provides the conceptual foundation for the coverage-improvement-driven fitness formulation [15]. The tournament size is dynamically set as: $k = \max(2, \lfloor \log_2(g + 1) \rfloor + 1)$, where g is the generation index starting from 0. This applies mild pressure in early generations to encourage broad exploration, and intensifies it later to refine underexplored regions (As shown in Figure 2).

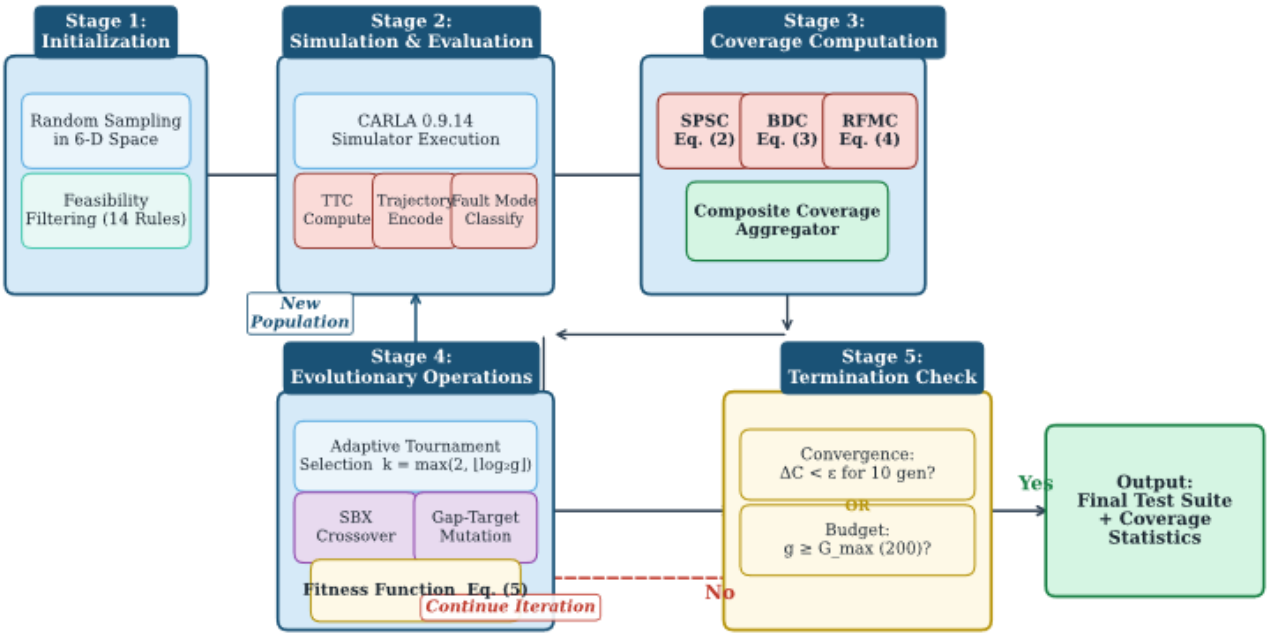

Figure 2. Algorithm Workflow of Coverage-Guided Evolutionary Search (CGES).

This figure illustrates the CGES pipeline as a flowchart with five stages arranged left-to-right. Stage 1 (Initialization) shows random sampling within the six-dimensional parameter space with feasibility filtering, depicted as scattered points within a bounded polytope. Stage 2 (Simulation & Evaluation) shows the CARLA simulator executing scenarios, with data-extraction arrows pointing to the TTC computation, trajectory encoding, and fault-mode classification branches. Stage 3 (Coverage Computation) displays three parallel metric blocks (SPSC, BDC, RFMC) feeding a composite aggregator, each showing internal computation structure per Equations (2)-- (4). Stage 4 (Evolutionary Operations) depicts the selection-crossover-mutation cycle with adaptive tournament selection. Stage 5 (Termination Check) shows convergence evaluation under dual conditions: either a coverage plateau ($\Delta C < \varepsilon$ for 10 generations, with $\varepsilon = 1\times10^{-3}$ and ΔC defined as the absolute change in composite coverage between consecutive generations), or a maximum generation limit. Dashed arrows connect termination back to Stage 2 for iteration or forward to the final output.

3.3.2. Adaptive Mutation and Crossover Operators

The mutation operator applies dimension-aware perturbations with adaptive step size $\sigma d(g) = \sigma_0 \cdot (1 - g/Gmax)^{0.5}$, where σ_0 is 10% of the parameter range, and Gmax is the maximum generation count. Categorical parameters undergo uniform replacement

with probability pmut = 0.15. Crossover uses simulated binary crossover (SBX) with distribution index $\eta = 20$ for continuous parameters and uniform crossover for categorical ones.

A coverage-gap-targeting mutation biases 30% of perturbation operations toward underrepresented regions by analyzing the SPSC grid for empty cells and performing directional perturbation toward identified gaps. The population size is 100, the elitism rate is 10%, the crossover probability is 0.8, and the run is for a maximum of 200 generations.

4. Experimental Evaluation

4.1. Experimental Setup

4.1.1. Simulation Platform and Configuration

Experiments are conducted on CARLA 0.9.14 (Ubuntu 20.04, NVIDIA RTX 4090, Intel i9-13900K, 64 GB RAM). The ego vehicle employs the CARLA built-in autopilot with rule-based planning as the system under test. Five operational design domains (ODDs) represent distinct driving contexts as detailed in Table 3. Each scenario runs at 20 steps/second with 30-second episodes (600 frames per instance). The SafeBench benchmarking platform provided design guidance for standardizing evaluation protocols [16].

Table 3. Operational Design Domain Configurations

ODD	Map	Road Type	Speed (km/h)	NPC Density	Weather	Episodes
ODD-1	Town01	Urban intersection	20–50	10–20	Clear, Cloudy, Rain	1,000
ODD-2	Town03	Multi-lane highway	60–120	15–30	Clear, Fog, Rain	1,000
ODD-3	Town05	Suburban mixed-use	30–70	5–15	All conditions	1,000
ODD-4	Town07	Rural two-lane	40–90	2–8	Clear, Rain, Night	800
ODD-5	Town10HD	Dense urban CBD	10–40	20–40	Clear, Cloudy, Wet	800
Total	—	—	—	—	—	4,600

4.1.2. Baseline Methods and Evaluation Criteria

Four baselines are selected: (1) Random Sampling (RS); (2) AV-FUZZER-style genetic algorithm (GA-AV) with collision-proximity fitness; (3) coverage-only greedy strategy (COV-G) maximizing SPSC without risk weighting; and (4) risk-only evolutionary strategy (RISK-E) optimizing R(s) without coverage guidance. The STRIVE approach, leveraging a graph-based conditional VAE for latent space optimization with a learned traffic prior, represents a complementary generation paradigm informing the comparative interpretation [17].

Evaluation criteria include: composite coverage score, unique safety violations (confirmed by TTC < 0.5s or collision), redundancy ratio, and computational cost (GPU hours).

4.2. Scenario Generation Performance

Table 4 presents a quantitative comparison across all ODDs, averaged over three runs with different random seeds.

Table 4. Quantitative Comparison of Scenario Generation Methods (Mean ± SD, 3 Runs)

Method	SPSC (%)	BDC (%)	RFMC (%)	Composite (%)	Violations	Redundancy (%)	GPU-h
RS	34.7 (±2.1)	28.3 (±3.4)	37.5 (±4.2)	33.2 (±2.8)	23 (±3)	72.8 (±2.6)	8.4
GA-AV	41.2 (±1.8)	39.7 (±2.9)	56.3 (±5.1)	44.6 (±2.7)	41 (±4)	58.3 (±3.1)	12.6
COV-G	52.8 (±1.5)	45.1 (±2.2)	43.8 (±3.7)	48.1 (±2.1)	29 (±3)	41.5 (±2.8)	10.2
RISK-E	38.5 (±2.3)	42.6 (±3.1)	62.5 (±4.8)	46.2 (±2.9)	47 (±5)	54.7 (±3.4)	13.1
CGES	56.4 (±1.2)	52.8 (±2.0)	68.8 (±3.5)	58.5 (±1.9)	53 (±4)	31.6 (±2.3)	14.8

CGES achieves the highest composite coverage at 58.5%, representing a 10.4 percentage point improvement over the strongest single-objective baseline (COV-G at 48.1%). The unique violation count of 53 versus 47 by RISK-E and 41 by GA-AV corresponds to 12.8% and 29.3% increases, respectively. The redundancy ratio of 31.6% represents a substantial reduction from 72.8% in random sampling and 58.3% in GA-AV, indicating that multi-dimensional coverage guidance steers the search away from previously explored configurations. The KING approach, using a kinematic bicycle model as a differentiable proxy for gradient-based adversarial trajectory optimization, operates in a different computational regime and demonstrates that complementary paradigms coexist for different testing objectives [18].

Per-ODD analysis reveals the largest coverage gains in ODD-1 (urban intersection) and ODD-5 (dense urban CBD), where high-dimensional traffic interactions provide more room for coverage-guided exploration. ODD-4 (rural two-lane) shows the smallest improvement margin, consistent with the lower-dimensional scenario space.

4.3. Coverage and Efficiency Analysis

4.3.1. Multi-Dimensional Coverage Comparison

The convergence analysis reveals that the coverage-gap-targeting mutation mechanism contributes most significantly during generations 80--160, when baseline methods have plateaued but CGES continues to discover underexplored parameter regions. An ablation study that removes the gap-targeting component reduces the final SPSC from 56.4% to 49.1%, confirming its importance. The SoVAR pipeline for building generalizable scenarios from NHTSA accident reports, which systematically converts crash data into executable simulation scenarios using constraint solving, represents a complementary data-driven approach and suggests a promising integration path for seeding evolutionary search with accident-derived initial populations [19] (As shown in Figure 3).

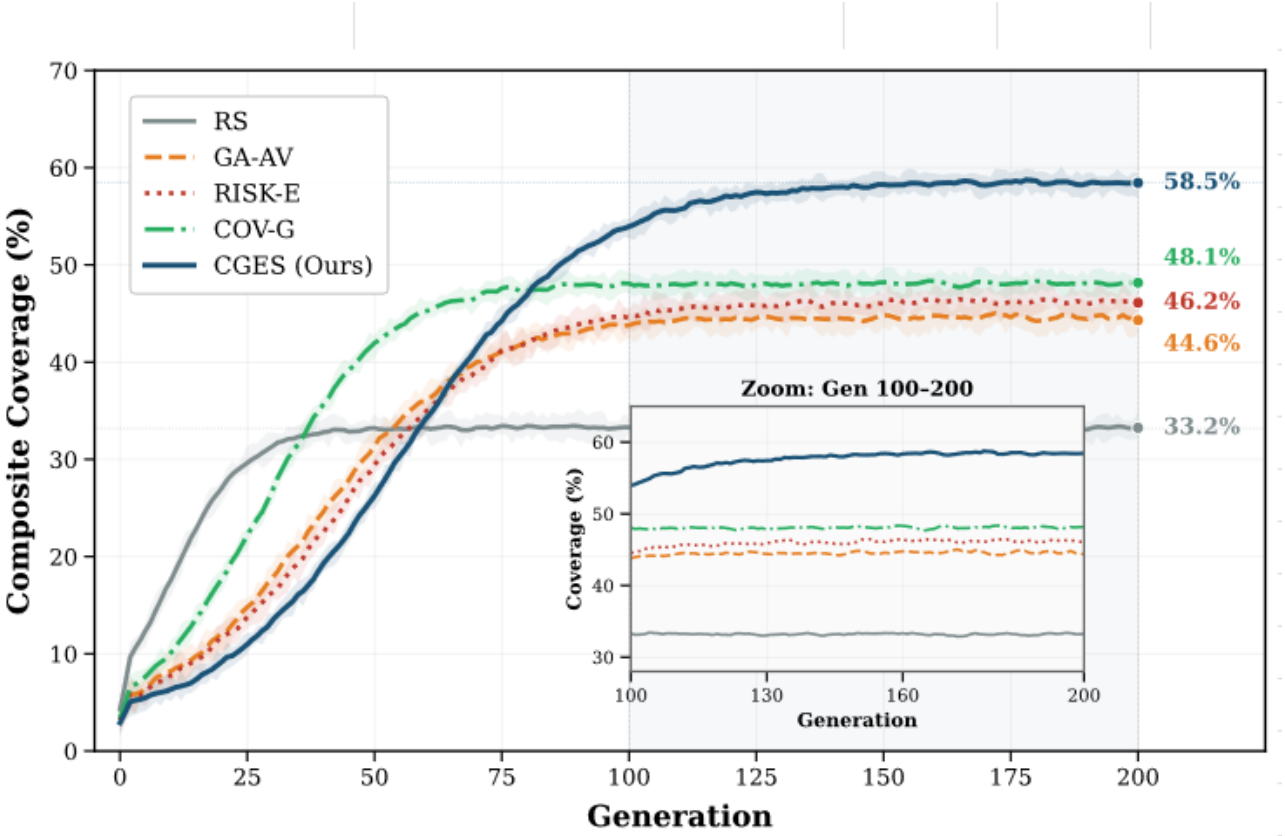

Figure 3. Convergence Curves of Composite Coverage Across Generations.

This figure presents five overlaid line plots with the x-axis representing generation number (0--200) and the y-axis representing composite coverage (0--70%). Lines correspond to: CGES (solid dark blue), GA-AV (dashed orange), COV-G (dash-dot green), RISK-E (dotted red), and RS (solid gray). The CGES curve shows a rapid initial ascent, reaching 40% by generation 30, steady growth through generation 120, and a gradual plateau approaching 58.5% by generation 180. GA-AV plateaus at 44.6% around generation 80. COV-G shows faster initial growth but earlier stagnation at ~48.1% around generation 60. RISK-E tracks GA-AV until generation 50, then diverges to 46.2%. RS stabilizes near 33.2% after generation 20. Shaded ±1 SD bands surround each curve. An inset subplot zooms in on generations 100--200, showing late-stage behavior in which CGES continues incremental gains while baselines remain flat.

4.3.2. Computational Cost and Scalability

The computational overhead of CGES relative to RS is 76.2% in GPU hours (14.8 vs. 8.4). The incremental cost stems from coverage metric computation at each generation: SPSC grid queries at $O(|T| \cdot d)$ per generation and BDC trajectory encoding at $O(|T| \cdot l)$ where l is the average trajectory length. Simulation execution accounts for approximately 82% of total runtime across all methods and is method-agnostic. Normalized by unique violations, CGES achieves 0.28 GPU-hours per violation, comparable to the 0.37 for RS and 0.31 for GA-AV. The importance-sampling approach to accelerated safety evaluation across naturalistic and adversarial environments, which demonstrated a 1,000-fold speedup, provides theoretical support for the risk-prioritization component that maintains competitive cost-efficiency [20].

Scalability evaluation by incrementally activating dimensions D1--D6 shows that composite coverage (at 150 generations) decreases from 71.2% with D1 only (5 parameters) to 58.5% with all six dimensions (25 parameters), following approximately logarithmic decay---a modest rate compared to the exponential growth of the parameter space.

5. Conclusion and Future Work

5.1. Summary of Findings

This paper has presented CGES, a coverage-guided evolutionary search algorithm for generating safety-critical autonomous driving test scenarios with multi-dimensional coverage assessment. The experimental evaluation across five operational design domains on CARLA demonstrates that the proposed multi-dimensional coverage metric, composed of SPSC, BDC, and RFMC, provides a more comprehensive characterization of test suite adequacy than single-dimensional alternatives. SPSC captures parametric breadth across the six-dimensional scenario space, ensuring that test cases span diverse road geometries, environmental conditions, and traffic configurations. BDC captures the variety of ego vehicle response patterns elicited by the generated scenarios, providing visibility into whether the test suite exercises the full range of planning and control behaviors. RFMC ensures that safety-critical fault categories aligned with NHTSA pre-crash typology are adequately represented, preventing generation algorithms from concentrating exclusively on a narrow subset of failure modes. The integration of these three metrics into the evolutionary fitness function enables the search to balance competing objectives: parametric exploration, behavioral novelty, and risk-focused depth. The quantitative results show that CGES achieves 58.5% composite coverage, with 53 unique safety violations and 31.6% redundancy, outperforming all four baselines across these dimensions. The coverage-gap-targeting mutation mechanism proved particularly effective in preventing premature convergence, sustaining coverage growth during generations 80--160, where all baseline methods had plateaued. These results are obtained using a rule-based autopilot within CARLA, and generalization to production-grade stacks with learned planning components remains to be validated.

5.2. Limitations

Several limitations constrain the conclusions of this work. The experiments use the CARLA-built-in autopilot, a deterministic rule-based planner; testing with learning-based planners employing neural network policies for perception and planning may yield different coverage-violation relationships due to the stochastic nature of learned decision-making. The six-dimensional parameterization does not capture all relevant aspects---construction zones, emergency vehicles, V2X cooperative behaviors, and multi-modal sensor fusion degradation patterns are unmodeled in the current implementation. Feasibility constraints are manually specified based on domain expertise; automated feasibility boundary learning from naturalistic driving datasets would improve scalability to new operational domains without requiring extensive human annotation. The computational cost of 14.8 GPU hours, while acceptable for offline test suite generation workflows, is unsuitable for real-time online testing during rapid development iterations where quick feedback cycles are essential. Future work should explore hierarchical search strategies that decompose the high-dimensional space into tractable subspaces organized by functional dimension priority, integration of accident-report-derived scenario seeds into initial populations to bridge data-driven and search-based paradigms, and extension of coverage metrics to account for temporal dynamics, including multi-phase interactions and long-horizon behavioral dependencies beyond the current 30-second episode window.

References

1. S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 8, pp. 87456–87477, 2020.
2. E. Thorn, S. Kimmel, and M. Chaka, "A framework for testable cases and scenarios for automated driving systems," National Highway Traffic Safety Administration, Report No. DOT HS 812 623, 2018.
3. S. Feng, Y. Feng, C. Yu, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, Part I: Methodology," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1573–1582, 2021.
4. U.S. Department of Transportation, *Automated vehicles comprehensive plan*, Office of the Secretary, 2021.
5. G. Li, Y. Li, S. Jha, T. Tsai, M. Sullivan, S. K. S. Hari, Z. Kalbarczyk, and R. Iyer, "AV-FUZZER: Finding safety violations in autonomous driving systems," in *Proceedings of the 31st IEEE International Symposium on Software Reliability Engineering (ISSRE)*, pp. 25–36, 2020.

6. Z. Zhong, G. Kaiser, and B. Ray, "Neural network guided evolutionary fuzzing for finding traffic violations of autonomous vehicles," *IEEE Transactions on Software Engineering*, vol. 49, no. 4, pp. 1992–2007, 2023.
7. H. Tian, G. Wu, J. Yan, Y. Jiang, J. Wei, W. Chen, S. Li, and D. Ye, "Generating critical test scenarios for autonomous driving systems via influential behavior patterns," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2022.
8. S. Feng, H. Sun, X. Yan, H. Zhu, Z. Zou, S. Shen, and H. X. Liu, "Dense reinforcement learning for safety validation of autonomous vehicles," *Nature*, vol. 615, pp. 620–627, 2023.
9. C. Hildebrandt, M. von Stein, and S. Elbaum, "PhysCov: Physical test coverage for autonomous vehicles," in *Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)*, pp. 449–461, 2023.
10. M. Cheng, Y. Zhou, and X. Xie, "BehAVExplor: Behavior diversity guided testing for autonomous driving systems," in *Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)*, pp. 488–500, 2023.
11. D. J. Fremont, T. Dreossi, S. Ghosh, X. Yue, A. L. Sangiovanni-Vincentelli, and S. A. Seshia, "Scenic: A language for scenario specification and data generation," in *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, 2019.
12. Y. Huai, S. Almanee, Y. Chen, X. Wu, Q. A. Chen, and J. Garcia, "scenoRITA: Generating diverse, fully mutable, test scenarios for autonomous vehicle planning," *IEEE Transactions on Software Engineering*, vol. 49, no. 10, pp. 4656–4676, 2023.
13. W. Ding, C. Xu, M. Arief, H. Lin, B. Li, and D. Zhao, "A survey on safety-critical driving scenario generation—A methodological perspective," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, pp. 6971–6988, 2023.
14. S. Kim, M. Liu, J. Rhee, Y. Jeon, Y. Kwon, and C. H. Kim, "DriveFuzz: Discovering autonomous driving bugs through driving quality-guided fuzzing," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2022.
15. T. Zohdinasab, V. Riccio, A. Gambi, and P. Tonella, "DeepHyperion: Exploring the feature space of deep learning-based systems through illumination search," in *Proceedings of the 30th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)*, 2021.
16. C. Xu, W. Ding, W. Lyu, Z. Liu, S. Wang, Y. He, H. Hu, D. Zhao, and B. Li, "SafeBench: A benchmarking platform for safety evaluation of autonomous vehicles," in *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2022.
17. D. Rempe, J. Philion, L. J. Guibas, S. Fidler, and O. Litany, "Generating useful accident-prone driving scenarios via a learned traffic prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17305–17315, 2022.
18. N. Hanselmann, K. Renz, K. Chitta, A. Bhattacharyya, and A. Geiger, "KING: Generating safety-critical driving scenarios for robust imitation via kinematics gradients," in *Proceedings of the European Conference on Computer Vision (ECCV)*, LNCS Vol. 13698, Springer, 2022.
19. A. Guo, H. Sun, Y. Yao, S. Chen, Z. Zhang, Y. Liu, and X. Xie, "SoVAR: Building generalizable scenarios from accident reports for autonomous driving testing," in *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2024.
20. S. Feng, X. Yan, H. Sun, Y. Feng, and H. X. Liu, "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment," *Nature Communications*, vol. 12, Article 748, 2021.