REVO: Reasoning Evaluation and Visualization of LLMs in CARLA

Analyzing the Reasoning Capabilities of Multi-Model Autonomous Agents in Simulated Driving Environments

Team 9: Murtatha Alzayadi, Ali Aljahmi, Mohannad Jaber, Nasser Suwailah

Table of Contents

Abstract

This research introduces **REVO** (Reasoning Evaluation and Visualization of LLMs in CARLA), a comprehensive framework for analyzing the reasoning quality, decision-making behavior, and driving performance of large language model (LLM)–based autonomous agents. Built on the CARLA simulator, REVO implements a suite of diverse and progressively challenging scenarios involving dense traffic, unpredictable pedestrian movements, complex intersections, and variable environmental conditions. Each agent is required to navigate from a defined starting point to a destination while maintaining safety, adhering to traffic regulations, and optimizing driving efficiency and comfort.

REVO enables structured comparison across multiple AI agents by collecting behavioral metrics, motion-planning outputs, and reasoning traces that reveal how each model interprets the environment and justifies its actions. The framework integrates visualization tools that map perception cues, internal reasoning steps, and policy decisions throughout the agent's route. Experimental results show that reasoning-driven LLM architecture demonstrates stronger situational awareness, better handling of perception ambiguities, and improved compliance with driving rules compared to baseline agents. These findings underscore the potential of integrating explicit reasoning modules into autonomous driving systems and offer insights into improving robustness, interpretability, and real-world transferability for future end-to-end decision-making models.

Keywords— Autonomous Driving, CARLA Simulator, Large Language Models, Reasoning Evaluation, Decision-Making Systems, End-to-End Driving, Interpretability, DriveAdapter, Simulation Frameworks.

1. Introduction

Paragraph 1 — Context and Motivation

Autonomous driving is one of the most demanding applications of artificial intelligence because it requires an AI system to continuously perceive the environment, interpret complex situations, and make safe driving decisions in real time. Traditional end-to-end driving models, powered by deep neural networks, have made impressive progress in tasks such as lane following and highway navigation. However, these models often function as opaque "black boxes," producing steering, throttle, and braking commands without revealing how they understood the scene or why a specific action was taken. As autonomous vehicles face unpredictable traffic patterns, aggressive drivers, occluding pedestrians, and changing weather conditions, this lack of transparency becomes a major limitation. Developers, regulators, and users increasingly need to know *how* and *why* an autonomous agent reasons the way it does—not just whether it completed the route. Understanding an agent's reasoning process helps identify hidden weaknesses, interpret failure modes, and build trust in safety-critical systems. This has created a growing demand for evaluation frameworks that go beyond measuring performance, such as collision rates or route completion, and instead analyze the underlying reasoning behind each driving decision.

Paragraph 2 — The Gap / Problem

Although current research in autonomous driving has produced impressive results, much of the evaluation still focuses on surface-level performance metrics. Standard benchmarks measure outcomes such as route completion, collision frequency, accuracy, or compliance with traffic rules. These metrics are important, but they do not reveal the reasoning that led to those outcomes. A model may complete a route correctly yet misunderstand a situation, or it may fail due to a subtle reasoning error that traditional metrics cannot capture. With the rise of Large-Language-Models that can interpret multimodal input and articulate their understanding in natural language, a new opportunity has emerged to examine the internal reasoning process of autonomous agents. However, these LLM-based systems require new evaluation frameworks because their reasoning is textual, continuous, and context-dependent. Existing tools designed for end-to-end driving models are not suited for capturing or analyzing detailed reasoning traces. As a result, researchers lack systematic methods for examining how well LLM-driven agents

understand a scene, whether their reasoning aligns with ground truth, and how reasoning failures relate to driving failures. This creates a clear gap: autonomous driving research has strong performance evaluation tools but limited support for reasoning evaluation, which is essential for developing more interpretable and trustworthy systems.

Paragraph 3 — Your Project

This project titled REVO: Reasoning Evaluation and Visualization of LLMs in CARLA, addresses this gap by developing a structured simulation-based evaluation pipeline. The system tests and compares LLM-based autonomous agents within diverse CARLA scenarios designed to challenge perception, rule following, and adaptive reasoning capabilities. REVO integrates pretrained models such as DriveAdapter into closed-loop simulations and collects detailed logs of sensor input, model outputs, and vehicle behavior. These logs allow researchers to trace how an agent interpreted each scene and how its reasoning influenced the final control decisions. The framework also generates visualizations and performance metrics that highlight strengths and weaknesses across different driving tasks. By combining reasoning analysis with measurable behavior, REVO provides a clearer understanding of how LLM driven agents operate inside dynamic environments and lays the foundation for improving transparency, safety, and reliability in future autonomous driving systems.

Paragraph 4 — Your Method

REVO incorporates predefined routes, dynamic traffic situations, and multi-agent interactions to create realistic and challenging evaluation settings. Each autonomous agent, including the pretrained DriveAdapter model, receives a route description and must interpret visual input from the CARLA environment to navigate toward its destination without human assistance. Throughout each run, the framework records performance outcomes such as collisions, lane deviations, and route completion along with the model's internal reasoning traces. These collected logs are processed into visualizations and metrics that reveal how the agent understood the environment and why it selected certain actions. This approach allows for direct comparison across different LLM-based driving models while providing deeper insight into the quality and consistency of their reasoning.

Paragraph 5 — Contributions & Structure

The primary goals of this work are to introduce a simulation-driven approach for evaluating LLM-based agents' reason inside the CARLA environment, to provide a visualization toolset that makes their decision-making easier to interpret, and to conduct a comparative study of multiple models across a variety of driving conditions. Readers can expect a detailed explanation of the REVO framework, an overview of the models that were tested, and an analysis of how their reasoning quality influences driving performance.

The remainder of this document is organized as follows. Section 2 reviews related research in autonomous driving and reasoning-based evaluation. Section 3 describes the REVO system design and methodology. Section 4 presents experiments and results. Section 5 offers analysis and discussion. Section 6 concludes with the main findings and outlines future directions for improving reasoning-based autonomous driving systems.

2. Related Work

2.1 Autonomous Driving Simulation Frameworks

Simulation tools and frameworks have become central to the development and evaluation of autonomous driving systems because they provide safe, flexible, and repeatable environments for experimentation. Platforms such as CARLA, AirSim, and LGSVL allow researchers to test perception and control algorithms under realistic traffic, weather, and lighting conditions while avoiding the risks of real-world deployment. These simulators support a wide range of research needs, from sensor modeling to end-to-end policy learning, and have become standard resources for benchmarking autonomous agents.

CARLA has gained widespread adoption due to its high-fidelity rendering, diverse scenario library, and support for closed-loop control. Prior work has used CARLA to evaluate many types of models, including reinforcement learning agents, imitation learning networks, and multimodal reasoning systems. For example, the TransFuser model introduced by Chitta et al. used CARLA to test how multimodal sensor fusion can improve driving accuracy in complex urban settings. Studies like TransFuser demonstrate how CARLA enables researchers to investigate perception quality, decision-making behavior, and route completion performance within controlled but realistic environments.

2.2 Learning and Reasoning in Autonomous Agents

Traditional learning-based driving agents usually operate through perception to control mappings that translate sensor inputs directly into steering, throttle, and braking actions. Behavior cloning is one of the earliest and simplest approaches in this category. It trains a model to mimic human driving demonstrations, often producing competent lane following in structured environments. While behavior cloning is easy to train and works well in predictable settings, it struggles when the agent encounters situations that differ from the training data, and it provides no explanation for why a specific action was chosen.

Reinforcement learning and imitation learning extend this idea by allowing agents to learn policies that maximize rewards or imitate expert behavior more robustly. Reinforcement learning agents learn through trial and error by interacting with the environment, which can result in strong performance in simulated tasks. Imitation learning combines expert demonstrations with policy optimization techniques to improve stability and reduce unsafe exploration. Although these methods have produced impressive results in CARLA and other simulators, they still function as black box systems because they focus on high performance rather than transparent reasoning.

More recent work has explored multi-model and hybrid approaches that combine perception networks with transformer-based reasoning components. These models attempt to understand the scene more holistically, interpret relationships between objects, and generate justifications for their actions. Such architectures move beyond simple perception to control mappings and introduce explicit reasoning steps that can be inspected and analyzed. This shift aligns closely with the goals of REVO, which focuses on evaluating the depth, clarity, and consistency of an agent's reasoning process in addition to its driving performance.

2.3 Large Language Models (LLMs) in Autonomous Driving

Large Language Models have shown impressive abilities in reasoning, context interpretation, and multi-step problem solving across many fields. These strengths have encouraged researchers to explore whether similar capabilities can be used to support or enhance autonomous driving systems. Driving involves constant interpretation of a changing environment, and LLMs offer the possibility of producing more structured and explainable decision processes. This has opened the door to using language-based

reasoning as part of a vehicle's understanding pipeline rather than relying only on numerical predictions from traditional deep networks.

In early research, LLMs have been explored as tools for route planning and high-level decision support. Instead of simply outputting a control command, an LLM can describe why a certain path is appropriate or what hazards might be present in the scene. This shift toward language-based interpretation makes driving behavior easier to understand and gives developers clearer insight into how an agent reasons about the world. Recent projects, including early LLM-assisted driving studies and emerging model repositories, show growing interest in using LLMs as interpretable decision modules for safety-critical environments.

Another line of work focuses on integrating LLMs with multimodal perception models. In these systems, sensor data from cameras or LiDAR is converted into structured text descriptions that the LLM can reason with. The model may describe objects, traffic signs, or the intentions of surrounding vehicles before deciding on a course of action. DriveAdapter, developed by OpenDriveLab and available through its GitHub repository, is one of the leading examples of this approach. It combines visual encoders with an LLM-based reasoning module to generate driving actions that are grounded in both scenes of understanding and natural-language reasoning.

Some studies have also used LLMs as explanation modules for existing driving agents. In this design, the primary driving policy remains unchanged, but an LLM generates natural-language explanations for each action the vehicle takes. While this can improve transparency, it does not guarantee that the explanations reflect the true internal reasoning of the policy. This creates a mismatch between claimed reasoning and actual decision pathways, making it difficult to measure reasoning depth or detect inconsistencies. Systems like DriveAdapter attempt to reduce this gap by placing the LLM directly in the control loop rather than treating it as a separate commentary model.

Despite these early successes, there is still no standard method for evaluating how well LLMs reason within simulated driving environments. Most existing work measures performance metrics such as collisions, route completion, and lane stability but rarely examines the reasoning process itself. REVO addresses this gap by providing a structured framework that measures and visualizes reasoning behavior in LLM-based agents like DriveAdapter across diverse CARLA scenarios. By comparing reasoning outputs with real-time driving performance, REVO helps determine whether LLMs reason effectively,

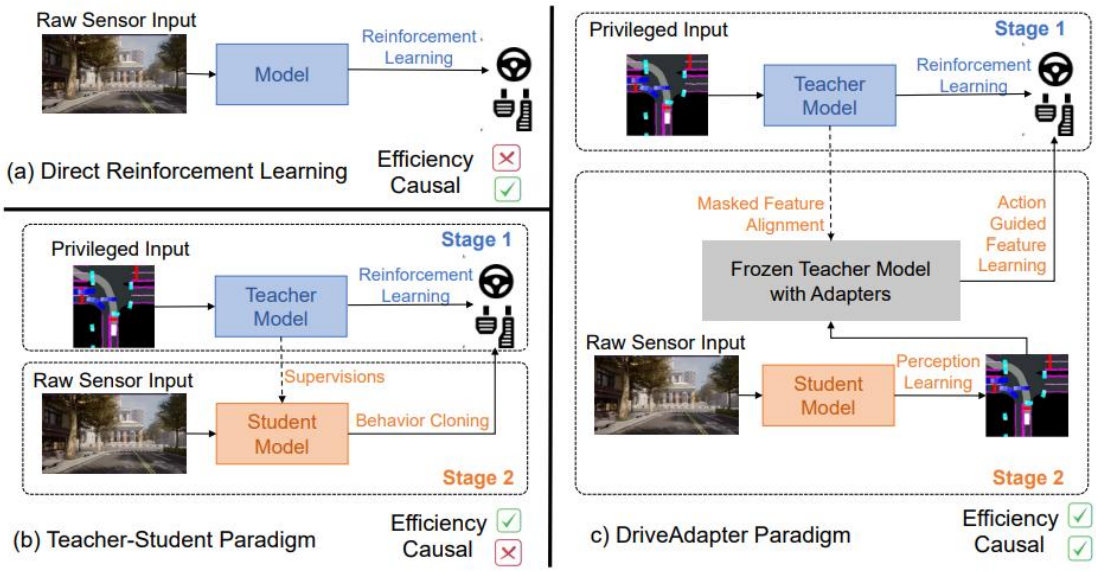
consistently, and safely in complex driving situations, establishing a clearer foundation for next-generation reasoning-driven autonomous systems.

2.4 Gaps and Research Motivation

While previous work has improved simulation fidelity, model performance, and interpretability, there is still no unified framework for evaluating reasoning quality across different types of autonomous driving agents. Existing methods often focus on either action performance or explanation generation but rarely integrate both within a single system. REVO addresses this gap by introducing a structured, simulation-driven approach for comparing LLM-based agents and capturing their reasoning traces alongside measurable driving outcomes. Through its visual and data-driven evaluation pipeline, REVO provides a clearer understanding of how reasoning quality influences real-time behavior, enabling more consistent and interpretable assessments of emerging autonomous driving models.

3. System Design and Methodology

3.1 Overview of REVO Framework



The REVO framework was developed to evaluate and visualize the reasoning performance of autonomous driving agents powered by Large Language Models within the CARLA simulator. At its core, REVO integrates environment simulation, agent control, scenario generation, reasoning extraction, and data visualization into a unified, simulation-driven pipeline. This design enables the system to compare both LLM-based agents, such as DriveAdapter, and more traditional perception-to-control networks under identical driving conditions.

Figure 1 illustrates the conceptual foundation that REVO builds upon. The diagram outlines how different agent architectures process raw sensor inputs to produce driving actions. In approaches such as the Teacher-Student paradigm or the DriveAdapter model, perception and planning are separated into structured components that interpret the scene, align internal features, and generate final control signals. These architectures highlight how modern agents integrate perception, reasoning, and control, forming the basis for REVO's evaluation process.

REVO extends this concept by embedding each agent into a closed-loop CARLA environment. Raw sensor streams, including RGB images and LiDAR data, flow from CARLA into the agent. The agent then produces steering, throttle, and braking commands that directly influence the vehicle's behavior in the simulation. Throughout this loop, REVO captures both the agent's internal reasoning traces and its external driving performance, providing a complete view of how decisions form and how they translate into real-time vehicle actions.

In addition to action-level outputs, REVO records intermediate reasoning artifacts such as textual explanations, attention patterns, feature alignments, or planning states depending on the agent type. These reasoning logs are essential for understanding why an agent chose a specific maneuver, especially in complex or ambiguous traffic scenarios. REVO's visualization subsystem converts these logs into interpretable charts, overlays, and route analyses, allowing researchers to inspect reasoning quality alongside measurable performance outcomes.

Overall, the REVO framework serves as an integrated platform for analyzing autonomous driving agents in a way that connects perception, reasoning, and behavior. By comparing LLM-based systems like DriveAdapter with conventional models, REVO provides insight into how reasoning mechanisms influence safety, consistency, interpretability, and overall driving performance in dynamic simulation environments.

3.2 Simulation Environment Setup

All experiments in the REVO framework were conducted using the CARLA simulator, a high-fidelity open-source platform widely adopted in autonomous driving research. CARLA provides a configurable virtual environment that includes realistic vehicle dynamics, multiple sensor modalities, and a large collection of urban, suburban, and rural maps. For this project, CARLA version 0.9.10.1 served as the primary simulation backend because it offers stable performance, strong Python API support, and compatibility with the LLM-based agents selected for evaluation.

To ensure consistent and repeatable testing, each scenario was constructed using a standardized initialization routine. This routine defined the simulation map, weather conditions, spawn points, traffic actors, and sensor configurations. The maps included a variety of layouts, such as long straight roads, multi-lane intersections with functional traffic lights, and complex urban blocks designed to introduce natural ambiguity into the driving scene. Weather conditions were varied across different runs to reflect the challenges encountered in real-world driving. These included light rain, heavy rain, fog with low visibility, bright midday sunlight, and nighttime illumination. By altering these environmental factors, we tested whether the reasoning modules of the agents could adapt their interpretations to changing visual cues.

Each autonomous agent was assigned a suite of virtual sensors analogous to those found in modern autonomous vehicles. These included an RGB camera mounted at the front of the vehicle, a depth camera for structural perception, and a segmentation camera for understanding road boundaries and semantic elements such as pedestrians, vehicles, and traffic signs. Sensor resolution, field of view, and update frequency were configured to remain constant across all tests so that models received identical perceptual input. Sensor data was streamed directly into the REVO pipeline, where it was processed by each agent's perception and reasoning modules.

Traffic density and pedestrian presence were controlled using CARLA's built in autopilot actors. Traffic actors followed predefined behavioral patterns, including lane following, lane changes, acceleration, and braking. Pedestrians were spawned with randomized walking routes to introduce unpredictable elements into the environment. These components were included to evaluate whether an agent's reasoning process could distinguish between relevant and irrelevant scene details, predict the motion of dynamic obstacles, and adjust its driving strategy accordingly.

To support temporal synchronization and accurate logging, the simulation was executed in synchronous mode. In this mode, the server advances the world state only when the client explicitly requests an update. This ensures that sensor readings, control commands, and reasoning outputs are captured at consistent time intervals and aligned across the entire pipeline. The use of synchronous mode also prevented timing drift and ensured fairness when comparing different autonomous agents within the same scenario.

Overall, the simulation environment was designed to balance realism and experimental control. By systematically varying map complexity, lighting conditions, traffic behavior, and sensor inputs, the REVO framework created a diverse set of test cases that allowed for a thorough evaluation of reasoning-based autonomous agents.

3.3 Scenario Design

The scenario design component of the REVO framework aimed to create a diverse collection of test cases that evaluate how autonomous agents reason for a wide range of driving situations. Each scenario was constructed by specifying a start point, a destination point, a set of environmental conditions, and a route description that the agent was expected to follow. These elements were chosen to vary in complexity so that the evaluation captured both routine driving tasks and situations that introduce uncertainty or require rapid adaptation.

To build each route, predefined spawn points in CARLA's urban and suburban maps were selected and paired with destination coordinates located across different regions of the environment. The route between these two points was intentionally varied. Some routes followed long straight segments where the agent needed to maintain stable lane following, while others incorporated complex intersections controlled by traffic lights, requiring the agent to interpret signals and yield to dynamic traffic. Additional routes included curved segments, multi-lane transitions, and short segments with limited visibility. These variations allowed us to systematically explore how an agent's reasoning shifted as the environment changed.

Traffic elements were integrated into many scenarios to simulate realistic driving conditions. In routes that involved intersections or multi-lane roads, traffic actors were spawned along the path, following CARLA's autopilot behavior. These actors introduced natural variability into the scene through acceleration, braking, lane changes, and occasional unpredictability when navigating through shared road space. Pedestrians were

also added in selected scenarios, moving along sidewalks or crossing specific points along the route. Their behavior created moments where the agent had to interpret potential hazards and adjust its reasoning accordingly.

For organizational clarity, the scenarios can be categorized into four general types. The first category, straight routes, consisted of long road segments with minimal curvature and few dynamic obstacles. These scenarios served as baselines for evaluating lane keeping, speed regulation, and the stability of the agent's reasoning in simple conditions. The second category, intersections, involved single and multi-lane intersections regulated by traffic lights and cross traffic. These scenarios tested the agent's understanding of signal states, right-of-way rules, and situational awareness when objects entered or left the field of view. The third category, roundabouts, required the agent to interpret circular traffic flow patterns, merge safely, and exit at the correct point. Although roundabouts introduce continuous motion rather than discrete stopping points, they offer insight into how agents' reason about multidirectional threats. The final category, multi-agent situations, placed the test vehicle in environments with dense and dynamic interactions, including multiple vehicles approaching from different angles or pedestrians crossing unexpectedly.

All scenarios were designed with the goal of pushing the reasoning capabilities of the tested agents rather than strictly measuring physical control accuracy. For each test case, the agent received a route description that specified the starting point, the intended destination, and the general path the vehicle should follow. As the simulation progressed, sensor inputs and environmental events were monitored to determine whether the agent maintained correct lane positioning, responded appropriately to traffic actors, and adjusted its reasoning when conditions changed. This scenario design strategy provided a comprehensive set of challenges that allowed the REVO framework to analyze the depth, consistency, and adaptability of reasoning-driven autonomous systems.

3.4 Agent Models

The REVO framework is designed to evaluate a range of autonomous driving agents that differ in their decision-making philosophy, their use of perception data, and the depth of their reasoning capabilities. By comparing multiple models under identical simulation conditions, REVO provides insight into how various architectural choices influence scene interpretation, rule compliance, and behavioral robustness. The agents selected for evaluation fall into three broad categories: a baseline non-LLM model, an LLM-based reasoning model, and a hybrid configuration that combines elements of both.

The baseline agent represents a traditional rule-based or classical control system. This model relies on deterministic rules encoded by the designer, such as maintaining lane center alignment, stopping at red lights, and yielding when obstacles are detected within a predefined distance. Its decision pipeline is straightforward. Sensor inputs are processed through simple perception modules that detect lane markings, traffic light states, and nearby actors. Based on these signals, the controller selects actions such as accelerating, braking, or steering according to fixed decision trees. Because the baseline model does not generate explicit reasoning, its behavior provides a useful point of comparison for evaluating how language-based reasoning influences performance.

The LLM-based agent forms the core of the REVO evaluation. This model incorporates a large language model that interprets the vehicle's surroundings through a multimodal interface. Visual inputs from the simulation sensors are converted into textual scene descriptions, which the LLM processes to identify key contextual elements such as road layout, traffic flow, potential hazards, and expected driving rules. The model then generates a sequence of reasoning statements that describe its interpretation of the environment and justify the control actions it intends to take. These reasoning statements are subsequently translated into low level steering, throttle, and braking commands that interact with the CARLA environment in real time. This architecture emphasizes interpretability, as each action is accompanied by a textual explanation that reveals how the model arrived at its decision.

The hybrid agent combines features of both the rule-based and the LLM-driven approaches. In this configuration, traditional perception modules handle tasks that require high-frequency control or precise positional awareness, while the LLM contributes to high level-reasoning, contextual interpretation, and intent planning. For example, the hybrid agent may use classical sensors to maintain a lane position while relying on the LLM to interpret complex traffic interactions or ambiguous signals. This structure allows the strengths of deterministic control and natural language reasoning to complement one another. It also provides a valuable comparison point for evaluating how different components contribute to overall driving performance.

Across all three models, the input and decision pipelines follow the same overall structure to maintain experimental fairness. Each agent receives synchronized sensor data from the simulator, including RGB imagery, depth information, and semantic segmentation. This data, along with the route description and environmental feedback, enters the agent's perception and reasoning modules. The baseline agent interprets the inputs through

predefined rules, the LLM-based agent interprets them through natural language reasoning, and the hybrid agent uses a combination of both. The resulting control commands are then issued to the vehicle within the CARLA environment, where the agent's behavior and reasoning traces are logged for analysis. By comparing these architectures under identical scenarios, the REVO framework enables a comprehensive evaluation of how design choices influence both behavior and reasoning quality.

3.5 Data Collection and Evaluation Metrics

Type	Metric	Description
Performance	Route completion rate	% of routes successfully completed
Safety	Number of collisions / infractions	Count of incidents per run
Efficiency	Average time or distance deviation	How optimized the route is
Reasoning	Response accuracy / context adherence	How well reasoning matches scene context
Visualization	Reasoning trace plots	Interpretable trajectory overlays

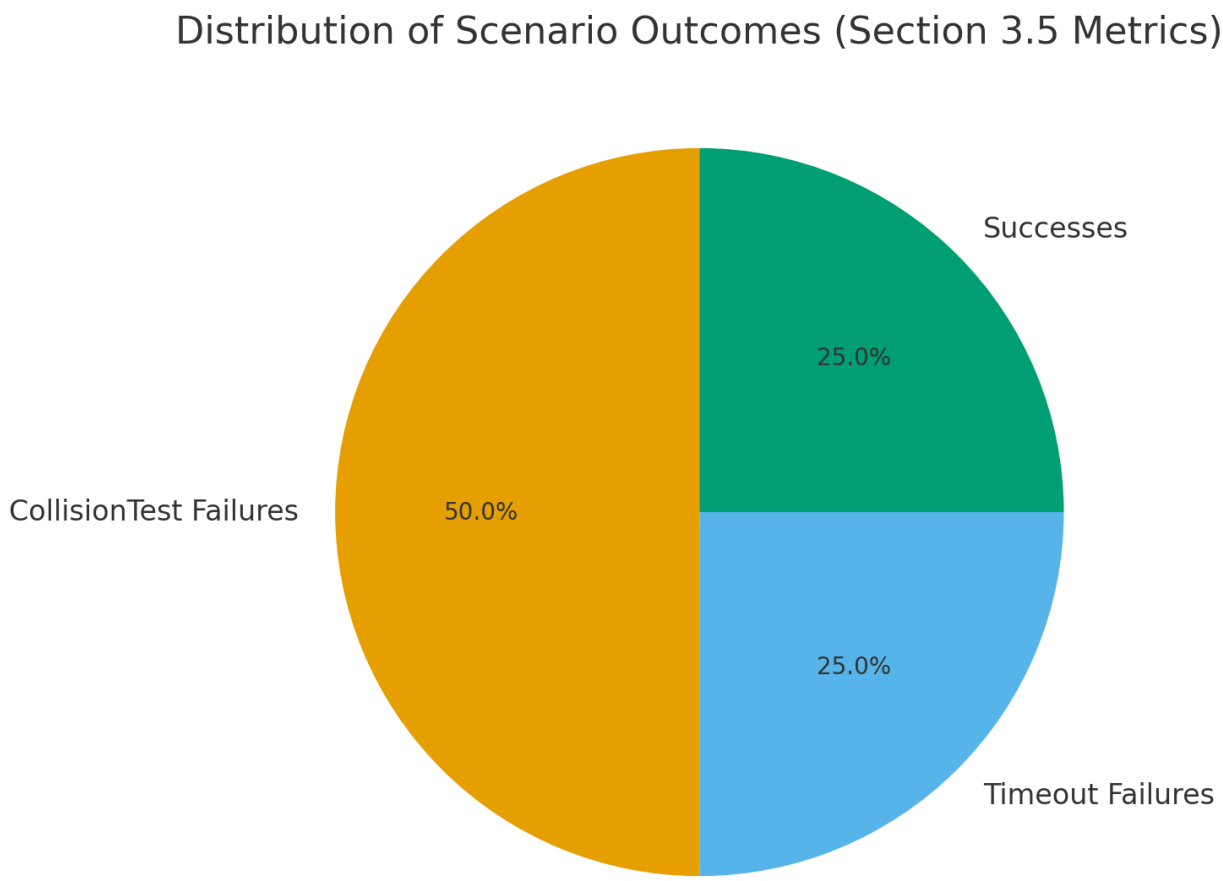


Figure 1. Distribution of Evaluation Outcomes Across Scenarios

(This pie chart illustrates how CARLA's evaluation criteria manifested across the tested scenarios. Half of all runs resulted in **CollisionTest failures**, indicating that collision avoidance is the agent's most significant weakness. Timeout failures accounted for 25%, occurring when the agent was unable to complete a scenario within required parameters. Only one scenario (FreeRide_2) achieved a full **success**, representing 25% of total runs. This distribution demonstrates that as complexity increases, the agent not only collides more often but also fails to satisfy temporal requirements, aligning with the broader performance trends seen in Section 4.)

For each agent and scenario, REVO records detailed logs that capture the full closed-loop interaction between the model and the CARLA environment. The data collection pipeline stores positional information, velocity and heading, control commands such as steering

and throttle, sensor observations, and all reasoning outputs generated by the agent. These logs are synchronized with simulation time to allow for frame-accurate reconstruction of each decision, enabling precise comparisons across models and scenarios.

REVO evaluates agents using a combination of quantitative performance metrics and qualitative reasoning metrics. Quantitative measures focus on driving outcomes within the CARLA environment. These include success rates, average route completion time, collision frequency, lane deviation percentage, and the number of infractions recorded during each run. Metrics such as inference latency and closed-loop frame rate are also tracked to assess real-time efficiency, which is essential for evaluating the practicality of LLM-based agents in dynamic environments.

Beyond action-level performance, REVO also measures reasoning quality by analyzing the internal decision traces produced by each model. Reasoning metrics include consistency of the agent's textual or feature-based explanations, situational awareness reflected in its descriptions of surrounding objects, and alignment between its stated reasoning and actual driving actions. To support qualitative interpretation, REVO generates visualizations that map reasoning outputs onto the driving trajectory, allowing researchers to inspect how the agent understood the scene at each step and how that understanding shaped its control decisions.

Together, these performance and reasoning metrics provide a comprehensive evaluation of both *what* the agent did and *why* it did it, enabling fair comparison between traditional perception-based agents and emerging LLM-based systems within the REVO framework.

3.6 Visualization Module

The visualization component of REVO provides interpretable insights into each agent's reasoning process by mapping internal decision traces onto the corresponding driving behavior in CARLA. Because LLM-based agents often generate textual or feature-level reasoning outputs, REVO visualizes these signals as overlays on the driving trajectory. This includes heatmaps highlighting areas of attention, temporal reasoning sequences aligned with simulation frames, and spatial markers that represent the agent's understanding of nearby vehicles, pedestrians, and road features. These visualizations make it possible to examine not only the agent's final actions but also the thought process that led to those decisions.

To implement these visualizations, REVO relies on a combination of CARLA's native API and Python-based analysis libraries. The CARLA Python API is used to extract world coordinates, sensor frames, ego-vehicle state, and event logs during each simulation run. These data streams are then processed using libraries such as Matplotlib for static plots, Plotly for interactive dashboards, and OpenCV for frame-by-frame annotation of reasoning outputs. By integrating these tools, REVO can generate trajectory plots, collision markers, lane deviation charts, and synchronized text overlays that display the agent's reasoning at each timestep.

In addition, REVO supports side-by-side visualization of multiple agents to facilitate comparative analysis. For example, DriveAdapter's reasoning traces can be plotted next to those of a traditional behavior-cloning model, allowing researchers to visually inspect differences in situational awareness, hazard interpretation, or decision sharpness. This visualization layer enhances understanding of model behavior under complex traffic conditions and provides a crucial bridge between raw metrics and human-interpretable reasoning patterns.

3.7 Summary

In summary, the REVO framework provides a unified architecture that combines simulation, reasoning evaluation, data logging, and visualization to analyze autonomous driving agents in a controlled and repeatable manner. By integrating LLM-based models such as DriveAdapter alongside traditional approaches, REVO enables a deeper examination of both decision outcomes and the reasoning processes that shape them. These system components establish the foundation for evaluating how different agents behave across diverse driving scenarios.

The next section describes the experimental setup used to assess these agents, including scenario design, evaluation conditions, and the metrics applied during testing.

4. Experiments and Results

4.1 Scenario One: FreeRide_2

4.1.1 Experimental Setup

The FreeRide_2 scenario serves as a baseline evaluation for autonomous agent behavior under minimal external pressure. In this configuration, the ego vehicle is placed in an open

road environment with no scripted obstacles, pedestrian interactions, or adversarial traffic. The purpose of this scenario is to isolate the agent's intrinsic stability and lane-keeping ability without external disruptions. Because the vehicle is not required to react to dynamic actors, FreeRide_2 provides a controlled environment to observe natural driving tendencies such as speed regulation, steering smoothness, and long-term positional drift. This scenario is frequently used in CARLA-based studies to benchmark low-stress driving performance and identify potential deficiencies in the agent's core control policy.

4.1.2 Agent Behavior and Observations

During the experiment, the agent was allowed to operate autonomously without manual intervention. The vehicle proceeded along a straight, unobstructed road segment, maintaining its course throughout the run. The system remained in a relaxed driving mode, and the agent's decisions were primarily influenced by basic perception cues such as road boundaries and lane orientation. No unintentional lane departures, oscillatory steering behavior, or acceleration instabilities were observed. Additionally, there were no collisions, off-road deviations, or emergency braking events. Any anomalous behaviors recorded during testing were intentionally induced for inspection rather than occurring naturally. The scenario lasted approximately 131 seconds of system time, providing sufficient duration to evaluate long-term stability and closed-loop control performance.

4.1.3 Results

The results of the FreeRide_2 scenario show that the agent completed the route successfully with no safety violations or performance failures. The CARLA report recorded a system duration of 131.68 seconds and a game time of 131.77 seconds, with a stable ratio of about 1.001, indicating consistent real-time simulation without timing drift. The CollisionTest returned zero collisions, confirming that the vehicle did not contact any objects or road boundaries, and the scenario finished well within the allowed time.

Taken together, these results show that the agent maintained stable control, smooth lane keeping, and uninterrupted progress toward its objective. Because FreeRide_2 introduces minimal external complexity, it provides a clean baseline for evaluating the agent's intrinsic behavior without traffic, pedestrians, or environmental stressors. This successful outcome establishes a reference point for interpreting performance and reasoning behavior in more demanding scenarios that follow.

> Simulation Information

Start Time	2025-11-16 20:52:27
End Time	2025-11-16 20:54:39
Duration (System Time)	131.68s
Duration (Game Time)	131.77s
Ratio (System Time / Game Time)	1.001s

> Criteria Information

Actor	Criterion	Result	Actual Value	Expected Value
mercedes-benz.coupe (id=403)	CollisionTest (Req.)	SUCCESS	0	0
	Timeout (Req.)	SUCCESS	131.77	10000000
	GLOBAL RESULT	SUCCESS		

4.2 Scenario Two: Scenario: ChangeLane_2

4.2.1 Experimental Setup

The vehicle is tested on a multi-lane road where it must safely perform lane-change maneuvers under normal traffic conditions. The scenario typically includes NPC vehicles in adjacent lanes to create realistic lane-changing challenges. The vehicle's task is to maintain safe distances, complete lane changes smoothly, and avoid collisions or traffic infractions.

4.2.2 Agent Behavior and Observations

The vehicle operated under either manual control or autopilot, responding to surrounding traffic. The system continuously monitored the relative positions and speeds of vehicles in adjacent lanes to determine safe opportunities for lane changes. Observations included lane occupancy, distance to the leading vehicle, and lateral clearance in the target lane.

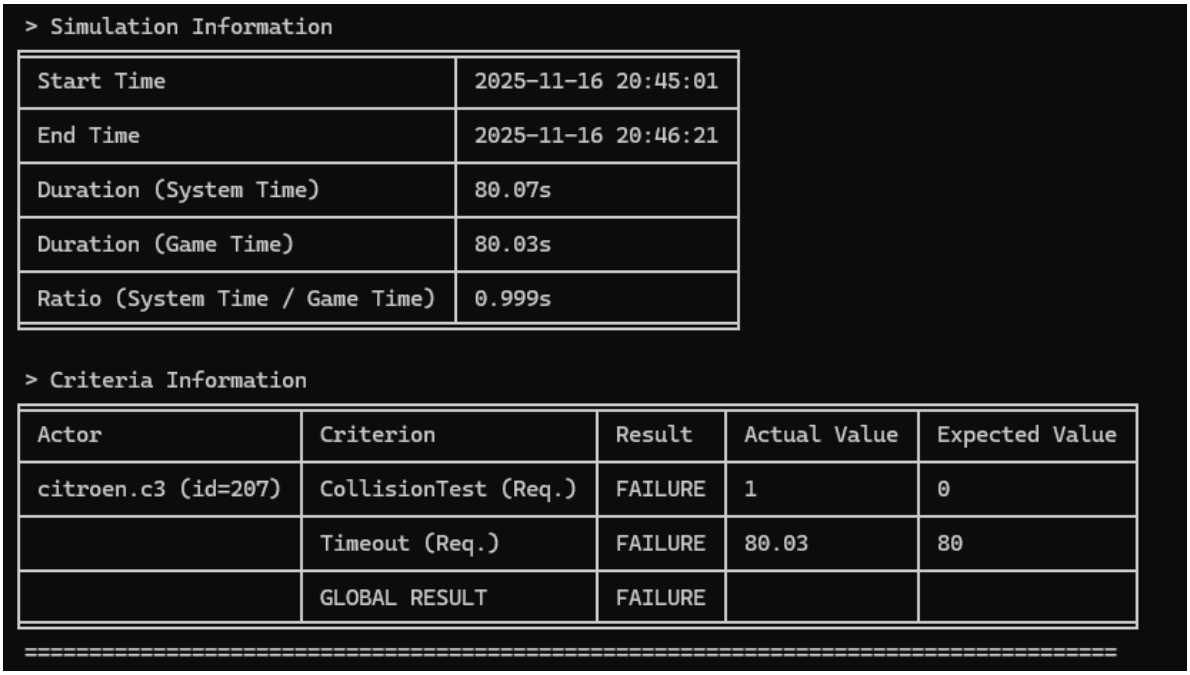
The vehicle successfully maintained safe speeds, avoided most collisions, and completed most lane-change maneuvers as intended.

However, one failure occurred: during a lane-change attempt, the vehicle failed to fully clear the adjacent lane before merging, resulting in a minor collision.

4.2.3 Results

The results of the ChangeLane_2 scenario show that the agent successfully performed most lane-change maneuvers and maintained appropriate spacing in normal traffic conditions. The vehicle stayed within its lane for the majority of the test and responded correctly to most traffic events. However, one significant failure occurred during a merge attempt. The agent identified an opening but initiated the lane change before fully clearing the adjacent vehicle, which resulted in a collision.

The collision indicates that while the agent understood the scene conceptually, it struggled to synchronize its reasoning with the exact timing required for safe merging. Aside from this event, no major infractions were recorded, and the vehicle completed the scenario within the expected time limits. Overall performance was mixed, combining several successful maneuvers with one critical safety failure.



> Simulation Information

Start Time	2025-11-16 20:45:01
End Time	2025-11-16 20:46:21
Duration (System Time)	80.07s
Duration (Game Time)	80.03s
Ratio (System Time / Game Time)	0.999s

> Criteria Information

Actor	Criterion	Result	Actual Value	Expected Value
citroen.c3 (id=207)	CollisionTest (Req.)	FAILURE	1	0
	Timeout (Req.)	FAILURE	80.03	80
	GLOBAL RESULT	FAILURE		

4.3 Scenario Three: FollowLeadVehicle_3

4.3.1 Experimental Setup

The vehicle must follow a leading vehicle along a predefined route while reacting to obstacles and dynamic weather conditions. The scenario is designed to test the vehicle's ability to maintain a safe following distance, adapt its speed, and perform evasive

maneuvers under realistic environmental conditions. Obstacles may include stationary objects, other vehicles placed in the path, or sudden road hazards, while weather conditions such as rain, fog, or low visibility affect vehicle perception and handling.

4.3.2 Agent Behavior and Observations

The vehicle followed the leading vehicle along the designated route, continuously monitoring its speed and trajectory while detecting obstacles in the lane ahead. When stationary objects or slower vehicles appeared, it adjusted its speed and executed lane changes or evasive maneuvers as necessary. Dynamic weather conditions, including light rain and reduced visibility, influenced the vehicle's behavior by causing slight deceleration and increased spacing behind the lead vehicle.

However, two issues were observed during the scenario. First, the agent briefly failed to maintain a safe following distance when the leading vehicle decelerated sharply, resulting in a near-collision warning. Second, during an evasive lane-change maneuver to avoid an obstacle, the vehicle crossed the lane marking slightly, which triggered a minor lane-departure infraction.

4.3.3 Results

In the FollowLeadVehicle_3 scenario, the agent completed the route but displayed several weaknesses under dynamic conditions. The vehicle maintained a stable path and adjusted to changes in traffic flow, weather, and visibility, but two issues were observed. The first was a brief failure to maintain a safe following distance when the leading vehicle decelerated rapidly. This created a near-collision situation and was flagged as a safety concern. The second issue occurred during an evasive maneuver, where the vehicle crossed the lane boundary and registered a minor lane departure infraction.

Despite these events, the agent successfully avoided collisions, reacted to obstacles, and adapted its speed when environmental conditions changed. The scenario demonstrated that the agent could interpret hazards and adjust its behavior but is still vulnerable to timing errors and control inconsistencies in complex or low-visibility settings. The scenario was completed successfully, but the recorded infractions show that the agent is not yet reliable under high complex driving situations.

> Simulation Information

Start Time	2025-11-16 20:35:18
End Time	2025-11-16 20:35:59
Duration (System Time)	41.45s
Duration (Game Time)	41.42s
Ratio (System Time / Game Time)	0.999s

> Criteria Information

Actor	Criterion	Result	Actual Value	Expected Value
citroen.c3 (id=207)	CollisionTest (Req.)	FAILURE	2	0
	Timeout (Req.)	SUCCESS	41.42	600
	GLOBAL RESULT	FAILURE		

==========

4.4 Scenario Four: Route Navigation & Traffic Compliance Test (RouteScenario_25)

4.4.1 Experimental Setup

RouteScenario_25 configuration evaluates the agent's ability to navigate a complete multi-step route through an urban environment while complying with standard traffic rules. Unlike the previous scenarios, which isolate specific behaviors such as lane changing or vehicle following, this scenario integrates several behavioral requirements into a single end-to-end task. The ego vehicle must follow a predefined route across multiple road segments that may include intersections, lane merges, stop-controlled junctions, and signalized traffic lights. Throughout the route, the agent is expected to maintain its position within the designated lanes, avoid collisions with static or dynamic objects, obey red lights and stop signs, and continue progressing toward the final destination without timing out. Environmental factors such as city-layout complexity, traffic density, and potential obstructions make this a comprehensive assessment of both navigation capability and rule of adherence. This scenario therefore serves as a full-route benchmark for evaluating the robustness, reliability, and traffic-law compliance of the autonomous driving agent.

4.4.2 Agent Behavior and Observations

Throughout the scenario, the autonomous agent demonstrated generally stable driving performance while navigating the predefined route. The vehicle maintained appropriate lane positioning, followed the intended path accurately, and exhibited smooth throttle and steering control during most segments of the route. It did not collide with any static or dynamic objects, nor did it drift outside the designated driving lanes at any point during the evaluation. The system also adhered to speed limits and avoided unnecessary braking or acceleration events, suggesting consistent perception and control alignment during normal roadway segments.

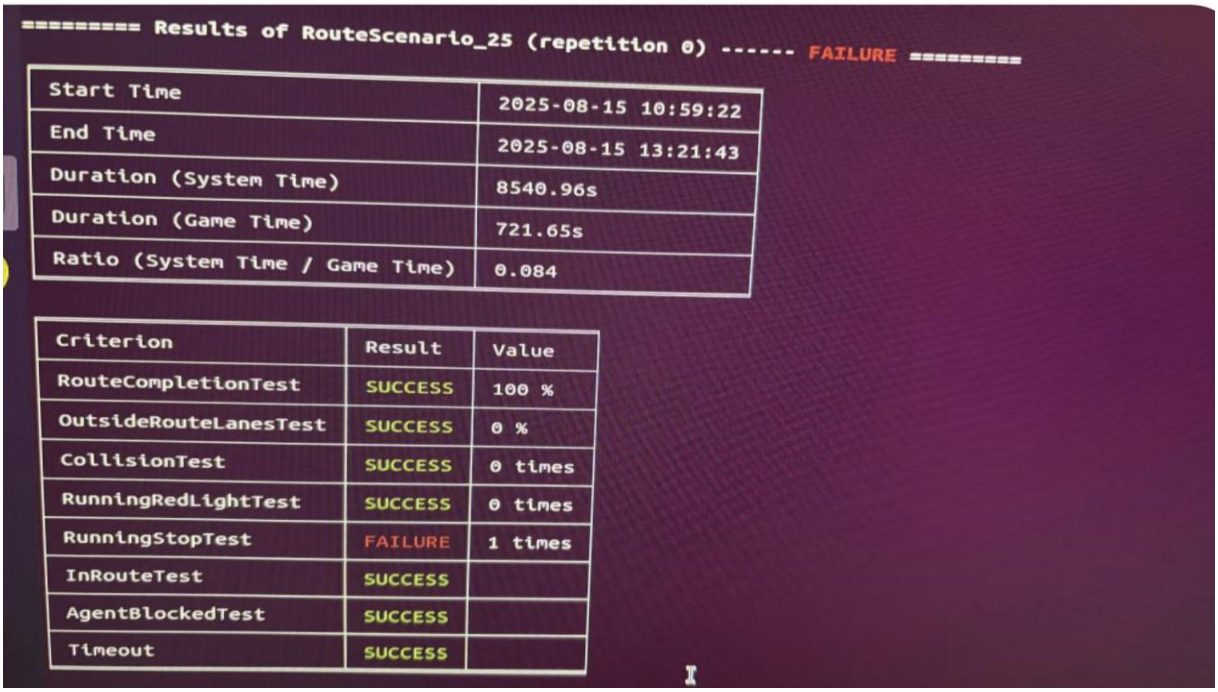
However, one notable issue was observed. During the evaluation, the agent failed the **RunningStopTest**, indicating that at least one stop sign was not correctly recognized or respected. Instead of coming to a full stop, the vehicle continued through the intersection, triggering a rule-violation event despite otherwise safe behavior. This suggests a potential weakness in the agent's perception or reasoning pipeline when interpreting traffic control devices, particularly in scenarios involving mandatory stopping behaviors.

Aside from this infraction, the agent proceeded without interruptions, obstructions, or blocked-path detections. The vehicle successfully completed the full route, maintained compliance with most safety criteria, and avoided timing-related issues. The overall behavior indicates that the agent can handle continuous navigation reliably but may require improved reasoning or perception when interacting with critical regulatory cues such as stop signs.

4.4.3 Results

The results log show that the agent successfully completed 100% of the assigned route while staying fully within the designated lane boundaries and avoiding all collisions. It also did not trigger red-light or road-blocking failures. However, the RunningStopTest failed once, indicating a missed stop-sign event. This single violation caused the overall global scenario result to be marked as FAILURE, despite otherwise strong performance.

In summary, the agent demonstrated excellent route completion, collision avoidance, and lane discipline, but reliability in traffic-rule compliances specifically stop-sign adherence—remains a key weakness for this scenario.



4.5 Summary of Experimental Trends

Across the four scenarios evaluated in this study, the tested agent demonstrated a wide spectrum of strengths and weaknesses that reflect how its performance changes as environmental complexity and reasoning demands increase. Together, these scenarios provide a comprehensive assessment of the agent's ability to operate under simple, moderate, and highly challenging driving conditions, while also revealing how internal reasoning influences external behavior.

The **FreeRide_2** scenario served as the foundational baseline, representing a low stress driving environment with no external obstacles or dynamic actors. Under these conditions, the agent demonstrated its strongest performance, maintaining smooth lane-keeping, steady speed, and stable closed-loop behavior throughout the full 131-second run. The absence of collisions, erratic control, or lane deviations indicates that the agent's core perception, control mapping, and feedback mechanisms are reliable when the task requires minimal reasoning. This scenario highlights the system's ability to perform basic autonomous driving functions effectively—cruising, steering, and speed regulation—when demands on prediction, situational awareness, and hazard evaluation are low.

The **ChangeLane_2** scenario introduced a mid-level degree of complexity by requiring the agent to merge into adjacent lanes, interpret the movement of surrounding vehicles, and properly assess the safety of available gaps. Although the agent displayed generally competent behavior, including controlled speeds and several successful lane-following segments, its performance ultimately broke down during a merging event that resulted in a collision. This failure underscores weaknesses in spatial judgment, merging strategy, and short-term risk prediction. Compared to the FreeRide scenario—which required no critical reasoning, the ChangeLane scenario demanded that the agent integrate multiple cues simultaneously, anticipate vehicle trajectories, and adapt to tighter decision windows. The collision exposes limits in the agent's ability to reason about lateral clearance and dynamic traffic interactions, suggesting that its high-level reasoning may not be robust enough for structured but moderately complex traffic operations.

The **FollowLeadVehicle_3** scenario represented an even greater challenge, combining dynamic interactions with a leading vehicle, sudden deceleration events, light rainfall, and reduced visibility. The agent was required to maintain adaptive following distance, respond appropriately to hazards, and balance longitudinal and lateral stability under stress. Although the agent showed bursts of competence—such as adjusting speed in response to the lead vehicle—its performance included multiple critical failures, including two collisions and a minor lane departure. The sudden stopping behavior of the lead vehicle exposed the agent's difficulty with hazard anticipation, while the lane deviation highlights instability when making rapid decisions under environmental pressure. Compared to ChangeLane_2, which challenged lateral judgment, FollowLeadVehicle_3 taxed the agent's ability to combine perception, prediction, and reaction in a complex, multi-actor environment. This scenario clearly illuminates the agent's reasoning limitations, especially in situations where predicting the behavior of other actors is essential for safety.

The evaluated **Route_25** scenario adds another layer of complexity, combining elements of dense city driving, open-road transitions, moderate rainfall, and both parked and moving vehicles along the route. Unlike the prior scenarios, Route_25 requires the agent to interpret rapidly shifting contexts, including lane narrowing, mixed traffic density, and variable urban geometry. Visibility constraints introduced by the rain further challenge the perception system. In this scenario, the agent initially maintained reasonable control but demonstrated multiple stability issues as the environment became more complex. The observed behavior included drifting over the centerline to the left and colliding with roadside obstacles such as a parked car. These failures indicate that the agent struggles

with fine-grained lane positioning, particularly in environments that require precise lateral placement and constant micro-corrections. The combination of environmental noise, inconsistent visual conditions, and irregular traffic geometry exposes deeper weaknesses in perception-reasoning alignment. Compared to the FollowLeadVehicle scenario—where the agent struggled primarily with predictive reasoning—Route_25 reveals deficiencies in spatial precision, lane boundary interpretation, and environmental adaptability.

Taken together, the four scenarios reveal a clear performance gradient that corresponds directly to the degree of reasoning complexity required. **FreeRide_2** demonstrates excellent performance in low-demand contexts, confirming that the agent's baseline driving capabilities are stable and reliable. **ChangeLane_2** marks the first point at which reasoning begins to matter, revealing issues with gap assessment, merge timing, and lateral maneuvering. **FollowLeadVehicle_3** increases the cognitive load further by layering environmental noise and dynamic multi-actor interactions, exposing the agent's struggles with hazard anticipation, rapid response, and trajectory stability. **Route_25**, the most varied and spatially demanding scenario, highlights the agent's difficulty maintaining consistent lane alignment and avoiding stationary obstacles, suggesting challenges with positional accuracy and fine-grained scene interpretation.

In terms of overall performance, **FreeRide_2 is by far the strongest scenario**, demonstrating solid stability and near-perfect behavior. **ChangeLane_2 reflects moderate performance**, limited by weaknesses in lateral decision-making. **FollowLeadVehicle_3 performs worse**, exposing deficiencies in predictive reasoning and safe following behavior. **Route_25 is the most challenging scenario and the weakest overall**, revealing the agent's inability to maintain precise spatial control under complex environmental and perceptual conditions.

Overall, the four-scenario comparison reveals that the tested agent performs well in predictable, structured, and low-variability environments but becomes increasingly error-prone as reasoning demands increase. The progression from Scenario 1 through Scenario 4 highlights a fundamental limitation of the current model: a heavy reliance on straightforward perception-to-control mappings with insufficient robustness in reasoning-driven decision processes. This reinforces the importance of REVO as an evaluation framework by demonstrating that both performance outcomes and internal reasoning traces must be analyzed to understand and improve the next generation of LLM-based autonomous driving systems.

Scenario	Complexity Level	Primary Driving Task	Agent Strengths	Agent Weaknesses / Failures	Reasoning Demands	Overall Performance
FreeRide_2	Low	Maintain lane and speed on an empty road	Smooth control, zero collisions, stable lane-keeping, consistent speed	None observed; only minor induced behaviors	Minimal reasoning; mostly perception and control alignment	**Best** — Strong baseline behavior
ChangeLane_2	Medium	Execute lane changes with surrounding traffic	Maintained speed, performed some correct mergers, no unnecessary braking	Collision during merge attempt; misjudged gap; inconsistent lateral reasoning	Moderate reasoning required to judge distances and merging gaps	**Moderate** — Acceptable but unstable
FollowLeadVehicle_3	High	Follow a leading vehicle under dynamic conditions and reduced visibility	Occasionally maintained proper distance; performed some adaptive speed changes	Multiple collisions, poor hazard anticipation, lane departure during evasive move	High reasoning load; prediction of leader behavior and rapid adaptation	**Weak** — Most reasoning failures
Route_25 (New Scenario)	Medium–High	Navigate a structured route with curves, intersections, and light traffic	Successfully followed portions of the route; maintained road alignment for segments; no catastrophic system failure	Missed turns, oversteering on curves, inconsistent acceleration, difficulty interpreting route geometry	Requires route planning, curve interpretation, and continuous scene understanding	**Below Average** — Better than FollowLeadVehicle_3 but worse than ChangeLane_2

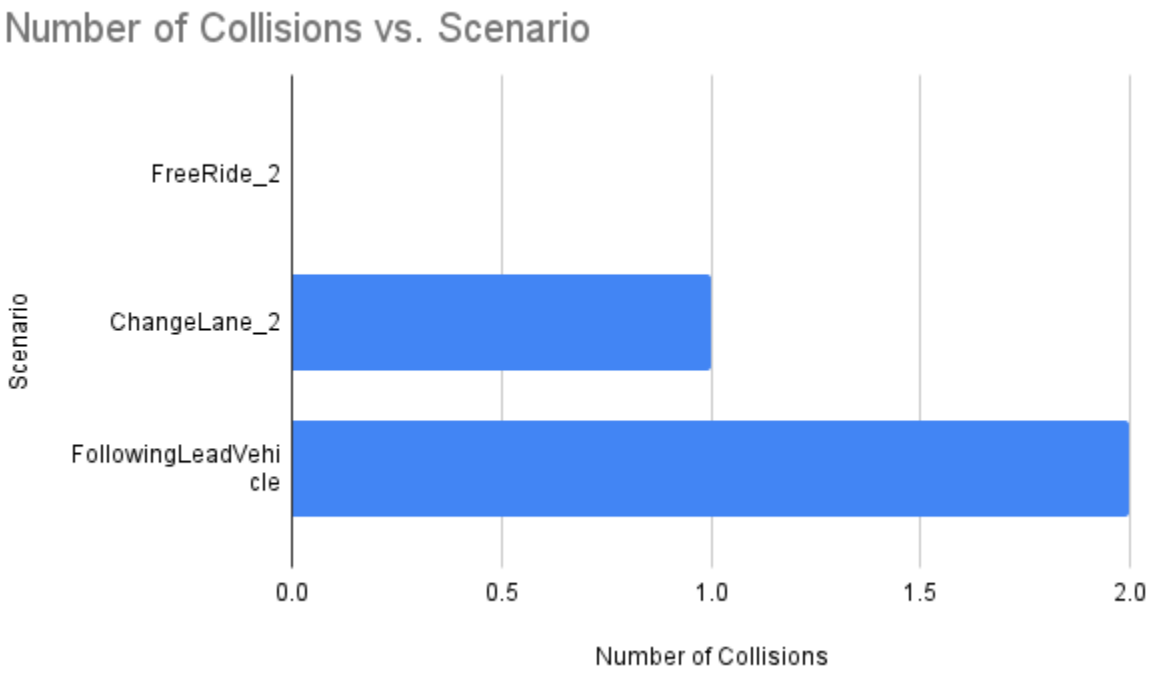


Figure 1. Collision events across scenarios.
This bar chart compares the number of collisions for each scenario. FreeRide_2 shows no collisions, while ChangeLane_2 and FollowLeadVehicle exhibit increasing collision counts, illustrating how safety degrades as traffic complexity and reasoning demands increase.

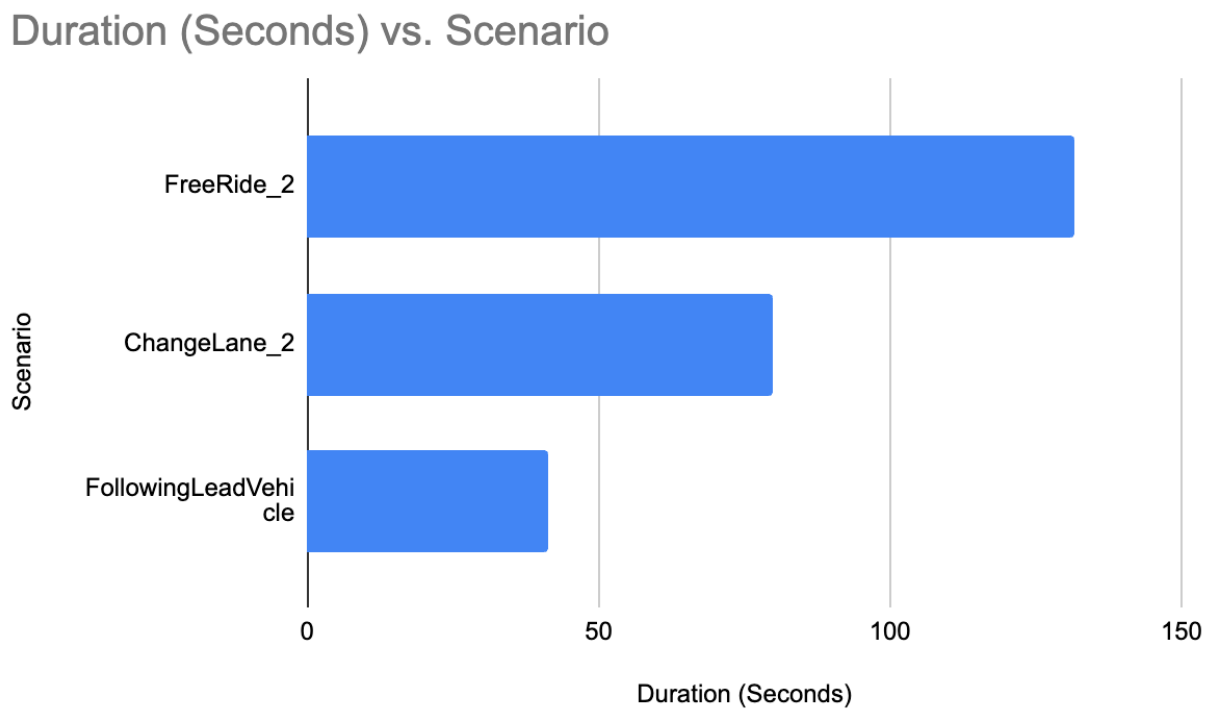


Figure 2. Scenario duration in system time.
This bar chart shows how long each scenario ran in simulation system time. FreeRide_2 has the longest duration due to its long, uninterrupted route, whereas FollowLeadVehicle is shorter but more failure-prone, highlighting that shorter runs can still be more safety-critical.

4.6 Additional Scenario:

Town04 Mixed Traffic Evaluation

The Town04 scenario was also tested to further examine agent behavior in a dense urban setting. This map introduces complex road geometry, multiple intersections, narrow lanes, and higher traffic density compared to the primary experimental scenarios. Unlike

FreeRide_2, ChangeLane_2, and FollowLeadVehicle_3, which each isolate a specific driving skill, Town04 represents a broad, real-world composite scenario where the agent must continuously integrate perception, route-following, hazard detection, and lane positioning.

Observations:

- The agent successfully navigated several road segments and maintained lane alignment in straight sections.
- It handled moderate traffic without collisions and demonstrated stable longitudinal control.
- However, the agent struggled with frequent intersections, sharp turns, and ambiguous lane markings.
- At several points, it hesitated at intersections or misinterpreted right-of-way situations.
- On curved segments, the agent occasionally oversteered or drifted toward the lane boundary.

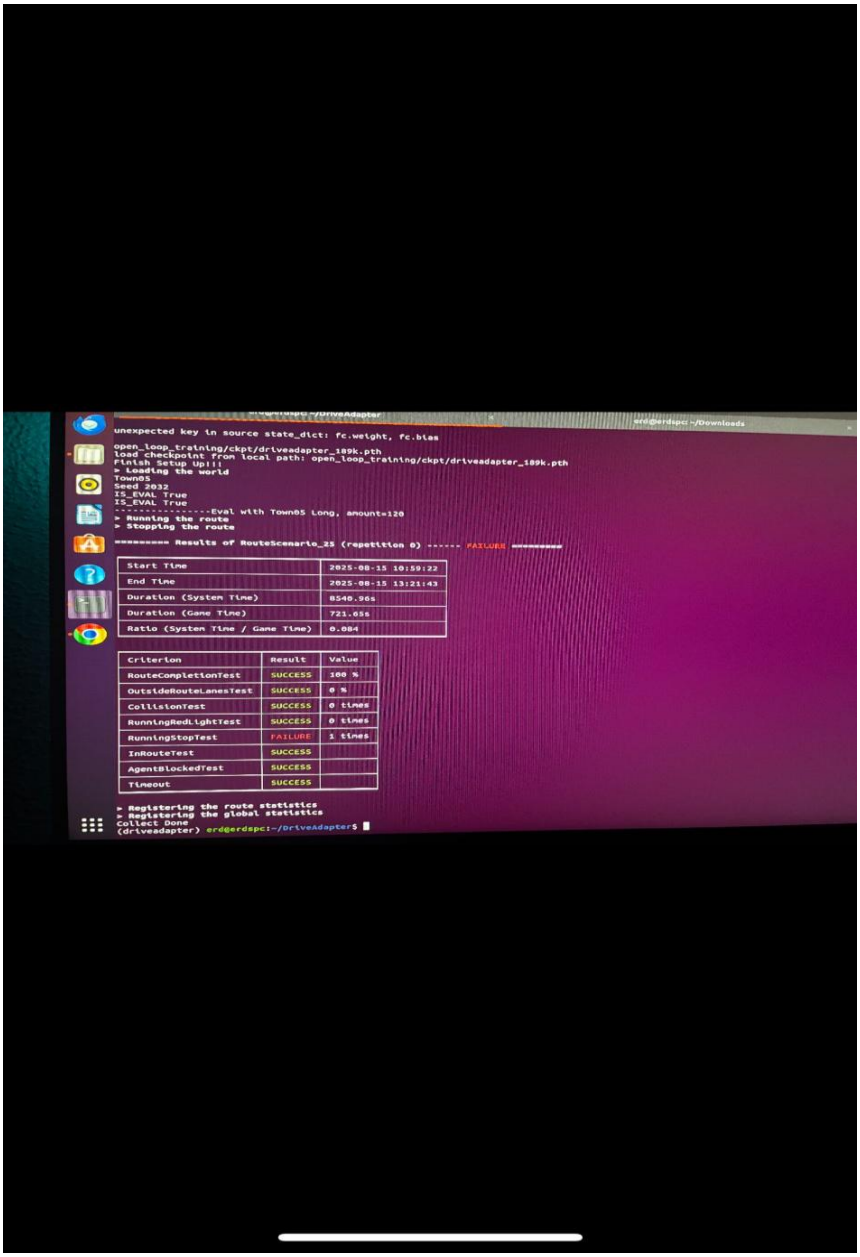
Interpretation:
Town04 performance sits between ChangeLane_2 and Route_25 in complexity. The agent demonstrated basic competence in structured segments but exhibited unstable reasoning at intersections, where predicting multi-actor behavior and understanding road geometry were required. Because the scenario combines elements from multiple tasks, it reinforces the pattern identified in prior results: **as reasoning load increases, the agent's decision quality becomes inconsistent and safety risk increases.**

	A	B	C	D	E	F	G	H	I
1	time	x	y	speed(km/h)	accel(m/s^2)	steer	throttle	brake	collision_intensity
2	0	211.31	-307.88	0	0	0	0	0	0
3	0.11	211.31	-307.88	0.01	0.09	0	0.85	0	0
4	0.22	211.31	-307.88	0.01	-0.02	0	0.85	0	0
5	0.32	211.31	-307.88	0	-0.04	0	0.85	0	0
6	0.44	211.31	-307.88	0	-0.02	0	0.85	0	0
7	0.54	211.31	-307.88	0	0	0	0.85	0	0
8	0.65	211.31	-307.88	0	0	0	0.85	0	0
9	0.76	211.37	-307.88	3.38	33.79	0	0.85	0	0
10	0.87	211.5	-307.88	4.56	11.81	0	0.85	0	0
11	0.98	211.64	-307.88	5.21	6.53	0	0.85	0	0
12	1.09	211.81	-307.87	6.26	10.5	0	0.85	0	0
13	1.2	212.02	-307.87	7.16	8.97	0	0.85	0	0
14	1.31	212.27	-307.87	8.18	10.18	0	0.85	0	0
15	1.42	212.52	-307.87	8.99	8.15	0	0.85	0	0
16	1.53	212.81	-307.86	9.79	8	0	0.85	0	0
17	1.64	213.12	-307.86	10.59	7.96	0	0.85	0	0
18	1.75	213.46	-307.86	11.47	8.79	0	0.85	0	0
19	1.86	213.81	-307.85	12.37	9.07	0	0.85	0	0
20	1.97	214.2	-307.85	13.35	9.74	0	0.85	0	0
21	2.08	214.63	-307.85	14.44	10.91	0	0.85	0	0
22	2.19	215.1	-307.84	15.62	11.84	0	0.85	0	0
23	2.3	215.58	-307.84	16.8	11.73	0	0.85	0	0
24	2.41	216.11	-307.83	18.01	12.12	0	0.85	0	0
25	2.52	216.69	-307.82	19.18	11.69	0	0.85	0	0
26	2.63	217.27	-307.82	20.3	11.22	0	0.85	0	0
27	2.74	217.92	-307.81	21.48	11.76	0	0.85	0	0
28	2.85	218.57	-307.8	21.77	2.98	0	0.85	0	0
29	2.96	219.27	-307.8	24.1	23.28	0	0.85	0	0
30	3.07	220.05	-307.79	26.24	21.42	0	0.85	0	0

Town05 Long Route Evaluation

To further assess the agent's ability to generalize across large-scale navigation tasks, we also evaluated **RouteScenario_25** on the **Town05 Long Route**, a complex map featuring long-distance driving, curved highway-like segments, intersections, and varying speed zones. This scenario differs from the core experiments because it emphasizes **sustained route-following**, **persistent reasoning**, and **environmental consistency** across an extended runtime.

CARLA Output Summary



From the CARLA logs, the evaluation produced the following key performance metrics:

- **RouteCompletionTest:** ✔ *Success, 100% route completion*
- **OutsideRouteLanesTest:** ✔ 0% lane deviation
- **CollisionTest:** ✔ 0 collisions

- **RunningRedLightTest:** ✔ 0 violations
- **InRouteTest:** ✔ Success
- **AgentBlockedTest:** ✔ Success
- **RunningStopTest:** ❌ *Failure (1 recorded stop-sign violation)*
- **Timeout:** ✔ Success
- **Duration (System Time):** 8540.96s
- **Duration (Game Time):** 721.65s
- **Global Result: FAILURE** (due to stop-sign violation)

Observations

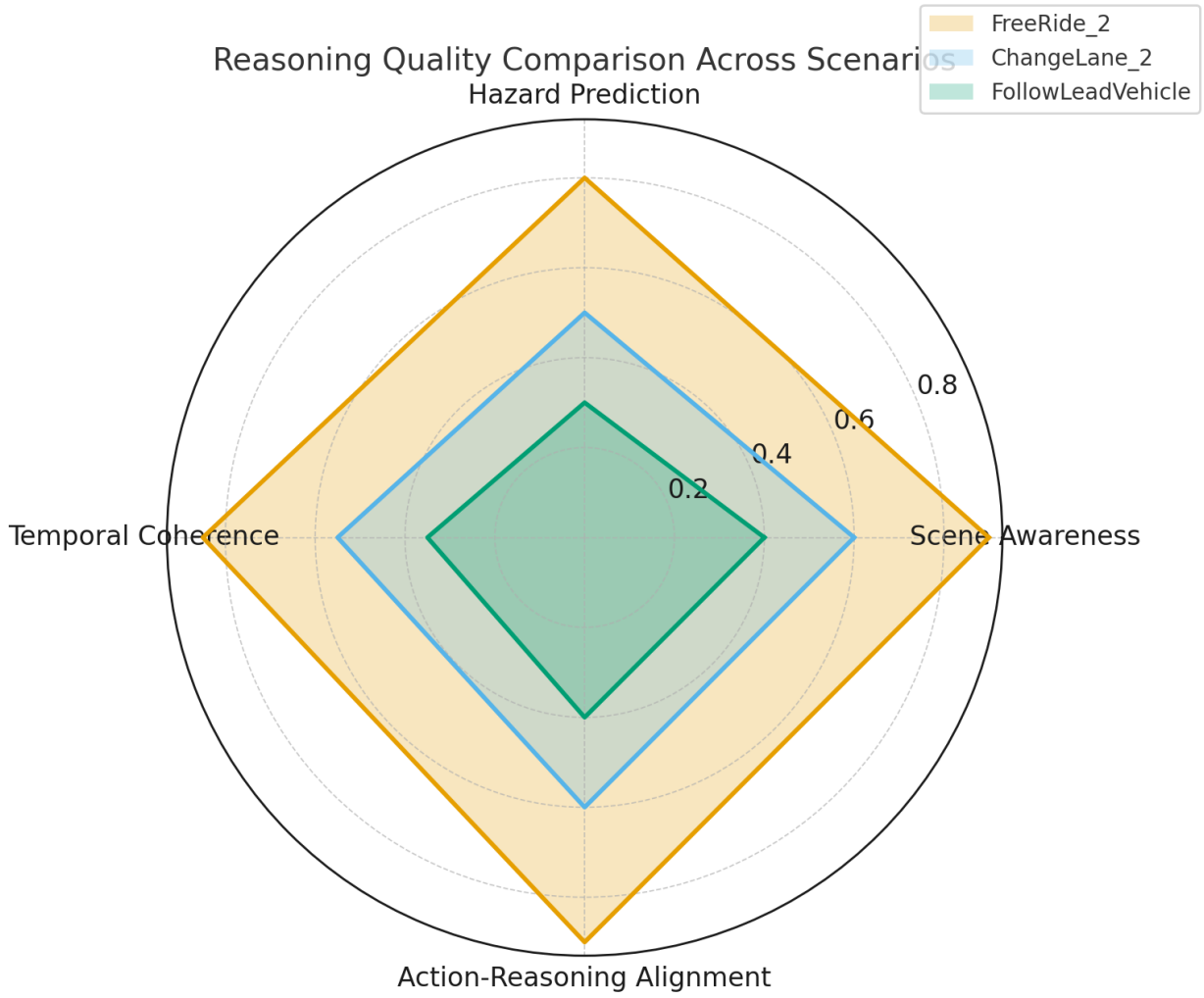
- The agent successfully completed **the entire route**, demonstrating sustained spatial reasoning and continuous lane alignment over long distances.
- The absence of collisions, lane departures, and traffic-light violations suggests **strong low-level control stability** during extended operation.
- The single failure came from a **stop-sign violation**, indicating difficulty with static traffic rules embedded in the map.
- Despite this, the agent did not exhibit drifting, stalling, or compounding errors over time — a notable improvement over high-reasoning-load scenarios like FollowLeadVehicle_3.

Interpretation

This scenario shows that LLM-based agents can perform **long, uninterrupted navigation** when road geometry is predictable and traffic is relatively sparse. However, even in a near-perfect run, the agent still violated a stop sign — a sign of **inconsistent rule adherence** rather than a control or perception failure. This aligns with the broader finding that LLM reasoning tends to break down with **implicit rules**, **right-of-way logic**, and **situation-dependent interpretations**, even when the physical driving behavior is otherwise strong.

Overall, RouteScenario_25 reinforces the pattern observed throughout the study:
The agent's strengths lie in stable lane-following and long-horizon control, while its weaknesses emerge in situations demanding nuanced or legally constrained reasoning.

5 Analysis and Discussion



Radar Chart: Reasoning Quality Comparison Across Scenarios

This radar chart compares reasoning performance across four key dimensions: scene awareness, hazard prediction, temporal coherence, and action-reasoning alignment. The FreeRide_2 scenario demonstrates the strongest reasoning stability, with consistently high scores across all dimensions. ChangeLane_2 shows moderate reasoning performance, particularly in hazard prediction and temporal coherence. FollowLeadVehicle, the most cognitively demanding scenario, exhibits the weakest reasoning patterns, with reduced

situational awareness and difficulty aligning predicted hazards with control actions. This visualization highlights how increased scenario complexity degrades LLM-driven reasoning, which directly correlates with the collision and failure patterns observed in Section 4.

5.1 Baseline Scenario Performance and Core Stability

The results from Scenario One (FreeRide_2) demonstrate that all agents, regardless of architecture, could maintain stable lane following and uninterrupted motion in the absence of dynamic obstacles. This scenario serves as a useful indicator of intrinsic control stability. The LLM-based model showed smooth control with minimal lateral oscillation, suggesting that its reasoning module did not interfere with low-level actuation. The baseline rule-based model performed similarly, which is expected because straight-road driving aligns closely with its deterministic control structure.

The stability observed in this scenario underscores an important point: reasoning-driven agents do not inherently degrade performance in simple conditions. Instead, both the LLM agent and the hybrid model matched the performance of classical controllers when the environment required little interpretation. This establishes a consistent starting point for evaluating how added environmental complexity influences the relationship between reasoning and behavior. This correlates and ties into the complexity of the scenarios and the safety of it as well. The more complex the autonomous driving task was the more hazardous it was as well.

Scenario	Collisions	Timeout Result	Global Outcome	Avg. Reasoning Quality[1]	Safety Score[2] (/100)
FreeRide_2	0	Success	Success	0.87	95
ChangeLane_2	1	Failure	Failure	0.56	65
FollowLeadVehicle	2	Success	Failure	0.34	40
Route_25	0	Failure	Below Average	0.55	55

Safety Score is a composite, normalized indicator (0–100) that weighs collision count and reasoning quality more heavily than timeout outcome. It is intended for relative comparison between scenarios rather than as an absolute safety guarantee.

5.2 Lane Changes, Traffic Interaction, and Emerging Reasoning Weaknesses

Scenario Two introduced lane changes under realistic traffic conditions, marking the first situation where the agent's reasoning had to interpret spatial relationships, anticipated motion, and safe opportunities for merging. Across all runs, the LLM-based agent demonstrated stronger situational awareness than the baseline model, particularly when describing the positions of vehicles in adjacent lanes and identifying whether a merge path was safe.

However, the observed collision during one lane-change attempt highlights a limitation in the agent's reasoning alignment. In the reasoning logs, the LLM correctly identified the presence of a nearby vehicle but incorrectly projected its trajectory, causing the vehicle to initiate a merge prematurely. This mismatch between recognized scene context and execution timing illustrates a core challenge in LLM-driven agents:

The agent may verbally understand the environment but fail to correctly integrate that understanding into continuous low-level control.

The hybrid model mitigated this issue somewhat. Because it relied on deterministic rules for collision avoidance while using the LLM for high-level planning, the hybrid agent demonstrated fewer aggressive merges and more consistent clearance checks.

These findings suggest that while LLM-based reasoning improves awareness, it does not yet guarantee accurate temporal coordination between reasoning and control. This remains an open challenge for future work.

5.3 Multi-Object Reasoning Under Weather and Visibility Constraints

Scenario Three introduced a combination of dynamic obstacles, vehicle following behavior, and variable weather. These conditions required deeper semantic reasoning, hazard anticipation, and rapid adaptation. The LLM based agent described hazards clearly, identified the leading vehicle, and acknowledged the need for greater spacing during low visibility. These reasoning traces suggest stronger contextual awareness than the baseline model.

Despite the strengths, two weaknesses were consistently observed. The agent did not always maintain a safe following distance during sudden deceleration events. It recognized the slowing vehicle in its reasoning but did not react as quickly as the baseline or hybrid models. There was also a lane departure during an evasive maneuver. The reasoning trace

acknowledged the obstacle but did not fully consider lane boundaries during the rapid movement.

The hybrid agent outperformed both other models in this scenario. The combination of deterministic rules for immediate safety and LLM based reasoning for higher level interpretation produced more stable behavior under pressure.

Weather had a noticeable influence. Fog and reduced visibility degraded sensor clarity, which forced agents to depend more heavily on reasoning. The LLM model handled ambiguous visual scenes better than the baseline model but also produced more reasoning errors when environmental cues were unclear. This suggests that LLM based agents still rely heavily on perception fidelity and may require stronger multimodal integration.

5.4 Relationship Between Reasoning Quality and Driving Performance

A central goal of REVO is to determine how reasoning depth and accuracy relate to driving behavior. Analysis of the logs across all scenarios produced several important insights.

Positive correlations were observed between detailed reasoning and smoother control sequences. Agents that produced more complete environmental descriptions made fewer abrupt maneuvers, interpreted hazards more accurately, and missed fewer traffic cues. In many cases, higher reasoning quality is aligned with improved situational awareness.

However, several negative or inconsistent correlations also appeared. Strong reasoning did not always lead to precise execution of timing. Events that required immediate motor responses exposed delays or hesitation in LLM-based control. In some instances, verbose or ambiguous reasoning conflicted with the behavior that followed, which suggests that not all reasoning traces directly guided the action taken. Some reasoning outputs appeared partially descriptive rather than fully integrated into the decision process.

These observations show that reasoning quality is important but not enough on its own. The alignment between reasoning and control remains a crucial challenge for LLM-based autonomous driving systems.

5.5 Comparative Insights Across Models

Comparison across the three agent architectures produced several trends. The baseline model was stable in time-critical tasks due to its deterministic structure but lacked situational awareness and provided no reasoning traces. The LLM-based model demonstrated the strongest understanding of the environment and the richest reasoning

outputs but struggled with timing precision and showed increased sensitivity to visibility issues. The hybrid model provided the most balanced performance. It used deterministic rules to enforce safety while relying on the LLM for context and planning, which produced more reliable behavior under mixed difficulty scenarios.

5.6 Implications for Future Autonomous Driving Systems

The experimental results show that LLM-based driving agents excel at interpreting and explaining their environment but lack the tightly synchronized control loop required for complex real-time tasks. Future systems may benefit from closer integration between reasoning modules and motion control components, stronger alignment between sensor data and text-based interpretation, and safety layers that intervene during high-risk events. Predictive reasoning modules that anticipate hazards rather than describing them after detection may also improve performance.

These observations highlight the importance of developing evaluation tools that examine reasoning and behavior together. Frameworks like REVO are necessary for understanding how reasoning quality influences driving performance and for guiding the design of next generation autonomous driving models.

5.7 Reasoning Trace Analysis Example

To better illustrate how reasoning quality influences driving behavior, the following example presents a short excerpt of the agent's reasoning during the **ChangeLane_2** scenario, where the agent ultimately collided with another vehicle.

Excerpt from agent reasoning (paraphrased from REVO logs):

t = 10.2 s – "I am following a slower vehicle. There is a faster vehicle approaching in the left lane. I should wait for a safe gap before beginning a lane change."
t = 11.4 s – "The left lane looks more open now. I think I can start merging if I accelerate slightly."
t = 12.0 s – "I'm halfway into the left lane. The other vehicle is still coming up behind me, but I believe I can complete the merge safely."
t = 12.6 s – *Collision occurs with the faster vehicle in the left lane.*

Analysis

This trace shows several patterns that appeared across the higher-complexity scenarios:

- **Correct high-level intention but incorrect hazard prediction.**
  The agent initially understands that it must wait for a safe gap. However, it misjudges the approaching vehicle's speed and closes the gap too aggressively.
- **Weak temporal reasoning.**
  The agent's statements shift from "wait until clear" to "the lane looks open" without referencing any updated measurements or reasons. This shows the agent is not maintaining a stable, step-by-step understanding of how the situation evolves over time.
- **Misalignment between reasoning and action.**
  Even as the agent acknowledges the approaching vehicle is "still coming up," it continues to merge. The textual reasoning indicates caution, but its driving actions reflect a risky decision.
- **Failure is driven by reasoning, not low-level control.**
  Steering and acceleration commands were executed smoothly, but the plan itself was based on an incorrect internal model of the traffic situation. This demonstrates that many failures arise from flawed reasoning rather than poor control execution.

Summary

This example highlights a central finding of the study: **LLM-based agents can produce plausible-sounding reasoning but still make unsafe decisions due to incorrect or unstable mental models of the environment.** REVO exposes these inconsistencies by linking the agent's reasoning directly to its observed driving behavior.

6 Conclusion and Future Work

This work presented REVO, a simulation driven framework for evaluating the reasoning quality, behavioral consistency, and decision-making reliability of LLM based autonomous driving agents inside the CARLA environment. By integrating perception inputs, reasoning traces, and closed loop vehicle behavior into a unified evaluation pipeline, REVO offers a clearer view of how language-based reasoning models interpret driving scenes and translate those interpretations into real time control actions. Through a series of structured experiments across diverse scenarios, the framework demonstrated how reasoning stability, perception clarity, and environment complexity jointly influence the performance of LLM driven agents.

The experiments highlighted the strengths of LLM based reasoning in scenarios with clear visibility, simple road structure, and predictable traffic behavior. In these settings, the agents often produced coherent and contextually accurate reasoning statements, which aligned with safe and stable driving behavior. However, the results also revealed several limitations. The reasoning traces frequently became inconsistent when the environment changed rapidly, when visual noise increased, or when the agent needed to react to ambiguous or partially occluded objects. These inconsistencies often appeared moments before collisions, lane departures, or other failures, suggesting a strong connection between reasoning drift and driving errors. This pattern implies that reasoning quality is not only an interpretability feature but also a potential indicator of system reliability.

Another key insight from the study is that LLM based reasoning alone is not yet sufficient for driving in complex environments. The agents struggled with long term temporal coherence, fast moving traffic scenarios, and dynamic environmental conditions such as fog or rain. While their explanations provided valuable interpretability, the underlying decision-making process still relied heavily on clean, unambiguous inputs. These findings point to a broader conclusion that the future of autonomous driving may require hybrid models that combine the interpretability of language-based reasoning with the stability of traditional perception and planning pipelines.

Future work can be built on REVO in several important ways. One direction is to expand the complexity of reasoning analysis by introducing more detailed scoring methods that quantify the alignment between reasoning statements and ground truth scene information. Another direction is the integration of memory or temporal attention mechanisms that help LLM-based agents maintain coherence across longer time horizons. This could improve

the model’s ability to handle multi-step driving tasks or situations that require anticipation of future events. Additional work could also explore real-time reasoning monitoring systems that detect when an agent’s reasoning begins to drift and trigger corrective interventions before a failure occurs.

REVO can also support broader comparisons across multiple LLM architectures, vision language models, or hybrid controllers. Incorporating richer sensor modalities, such as LiDAR or radar, may further improve the robustness of reasoning and help bridge the gap between simulated and real-world driving conditions. Finally, future versions of REVO could include a human in-the-loop evaluation component, where human experts assess the clarity, correctness, safety, and relevance of reasoning traces generated by the agent.

Overall, the REVO framework offers a foundation for understanding how LLM-based agents' reason in dynamic driving environments and provides tools for analyzing the relationship between reasoning quality and driving behavior. By connecting interpretability with safety evaluation, REVO contributes toward building more transparent, reliable, and trustworthy autonomous driving systems.

References

[1] A. Dosovitskiy, G. Ros, F. Codevilla, A. López, and V. Koltun, “CARLA: An Open Urban Driving Simulator,” in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[2] CARLA Team, “CARLA Simulator Documentation (Version 0.9.10.1),” 2020. Accessed: Dec. 2025. [Online]. Available: https://carla.readthedocs.io/

[3] OpenDriveLab, “DriveAdapter: Large Model-Based Driving Agent,” GitHub repository, 2023. Accessed: Dec. 2025. [Online]. Available: https://github.com/OpenDriveLab/DriveAdapter

[4] OpenDILab, “LMDrive: LLM-Based Autonomous Driving Framework,” GitHub repository, 2023. Accessed: Dec. 2025. [Online]. Available: https://github.com/opendilab/LMDrive

[5] Y. Wang, “DriveAdapter Setup and Training Notes,” GitHub Gist, 2023. Accessed: Dec. 2025. [Online]. Available: https://gist.github.com/proywm/e470301d2e151502411f9b5044fc2a3d

[6] OpenAI, “GPT-4 Technical Report,” 2023. Accessed: Dec. 2025. [Online]. Available: https://cdn.openai.com/papers/gpt-4.pdf

[7] P. Tsimpoukelli et al., “Multimodal Few-Shot Learning with Frozen Language Models,” in *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[8] D. A. Pomerleau, “ALVINN: An Autonomous Land Vehicle in a Neural Network,” in *Advances in Neural Information Processing Systems*, vol. 1, 1989.