

AgentDrive: An Open Benchmark Dataset for Agentic AI Reasoning with LLM-Generated Scenarios in Autonomous Systems

Mohamed Amine Ferrag^{*§}, Senior Member, IEEE, Abderrahmane Lakas*, Senior Member, IEEE,
and Merouane Debbah¹, Fellow, IEEE

Abstract—The rapid advancement of Large Language Models (LLMs) has sparked growing interest in their integration within autonomous systems to enable reasoning-driven perception, planning, and decision-making. However, evaluating and training such agentic AI models remains challenging due to the lack of large-scale, structured, and safety-critical benchmarks. This paper introduces AgentDrive, an open benchmark dataset containing 300,000 LLM-generated driving scenarios designed for the training, fine-tuning, and evaluation of autonomous agents under diverse conditions. AgentDrive formalizes a factorized scenario space across seven orthogonal axes—scenario type, driver behavior, environment, road layout, objective, difficulty, and traffic density—and employs an LLM-driven prompt-to-JSON pipeline to produce semantically rich, simulation-ready specifications that are validated against physical and schema constraints. Each scenario undergoes simulation rollouts, surrogate safety metric computation, and rule-based outcome labeling. To complement simulation-based evaluation, we introduce AgentDrive-MCQ, a 100,000-question reasoning benchmark spanning five reasoning dimensions—physics, policy, hybrid, scenario, and comparative—to systematically assess the cognitive and ethical reasoning of LLM-based agents. We conducted a large-scale evaluation of fifty leading LLMs on the AgentDrive-MCQ benchmark to measure their reasoning capabilities across these five dimensions, covering models such as GPT-5, ChatGPT 4o, Gemini 2.5 Flash, DeepSeek V3, Qwen3 235B, ERNIE 4.5 300B, Grok 4, Mistral Medium 3.1, and Phi 4 Reasoning Plus. Results reveal that while proprietary frontier models dominate in contextual and policy reasoning, advanced open models are rapidly closing the gap in structured and physics-grounded reasoning. To support open science and reproducibility, we release the AgentDrive dataset (including labeled data), the AgentDrive-MCQ benchmark, evaluation scripts, and all related materials on GitHub: <https://github.com/maferrag/AgentDrive>.

Index Terms—Autonomous Driving, Large Language Models, Autonomous AI Agents, Benchmark Datasets, Reasoning.

I. INTRODUCTION

The rise of large language models (LLMs) has sparked broad interest in applying them to autonomous driving (AD) due to their strong reasoning and conversational abilities [1]. Researchers have introduced the concept of LLM4AD, designing AD systems that leverage LLMs for tasks ranging from perception and scene understanding to decision-making [2], [3], [4]. For example, Cui *et al.* [2] proposed an LLM4AD framework with a comprehensive simulation benchmark to

evaluate how well LLMs follow driving instructions. Initial experiments in both simulation and real-world driving showed that LLMs can indeed enhance an autonomous vehicle’s understanding of complex environments and improve human–vehicle interactions. These findings, echoed by recent surveys [1], [3], suggest that combining LLMs with vision models could enable more open-world perception, logical reasoning, and adaptive learning than traditional rule-based or end-to-end systems. At the same time, the LLM4AD concept paper and surveys emphasize that this nascent field faces significant challenges, such as ensuring real-time performance and safety, which must be addressed as development continues.

Integrating LLMs into an autonomous driving agent raises new complexities in how the vehicle interprets instructions, interacts with humans, and adheres to traffic rules. One approach to this is DriVLM, an LLM-based driving agent augmented with both embodied experiences (learned via a driving simulator) and social experiences from real human dialogues [5]. The goal of DriVLM is to facilitate natural communication between humans and self-driving cars and to handle long-horizon navigation tasks through free-form dialogue. DriVLM demonstrated competitive performance in open-loop simulations and closed-loop user studies; however, it also revealed several limitations, including unacceptably high inference times, imbalanced training data, and difficulty handling sudden environmental changes. To ensure LLM-driven vehicles make decisions that are not only effective but also legally and ethically aligned, other works focus on injecting domain knowledge and safeguards into the decision process. For example, Cai *et al.* [6] present a retrieval-augmented reasoning framework where a Traffic Regulation Retrieval module automatically fetches relevant traffic laws and safety guidelines based on the vehicle’s situation, and an LLM then interprets these rules to assess each action’s legality and safety. This approach yields an interpretable decision-making pipeline that adapts to regional regulations and provides transparency into why the AI chooses a certain action. Similarly, Kong *et al.* [7] introduce a multi-agent superalignment framework to enforce data security and policy compliance in LLM-driven cars. Their system safeguards sensitive vehicle data (e.g., precise locations, camera feeds) from potential leaks and filters the LLM’s queries and outputs, ensuring that driving commands do not violate safety rules or human values. These efforts demonstrate how researchers are enhancing LLM-based driving agents with embodied learning and alignment

*Department of Computer and Network Engineering, College of Information Technology, United Arab Emirates University, Al Ain, United Arab Emirates.

¹Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates.

[§]Corresponding author: mohamed.ferrag@uaeu.ac.ae

mechanisms, enabling autonomous vehicles to interact naturally while adhering to legal, safety, and trust constraints.

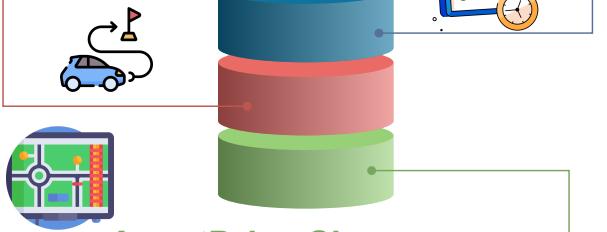
Beyond high-level dialogue and compliance, LLMs and multimodal models are being used to tackle core perception and planning challenges in autonomous driving. Bai *et al.* [8] argue that prior vision–language planning approaches, which tokenize only 2D images, cannot reliably perceive a 3D world. They propose a planner that uses DETR-style 3D perceptrons as tokenizers, feeding the LLM with 3D object queries from multi-view camera images. This strategy provided rich geometric context, yielding superior performance in 3D object detection and end-to-end planning on the nuScenes dataset, suggesting that 3D-tokenized LLMs could be key to more reliable path planning. Sah *et al.* [9] integrate deep learning models for traffic sign recognition and lane detection with a lightweight multimodal LLM that can reason about the scene. This hybrid system achieved near-perfect accuracy in clear conditions and maintained robustness under adverse weather and occlusions, demonstrating the LLM’s ability to interpret contextual clues. Other studies leverage LLMs for specialized tasks such as parking in mixed traffic [10] and interpretable end-to-end control. Chen *et al.* [11] fuse object-level vector modalities with LLM reasoning to provide explainable driving decisions, while Xu *et al.* [12] propose DriveGPT4, which generates both low-level control signals and natural language explanations for each action. These advances highlight LLMs as powerful engines for enhancing perception, planning, and transparency in AD systems.

To validate and further develop LLM-driven autonomous vehicles, the community is creating new benchmarks and evaluation tools to enhance their development. One notable effort is Bench2ADVLM, a closed-loop evaluation framework introduced by Zhang *et al.* [13] to test vision–language AD models in interactive settings. Bench2ADVLM enables real-time closed-loop testing across simulators and physical vehicles, with a scenario generator that uncovers failure modes. Results indicate that current systems still exhibit limitations under interactive conditions. In parallel, MAPLM [14] provides a large-scale multimodal dataset combining maps, LiDAR, panoramic images, and Q&A annotations, highlighting the importance of domain-specific data for grounding LLMs in driving contexts. Gao *et al.* [4] survey how foundation models can generate and analyze diverse scenarios, showing their potential for evaluating and handling safety-critical cases. Despite progress, challenges persist in latency, interpretability, privacy, regulatory compliance, and robustness. Addressing these motivates the creation of AgentDrive, an open benchmark that exposes LLM-based agents to diverse, safety-critical, and long-horizon scenarios through systematic closed-loop evaluation.

Building on the challenges of scalable and realistic scenario generation, our work poses two key research questions for agentic AI evaluation:

AgentDrive-Gen AgentDrive-MCQ

A large-scale dataset of 300 K **LLM-generated driving scenarios** encoded in structured JSON, covering seven orthogonal axes — scenario type, driver behavior, environment, road layout, objective, difficulty, and traffic density — ensuring semantic richness and simulation readiness.



AgentDrive-Sim

A simulation and labeling dataset with 300 K scenario executions, each producing surrogate safety metrics (e.g., time-to-collision, headway distance) and interpretable outcome labels (safe_goal, safe_stop, inefficient, unsafe). Enables supervised learning and benchmarking.

Fig. 1: Overview of the *AgentDrive* benchmark suite, which comprises three complementary datasets: *AgentDrive-Gen* for LLM-based driving scenario generation, *AgentDrive-Sim* for simulation and outcome labeling, and *AgentDrive-MCQ* for reasoning and decision-making evaluation.

Research Questions

RQ1: How can Large Language Models (LLMs) be systematically leveraged to generate diverse, semantically rich, and safety-critical autonomous driving scenarios that enable large-scale training, fine-tuning, and evaluation of agentic AI systems?

RQ2: How can structured simulation data and reasoning-based benchmarks be unified to comprehensively assess the cognitive, ethical, and physics-grounded reasoning capabilities of LLMs in autonomous systems?

RQ3: To what extent can state-of-the-art LLMs—both proprietary and open-source (e.g., GPT-5, ChatGPT 4o, DeepSeek V3, Qwen3 235B, ERNIE 4.5 300B, Grok 4)—demonstrate consistent and reliable reasoning performance across multiple driving-related reasoning dimensions?

In response to these research questions, we introduce AgentDrive, an open benchmark dataset constructed from LLM-generated scenarios for evaluating and training agentic AI models in autonomous systems, as presented in Fig. 1. AgentDrive is built around an end-to-end generation and evaluation pipeline that samples from a factorized scenario space, encodes each instance into a structured specification via an LLM-driven prompt, validates and executes it in simulation, computes surrogate safety metrics, and assigns interpretable outcome labels. Furthermore, we extend this framework with

AgentDrive-MCQ, a reasoning-oriented benchmark designed to evaluate the cognitive and decision-making capabilities of LLMs. The key contributions of this work are summarized as follows:

- 1) **Factorized Scenario Space:** We formalize the autonomous driving scenario space across seven orthogonal axes—*scenario type, driver behavior, environment, road layout, objective, difficulty, and traffic density*—to ensure comprehensive coverage of both routine and rare safety-critical situations.
- 2) **AgentDrive-Gen:** We developed AgentDrive-Gen, a large-scale benchmark comprising 300,000 LLM-generated driving scenarios. It is built upon a prompt-engineering and schema-validation framework that transforms abstract scenario tuples into valid JSON specifications. The pipeline leverages large language models to generate semantically rich, physically consistent, and simulation-ready representations suitable for both training and evaluation.
- 3) **AgentDrive-Sim:** Each generated scenario from AgentDrive-Gen is executed within a simulator to produce dynamic rollouts. Surrogate safety metrics (e.g., minimum time-to-collision and headway distance) are computed, and a rule-based labeling system assigns interpretable outcome categories (*safe_goal, safe_stop, inefficient, unsafe*). The resulting dataset, denoted AgentDrive-Sim, serves as a large-scale benchmark for supervised learning and performance evaluation in safety-critical driving scenarios.
- 4) **AgentDrive-MCQ:** We introduced AgentDrive-MCQ, a reasoning-focused extension of AgentDrive comprising 100,000 multiple-choice questions. Each question is systematically derived from structured scenarios and categorized into five reasoning styles—*physics, policy, hybrid, scenario, and comparative*. This benchmark enables large-scale, fine-grained evaluation of LLM reasoning and decision-making capabilities across diverse driving conditions, complementing simulation-based performance with structured reasoning assessment.
- 5) **Large-Scale LLM Evaluation:** We conduct a comprehensive evaluation of fifty leading LLMs—including GPT-5, ChatGPT 4o, Gemini 2.5 Flash, DeepSeek V3, Qwen3 235B, ERNIE 4.5 300B, Grok 4, and Mistral Medium 3.1—on the AgentDrive-MCQ benchmark to measure their reasoning capabilities across five dimensions: *comparative, physics, policy, hybrid, and scenario*. Results demonstrate that while proprietary frontier models lead in contextual and policy reasoning, advanced open models are rapidly closing the gap in structured and physics-grounded reasoning.

The remainder of this paper is organized as follows. Section II reviews related work on LLM-based autonomous driving benchmarks and reasoning evaluation. Section III details the AgentDrive dataset generation methodology, including scenario design, prompt engineering, simulation, and labeling processes. Section IV introduces the AgentDrive-MCQ benchmark, which extends AgentDrive into structured reasoning

evaluation. Section V presents the reasoning performance of fifty leading large language models on AgentDrive-MCQ across multiple reasoning dimensions. Finally, Section VI concludes the paper and outlines future research directions.

II. RELATED WORK

Recent research has explored a wide range of approaches to integrating large language models into autonomous driving systems. These works span from instruction-following frameworks and cooperative driving systems to benchmarks for scenario understanding, theory testing, and automated scenario generation. For clarity, we group the discussion thematically into five main categories: (i) LLM-augmented autonomous driving agents, (ii) LLM-based scenario and code generation, (iii) benchmarks for driving scenario reasoning, (iv) driving theory tests for LLMs, and (v) our positioning of AgentDrive in this landscape. A comparative overview of these works, including their focus, key features, and limitations, is provided in Table I.

A. LLM-Augmented Autonomous Driving Agents

Ma *et al.* [15] introduced LaMPilot, a framework that integrates LLMs into autonomous driving systems to interpret high-level instructions and translate them into low-level driving code using predefined primitives. Alongside the framework, they released LaMPilot-Bench, a benchmark to evaluate how well LLM-based agents follow such instructions across various scenarios. Their results showed that large models can generalize across multiple driving tasks, highlighting the potential of LLMs for AD instruction-following. In the multi-vehicle context, Chiu *et al.* [17] proposed V2V-LLM, which leverages multimodal LLMs for vehicle-to-vehicle cooperative driving. By merging perception data from multiple connected cars, V2V-LLM enables reasoning about occluded hazards and planning joint maneuvers. Their evaluations demonstrated that the approach outperforms traditional early-fusion baselines in grounding objects and supporting collaborative driving strategies.

B. LLM-Based Scenario and Code Generation

Beyond direct control, LLMs have also been used to generate simulation content. Lebioda *et al.* [18] studied whether requirements in natural language can be transformed into configuration code for driving simulators. Using a case study in CARLA, they demonstrated that while LLMs can generate functional code, human intervention is still required to address errors and omissions, underscoring the limitations of current models in automated scenario coding. In contrast, Yao *et al.* [19] introduced AGENTS-LLM, a framework for augmenting real-world traffic scenarios with LLM-driven modifications. Their agentic approach allows incremental scenario editing (e.g., introducing a red-light violation) while maintaining physical plausibility. Human evaluations confirmed that AGENTS-LLM generates realistic yet rare edge cases, providing scalable stress-testing for autonomous planners.

TABLE I: Comparison of related works on LLM- and VLM-based autonomous driving benchmarks and evaluation.

Work	Year	Focus / Contribution	Key Features	Limitations
<i>LaMPilot</i> [15]	2024	Instruction-following AD with LLM-generated driving code	Benchmark and framework translating text to executable primitives	Limited to a pre-defined action space and code translation; does not generate structured simulation scenarios or reasoning benchmarks
Tang <i>et al.</i> [16]	2024	Driving theory evaluation of LLMs	500+ MCQs; comparison of GPT-3.5, GPT-4, etc.	Tests static theoretical knowledge only; lacks simulation-grounded tasks, generative scenario diversity, and reasoning integration
<i>V2V-LLM</i> [17]	2025	Cooperative vehicle-to-vehicle reasoning with multimodal LLMs	Shared perception fusion; V2V-QA dataset	Focused on connected vehicles and perception fusion; does not provide large-scale generative benchmarks or unified reasoning evaluation
Lebioda <i>et al.</i> [18]	2025	LLMs for automotive requirement-to-code translation	Case study: natural language → CARLA configuration	Semi-automated with frequent human correction; not designed for large-scale generative or reasoning-driven evaluation
<i>AGENTS-LLM</i> [19]	2025	Agentic LLM for scenario augmentation	Incremental modification of real traffic logs; realistic rare cases	Relies on existing driving data; lacks fully generative, simulation-based, or reasoning-oriented evaluation
Zhou <i>et al.</i> [20]	2025	Benchmarking LLMs on motorway driving scenarios	Text-based scenario understanding; comparison of six LLMs	Restricted to motorway conditions and textual QA; does not incorporate generative simulation or diverse reasoning styles
<i>STS-Bench</i> [21]	2025	Spatio-temporal reasoning benchmark	Multi-camera nuScenes QA; 43 scenario types, 971 MCQs	Focused on visual QA; not simulation-grounded and lacks LLM-generated structured scenarios or agentic reasoning coverage
<i>AD²-Bench</i> [22]	2025	Hierarchical CoT benchmark in adverse conditions	5.4k annotated reasoning chains; step-by-step evaluation	Limited to perception and reasoning under predefined conditions; no generative scenario modeling or simulation-based evaluation
Pei <i>et al.</i> [23]	2025	Theory + hazard perception benchmark	700 MCQs, 54 hazard videos; real exam format	Focused on driver testing and hazard perception; lacks generative scenarios, simulation rollouts, and reasoning diversity
<i>DriveBench</i> [24]	2025	Evaluation of Vision-Language Models (VLMs) for reliability and grounding	<i>DriveBench</i> (VLM): 19,200 frames, 20,498 QA pairs across perception, prediction, planning, and behavior; 12 VLMs tested under 17 conditions (clean, corrupted, text-only); refined robustness metrics	Focuses on VLM robustness and reliability; not a generative benchmark—uses fixed datasets; emphasizes perception/prediction rather than textual scenario generation or reasoning diversity
AgentDrive (<i>ours</i>)	2025	Unified benchmark suite for generative simulation and reasoning-driven evaluation of agentic AI models	AgentDrive: 300K LLM-generated, simulation-ready driving scenarios across seven orthogonal axes (scenario type, driver behavior, environment, road layout, objective, difficulty, traffic density); structured JSON schema, surrogate safety metrics, and rule-based outcome labeling. AgentDrive-MCQ: 100K reasoning questions covering five dimensions (<i>physics, policy, hybrid, scenario, comparative</i>); large-scale evaluation of 50 LLMs with open-source scripts for training and assessment.	Currently limited to text-based reasoning where visual contexts are represented as textual descriptions; multimodal (visual-language) LLMs to be integrated in future work. Unlike prior works, it introduces the first fully generative, simulation-grounded, and reasoning-oriented benchmark for evaluating autonomous agent cognition.

C. Benchmarks for Driving Scenario Reasoning

Zhou *et al.* benchmarked six LLMs on motorway driving scenario understanding, focusing on their ability to interpret textual descriptions of traffic situations [20]. Their results offer insights into the relative strengths of LLMs for functional driving scenario reasoning. Fruhwirth-Reisinger *et al.* developed STSBench, a spatio-temporal benchmark derived from nuScenes that evaluates multimodal LLMs on reasoning across time and multiple views [21]. Covering 43 scenario types with nearly 1,000 human-verified multiple-choice questions, STSBench revealed that state-of-the-art vision-language models still struggle with spatio-temporal reasoning in traffic. Similarly, Wei *et al.* presented AD²-Bench, a benchmark targeting adverse driving conditions such as dense traffic, fog, and nighttime environments [22]. With more than 5,000 annotated reasoning chains, AD²-Bench highlights the difficulty multimodal current LLMs face in step-by-step reasoning under safety-critical conditions.

D. Driving Theory Tests for LLMs

Several works have drawn inspiration from human licensing tests to evaluate LLMs' knowledge of driving rules. Tang *et al.* tested multiple LLMs on driving theory questions, reporting that only GPT-4 consistently exceeded the passing threshold, while other models lacked sufficient domain knowledge [16]. Pei *et al.* expanded this approach by creating a benchmark of over 700 multiple-choice questions and 54 hazard perception videos [23]. Their findings showed that while GPT-4 performed strongly on theoretical questions, no model successfully passed the hazard perception component, revealing current limitations in dynamic traffic understanding.

E. AgentDrive in Context

Recent research has demonstrated significant progress in applying large language models to autonomous driving, spanning instruction following, cooperative reasoning, requirement translation, and domain-specific benchmarking. However, existing studies typically address isolated components of the autonomy

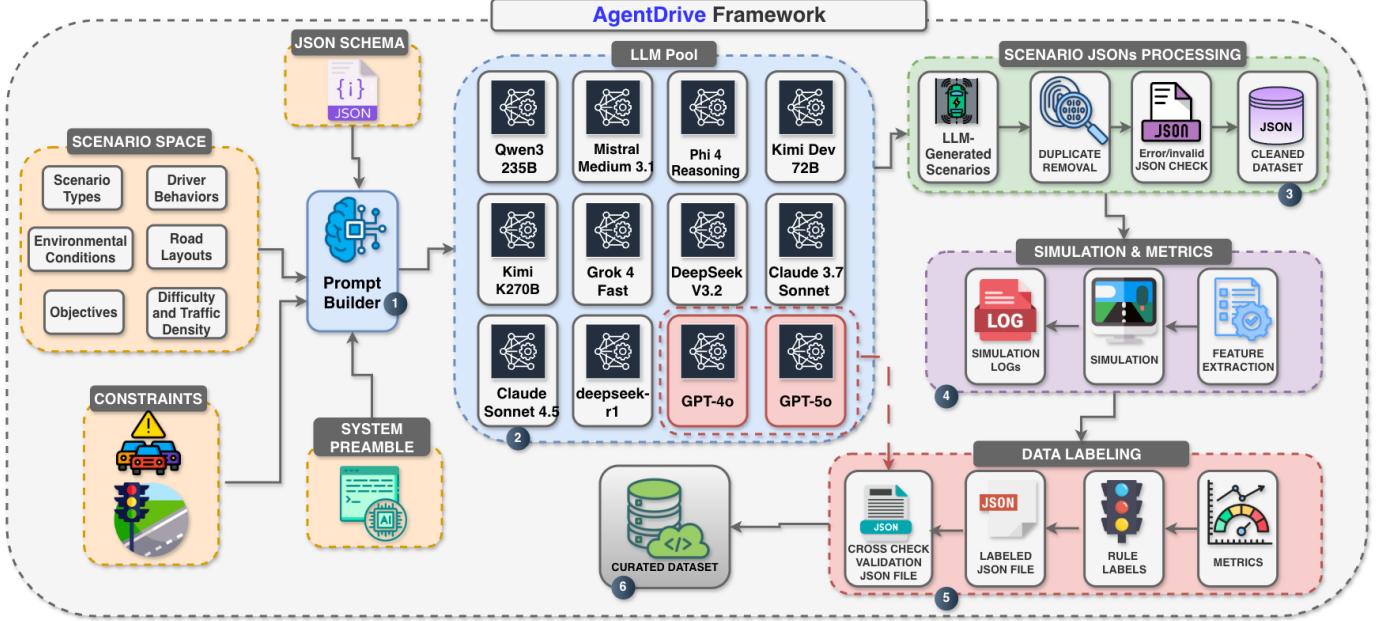


Fig. 2: AgentDrive Framework: End-to-End Pipeline for LLM-Generated Autonomous Driving Scenarios.

pipeline—such as perception, code synthesis, or theoretical reasoning—without providing a unified and generative framework that integrates scenario creation, simulation, and reasoning evaluation.

AgentDrive advances this landscape by introducing a fully generative, simulation-grounded, and reasoning-oriented benchmark that bridges these dimensions. It defines a formal compositional scenario space and employs LLM-driven prompt engineering to generate 300K structured driving scenarios validated through a JSON schema. Each scenario is grounded in high-fidelity simulation rollouts, where surrogate safety metrics are computed, and rule-based labeling provides interpretable outcome categories. In addition, entropy-maximized sampling ensures balanced coverage across scenario dimensions, including rare safety-critical conditions.

Beyond simulation, AgentDrive-MCQ extends the benchmark into the reasoning domain with 100K multiple-choice questions that assess LLM understanding across physics, policy, hybrid, scenario, and comparative reasoning styles. This unified framework—combining generative scenario synthesis with structured reasoning evaluation—positions AgentDrive as the first principled benchmark for assessing the robustness, safety, and generalization capabilities of agentic AI systems in autonomous driving.

III. AGENTDRIVE: DATASET GENERATION METHODOLOGY

This section presents our proposed pipeline for generating, simulating, and labeling autonomous driving scenarios. The methodology is structured into six key components: (i) definition of the scenario space, (ii) LLM-driven specification, (iii) simulation rollout generation, (iv) computation of surrogate safety metrics, (v) rule-based labeling, and (vi) construction of the data set. Each component is described in detail with a mathematical formalization that clarifies the problem setting

and highlights the reproducibility of our framework. The primary goal is to transform abstract scenario descriptions into structured, simulation-ready inputs that yield diverse, safety-critical driving trajectories. Table II presents the key symbols, their corresponding methodological blocks, and the concise definitions used throughout the dataset generation pipeline.

Figure 2 illustrates the complete workflow of AgentDrive, which transforms abstract scenario specifications into curated, simulation-ready datasets. The process begins with a factorized scenario space that encodes driver behaviors, road layouts, environmental conditions, objectives, difficulty levels, and traffic density. Structured prompts are then built and passed to a diverse pool of large language models (LLMs), which generate candidate scenario JSONs. These outputs undergo schema validation, error checks, duplicate removal, and cross-model consistency verification before being added to the cleaned dataset. Validated scenarios are executed in simulation, during which logs and surrogate safety metrics are collected. This is followed by rule-based labeling to assign interpretable outcome categories such as safe stop, safe goal, unsafe, or inefficient. The final curated dataset provides a diverse, safety-critical benchmark that supports the training and evaluation of agentic AI systems for autonomous driving.

A. Scenario Space Definition

The foundation of reliable autonomous driving evaluation lies in constructing a representative and diverse scenario space. To formalize this, we define a scenario as a tuple:

$$s = (t, b, e, r, o, d, q), \quad (1)$$

where $t \in \mathcal{T}$ represents the scenario type (e.g., lane change, intersection crossing, merging), $b \in \mathcal{B}$ denotes the driver behavior (e.g., compliant, distracted, or aggressive), $e \in \mathcal{E}$ captures environmental conditions (e.g., rain, fog, or low

visibility), $r \in \mathcal{R}$ defines the road layout (e.g., straight highway, urban intersection, or roundabout), $o \in \mathcal{O}$ specifies the ego-vehicle objective (e.g., safe navigation, overtaking, or emergency stop), $d \in \mathcal{D}$ quantifies the difficulty level (e.g., easy, moderate, or complex), and $q \in \mathcal{Q}$ encodes traffic density (e.g., sparse, medium, or congested).

The overall scenario space is expressed as the Cartesian product of these seven axes:

$$\mathcal{S} = \mathcal{T} \times \mathcal{B} \times \mathcal{E} \times \mathcal{R} \times \mathcal{O} \times \mathcal{D} \times \mathcal{Q}. \quad (2)$$

This formalization provides a structured and extensible framework for representing the heterogeneity of real-world driving contexts. By sampling along each axis, we ensure broad coverage of both nominal and non-nominal conditions. Crucially, this includes rare but critical safety scenarios, such as malfunctioning traffic signals, sensor-blinding weather, or adversarial driver behaviors. These edge cases, though infrequent in naturalistic datasets, are disproportionately influential in assessing system robustness and generalization.

Moreover, the factorized design of \mathcal{S} enables systematic stress-testing across dimensions. For example, combining aggressive driver behavior (b) with high traffic density (q) under adverse weather (e) can generate compound scenarios that expose the limitations of perception, prediction, and planning modules. Hence, the scenario space not only supports comprehensive coverage but also facilitates controlled experimentation to benchmark agentic AI systems driven by LLM-generated scenarios.

B. Scenario Taxonomy and Diversity

The richness of the scenario space \mathcal{S} (Eq. 2) emerges from its decomposition into multiple orthogonal axes, each capturing a distinct facet of the driving task. This factorized view not only supports structured exploration of safety-critical situations but also enables controlled variation along individual dimensions. Below, we outline the key axes and their corresponding taxonomies.

1) Scenario Types.: We define a comprehensive set of driving situations that span both routine and edge-case conditions. Representative examples include *red-light dilemma zones*, *yield/priority confusion*, *highway merging at high speed*, *icy bridge segments*, and *pedestrian jaywalking*. The taxonomy explicitly incorporates vulnerable road users (VRUs), environmental hazards, and mechanical failures (e.g., brake loss, tire blowouts). By explicitly encoding corner cases, the taxonomy ensures systematic stress-testing of autonomous agents in situations rarely observed in naturalistic driving logs but disproportionately critical for safety validation.

2) Driver Behaviors.: To capture the heterogeneity of human driving styles, we define behavioral profiles ranging from *calm, cooperative, or rule-following* to *aggressive, impatient, distracted, or impaired*. We also account for role-specific driving modes, such as *taxi-style stop-and-go, delivery routing under time pressure*, and *convoy following*. This axis ensures that autonomous systems are evaluated not only against compliant traffic participants but also against unpredictable, adversarial, and non-standard agents, reflecting the variability of real-world traffic ecosystems.

TABLE II: Notation used throughout the dataset generation methodology.

Symbol	Block	Definition
s	Scenario Space	Scenario tuple $s = (t, b, e, r, o, d, q)$
$t \in \mathcal{T}$	Scenario Space	Scenario type (e.g., lane change, intersection)
$b \in \mathcal{B}$	Scenario Space	Driver behavior (e.g., compliant, aggressive)
$e \in \mathcal{E}$	Scenario Space	Environmental conditions (e.g., rain, fog)
$r \in \mathcal{R}$	Scenario Space	Road layout/topology
$o \in \mathcal{O}$	Scenario Space	Ego objective (e.g., overtake, safe stop)
$d \in \mathcal{D}$	Scenario Space	Difficulty level {easy, medium, hard}
$q \in \mathcal{Q}$	Scenario Space	Traffic density {low, medium, high}
\mathcal{S}	Scenario Space	Cartesian product $\mathcal{T} \times \mathcal{B} \times \mathcal{E} \times \mathcal{R} \times \mathcal{O} \times \mathcal{D} \times \mathcal{Q}$
$H(d)$	Scenario Space	Difficulty-specific numeric constraints (e.g., min_ttc, max_accel)
$P(s, H(d))$	LLM Spec	Structured prompt combining $f(s)$ with $H(d)$
$f(s)$	LLM Spec	Natural-language encoding of the scenario tuple s
\mathcal{G}_ϕ	LLM Spec	LLM generator with parameters ϕ
\hat{j}	LLM Spec	Raw JSON scenario generated by \mathcal{G}_ϕ
C	LLM Spec	JSON schema / structural constraints
Π	LLM Spec	Post-processing/repair module enforcing C
R	LLM Spec	Maximum repair/retry attempts for Π
j^*	LLM Spec	Repaired/validated JSON after Π
\mathcal{M}	Simulation	Simulator configured by \hat{j}^*
$\tau = \{x_t\}_{t=0}^{T-1}$	Simulation	Rollout (state trajectory) of horizon T
T	Simulation	Number of discrete time steps (duration_steps)
$x_t \in \mathbb{R}^d$	Simulation	State vector at time t (ego, traffic, signals, environment)
Δt	Simulation	Time step, $\Delta t = 1/\text{policy_frequency}$
$\text{TTC}(t)$	Metrics	Instantaneous time-to-collision at time t
TTC_{\min}	Metrics	Episode-level minimum TTC, $\min_t \text{TTC}(t)$
$x_E(t), x_V(t)$	Metrics	Longitudinal positions of ego E and lead vehicle V
$v_E(t), v_V(t)$	Metrics	Velocities of ego E and lead vehicle V
θ_{unsafe}	Metrics	Unsafe TTC threshold (e.g., 0.5 s)
θ_{near}	Metrics	Near-miss TTC threshold (e.g., 1.0 s)
collision	Labeling	Detected crash event (boolean)
red_violation	Labeling	Crossed stopline on red (boolean)
stopped_on_red	Labeling	Full stop before stopline on red (boolean)
crossed_green	Labeling	Crossed stopline on green (boolean)
Y	Labeling	Outcome label $\in \{\text{unsafe}, \text{safe_goal}, \text{safe_stop}, \text{inefficient}\}$
ℓ	Labeling	Assigned categorical label for a rollout
\mathcal{D}	Dataset	Final dataset $\{(s_i, \hat{j}_i, \tau_i, \ell_i)\}_{i=1}^N$
N	Dataset	Number of generated scenarios/entries
$\mathcal{H}(\cdot)$	Dataset	Entropy used to encourage axis-wise coverage
$\{axis_i\}$	Dataset	Sequence of sampled values along a given axis ($i = 1..N$)
policy_frequency	Params	Control frequency (Hz) determining Δt
duration_steps	Params	Discrete horizon length for each scenario

3) Environmental Conditions.: Real-world conditions vary widely across geography, climate, and time of day. Our taxonomy spans clear daytime driving, low-visibility situations such as *foggy night driving*, and extreme weather events including *hailstorms*, *sandstorms*, and *blinding low-sun glare*.

We also incorporate urban-specific challenges such as *neon light glare at night*, which degrade visual perception systems. These variations directly reflect the operational design domains (ODDs) outlined by regulatory bodies and expose the fragility of perception and control pipelines under degraded sensing conditions.

4) *Road Layouts*.: Road geometry constitutes another axis of complexity. We include highways, interchanges, roundabouts, rural roads, mountain passes, tunnels, and bridges with varying constraints such as *narrow lanes*, *limited visibility*, or *sharp curves*. By systematically varying these topologies, the framework probes the limits of map-based localization, lane-level reasoning, and motion planning under non-trivial geometric constraints.

5) *Objectives*.: The ego vehicle’s intended task defines the contextual goal for each scenario. Objectives range from regulatory compliance (e.g., *obey red light*, *respect pedestrian crossings*, *use turn signals*) to maneuvering (e.g., *merge smoothly*, *overtake safely*, *execute U-turns*). Defensive objectives, such as *maintaining safe headway* or *avoiding collision*, are also integrated, aligning with safety-critical evaluation metrics. By conditioning on objectives, scenarios can be tailored to assess specific capabilities of planning and decision-making modules.

6) *Difficulty and Traffic Density*.: To control scenario challenge in a principled way, we vary difficulty levels {easy, medium, hard} and traffic densities {low, medium, high}. Each difficulty level is mapped to structured constraints $H(d)$ that define numerical thresholds for safety-critical quantities such as minimum time-to-collision (TTC) and maximum admissible acceleration:

$$H(\text{easy}) = \{\min_{\text{ttc}} = 4.0 \text{ s}, \max_{\text{accel}} = 2.5 \text{ m/s}^2\}, \quad (3)$$

$$H(\text{medium}) = \{\min_{\text{ttc}} = 2.5 \text{ s}, \max_{\text{accel}} = 4.0 \text{ m/s}^2\}, \quad (4)$$

$$H(\text{hard}) = \{\min_{\text{ttc}} = 1.5 \text{ s}, \max_{\text{accel}} = 6.0 \text{ m/s}^2\}. \quad (5)$$

These constraints serve as structured hints to the LLM during scenario generation, ensuring that the resulting instances conform to quantifiable levels of risk and challenge.

By combining these axes, AgentDrive systematically generates scenarios that balance routine coverage with exposure to rare, safety-critical events. This design positions AgentDrive not only as a diverse dataset but also as a principled benchmark for robustness, safety, and generalization in autonomous systems powered by agentic AI.

C. Scenario Specification via LLM

Once a scenario tuple is sampled, the next challenge is to transform its abstract representation into a structured, semantically rich specification suitable for simulation. This transformation requires bridging symbolic scenario definitions with executable formats that respect both linguistic expressiveness and physical realism.

Formally, for each sampled scenario s , we construct a prompt:

$$P(s, H(d)) = f(s) \cup H(d), \quad (6)$$

where $f(s)$ encodes the axis tuple into a natural language description, and $H(d)$ provides difficulty-specific numerical hints (e.g., minimum time-to-collision thresholds, maximum acceleration bounds). The prompt thus combines contextual semantics with quantitative constraints, guiding the generative process.

A large language model (LLM) \mathcal{G}_ϕ then produces a structured JSON representation:

$$\hat{j} = \mathcal{G}_\phi(P(s, H(d))), \quad (7)$$

which must conform to a schema \mathcal{C} . Generations that violate \mathcal{C} are flagged as invalid and corrected by a post-processing module Π , which retries up to R attempts:

$$\hat{j}^* = \Pi(\hat{j}, \mathcal{C}, R). \quad (8)$$

To operationalize the process, we define a structured prompt consisting of three components:

- **System preamble:**

You are a driving scenario planner for a highway simulator. Return a SINGLE valid JSON object that exactly matches the schema. Do not include comments or explanations|JSON ONLY.

- **Scenario description $f(s)$ and difficulty hints $H(d)$, e.g.,** “Scenario type: red-light dilemma, driver: aggressive, environment: foggy night, road: urban intersection, objective: avoid collision, difficulty: hard, traffic density: high. Constraints: min_ttc=1.5s, max_accel=6.0 m/s²”

- **Constraints:**

Respect physics: |accel| ≤ 6 m/s², v_mps ≥ 0, lanes are 0-indexed. If traffic_light present: ego spawn.x ≥ 30 m before stopline_x. duration_steps ≥ red_steps + green_steps. Include seed, policy_frequency, and all required fields.

The generated JSON must conform to the schema \mathcal{C} , summarized in Table III.

Figure 3 presents an overview of the AgentDrive scenario dataset, which comprises around 300,000 samples, highlighting its diversity across difficulty levels, road layouts, and environmental conditions. Subfigure (a) shows a well-balanced distribution of scenario difficulty, with roughly one-third of the samples categorized as *easy*, *medium*, and *hard*. Subfigure (b) illustrates the correlation between layout type and difficulty, indicating that complex road structures—such as *hairpin turns* and *cloverleaf interchanges*—are generally associated with higher difficulty levels. Subfigure (c) lists the ten most frequent road layouts, showcasing the dataset’s variety of geometric configurations. Subfigures (d)–(f) capture temporal and environmental diversity, showing that most scenarios occur during *daytime*, under *good visibility*, and in *clear weather*—conditions that ensure both realism and safety-critical coverage for evaluating autonomous driving systems.

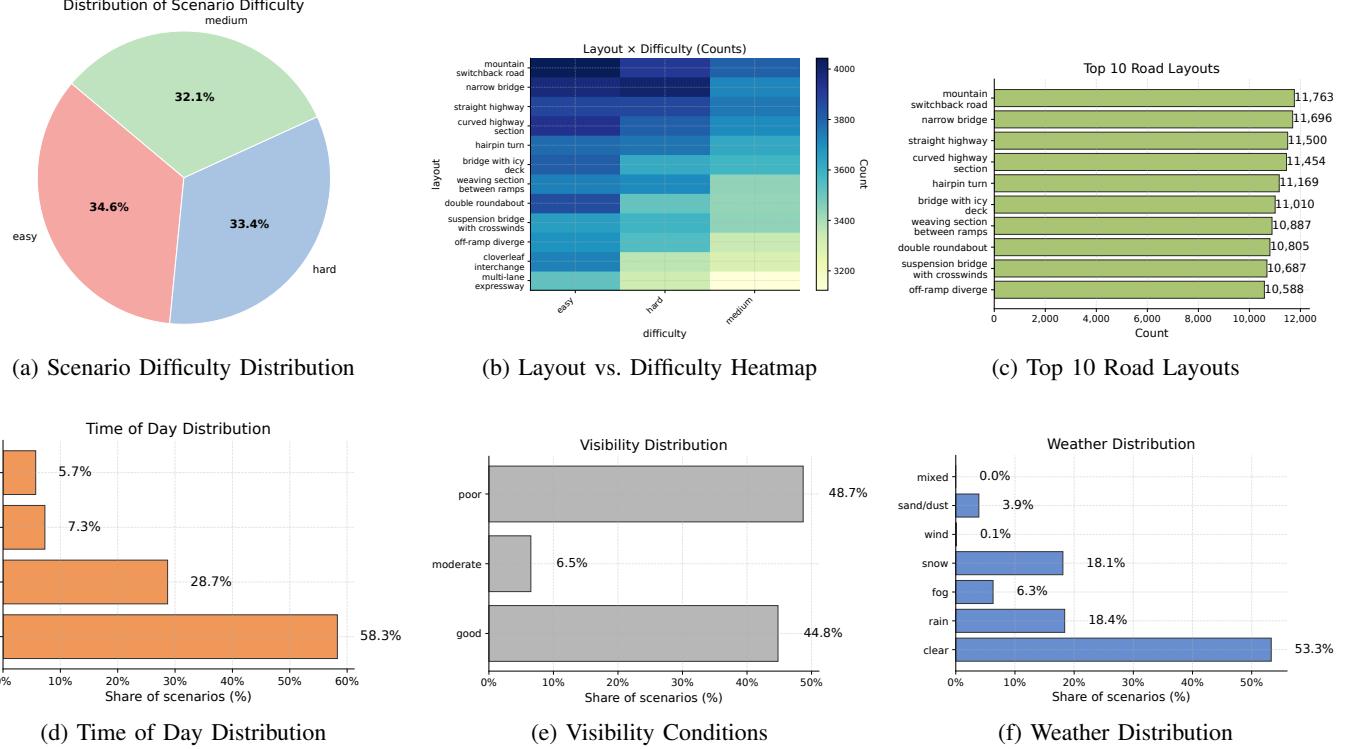


Fig. 3: Distributions and layout statistics across the AgentDrive scenario dataset, showing difficulty levels, layout correlations, temporal and environmental characteristics.

TABLE III: Schema for LLM-generated scenarios .

Field	Description / Constraints
name	Short slug, e.g., RedLight_Merge_v1
seed	Non-negative integer
duration_steps	Integer ≥ 60 (0.1s per step at policy_frequency=10)
road	{lanes ≥ 2 , speed_limit_kph $\in [60, 140]$ }
traffic_light	Stopline position, red/green durations
environment	Weather, time of day, visibility
layout	Text description of road geometry
objective	Ego-vehicle task (e.g., avoid collision)
difficulty	easy medium hard
traffic_density	low medium high
ego_spawn	{x, y, v_mps ≥ 0 }, plus goal description
traffic[]	List of agents (type, behavior, lane, spawn, velocity)
events[]	Optional timed events (e.g., sudden brake, cut-in)
metrics[]	Evaluation metrics (TTC, headway, collisions)

D. Simulation Rollout Generation

Once a valid JSON specification \hat{j} has been generated, the simulator \mathcal{M} is implemented using the highway-env framework [25] and configured to produce a trajectory (or rollout):

$$\tau = \{x_t\}_{t=0}^{T-1}, \quad (9)$$

where T is the horizon determined by duration_steps, and each state $x_t \in \mathbb{R}^d$ encodes ego-vehicle kinematics (position, velocity, acceleration), surrounding traffic states, traffic light signals, and relevant environmental parameters. The simulation time step is governed by the policy frequency:

$$\Delta t = \frac{1}{\text{policy_frequency}}. \quad (10)$$

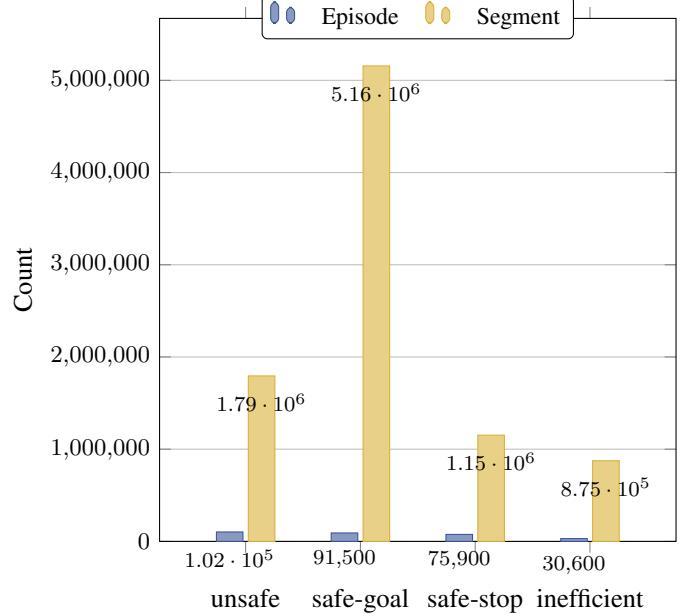


Fig. 4: Label distribution of AgentDrive-Sim across episode and segment levels.

The rollout τ captures the temporal dynamics of vehicle interactions, providing a reproducible and controllable environment for downstream analysis. The use of highway-env [25] allows for efficient prototyping and standardized simulation of diverse driving scenarios, including highway, intersection, and merging behaviors.

E. Surrogate Safety Metrics

To evaluate safety outcomes at scale without requiring costly real-world testing, we rely on surrogate safety metrics. Among these, *Time-to-Collision (TTC)* is the primary indicator of criticality. The instantaneous TTC between ego vehicle E and its nearest lead vehicle V is:

$$\text{TTC}(t) = \begin{cases} \frac{x_V(t) - x_E(t)}{v_E(t) - v_V(t)}, & v_E(t) > v_V(t) \wedge x_V(t) > x_E(t), \\ +\infty, & \text{otherwise,} \end{cases} \quad (11)$$

where $x_E(t)$, $x_V(t)$ are longitudinal positions, and $v_E(t)$, $v_V(t)$ are velocities of ego and lead vehicles. The episode-level critical TTC is given by:

$$\text{TTC}_{\min} = \min_t \text{TTC}(t). \quad (12)$$

We define thresholds $\theta_{\text{unsafe}} = 0.5\text{s}$ and $\theta_{\text{near}} = 1.0\text{s}$, consistent with established traffic safety analysis standards. These thresholds allow us to classify events into unsafe, near-miss, and safe zones. Such surrogate metrics provide interpretable signals that map closely to real-world safety concepts, enabling automated large-scale safety assessment.

F. Rule-Based Labeling

While continuous metrics quantify risk, categorical labels improve interpretability and facilitate supervised learning. From each rollout τ , we detect discrete events:

$$\begin{aligned} \text{collision} &= (\text{ego crashed}), \\ \text{red_violation} &= (\text{crossed stopline} \wedge (\text{light} = \text{Red})), \\ \text{stopped_on_red} &= (\neg \text{crossed stopline} \wedge (v_E < \epsilon) \wedge (\text{light} = \text{Red})), \\ \text{crossed_green} &= (\text{crossed stopline} \wedge (\text{light} = \text{Green})). \end{aligned} \quad (13)$$

These events are mapped into high-level outcome labels:

$$\begin{aligned} Y = \text{unsafe}, & \quad \text{if collision} \vee \text{red_violation} \vee \text{TTC}_{\min} < \theta_{\text{unsafe}}, \\ Y = \text{safe_goal}, & \quad \text{if stopped_on_red} \wedge \text{crossed_green}, \\ Y = \text{safe_stop}, & \quad \text{if stopped_on_red} \wedge \neg \text{crossed_green}, \\ Y = \text{inefficient}, & \quad \text{otherwise.} \end{aligned} \quad (14)$$

This labeling strategy strikes a balance between precision and interpretability. It assigns outcomes recognizable to human experts (e.g., *unsafe*, *safe stop*), thereby complementing continuous metrics with discrete ground truth categories. Fig. 4 presents the label distribution of AgentDrive-Sim across both episode and segment levels. The results show that at the segment level, safe-goal instances dominate with more than five million samples, followed by unsafe and inefficient outcomes, whereas the episode-level distribution remains more balanced across all categories. This clear disparity highlights the higher granularity and diversity of behaviors captured during segment-level analysis compared to aggregated episode-level evaluations.

G. Dataset Construction

After N iterations, we obtain the benchmark dataset:

$$\mathcal{D} = \{(s_i, \hat{j}_i, \tau_i, \ell_i)\}_{i=1}^N, \quad (15)$$

where each entry contains: the sampled scenario axes s_i , the generated JSON \hat{j}_i , the resulting rollout τ_i , and the assigned label ℓ_i . To ensure coverage, we maximize entropy across scenario axes:

$$\max \sum_{\text{axis} \in \{t, b, e, r, o, d, q\}} \mathcal{H}(\{\text{axis}_i\}), \quad (16)$$

subject to schema compliance $\hat{j}_i \in \mathcal{C}$ and valid rollouts τ_i .

This entropy maximization prevents the dataset from collapsing into trivial or repetitive cases, ensuring diversity across scenario types, behaviors, environments, and road layouts. The resulting dataset thus captures both routine and rare conditions, providing a rigorous benchmark for training and evaluating autonomous driving models under varied and safety-critical contexts.

Algorithm 1: AgentDrive Scenario Generation and Labeling Pipeline

- 1: Sample scenario tuple $s \sim \mathcal{S}$
 - 2: Construct enriched prompt $P = P(s, H(d))$
 - 3: Generate JSON $\hat{j} = \mathcal{G}_\phi(P)$
 - 4: Validate \hat{j} against schema \mathcal{C} ; repair via Π up to R retries
 - 5: Simulate rollout $\tau = \mathcal{M}(\hat{j})$
 - 6: Compute surrogate safety metrics (e.g., TTC_{\min})
 - 7: Detect events and assign categorical label ℓ via Eq. 14
 - 8: Store (s, \hat{j}, τ, ℓ) into dataset \mathcal{D}
-

H. End-to-End Scenario Generation Pipeline

The overall pipeline is summarized in Algorithm 1. Each stage progressively refines the representation of a scenario, moving from abstract symbolic specifications to concrete simulation trajectories with interpretable safety labels. This layered transformation ensures that high-level abstractions such as “*aggressive merging during rain*” can be systematically translated into structured inputs, simulated behaviors, and quantitative safety outcomes.

The process begins by sampling scenario tuples from the multidimensional space \mathcal{S} (Eq. 2), guaranteeing balanced coverage across scenario types, driver behaviors, environmental conditions, and traffic densities. This systematic sampling provides exposure to both common conditions and rare, safety-critical cases. Each tuple is then encoded into a prompt enriched with difficulty-specific hints $H(d)$ (Eq. 6), which conditions the large language model (LLM) to generate structured specifications that remain semantically meaningful while adhering to physics and traffic safety constraints.

The LLM produces a candidate JSON specification (Eq. 7), which is validated against the schema \mathcal{C} . Invalid generations are corrected by the post-processor Π , with retries up to R attempts. This stage acts as a safeguard, ensuring that only semantically correct and physically plausible specifications propagate downstream.

Once a valid JSON is obtained, the simulator \mathcal{M} generates a rollout trajectory τ (Eq. 9) that captures the temporal evolution of the ego vehicle and surrounding agents under the defined scenario. This simulation grounds abstract descriptions in dynamic, reproducible trajectories. From τ , surrogate safety metrics

TABLE IV: Notation used throughout the AgentDrive-MCQ dataset generation methodology.

Symbol	Block	Definition
\mathcal{S}	Scenario Input	Validated scenario JSON containing structured fields
d	Description	Natural language narrative derived from \mathcal{S}
m	Model	Large language model used for description/MCQ generation
R	Control	Retry budget for error-guided synthesis attempts
Q	MCQ Synthesis	Question text derived from description d
\mathcal{O}	MCQ Synthesis	Set of four candidate answer options
i^*	MCQ Synthesis	Index of the correct choice ($i^* \in \{A, B, C, D\}$)
r	MCQ Synthesis	Concise textual rationale supporting the correct answer
s	Style Control	Reasoning style $\in \{\text{physics}, \text{policy}, \text{hybrid}, \text{scenario}, \text{comparative}\}$
δ	Validation	Difficulty level $\in \{\text{Easy}, \text{Medium}, \text{Hard}\}$
M_s	Assembly	Final MCQ object for style s linked to scenario \mathcal{S}
h	Assembly	SHA-256 hash for persistent identifier of M_s
v	Physics	Ego vehicle speed (m/s or km/h)
t_r	Physics/Policy	Driver or agent reaction time (s)
a_{ego}	Physics	Ego vehicle deceleration capability (m/s^2)
a_{lead}	Physics	Lead/cut-in vehicle deceleration (m/s^2)
g	Physics/Hybrid	Computed minimum safe headway distance (m)
τ	Policy	Time-gap rule (s), e.g., 2–4 seconds depending on conditions
g_{hyb}	Hybrid	Headway including both physics-based minimum and policy margin (m)
A	Comparative	Candidate driving maneuver (e.g., brake, accelerate, change lane, maintain speed)

such as TTC_{\min} (Eq. 12) are computed, and discrete rule-based event detection is performed. The outcome is assigned a categorical label ℓ (Eq. 14)—safe_goal, safe_stop, inefficient, or unsafe—providing interpretable targets for benchmarking autonomous driving models.

Finally, the complete record (s, j, τ, ℓ) is stored in the dataset \mathcal{D} (Eq. 15). Repeating this process across N iterations yields a diverse, large-scale benchmark that balances coverage and realism, forming the foundation of AgentDrive.

IV. AGENTDRIVE-MCQ BENCHMARK

A. Motivation

Traditional benchmarks for autonomous driving agents have primarily emphasized perception and low-level control, including object detection, trajectory prediction, and actuation accuracy. While these tasks are essential, real-world driving introduces *context-sensitive decision reasoning*, where multiple heterogeneous factors such as weather conditions, traffic signals, surrounding vehicle behavior, and road geometry must be simultaneously integrated. This dimension of reasoning is often underexplored in existing datasets, which limits their ability to evaluate agents in high-stakes, safety-critical scenarios.

To address this gap, we introduce AgentDrive-MCQ, a benchmark designed to probe the reasoning and decision-making capabilities of large language models (LLMs) when deployed as agentic controllers in autonomous driving. The central innovation lies in transforming structured driving scenarios into natural language descriptions and subsequently into multiple-choice questions (MCQs) that demand hard reasoning. Unlike simple factual Q&A, AgentDrive-MCQ

Algorithm 2: AgentDrive-MCQ Generation

```

Input: Validated scenario  $\mathcal{S}$  (JSON-conformant), model  $m$ ,  

        retry budget  $R \geq 1$ 
Output: Set of MCQ JSON objects  $\{M_{\text{style}}\}$  linked to  $\mathcal{S}$ 
// Stage 1: Scenario → Description
 $d \leftarrow \text{GENERATEDESCRIPTION}(\mathcal{S}, m);$ 
    Call LLM with constrained system prompt (10–12  

        sentences, plain text only; include units; no lists/code);
    Normalize whitespace; truncate to a maximum  

        token/word budget; attach as
     $\mathcal{S}[\text{description}] \leftarrow \{\text{model} : m, \text{text} : d\};$ 
    Persist augmented copy as
    <scenario_name>_withdesc.json;
// Stage 2: Description → MCQ Synthesis
Across Five Styles
foreach  $s \in \{\text{physics}, \text{policy}, \text{hybrid}, \text{scenario}, \text{comparative}\}$ 
do
     $(Q, \mathcal{O}, i^*, r) \leftarrow \text{MAKEMCQ}(d, m, s)$  with
        response_format=json_object;
    Validate style-specific constraints:
        physics/hybrid ⇒ numeric tokens in  $Q$  and  $\mathcal{O}$ ;
        scenario ⇒ qualitative, not all-numeric;
        comparative ⇒ ≥ 2 action-oriented options;
    Enforce general checks:  $|\mathcal{O}| = 4$ ; labels
         $\{A, B, C, D\}$ ; options distinct;  $|Q| \leq 25$  words;
         $r \neq \emptyset$ ;
    If validation fails, retry with error-guided hint up to
         $R$  times;
// Stage 3: Assembly & Persistence
foreach validated style  $s$  do
     $M_s \leftarrow \langle \mathcal{S}[\text{name}], d, Q, \mathcal{O}, i^*, r, s \rangle;$ 
     $h \leftarrow \text{SHA256}(d \parallel Q \parallel \mathcal{O} \parallel i^* \parallel s);$ 
    Save  $M_s$  as
    <scenario_name>_{h[0:8]}_mcq_{s}.json;
return  $\{M_s\}$ ;

```

requires agents to synthesize information across multiple scenario layers and to select strategies aligned with safety-critical objectives.

To capture the full spectrum of driving reasoning, AgentDrive-MCQ spans five complementary MCQ styles:

- **Physics-based:** numerically grounded questions requiring kinematic calculations such as time-to-collision or stopping distance.
- **Policy-based:** questions emphasizing traffic rules, defensive driving guidelines, and safety heuristics.
- **Hybrid:** items combining physics-derived calculations with additional policy or margin-based reasoning.
- **Scenario-interpreting:** qualitative reasoning about primary risks, priorities, and contextual constraints.
- **Comparative/optimization:** action-selection questions evaluating the relative safety of candidate maneuvers.

This taxonomy ensures that AgentDrive-MCQ not only evaluates mathematical reasoning, but also policy compliance, risk interpretation, and maneuver optimization—core facets of real-world driving intelligence. Correct answers are consistently aligned with safe driving strategies, anchoring the benchmark in risk-aware reasoning.

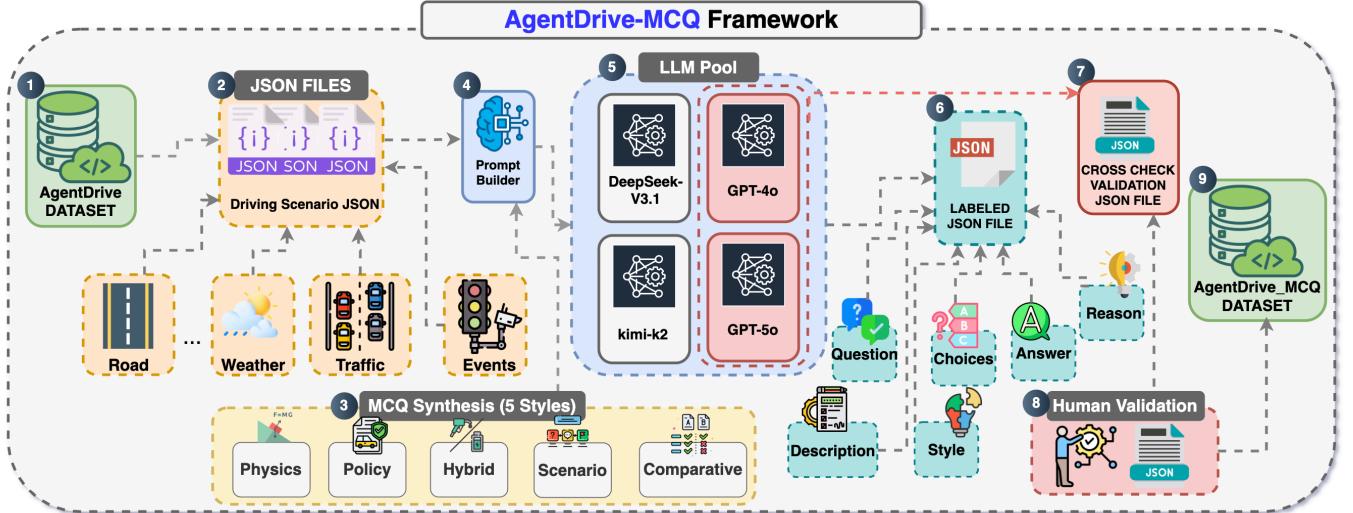


Fig. 5: The AgentDrive-MCQ framework.

B. AgentDrive-MCQ: Dataset Generation Methodology

We design a unified pipeline that consistently transforms structured scenarios into benchmark-ready MCQs across all five reasoning styles. The pipeline operates in three stages: (i) generating natural language descriptions, (ii) synthesizing reasoning-intensive MCQs in each style, and (iii) assembling validated MCQ objects with traceable identifiers. This staged design allows for scalable dataset expansion and systematic evaluation across diverse driving environments.

The entire workflow is formalized in Algorithm 2. Starting from a validated scenario JSON, the first stage generates a natural language description, constrained to 10–12 sentences, and enriched with contextual details, including road layout, environmental conditions, traffic controls, and dynamic vehicle maneuvers. The second stage applies an LLM-based synthesis process to convert the description into five structured MCQ JSONs, one per style. Each MCQ contains a concise question, four candidate answers, a single correct choice, and a textual rationale. Multiple validation checks (e.g., word length, option distinctness, label format, numeric presence when required, and action-orientation for comparative items) ensure structural consistency, with automated retries in place when failures occur. The final stage assembles the outputs into persistent MCQ objects, uniquely identified by content hashes, and stores them in the benchmark repository. This end-to-end design ensures traceability, reproducibility, and robustness of the generated dataset.

C. Framework Illustration

The overall architecture of the pipeline is depicted in Figure 5. The process begins with the AgentDrive dataset, which encodes scenario attributes such as road layout, weather conditions, traffic density, and event triggers. These features are formalized into JSON files that serve as the structured representation of driving scenarios. A dedicated prompt builder then reformulates the structured data into prompts for a pool of

LLMs, including DeepSeek-V3.1, GPT-4o, kimi-k2, and GPT-5o. The LLM pool generates narrative scenario descriptions and subsequently produces five reasoning-intensive MCQs per scenario, corresponding to the physics, policy, hybrid, scenario, and comparative styles. The intermediate outputs are stored as labeled JSON files and passed through a cross-check validation stage to ensure consistency and correctness. Finally, the validated items are aggregated into the AgentDrive-MCQ dataset, providing a robust benchmark for evaluating context-sensitive reasoning under multi-factor constraints.

D. Prompt Design and Control

Two carefully engineered prompts underpin the pipeline. The first, the *Description Prompt*, instructs the model to summarize the scenario in 10–12 sentences, covering road geometry, environmental conditions, traffic controls, and dynamic agent behaviors. The result is a compact, self-contained narrative that encodes all relevant constraints. The second, the *MCQ Prompt*, requires the model to generate a reasoning question of high difficulty, with exactly four candidate answers and one correct choice. Depending on the style, additional constraints enforce numeric presence (physics/hybrid), policy interpretation (policy), qualitative reasoning (scenario), or action comparison (comparative). Strict JSON formatting is imposed to enable automatic parsing and reproducibility. This design ensures that question difficulty emerges organically from scenario complexity and style-specific constraints, rather than from arbitrary distractor construction.

The pipeline yields two structured outputs. The first is a description-augmented scenario JSON that retains the original structured fields while enriching them with natural-language descriptions. The second is a set of AgentDrive-MCQ JSONs, each corresponding to one of the five reasoning styles. Each JSON includes a description, a reasoning-intensive question, four candidate answers, the correct answer, and a supporting rationale. This dual representation enables joint use in simulation-based evaluation and reasoning-based benchmarking.

TABLE V: Accuracy (%) results of 50 examined LLM reasoning models evaluated across multiple reasoning styles using 2k samples from AgentDrive-MCQ.

Model	Company	Size	License	Comparative	Hybrid	Physics	Policy	Scenario	Overall
ChatGPT 4o	OpenAI	N/A	Proprietary	90.0	72.5	55.0	100	95.0	82.5
GPT-5 Chat	OpenAI	N/A	Proprietary	92.5	70.0	50.0	100	92.5	81.0
Qwen3 235B A22B (2507)	Alibaba	235B	Open	92.5	60.0	67.5	87.5	97.5	81.0
Mistral Medium 3.1	Mistral AI	N/A	Proprietary	95.0	60.0	52.5	97.5	95.0	80.0
GPT-4.1 Mini	OpenAI	N/A	Proprietary	90.0	55.0	50.0	95.0	97.5	77.5
GPT-4.1	OpenAI	N/A	Proprietary	90.0	50.0	52.5	90.0	95.0	75.5
Phi 4 Reasoning Plus	Microsoft	14B	Open	95.0	45.0	47.5	92.5	97.5	75.5
Gemini 2.5 Flash	Google	391B	Proprietary	92.5	65.0	32.5	92.5	95.0	75.5
Qwen3 Max	Alibaba	N/A	Open	95.0	52.5	47.5	95.0	95.0	77.0
ERNIE 4.5 300B A47B	Baidu	300B	Open	85.0	45.0	52.5	95.0	97.5	75.0
DeepSeek Chat V3 (0324)	DeepSeek	685B	Open	77.5	55.0	45.0	95.0	95.0	73.5
GPT-4o (2024-11-20)	OpenAI	N/A	Proprietary	95.0	55.0	45.0	85.0	95.0	75.0
Qwen3 Next 80B A3B Ins	Alibaba	80B	Open	82.5	52.5	40.0	95.0	95.0	73.0
Kimi Dev 72B	Moonshot AI	72B	Open	80.0	57.5	42.5	87.5	92.5	72.0
Hermes 3 (Llama 3.1 405B)	Nous Research	405B	Open	82.5	40.0	57.5	87.5	90.0	71.5
Llama 3.1 70B Instruct	Meta	70B	Open	92.5	52.5	57.5	52.5	97.5	70.5
Llama-109B-MoE	DeepCogito	109B	Open	87.5	52.5	45.0	67.5	97.5	70.0
Kimi K2 (0905)	Moonshot AI	1T	Open	85.0	50.0	37.5	80.0	95.0	69.5
Grok 3 Mini	xAI	N/A	Proprietary	82.5	42.5	37.5	87.5	97.5	69.5
DeepSeek V3.1 Terminus	DeepSeek	N/A	Open	85.0	32.5	47.5	85.0	92.5	68.5
Grok 4 Fast	xAI	N/A	Proprietary	87.5	30.0	42.5	87.5	90.0	67.5
Kimi K2	Moonshot AI	1T	Open	85.0	32.5	37.5	90.0	92.5	67.5
DeepSeek V3.2 Exp	DeepSeek	N/A	Open	85.0	32.5	55.0	75.0	90.0	67.5
Claude 3.7 Sonnet (Thinking)	Anthropic	N/A	Proprietary	92.5	40.0	65.0	42.5	95.0	67.0
Claude Opus 4.1	Anthropic	N/A	Proprietary	92.5	42.5	10.0	95.0	92.5	66.5
GPT-4.1 Nano	OpenAI	N/A	Proprietary	87.5	17.5	35.0	97.5	90.0	65.5
Gemma 3 12B IT	Google	12B	Open	57.5	50.0	50.0	70.0	85.0	62.5
Llama 3.3 70B Instruct	Meta	70B	Open	90.0	47.5	55.0	27.5	95.0	63.0
Qwen Turbo	Alibaba	N/A	Open	87.5	22.5	30.0	75.0	90.0	61.0
Nova Micro V1	Amazon	N/A	Proprietary	75.0	15.0	35.0	90.0	87.5	60.5
Phi 3 Medium 128K Instruct	Microsoft	14B	Open	82.5	20.0	30.0	75.0	90.0	59.5
Gemma 3 27B IT	Google	27B	Open	92.5	27.5	35.0	35.0	97.5	57.5
OLMo 2 32B Instruct	AllenAI	32B	Open	57.5	25.0	45.0	62.5	90.0	56.0
GPT-4o Mini	OpenAI	N/A	Proprietary	85.0	20.0	52.5	30.0	90.0	55.5
GPT-4o Mini (2024-07-18)	OpenAI	N/A	Proprietary	85.0	20.0	52.5	25.0	90.0	54.5
WizardLM 2 8x22B	Microsoft	176B	Open	60.0	45.0	45.0	25.0	90.0	53.0
Claude Sonnet 4.5	Anthropic	468B	Proprietary	75.0	0.0	0.0	85.0	95.0	51.0
Llama 4 Scout	Meta	17B	Open	85.0	22.5	20.0	25.0	92.5	49.0
LFM 3B	Liquid AI	3B	Open	67.5	25.0	40.0	12.5	92.5	47.5
AFM 4.5B	Arcee AI	4.5B	Open	62.5	27.5	25.0	32.5	90.0	47.5
Llama 4 Maverick	Meta	17B	Open	77.5	2.5	5.0	55.0	95.0	47.0
Command R7B (12-2024)	Cohere	7B	Open	80.0	7.5	27.5	20.0	95.0	46.0
Llama 3.1 8B Instruct	Meta	8B	Open	62.5	20.0	27.5	10.0	90.0	42.0
Qwen 2.5 7B Instruct	Alibaba	7B	Open	47.5	12.5	20.0	50.0	57.5	37.5
Llama 3.2 3B Instruct	Meta	3B	Open	52.5	7.5	35.0	7.5	80.0	36.5
Llama 3.2 1B Instruct	Meta	1B	Open	62.5	10.0	30.0	7.5	72.5	36.5
Mistral Nemo	Mistral AI	12B	Open	65.0	5.0	12.5	10.0	90.0	36.5
GPT-3.5 Turbo Instruct	OpenAI	N/A	Proprietary	52.5	5.0	27.5	5.0	82.5	34.5
GLM 4.6	Z-AI	N/A	Open	60.0	5.0	2.5	45.0	37.5	30.0
GPT-OSS-120B	OpenAI	120B	Open	22.5	5.0	10.0	10.0	10.0	11.5
GPT-OSS-20B	OpenAI	20B	Open	17.5	7.5	0.0	15.0	7.5	9.5

LLM parameters: temperature = 0.0, which controls randomness (0 = deterministic output); max_tokens = 16, which defines the maximum number of tokens generated; top_p = 1.0, which is the nucleus sampling parameter (1.0 = all tokens considered); and max_retries = 5, which specifies the maximum number of retry attempts in case of LLM failure.

V. LLM REASONING PERFORMANCE ON AGENTDRIVE-MCQ

The AgentDrive-MCQ benchmark provides a rigorous evaluation framework for assessing the reasoning capabilities of large language models across diverse driving-related scenarios. This section presents a comprehensive analysis of 50 leading LLMs, encompassing both proprietary and open-source architectures, and evaluates them across five reasoning dimensions: comparative, hybrid, physics, policy, and scenario. The goal is to identify strengths and weaknesses in model reasoning under varied environmental, physical, and ethical constraints. The analysis highlights how model scale, architecture, and

training strategy influence performance, revealing that while frontier proprietary systems such as ChatGPT 4o and GPT-5 Chat dominate contextual and policy-based reasoning, advanced open models like Qwen3 235B A22B and ERNIE 4.5 300B A47B are rapidly closing the performance gap in structured, physics-grounded, and hybrid reasoning tasks.

A. Performance Metrics Definition

The reasoning performance of each large language model (LLM) in AgentDrive-MCQ is evaluated across five distinct reasoning dimensions: *Comparative*, *Hybrid*, *Physics*, *Policy*, and *Scenario*. Each dimension represents a specific cogni-

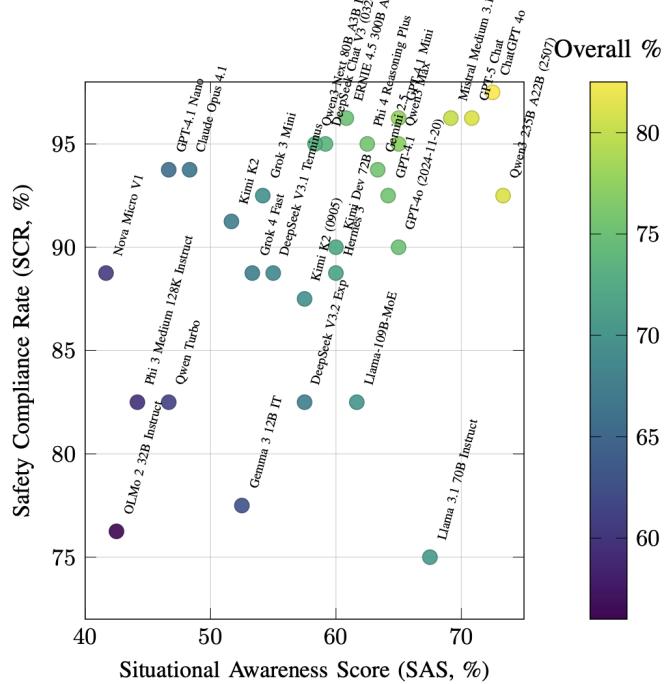


Fig. 6: Top models by highest Overall: SAS vs. SCR.

tive ability required for robust, safety-aligned reasoning in autonomous driving contexts.

a) *Overall Accuracy*: The **accuracy for each reasoning style** measures the proportion of correctly answered questions in that specific reasoning category:

$$\text{Accuracy}_j = \frac{\text{Correct Responses}_j}{\text{Total Questions}_j} \times 100, \quad (17)$$

where $j \in \{\text{Comparative, Hybrid, Physics, Policy, Scenario}\}$

The **Overall Accuracy** serves as a unified indicator of model performance across all reasoning styles and is computed as:

$$\text{Overall Accuracy} = \frac{1}{5} \sum_{j=1}^5 \text{Accuracy}_j \quad (18)$$

A higher overall accuracy reflects stronger reasoning consistency and generalization across heterogeneous driving scenarios.

Beyond these primary accuracies, two derived reliability metrics provide deeper insight into model safety and awareness.

b) *Safety Compliance Rate (SCR)*: The **Safety Compliance Rate (SCR)** assesses the model's adherence to normative driving policies and ethical safety principles embedded in regulatory or operational constraints. It integrates both procedural correctness (*Policy*) and behavioral consistency under realistic traffic situations (*Scenario*):

$$\text{SCR} = \frac{\omega_p \cdot \text{Policy} + \omega_s \cdot \text{Scenario}}{\omega_p + \omega_s}, \quad (19)$$

where ω_p and ω_s denote the relative importance weights of the *Policy* and *Scenario* dimensions (by default, $\omega_p = \omega_s = 1$).

A higher SCR reflects a model's capability to comply with explicit safety regulations (e.g., traffic rules, right-of-way laws) and implicit ethical norms (e.g., harm minimization, risk-aware behavior) under uncertain or dynamic conditions. This metric thus serves as an indicator of *safety alignment* and *regulatory robustness* in autonomous decision-making.

c) *Situational Awareness Score (SAS)*: The **Situational Awareness Score (SAS)** quantifies a model's capacity to perceive, interpret, and predict environmental dynamics—a core cognitive competence in safety-critical autonomous reasoning. It aggregates perception–reasoning alignment across three complementary reasoning styles:

$$\text{SAS} = \frac{\omega_c \cdot \text{Comparative} + \omega_h \cdot \text{Hybrid} + \omega_f \cdot \text{Physics}}{\omega_c + \omega_h + \omega_f}, \quad (20)$$

where ω_c , ω_h , and ω_f correspond to the relative weights assigned to *Comparative*, *Hybrid*, and *Physics* reasoning dimensions, respectively (default: $\omega_c = \omega_h = \omega_f = 1$). A higher SAS denotes enhanced situational comprehension—capturing spatial awareness, causal inference, and physical feasibility reasoning. Models achieving strong SAS values exhibit robust contextual understanding and predictive foresight, especially in edge-case or high-risk driving scenarios.

B. Overview of Model Performance

The comparative and categorical analysis of 50 evaluated LLMs on the AgentDrive–MCQ benchmark reveals significant diversity in reasoning performance across five distinct dimensions: comparative, hybrid, physics, policy, and scenario. As summarized in Table V, proprietary frontier models such as ChatGPT 4o (82.5%) and GPT-5 Chat (81.0%) from OpenAI dominated the benchmark, achieving perfect or near-perfect accuracy in policy (100%) and scenario (97.5%) reasoning tasks. These results underscore their superior contextual reasoning, ethical prioritization, and adaptability to complex decision-making scenarios. Among open-source systems, Qwen3 235B A22B reached a competitive 81.0% overall accuracy, leading in physics-driven reasoning (67.5%), while ERNIE 4.5 300B A47B achieved 75.0%, reflecting the growing maturity of Chinese foundation models. Other strong performers, including Mistral Medium 3.1 (80.0%) and GPT-4.1 Mini (77.5%), demonstrated balanced reasoning proficiency across multiple domains, highlighting the value of large-scale fine-tuning and domain adaptation.

In contrast, smaller, earlier-generation models exhibited significant performance degradation. Models such as Llama 3.1 8B Instruct (42.0%) and Qwen 2.5 7B Instruct (37.5%) underperformed in multi-modal reasoning tasks, particularly when physical or contextual inference was required. The hybrid reasoning category emerged as the most challenging, with even top-tier models—ChatGPT 4o (72.5%), GPT-5 Chat (70.0%), and Qwen3 235B A22B (60.0%)—struggling to integrate symbolic, numerical, and contextual information effectively. Conversely, mid-scale open models such as Kimj Dev 72B (72.0%) and DeepSeek V3.1 Terminus (68.5%) displayed encouraging stability, demonstrating that efficient architectures can approach the performance of frontier proprietary models.

when supported by focused fine-tuning. Overall, the results confirm that model scale, training strategy, and architecture design are pivotal for achieving robust reasoning, with open ecosystems rapidly closing the gap in structured and physics-aware reasoning.

C. SAS–SCR Correlation

Fig. 6 visualizes the relationship between the Situational Awareness Score (SAS) and the Safety Compliance Rate (SCR) for the top models ranked by Overall Accuracy from Table V. The upper-right region of the scatter plot denotes the ideal operational zone, where models achieve both high situational awareness and strong safety compliance.

The results clearly indicate that frontier proprietary models such as ChatGPT 4o, GPT-5 Chat, and Qwen3 235B A22B dominate overall performance, reaching accuracies above 80 %. Their SCR values exceed 95 %, and their SAS values remain in the 65–75 % range, demonstrating balanced reasoning between environmental understanding and safe decision-making. These findings highlight the maturity of large-scale, instruction-tuned architectures that integrate reasoning and physics-based inference effectively.

Mid-tier open-source models, including Mistral Medium 3.1 and ERNIE 4.5 300B A47B, exhibit competitive safety compliance but somewhat lower SAS, suggesting stronger rule adherence but limited depth in contextual or physical reasoning. Such patterns emphasize that mastering safety alignment does not automatically yield situational adaptability. Overall, a positive correlation emerges between SCR and SAS—models that understand complex physical environments tend also to make safer decisions. The clustering of high-performing models in the upper-right quadrant of Fig. 6 validates that reasoning precision and safety compliance are complementary competencies.

D. Physics-Style Challenges

The most demanding physics-style questions tested the quantitative reasoning depth of the evaluated models, involving complex physical dependencies such as traction, slope gradients, and variable deceleration rates. Even the top-performing models—GPT-5 Chat (50.0%), ChatGPT 4o (55.0%), and Qwen3 235B A22B (67.5%)—showed reduced accuracy, reflecting the inherent difficulty of reasoning about motion dynamics and braking trajectories under uncertainty. Despite their contextual fluency, these models often underestimated or overestimated stopping distances, revealing weaknesses in applying physical formulas to real-world conditions. The findings emphasize that LLMs, while linguistically proficient, still lack numerical grounding and consistency when dealing with multivariable safety equations critical to autonomous driving.

E. Policy-Style Challenges

The policy-style reasoning category primarily assessed the models’ ability to interpret traffic rules, safety margins, and prioritization policies under constrained visibility or dynamic

hazards. High-performing systems such as ChatGPT 4o and GPT-5 Chat achieved 100% accuracy, demonstrating mastery in policy alignment and context-sensitive judgment. However, smaller open models like Llama 3.1 8B Instruct (10.0%) and AFM 4.5B (32.5%) frequently failed to generalize safety rules, often choosing aggressive or unsafe responses. These results underscore the persistent gap between linguistic pattern recognition and actionable policy reasoning, suggesting that lightweight models require targeted, safety-aware fine-tuning to achieve reliable decision-making performance in regulatory and ethical reasoning domains.

F. Hybrid-Style Challenges

The hybrid-style reasoning dimension emerged as the most universally difficult in all model families—proprietary, open, and mid-scale. It required the simultaneous processing of symbolic policy cues and quantitative calculations, especially in multi-constraint decision-making scenarios (e.g., braking on gravel, handling downhill deceleration, or reacting under adverse weather). Even top-tier models such as ChatGPT 4o (72.5%) and GPT-5 Chat (70.0%) exhibited variability, while open counterparts such as Qwen3 235B A22B (60.0%), Kimi Dev 72B (57.5%), and DeepSeek Chat V3 (55.0%) struggled with precise multi-factor estimation. The observed pattern confirms that hybrid reasoning—requiring the fusion of conceptual policy understanding and numerical grounding—remains an unresolved challenge in current LLM architectures. Future model advancements must focus on improving cognitive compositionality and structured reasoning under uncertainty to achieve human-level reliability in hybrid decision environments.

G. Comparative-Style Challenges

The comparative-style challenges assessed each model’s capacity to contrast multiple decision alternatives and identify the safest course of action under uncertain conditions. This dimension yielded generally strong results, with several models surpassing 90% accuracy. Mistral Medium 3.1 and Phi 4 Reasoning Plus achieved the highest comparative reasoning scores (95.0%), followed closely by Qwen3 Max and GPT-5 Chat (92.5%). The dominance of large proprietary systems such as ChatGPT 4o (90.0%) and Gemini 2.5 Flash (92.5%) underscores the importance of fine-tuned instruction alignment for maintaining decision consistency. In contrast, small and mid-scale open models—including Qwen 2.5 7B Instruct (47.5%) and Llama 3.1 8B Instruct (62.5%)—struggled with comparative reasoning, often failing to weigh competing outcomes accurately. These results suggest that comparative reasoning, while linguistically less demanding than numerical estimation, still benefits from high-capacity contextual modeling and refined safety-aware alignment. Overall, this dimension highlights the maturity of top-tier LLMs in structured choice-making but also reveals that smaller open models lack the inferential depth required to handle multi-option trade-offs reliably.

H. Scenario-Style Challenges

Scenario-style challenges required a holistic understanding of complex, dynamic environments with multiple interacting

factors such as weather, visibility, and traffic density. This category produced some of the highest individual accuracies across the benchmark. Models like ChatGPT 4o, GPT-5 Chat, Phi 4 Reasoning Plus, Qwen3 Max, and ERNIE 4.5 300B A47B each achieved near-perfect scenario reasoning scores (97.5%), demonstrating exceptional situational awareness and context synthesis. These systems consistently identified safe maneuvers by integrating spatial, temporal, and regulatory cues. In contrast, mid-tier models such as DeepSeek Chat V3 (95.0%) and Kimi Dev 72B (92.5%) remained strong but slightly less consistent in handling unexpected hazards or environmental uncertainty. Smaller models—such as Llama 3.2 3B Instruct (80.0%) or Gemma 3 12B IT (85.0%)—showed substantial performance degradation, often missing implicit contextual dependencies. The overall trend indicates that scenario-based reasoning is the most stable and generalizable dimension for frontier models, as it combines perception, rule-following, and reasoning depth. However, achieving consistent real-world interpretability under diverse environmental conditions remains an open research challenge for the broader LLM ecosystem.

VI. CONCLUSION

In this paper, we presented AgentDrive, a comprehensive open benchmark dataset designed to advance the training, fine-tuning, and evaluation of agentic AI systems in autonomous driving. Built upon a large-scale corpus of 300,000 LLM-generated driving scenarios, AgentDrive introduced a factorized scenario space that captured the complexity and diversity of real-world environments through seven orthogonal dimensions: scenario type, driver behavior, environment, road layout, objective, difficulty, and traffic density. An LLM-driven prompt-to-JSON specification pipeline ensured semantic richness, physical validity, and simulation readiness, while surrogate safety metrics and rule-based labeling provided interpretable ground truth for supervised learning and reasoning analysis.

To complement simulation-based evaluation, we proposed AgentDrive-MCQ, a reasoning-oriented benchmark containing 100,000 multiple-choice questions designed to assess the cognitive, ethical, and physics-grounded reasoning capabilities of LLM-based agents. We conducted a large-scale evaluation of fifty state-of-the-art LLMs—including GPT-5, ChatGPT 4o, Gemini 2.5 Flash, DeepSeek V3, Qwen3 235B, ERNIE 4.5 300B, Grok 4, and Mistral Medium 3.1—which demonstrated that while proprietary models currently dominate in contextual and policy reasoning, advanced open models rapidly closed the gap in structured and physics-based reasoning.

AgentDrive and AgentDrive-MCQ together provided a framework for benchmarking and improving agentic AI reasoning in safety-critical autonomous systems. Beyond evaluation, these resources enabled large-scale model training, fine-tuning, and transfer learning across simulation and reasoning domains. In future work, we plan to extend AgentDrive to multi-agent and multimodal environments, integrate real-world sensor data, and explore alignment strategies to further enhance the reliability and interpretability of LLM-driven autonomous agents.

To support open science and reproducibility, we released the complete AgentDrive dataset, labeled scenarios, AgentDrive-MCQ benchmark, Google Colab evaluation scripts, and all related materials on GitHub.

REFERENCES

- [1] Z. Yang, X. Jia, H. Li, and J. Yan, “Llm4drive: A survey of large language models for autonomous driving,” *arXiv preprint arXiv:2311.01043*, 2023.
- [2] C. Cui, Y. Ma, Z. Yang, Y. Zhou, P. Liu, J. Lu, L. Li, Y. Chen, J. H. Panchal, A. Abdelraouf *et al.*, “Large language models for autonomous driving (llm4ad): Concept, benchmark, experiments, and challenges,” *arXiv preprint arXiv:2410.15281*, 2024.
- [3] C. Cui, Y. Ma, X. Cao, W. Ye, Y. Zhou, K. Liang, J. Chen, J. Lu, Z. Yang, K.-D. Liao *et al.*, “A survey on multimodal large language models for autonomous driving,” in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2024, pp. 958–979.
- [4] Y. Gao, M. Piccinini, Y. Zhang, D. Wang, K. Moller, R. Brusnicki, B. Zarrouki, A. Gambi, J. F. Totz, K. Storms *et al.*, “Foundation models in autonomous driving: A survey on scenario generation and scenario analysis,” *arXiv preprint arXiv:2506.11526*, 2025.
- [5] Y. Huang, J. Sansom, Z. Ma, F. Gervits, and J. Chai, “Drivlme: Enhancing llm-based autonomous driving agents with embodied and social experiences,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 3153–3160.
- [6] T. Cai, Y. Liu, Z. Zhou, H. Ma, S. Z. Zhao, Z. Wu, and J. Ma, “Driving with regulation: Interpretable decision-making for autonomous vehicles with retrieval-augmented reasoning via llm,” *arXiv preprint arXiv:2410.04759*, 2024.
- [7] X. Kong, T. Braunl, M. Fahmi, and Y. Wang, “A superalignment framework in autonomous driving with large language models,” in *2024 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2024, pp. 1715–1720.
- [8] Y. Bai, D. Wu, Y. Liu, F. Jia, W. Mao, Z. Zhang, Y. Zhao, J. Shen, X. Wei, T. Wang *et al.*, “Is a 3d-tokenized llm the key to reliable autonomous driving?” *arXiv preprint arXiv:2405.18361*, 2024.
- [9] C. K. Sah, A. K. Shaw, X. Lian, A. S. Baig, T. Wen, K. Jiang, M. Yang, and D. Yang, “Advancing autonomous vehicle intelligence: Deep learning and multimodal llm for traffic sign recognition and robust lane detection,” *arXiv preprint arXiv:2503.06313*, 2025.
- [10] Y. Jin and J. Ma, “Large language model as parking planning agent in the context of mixed period of autonomous vehicles and human-driven vehicles,” *Sustainable Cities and Society*, vol. 117, p. 105940, 2024.
- [11] L. Chen, O. Sinavski, J. Hünermann, A. Karnsund, A. J. Willmott, D. Birch, D. Maund, and J. Shotton, “Driving with llms: Fusing object-level vector modality for explainable autonomous driving,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 14 093–14 100.
- [12] Z. Xu, Y. Zhang, E. Xie, Z. Zhao, Y. Guo, K.-Y. K. Wong, Z. Li, and H. Zhao, “Drivegpt4: Interpretable end-to-end autonomous driving via large language model,” *IEEE Robotics and Automation Letters*, 2024.
- [13] T. Zhang, T. Jin, L. Wang, J. Liu, S. Liang, M. Zhang, A. Liu, and X. Liu, “Bench2adlvm: A closed-loop benchmark for vision-language models in autonomous driving,” *arXiv preprint arXiv:2508.02028*, 2025.
- [14] X. Cao, T. Zhou, Y. Ma, W. Ye, C. Cui, K. Tang, Z. Cao, K. Liang, Z. Wang, J. M. Rehg *et al.*, “Maplm: A real-world large-scale vision-language benchmark for map and traffic scene understanding,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 21 819–21 830.
- [15] Y. Ma, C. Cui, X. Cao, W. Ye, P. Liu, J. Lu, A. Abdelraouf, R. Gupta, K. Han, A. Bera *et al.*, “Lampilot: An open benchmark dataset for autonomous driving with language model programs,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 15 141–15 151.
- [16] Z. Tang, J. He, D. Pe, K. Liu, T. Gao, and J. Zheng, “Test large language models on driving theory knowledge and skills for connected autonomous vehicles,” in *Proceedings of the 19th Workshop on Mobility in the Evolving Internet Architecture*, 2024, pp. 1–6.
- [17] H.-k. Chiu, R. Hachiuma, C.-Y. Wang, S. F. Smith, Y.-C. F. Wang, and M.-H. Chen, “V2v-llm: Vehicle-to-vehicle cooperative autonomous driving with multi-modal large language models,” *arXiv preprint arXiv:2502.09980*, 2025.
- [18] K. Lebioda, N. Petrovic, F. Pan, V. Zolfaghari, A. Schamschurko, and A. Knoll, “Are requirements really all you need? using llms to generate configuration code: A case study in automotive simulations,” *IEEE Access*, 2025.

- [19] Y. Yao, S. Bhatnagar, M. Mazzola, V. Belagiannis, I. Gilitschenski, L. Palmieri, S. Razniewski, and M. Hallgarten, “Agents-ilm: Augmentative generation of challenging traffic scenarios with an agentic ilm framework,” *arXiv preprint arXiv:2507.13729*, 2025.
- [20] J. Zhou, Y. Zhao, A. Yang, and A. Eichberger, “Benchmarking large language models for motorway driving scenario understanding,” SAE Technical Paper, Tech. Rep., 2025.
- [21] C. Frühwirth-Reisinger, D. Malić, W. Lin, D. Schinagl, S. Schulter, and H. Possegger, “Stsbench: A spatio-temporal scenario benchmark for multi-modal large language models in autonomous driving,” *arXiv preprint arXiv:2506.06218*, 2025.
- [22] Z. Wei, C. Qiang, B. Jiang, X. Han, X. Yu, and Z. Han, “Ad[^] 2-bench: A hierarchical cot benchmark for mlm in autonomous driving under adverse conditions,” *arXiv preprint arXiv:2506.09557*, 2025.
- [23] D. Pei, Y. Wu, J. He, K. Liu, M. Chen, X. Xiao, S. Zhang, and J. Zheng, “Methodology and benchmark for automated driving theory test of large language models,” *IEEE Transactions on Intelligent Transportation Systems*, 2025.
- [24] S. Xie, L. Kong, Y. Dong, C. Sima, W. Zhang, Q. A. Chen, Z. Liu, and L. Pan, “Are vlms ready for autonomous driving? an empirical study from the reliability, data, and metric perspectives,” *arXiv preprint arXiv:2501.04003*, 2025.
- [25] E. Leurent, “An environment for autonomous driving decision-making,” <https://github.com/eleurent/highway-env>, 2018.