LVC: Augmenting Autonomous Driving via Language-based V2V Cooperation

Anonymous ACL submission

Abstract

Although Vehicle-to-Vehicle (V2V) communication is a promising solution for cooperative driving, existing approaches are largely limited to basic scenarios such as intersections and straight roads due to the lack of diverse benchmarks, failing to fully exploit its potential. In this work, we propose Comm2Interact, a novel and comprehensive V2V benchmark built on the CARLA simulator. The benchmark is specially curated to challenge agents with complex traffic scenarios, such as view occlusions, roundabouts, and tactical overtaking and requires diverse cooperative capabilities, ranging from perception sharing and right-of-way negotiation to maneuver coordination. To effectively navigate these intricate scenarios, we propose LVC, an LLM-based cooperative driving framework that transforms high-level intention into precise control commands. LVC leverages a set of interaction primitives to decompose complex scenarios into atomic, manageable sub-tasks, and employs a Memory Module to handle long-tail edge cases via reflection, ensuring safe operations. Extensive experiments on both the proposed challenging benchmark and existing V2V benchmarks demonstrate that LVC performs favorably against state-of-the-art methods in terms of both safety and success rates, showing its effectiveness in handling diverse traffic interactions.

1 Introduction

While autonomous driving has achieved remarkable progress, safety remains a critical bottleneck in high-interaction scenarios like unprotected intersections and occluded environments (Jiang et al., 2023; Hu et al., 2023; Shao et al., 2024; Renz et al., 2025). Accidents in these contexts often arise not from perception errors, but from failures in intent inference—simply detecting an oncoming vehicle does not reveal its strategic goals, such as the intent to yield or accelerate. This limitation highlights the urgent need for effective V2V cooperation, yet existing paradigms struggle to balance performance with practical deployment constraints.

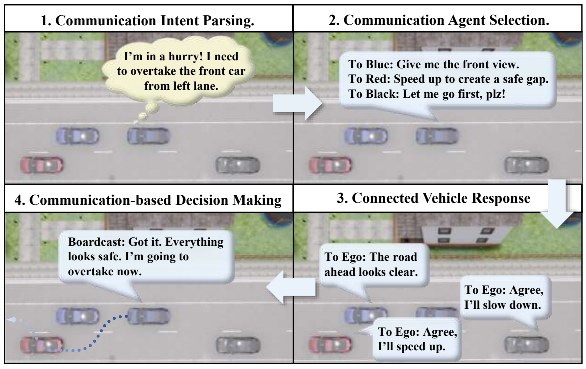


Figure 1: Completing complex driving maneuvers by breaking interactive scenarios down into multiple simple one-on-one tasks.

Existing research on cooperative autonomous driving focuses on two directions: cooperative perception and centralized planning. Early V2X approaches focused primarily on sharing raw sensor data or intermediate feature maps to expand the ego vehicle's field of view (Hu et al., 2022; Xu et al., 2022a,b, 2023). While these methods effectively mitigate the risk of blind spots, they suffer from excessive communication bandwidth and computational overheads. More importantly, they address only the perceptual layer; simply seeing another vehicle does not inherently resolve the conflict of right-of-way, nor does it facilitate the explicit negotiation required for complex maneuvers. On the other hand, centralized planning approaches coordinate multi-vehicle planning by formulating scenario-specific optimization strategies (Huang et al., 2023b,a; Liu et al., 2024a) or learning policies based on global reward functions (Zhan et al., 2019; Zhao et al., 2023; Zheng and Gu, 2024; Liu et al., 2024b). However, these methods typically depend on robust infrastructure support (e.g., RSUs or central servers) and suffer from limited scenario generalizability (Liu et al., 2025a).

The recent emergence of Large Language Models (LLMs) and Vision-Language-Action (VLA) models has introduced a promising paradigm: utilizing natural language as a semantic interface for vehicle communication. Unlike rigid data protocols that require a pre-defined schema for every possible event, natural language offers a universal interface to describe arbitrary driving contexts and unexpected intent. Furthermore, by leveraging the rich driving knowledge and reasoning capabilities inherent in LLMs, agents can emulate human-like negotiation logic to resolve conflicts intuitively. However, current language-based V2V methods (Gao et al., 2025; Hu et al., 2024; Cui et al., 2025; Fang et al., 2025) and benchmark (Liu et al., 2025a) focus mainly on simple tasks and scenarios such as right-of-way negotiation at intersections and road merges. They overlook the potential of communication in more practical and complex interaction scenarios, such as coordinated overtaking, which enables vehicles to proactively collaborate rather than merely avoid collisions.

To address the intricate challenges inherent in realistic driving scenarios, we propose Comm2Interact, a comprehensive cooperative driving benchmark built on CARLA, designed to evaluate agent communication in complex, high-stakes environments. By covering diverse road topologies and environmental conditions, Comm2Interact requires agents to engage in perception sharing, right-of-way negotiation, and maneuver coordination. The benchmark is structured into four distinct taxonomies: Intersection Negotiation, View Occlusion, Collaborative Overtaking, and Roundabout Navigation, providing a robust platform to validate the efficacy of cooperative driving in complex practical traffic scenarios.

To tackle the challenges within Comm2Interact, we accompany the benchmark with LVC, an LLM-based V2V Cooperative framework. LVC departs from monolithic policy learning by establishing a set of standardized driving primitives, which allows agents to decompose complex scenarios into manageable, atomic communication tasks. This primitive-based structure not only facilitates precise, target-oriented negotiation in high-stakes environments but also ensures extensibility: new scenarios in the benchmark can be readily addressed by defining corresponding primitives without retraining the entire system. Furthermore, to address long-tail edge cases beyond the scope of static primitives, we introduce a reflection-based Memory Module. This component allows the system to accumulate driving experience and retrieve successful strategies for complex, open-world interactions.

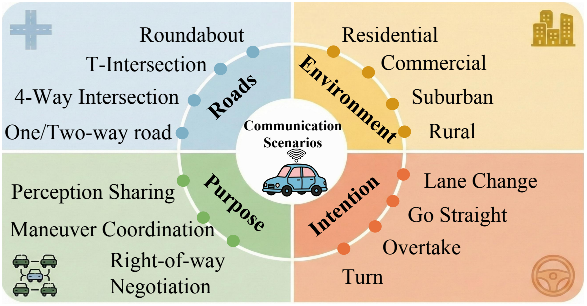


Figure 2: Comm2Interact covers various road environments, multiple driving intentions, and various communication tasks.

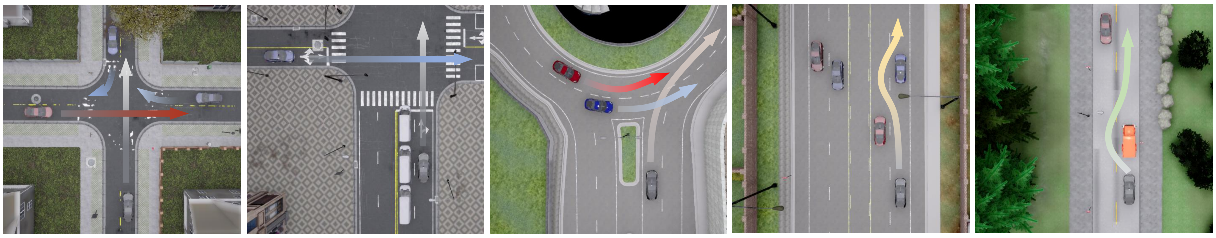
The main contributions of this work are:

- We propose Comm2Interact, a comprehensive benchmark featuring diverse road topologies and multifaceted interaction goals.
- We propose LVC, a baseline framework that standardized primitives for precise negotiation and a Memory Module to handle long-tail edge cases via reflection.
- Extensive experiments on both the proposed and existing benchmarks validate the significant advantages of cooperative driving in complex traffic scenarios, while confirming the superior effectiveness of LVC.

2 Related Work

2.1 V2V Autonomous Driving environment

Simulation platforms are critical for validating multi-agent interaction. Lightweight platforms like SMARTS (Zhou et al., 2020) and Highway-env (Leurent et al., 2018) facilitate multi-agent reinforcement learning but suffer from simplified physics and low-fidelity sensor simulation. Conversely, CARLA (Dosovitskiy et al., 2017) provides the high-fidelity dynamics essential for rigorous testing. To address collaborative driving needs, V2Xverse (Liu et al., 2025b) leverages such fidelity to offer a comprehensive platform supporting closed-loop, multi-agent evaluation beyond mere perception tasks. Built on V2Xverse, the InterDrive (Liu et al., 2025a) benchmark specifically



(a) Intersection Negotiation (b) View Occlusion (c) Roundabout Negotiation (d) Normal Overtaking (e) Opposing Overtaking

Figure 3: All scene categories included in Comm2Interact.

evaluates interaction capabilities, focusing on conflict resolution at intersection crossing, lane changing and lane merging. However, it lacks the environmental complexity required to test deep coordination under uncertainty. To bridge this gap, our Comm2Interact introduces high-stakes scenarios specifically designed to validate complex cooperative driving and linguistic negotiation.

2.2 Language-based V2V Autonomous Driving

Recently, several approaches have explored the use of language-based communication in autonomous driving. LangCoop (Gao et al., 2025) encapsulates perception and intent into text, effectively minimizing the bandwidth overhead associated with image feature transmission. CoLMDriver (Liu et al., 2025a) features a parallel driving pipeline anchored by an LLM-based negotiation module under an actor-critic paradigm. Through multi-round dialogue, it continuously refines cooperation policies based on feedback, allowing conflicting agents to reach a consensus on right-of-way. TalkingVehicles (Cui et al., 2025) introduces a post-hoc reflection mechanism to derive experience from past failures. However, these methods are predominantly confined to limited scenarios, failing to fully exploit the potential of communication.

3 The Comm2Interact Benchmark

We implement Comm2Interact in the CARLA simulator (version 0.9.15) to model cooperative driving under partial observability. As shown in Fig. 3, to capture diverse behaviors, we structure 80 scenarios into four classes based on topological complexity: **Intersection Negotiation**, **View Occlusion**, and **Roundabout Navigation** (16 cases each), and **Collaborative Overtaking** (32 cases, split between normal and opposing). This categorization ensures a comprehensive evaluation of communication strategies across different road structures. The specific generation protocols are detailed below.

3.1 Intersection Negotiation

Scenario Description: The scenarios are established at unsignalized intersections and feature multi-directional traffic with ambiguous right-of-way. The primary challenge arises from the high-density dynamic environment, where aggressive peer behaviors create potential deadlocks or imminent collision risks that cannot be safely resolved through passive observation.

Scenario Generation: To guarantee high-value interaction instances, we ensure that the ego vehicle's trajectory intersects with at least one peer vehicle. We implement an Adaptive Collision Mechanism: initially, conflicting peer vehicles dynamically adjust their velocities based on the ego vehicle's real-time state to maintain a deterministic collision course. This aggressive tracking behavior is designed to persist until the first V2V communication frame is successfully established. To ensure Behavioral Diversity, specific peer vehicles are programmed as Adversarial Agents; they persistently return a "Refuse" signal to all negotiation requests, maintaining their velocity regardless of risk.

3.2 View Occlusion

Scenario Description: The scenario is positioned at intersections characterized by severe visual occlusions. A line of stationary vehicles (e.g., waiting for traffic signals) on the adjacent lane creates a visual barrier, strictly blocking the ego vehicle's line-of-sight towards lateral traffic. The fundamental challenge is the physical limitation of reaction capability relative to occlusion. While the latent risk eventually enters the onboard sensor's field of view, the ego vehicle has already breached the safe braking distance, rendering a collision physically unavoidable without early warning. Therefore, the ego vehicle must communicate with other agents to obtain vision of the occluded area.

Scenario Generation: We utilize stationary traffic on the ego vehicle's left as an occlusion mask to conceal a Peer vehicle. This hidden agent employs the Adaptive Collision Mechanism to dynamically maintain a collision course with the approaching ego vehicle. Notably, if the ego vehicle executes a preventive stop, the adaptive mechanism acts to reduce the peer's velocity to its lower bound. Consequently, the peer continues to traverse the intersection at minimum speed, effectively preventing simulation deadlocks.

3.3 Roundabout Negotiation

Scenario Description: This scenario takes place in multi-lane roundabouts. We focus specifically on the Inner Ring Entry task. This is more complex than a simple merge because it requires a continuous lane change: the ego vehicle must first cross the outer ring and then immediately enter the inner ring. Therefore, the ego vehicle has to interact with vehicles on both the outer and inner lanes at the same time, dealing with two conflicts instead of one. The main challenge is Multi-Lane Coordination. The ego vehicle encounters a dual conflict scenario: securing right-of-way from the outer-ring vehicle is not enough if the inner-ring vehicle blocks the target position. If the ego vehicle only observes passively, it might get stuck between lanes or cause a collision. This situation requires proactive solutions to ensure both lanes are clear.
Scenario Generation: We place vehicles on multiple lanes of the roundabout. When the ego vehicle attempts to enter, the vehicles inside the roundabout use the Adaptive Collision Mechanism to ensure a conflict occurs. In addition, we include simpler cases, such as negotiating with entering vehicles while driving inside the ring, or negotiating with outer-ring vehicles while exiting.

3.4 Collaborative Overtaking

Scenario Description: This scenario covers two distinct operational modes Normal Overtaking and Opposing Overtaking, triggered when the ego vehicle is blocked by a slow-moving lead vehicle. The system must decide whether to use the adjacent lane or borrow the oncoming lane based on road structure. Crucially, this scenario introduces specific adversarial constraints: (a) In Normal Overtaking, the target lane presents a dual challenge: parallel vehicles require **Maneuver Coordination** to open a sufficient gap, while rear approaching traffic necessitates negotiation to secure a safe merge window. (b) In Opposing Overtaking, the large dimensions of the lead vehicle create severe visual occlusion, rendering the oncoming lane unobservable to the ego vehicle's local sensors alone.
Scenario Generation: To induce a valid overtaking incentive, we initialize a lead vehicle traveling significantly below ego speed. (a) For Normal Overtaking, we populate the adjacent lane with adversarial traffic, positioning Peer vehicles either moving parallel to the lead vehicle or approaching rapidly from the rear to intentionally close the merge window. (b) For Opposing Overtaking, we configure a "Blind Overtake" setup where the lead vehicle strictly blocks the ego vehicle's forward sensors, while we generate random oncoming traffic in the opposing lane that remains undetectable from the ego vehicle's perspective.

4 Method

4.1 Problem Formulation

We consider a multi-agent autonomous driving environment with a set of vehicles $\mathcal{V} = \{v_1, \ldots, v_N\}$. For each agent v_i, let $o_i \in \mathcal{O}$ denote the local observation and $g_i \in \mathcal{G}$ denote the high-level navigation instruction.
Baseline VLA Policy. Each agent is controlled by a fundamental Vision-Language-Action (VLA) policy, denoted as π_{vla}. Under standard conditions, the agent generates a control action a_i^{base} relying solely on its local sensory observation o_i and navigational instruction g_i:

$$a_i^{\text{base}} = \pi_{\text{vla}}(o_i, g_i). \tag{1}$$

However, π_{vla} suffers from partial observability regarding the latent intentions and future strategies of surrounding vehicles, often leading to conservative or conflicting behaviors in complex scenarios.
Communication-Augmented Formulation. To resolve the ambiguity of partial observability, a communication mechanism is integrated into the agent policy. The final collaborative action a_i is formulated as:

$$a_i = \pi_{\text{collab}}\Big(o_i, g_i, \mathcal{M}(o_i, g_i, \mathcal{N}_i)\Big), \tag{2}$$

where $\mathcal{M}(\cdot)$ represents the communication mechanism that generates negotiation contexts. This mechanism operates on the neighbor state set $\mathcal{N}_i$, defined as:

$$\mathcal{N}_i = \{(\text{ID}_j, p_j, v_j, \psi_j) \mid j \in V_{\text{comm}}\}. \tag{3}$$

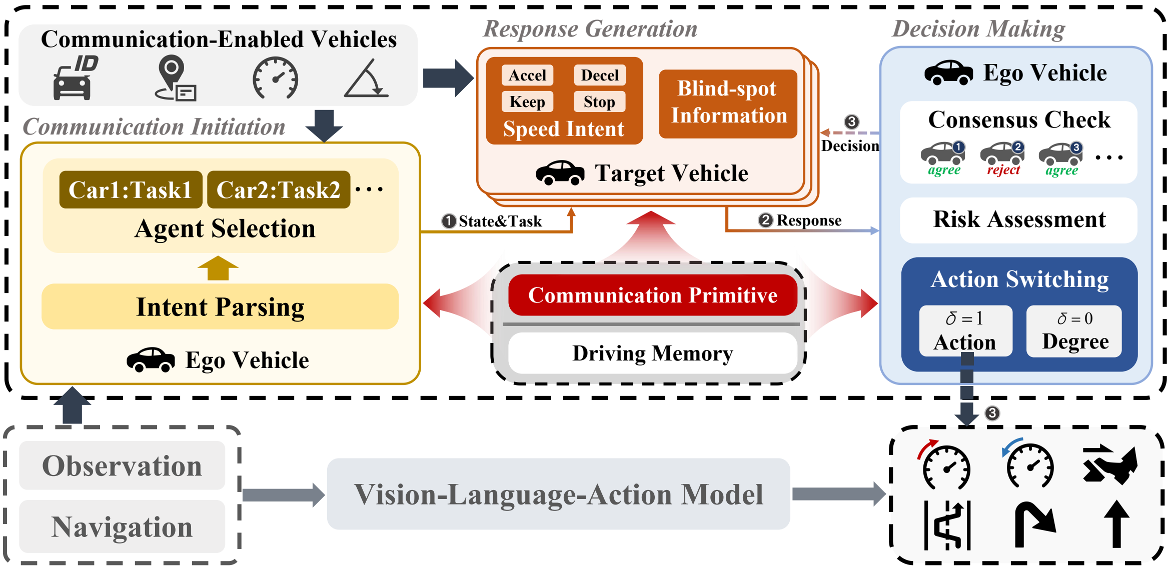


Figure 4: Overview of the LVC framework. The system executes a peer-to-peer three-way handshake to resolve conflicts: 1. The ego vehicle dispatches task requests; 2. target vehicles return responses; 3. the ego vehicle verifies consensus to formulate and broadcast the final decision.

Here, p_j, v_j, and ψ_j denote the position (GPS), velocity, and heading (IMU) of the neighbor j, respectively. Since these telemetry data are readily accessible via V2V protocols, they serve as a lightweight prior for target identification. V_{comm} denotes the set of communication-enabled vehicles within a fixed radius R of the ego vehicle that are directly detectable by the on-board radar.

4.2 Language-based V2V Cooperation

Our framework operates via a three-stage pipeline: *Communication Initiation*, *Connected Vehicle Response*, and *Decision Making*. Each stage is grounded in a corresponding primitive set ($\mathcal{L}_{\text{init}}$, $\mathcal{L}_{\text{resp}}$, $\mathcal{L}_{\text{dec}}$) to standardize collaboration. Additionally, a memory module is used to integrate past driving experiences to handle complex scenarios beyond the scope of these primitives.

4.2.1 Communication Initiation

The ego vehicle first analyzes the driving context to determine if communication is necessary and, if so, identifies the target vehicles and communication tasks.

Communication Intent Parsing. Given LLMs' limited spatial reasoning with raw coordinates, we utilize high-level navigation instructions (e.g., lane changes) as semantic inputs. However, relying solely on these static commands is insufficient, as they often fail to capture immediate tactical maneuvers or implicit conflicts inherent in converging road structures. Therefore, we derive the ego vehicle's intent by synthesizing navigation instructions with current environmental context, ensuring communication is triggered by actual interaction needs rather than mere topological changes.

The module derives the driving intent I_{ego} by synthesizing the global navigation instruction g_i with the VLM-generated scene description l_i and the set of communicating vehicles $\mathcal{N}_i$:

$$I_{\text{ego}} = \text{LLM}_{\text{intent}}(l_i, g_i, \mathcal{N}_i) \tag{4}$$

The output is categorized into four types: *Lane Keeping*, *Intersection Turning*, *Passive Lane Change* (navigation-based), and *Active Lane Change* (overtaking).

Communication Agent Selection. Utilizing the primitive corresponding to I_{ego}, we employ the LLM to analyze the scene and surrounding traffic, thereby precisely selecting the necessary communication targets and assigning specific tasks:

$$(V_{\text{target}}, T_{\text{comm}}) = \text{LLM}_{\text{target}}(l_i, g_i, \mathcal{N}_i, \mathcal{L}_{\text{init}}(I_{\text{ego}})) \tag{5}$$

where $V_{\text{target}} \subset \mathcal{N}_i$, T_{comm} are categorized into five types: *Negotiation for intersection* or lane changing, *Perception Sharing* for intersection or lane changing and *Cooperative Overtaking*. Subsequently, the ego vehicle sends individual requests to each identified target vehicle.

4.2.2 Connected Vehicle Response

Upon receiving the request, the target vehicle $v_j \in V_{\text{target}}$ processes it by referencing the corresponding primitive. For instance, in negotiation tasks, the agent evaluates whether a conflict of intent exists and assesses if yielding pose any safety risks. All responses contain shared perception information to help expand field of view. For negotiation and cooperation tasks, the target vehicle further provides a strategic decision (Accept/Reject) and a speed intent (Accelerate, Decelerate, Maintain, or Stop).

$$R_j = \text{LLM}_{\text{reply}}(l_j, g_j, \mathcal{N}_j, I_{\text{ego}}, \mathcal{L}_{\text{resp}}(T_{\text{comm}})) \tag{6}$$

4.2.3 Communication-based Decision Making

Based on collected $\{R_j\}$, the ego vehicle proceeds with the maneuver only if all targets accept the request and the shared blind-spot data reveals no threats. This final decision logic relies on the corresponding primitive:

$$(\delta, a_{\text{comm}}) = \text{LLM}_{\text{dec}}(l_i, g_i, \mathcal{N}_i, \{R_j\}, \mathcal{L}_{\text{dec}}(I_{\text{ego}})) \tag{7}$$

Here, $\delta = 1$ signifies that the collaborative maneuver is approved. The final control action a_i is determined by the following switching logic:

$$a_i = \begin{cases} a_{\text{comm}}, & \text{if } \delta = 1, \\ \pi_{\text{vla}}(o_i, g_i), & \text{otherwise.} \end{cases} \tag{8}$$

If the negotiation succeeds ($\delta = 1$), the agent executes the communication-derived action a_{comm} to perform the complex maneuver. Conversely, if the request is rejected or no communication is required, the system gracefully degrades to the standard VLA policy π_{vla}, ensuring robust basic driving behavior.

4.3 Reflection-based Memory Module

While the primitive set facilitates deterministic execution, a static set of rules cannot exhaustively cover the long-tail scenarios inherent in open-world driving. To address these edge cases, we propose a Memory Module that acts as a dynamic knowledge supplement to the fixed primitives. Drawing inspiration from the reflection mechanisms in DiLu (Wen et al., 2023), this module summarize historical driving experiences and explicitly learn from past failures. We provide 8 additional examples for each scenario to construct the memory bank, ensuring no ground truth leakage.

First, we construct a Negative Sample Set derived from training cases which include two types of failures: **Collisions:** Explicit safety failures where the vehicle hits an obstacle and **Abnormal Trajectories:** Planning failures caused by inconsistent decision-making. Next, an LLM analyzes the recent timesteps preceding a failure to pinpoint the cause. For collisions, it determines whether the error stemmed from a specific communication step or the lack of a request. For Abnormal Trajectories, the LLM analyzes decision trajectories to pinpoint where the inconsistency initiated and attributes the error to a specific stage within our pipeline. After human verification, corrected actions are stored in a memory bank. During inference, relevant memories are retrieved to assist the agent in handling complex scenarios beyond standard primitive rules.

5 Experiments

Simulation Environment.

We evaluate our framework on the InterDrive (Liu et al., 2025a) and the proposed Comm2Interact, in the CARLA simulator with a simulation frequency of 20 Hz. InterDrive is built upon the V2Xverse platform and features 46 scenarios specifically designed for right-of-way negotiation. It encompasses three core categories: Intersection Crossing with diverse entry/exit combinations, Lane Merging (spanning T-junctions and ramps), and Lane Changing involving parallel-to-intersecting maneuvers.

Metrics. We evaluate our method using standard closed-loop metrics: Driving Score (DS), Route Completion (RC), Infraction Score (IS), and Success Rate (SR). RC measures the percentage of the route distance completed. IS acts as a penalty coefficient (ranging from 0 to 1) based on collisions and traffic violations. DS is the primary metric, calculated as the product of RC and IS, balancing progress with safety. Finally, SR reports the percentage of episodes where the agent reaches the destination without any critical failures.

Implementation Details. We adopt SimLingo (Renz et al., 2025), a SOTA VLA model, as the backbone driving policy. SimLingo is selected for its robust capability in interpreting textual instructions and translating high-level actions into precise low-level control commands. To empower the LVC framework with advanced reasoning, we utilize a dual-LLM architecture: GPT-4o (Achiam et al., 2023) serves as the Scene Descriptor, extract-

Method	Intersection Negotiation				Roundabout Navigation				View Occlusion				Total			
	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑
LMDrive	70.57	70.57	**1.00**	0.50	66.50	66.50	**1.00**	0.56	78.44	78.44	**1.00**	0.50	71.84	71.84	**1.00**	0.52
SimLingo	88.30	98.64	0.88	0.81	88.00	**100.00**	0.88	0.68	49.33	90.12	0.51	0.00	75.21	96.25	0.76	0.50
Codriving	91.03	92.94	0.95	0.88	78.29	96.16	0.79	0.63	72.96	87.25	**1.00**	0.50	80.76	92.12	0.91	0.67
Colmdriver	55.26	55.26	**1.00**	0.31	70.91	**100.00**	0.71	0.56	25.06	25.06	**1.00**	0.00	50.41	60.11	0.90	0.29
SimLingo+Comm	69.20	97.90	0.72	0.25	82.50	**100.00**	0.83	0.56	62.77	**100.00**	0.63	0.38	71.49	99.30	0.73	0.40
LVC w/o memory	**100.00**	**100.00**	**1.00**	**1.00**	88.04	99.36	0.89	0.75	**100.00**	**100.00**	**1.00**	**1.00**	96.01	99.79	0.96	0.92
LVC	**100.00**	**100.00**	**1.00**	**1.00**	**97.50**	**100.00**	0.98	**0.94**	**100.00**	**100.00**	**1.00**	**1.00**	**99.17**	**100.00**	0.99	**0.98**

Table 1: Comprehensive quantitative comparison across multiple scenarios. Bold fonts indicate the best results.

Method	Normal Overtaking				Opposing Overtaking			
	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑
SimLingo	16.82	85.87	0.23	0.06	23.00	74.98	0.26	0.00
SimLingo+Comm	72.31	91.02	0.79	0.31	66.58	94.07	0.67	0.56
LVC w/o memory	90.76	96.13	**0.94**	0.75	82.28	94.95	0.86	0.62
LVC	**93.80**	**99.00**	**0.94**	**0.88**	**90.55**	**98.75**	**0.91**	**0.81**

Table 2: Quantitative comparison on Overtaking scenarios. Bold fonts indicate the best results.

ing semantic perception information from visual inputs, while Gemini-2.5-flash-thinking (Comanici et al., 2025) acts as the Reasoning Core for communication and decision-making, leveraged for its superior logical inference capabilities. Furthermore, a 50 ms communication latency is introduced to evaluate robustness against network delays.

5.1 Experiments on Comm2Interact

We evaluate our method against SOTA VLA baselines on the Comm2Interact. As mentioned in Sec. 3.1, we categorize negotiation scenarios into "Peer Yields" and "Peer Refuses." To ensure fair comparison, we standardize the peer vehicle behavior. For non-negotiation baselines, the peer vehicle is programmed to automatically yield in the "Peer Yields" scenarios. This ensures that the scene dynamics are consistent across all methods, regardless of their communication capabilities.

Furthermore, since many baseline methods lack explicit overtaking capabilities or cannot interpret overtaking intentions as input prompts, they typically default to car-following behavior in overtaking scenarios, resulting in task failure. Consequently, these methods are excluded from the overtaking comparison. We present the comparison results for the different scenarios in Tab. 1 and Tab. 2, respectively.

Non-Communication VLAs. The non-communicative baselines reveal the fundamental limitations of single-agent systems in handling social interactions. LMDrive (Shao et al., 2024) adopts an overly conservative policy, which ensures safety but frequently leads to **deadlocks** during negotiation, as it yields indefinitely even when the right-of-way is conceded. On the other hand, SimLingo exhibits aggressive behavior that prioritizes efficiency at the expense of safety, resulting in the lowest IS score due to late braking and dangerous maneuvers. These results highlight that without explicit intent sharing, single-agent methods struggle to balance safety and efficiency in complex scenarios.

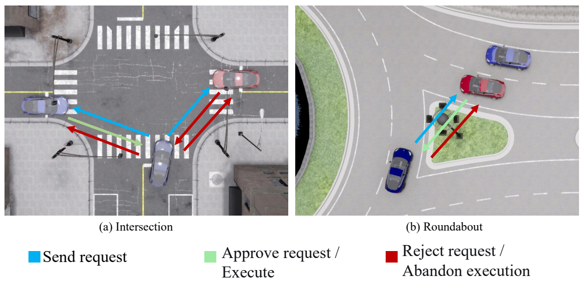


Figure 5: Communication in various scenarios. The three arrows represent our three-way handshake: send request, receive response, and broadcast decision.

Communication-Based VLAs. CoDriving (Liu et al., 2025b) shares perception data to mitigate blind spots but lacks explicit intent negotiation, failing to determine precedence when trajectories overlap. As a result, there is still a high probability of collisions in View Occlusion scenarios, proving that perception sharing alone is insufficient for conflict resolution.

CoLMDriver (Liu et al., 2025a) achieves high Safety Scores across most scenarios but suffers from low Route Completion due to an overly conservative negotiation strategy. Specifically, CoLMDriver aggregates all potentially conflicting vehicles into a single negotiation group. Within this group, the negotiated outcome typically favors al-

Method	InterDrive-total				InterDrive-IC				InterDrive-LM				InterDrive-LC			
	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑	DS↑	RC↑	IS↑	SR↑
CoDriving	74.13	96.31	0.76	0.57	66.32	90.57	0.71	0.61	96.18	**100.00**	0.96	0.75	36.57	**100.00**	0.37	0.00
Rule-based	78.38	91.85	0.80	0.72	80.06	95.93	0.81	**0.72**	94.44	**100.00**	0.94	0.90	34.43	62.29	0.42	0.25
CoLMDriver	**88.53**	94.05	**0.90**	0.77	82.07	88.78	0.86	**0.72**	**98.27**	99.93	**0.98**	0.92	59.21	82.50	0.60	**0.50**
LVC	86.91	**97.11**	0.89	**0.78**	**86.74**	**96.22**	**0.89**	**0.72**	96.80	**100.00**	0.97	**0.95**	**62.58**	92.24	**0.69**	**0.50**

Table 3: Performance comparison on InterDrive scenarios. Bold indicates the best result.

lowing only one vehicle to move while others remain stationary. This leads to inefficient driving: safety is preserved, but the ego vehicle often runs out of time before completing its route, especially in crowded intersections. As shown in Fig. 5(a), the ego vehicle unnecessarily yields to the red vehicle despite no direct conflict between them. Furthermore, relying solely on negotiation without shared perception introduces collision risks. As illustrated in Fig. 5(b), the lack of shared sensory information prevents the detection of occluded hazards.

Compared to existing methods, our approach adopts a decentralized framework and uses primitives to select communication targets accurately, avoiding unnecessary negotiation. In Fig. 5(a), the ego vehicle avoids communicating with the red vehicle. Furthermore, our method combines negotiation with shared perception. In Fig. 5(b), although the red vehicle yields, the ego vehicle detects a hidden threat in the red vehicle's shared view and decides to stop.

Ablation Study. We conducted ablation studies to validate our key components. We compare the following settings: SimLingo+Comm: Using SimLingo as the baseline with the communication module added, but without the Primitive Set and Memory Module. LVC (w/o Memory): Using the Primitive Set but without the Memory Module.

After adding the communication module to SimLingo, the LLM reasoning capability allows it to handle some simple scenarios. However, it still struggles to ensure safety in complex interaction scenarios, such as overtaking. Furthermore, it suffers from inefficient target selection, just like CoLMDriver. This makes communication counterproductive in some cases, leading to worse performance than having no communication. When the Primitive Set is used, the peer vehicle is explicitly assigned different tasks, and the ego vehicle makes judgments based on specific intents. This results in safer and more effective driving compared to the baseline. However, collisions and in-

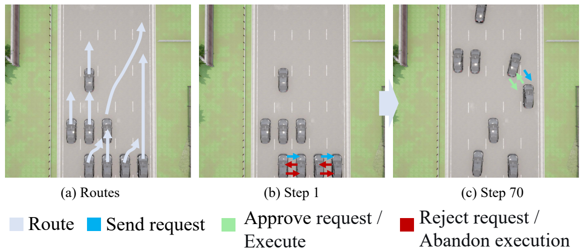


Figure 6: Result of LVC in InterDrive lane change scenario.

complete tasks still occur frequently. After adding the Memory Module (the full method), the model achieves the best performance across multiple scenarios. This demonstrates the effectiveness of our proposed framework.

5.2 Experiments on InterDrive

InterDrive includes scenarios with high traffic density, which poses a challenge for decentralized communication. As shown in the Lane Change (LC) example in the Fig. 6, the vehicle correctly selects the communication target based on its intention and successfully executes the maneuver. Notably, as shown in Tab. 3 we achieved results comparable to CoLMDriver without any data training.

6 Conclusion

In this paper, we address the limitations of existing V2V research, which is often confined to simplistic scenarios that fail to capture the complexity of real-world interactions. To bridge this gap, we propose Comm2Interact, a novel benchmark that establishes a rigorous standard for evaluating autonomous agents in complex, unstructured traffic scenarios. To navigate these challenges, we introduce LVC, an LLM-driven framework that utilizes interaction primitives and a reflection-based memory module to decompose and resolve intricate social conflicts safely. Quantitative results validate that LVC not only handles diverse communication demands but also achieves superior performance compared to existing baselines.

7 Limitations

Despite the promising results, our framework exhibits two primary limitations.

First, the reliance on general-purpose Large Language Models introduces substantial **inference latency**. This computational overhead poses a challenge for real-time deployment, particularly in high-speed scenarios where millisecond-level reaction is critical.

Second, our current approach operates in a zero-shot manner **without domain-specific training**. While general-purpose priors are effective for common tasks, the model lacks the specialized expertise derived from large-scale driving datasets. Consequently, it may struggle with highly specialized or extreme driving contexts that deviate from standard traffic rules.

To address these challenges, future work will focus on **fine-tuning LLMs** by leveraging expert driving data combined with Reinforcement Learning (RL). This direction aims to better align the model's reasoning with professional driving behaviors while potentially optimizing inference efficiency for practical deployment.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Jiaxun Cui, Chen Tang, Jarrett Holtz, Janice Nguyen, Alessandro G Allievi, Hang Qiu, and Peter Stone. 2025. Towards natural language communication for cooperative autonomous driving via self-play. *arXiv preprint arXiv:2505.18334*.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR.

Shiyu Fang, Jiaqi Liu, Mingyu Ding, Yiming Cui, Chen Lv, Peng Hang, and Jian Sun. 2025. Towards interactive and learnable cooperative driving automation: a large language model-driven decision-making framework. *IEEE Transactions on Vehicular Technology*.

Xiangbo Gao, Yuheng Wu, Rujia Wang, Chenxi Liu, Yang Zhou, and Zhengzhong Tu. 2025. Langcoop: Collaborative driving with language. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4226–4237.

Senkang Hu, Zhengru Fang, Zihan Fang, Yiqin Deng, Xianhao Chen, and Yuguang Fang. 2024. Agentscodriver: Large language model empowered collaborative driving with lifelong learning. *arXiv preprint arXiv:2404.06345*.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, and 1 others. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862.

Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. 2022. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 35:4874–4886.

Zhenmin Huang, Haichao Liu, Shaojie Shen, and Jun Ma. 2023a. Parallel optimization for cooperative autonomous driving at unsignalized roundabouts with hard safety guarantees. *arXiv preprint arXiv:2303.03090*.

Zhenmin Huang, Shaojie Shen, and Jun Ma. 2023b. Decentralized ilqr for cooperative trajectory planning of connected autonomous vehicles via dual consensus admm. *IEEE Transactions on Intelligent Transportation Systems*, 24(11):12754–12766.

Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. 2023. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350.

Edouard Leurent and 1 others. 2018. An environment for autonomous driving decision-making.

Changxing Liu, Genjia Liu, Zijun Wang, Jinchang Yang, and Siheng Chen. 2025a. Colmdriver: Llm-based negotiation benefits cooperative autonomous driving. In *ICCV*.

Genjia Liu, Yue Hu, Chenxin Xu, Weibo Mao, Junhao Ge, Zhengxiang Huang, Yifan Lu, Yinda Xu, Junkai Xia, Yafei Wang, and 1 others. 2025b. Towards collaborative autonomous driving: Simulation platform and end-to-end system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Haichao Liu, Zhenmin Huang, Zicheng Zhu, Yulin Li, Shaojie Shen, and Jun Ma. 2024a. Improved consensus admm for cooperative motion planning of large-scale connected autonomous vehicles with limited communication. *IEEE Transactions on Intelligent Vehicles*.

Jiaqi Liu, Peng Hang, Xiaoxiang Na, Chao Huang, and Jian Sun. 2024b. Cooperative decision-making for cavs at unsignalized intersections: A marl approach with attention and hierarchical game priors. *IEEE Transactions on Intelligent Transportation Systems*.

Katrin Renz, Long Chen, Elahe Arani, and Oleg Sinavski. 2025. Simlingo: Vision-only closed-loop autonomous driving with language-action alignment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11993–12003.

Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. 2024. Lmdrive: Closed-loop end-to-end driving with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15120–15130.

Licheng Wen, Daocheng Fu, Xin Li, Xinyu Cai, Tao Ma, Pinlong Cai, Min Dou, Botian Shi, Liang He, and Yu Qiao. 2023. Dilu: A knowledge-driven approach to autonomous driving with large language models. *arXiv preprint arXiv:2309.16292*.

Runsheng Xu, Xin Xia, Jinlong Li, Hanzhao Li, Shuo Zhang, Zhengzhong Tu, Zonglin Meng, Hao Xiang, Xiaoyu Dong, Rui Song, and 1 others. 2023. V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13712–13722.

Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. 2022a. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *European conference on computer vision*, pages 107–124. Springer.

Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. 2022b. Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2583–2589. IEEE.

Wei Zhan, Liting Sun, Di Wang, Haojie Shi, Aubrey Clausse, Maximilian Naumann, Julius Kummerle, Hendrik Konigshof, Christoph Stiller, Arnaud de La Fortelle, and 1 others. 2019. Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps. *arXiv preprint arXiv:1910.03088*.

Rui Zhao, Yun Li, Fei Gao, Zhenhai Gao, and Tianyao Zhang. 2023. Multi-agent constrained policy optimization for conflict-free management of connected autonomous vehicles at unsignalized intersections. *IEEE transactions on intelligent transportation systems*, 25(6):5374–5388.

Zhi Zheng and Shangding Gu. 2024. Safe multi-agent reinforcement learning with bilevel optimization in autonomous driving. *IEEE Transactions on Artificial Intelligence*.

Ming Zhou, Jun Luo, Julian Villella, Yaodong Yang, David Rusu, Jiayu Miao, Weinan Zhang, Montgomery Alban, Iman Fadakar, Zheng Chen, and 1 others. 2020. Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. *arXiv preprint arXiv:2010.09776*.

A Appendix

A.1 Use of AI Assistants

We used Gemini-3.0-flash for language polishing only.

A.2 Primitive Sets and Communication Examples

To provide a more intuitive understanding of our methodology and facilitate further analysis of the interaction logic, we provide comprehensive details in this supplementary material. We first explicitly list the System Prompts and Primitive Sets designed for each module of the LVC framework to ensure reproducibility. Furthermore, we present Qualitative Communication Examples from several challenging scenarios (including Overtaking and Roundabouts), showcasing the step-by-step negotiation process and the resulting cooperative behaviors.

Listing 1: Prompt for Communication Intent Parsing.

```
### Role Definition
As the V2V Communication Strategy Agent, your goal is to determine the ego vehicle's communicative intent. To do this, you must synthesize high−level navigation goals with real−time environmental context, ensuring communication is driven by genuine interaction needs, not just map updates.

### Input Data Structure
{
  "Ego_Intent": The initial intent code from the direction options list below (usually "4"),
  "Ego_State": speed, compass, location.
  "Surrounding_vehicles": List of vehicles.
    "Coordinate System": Ego is at (0, 0).
    "X" : Longitudinal distance (>0 ahead, <0 behind).
    "Y" : Lateral distance (Left is positive, Right is negative).
  "Road_Topology": The recommended direction If vehicle want to overtake. "left", "right", or "none".
  "Lane_Change_Recommend": One of "roundabout", "intersection", or "straight".
  "Memory_Knowledge": If not empty, it means the retrieved scenario is highly similar to the current scenario.
}

### Direction options
[1] go left at the next intersection
[2] go right at the next intersection
[3] go straight at the next intersection
[4] follow the road (Default)
[5] do a lane change to the left (Navigation/Mandatory)
[6] do a lane change to the right (Navigation/Mandatory)
[7] prepare to overtake from the left lane (Speed/Discretionary)
[8] prepare to overtake from the right lane (Speed/Discretionary)

### Position ClassIfication (Pre−processing)
For each vehicle, categorize location using Strict Coordinate Rules:
− Ego Lane: {−0.5∗Road_Width} < y < {0.5∗Road_Width}
− Target Lane (Left): {0.5∗Road_Width} < y < {1.5∗Road_Width} (Applicable If Intent involves Left)
− Target Lane (Right): {−1.5∗Road_Width} < y < {−0.5∗Road_Width} (Applicable If Intent involves Right)
− Longitudinal: ahead (x > 0) | behind (x < 0)

### Intent Re−evaluation
1. Intersection Priority:
If Intent is "4" and RoadTopology is "Roundabout" or "Intersection":
− Action: Upgrade Intent to "2" (Go right at intersection).
− Rationale: Considered as road merging.
2. Discretionary Overtake (Low Priority):
If (Intersection Condition NOT met) and (original Intent is "3" or "4") and (Ego is stuck behind a vehicle: speed < self_speed, Ego Lane, x < {Max_Safe_Driving_Distance}):
− If LaneChangeRecommend is "Left":  Upgrade Intent to "7" (Overtake Left).
− else If LaneChangeRecommend is "Right": Upgrade Intent to "8" (Overtake Right).
− else: Keep Input Intent.
3. Default:
− Action: Keep Input Intent unchanged.
### Output Format
{
  "reasoning": "Explain intent update and task assignment logic for each ID.",
  "original_intent_code": Int (one of the ∗∗DIRECTION OPTIONS∗∗ list),
  "final_intent_code": Int (one of the ∗∗DIRECTION OPTIONS∗∗ list),
  "final_intent_description": "String",
}
```

Listing 2: Primitive Set of Communication Agent Selection

```
### Role Definition
You are the V2V Communication Strategy Agent for an autonomous vehicle (Ego vehicle).Your goal is to generate a detailed Communication Plan. You must assign a specIfic Task
Type to each relevant surrounding vehicle based on its position and the Ego's final intent.

### Task Type Definitions (Strict Enum)
1. negotiation_intersection: Coordinate Right−of−Way at junctions.
2. Negotiation_line_change: Request yield/acknowledge for merge.
3. Cooperative_overtake: Request vehicle ahead to enable passing.
4. perception_intersection: Request blind−spot data at intersections.
5. perception_lane_change: Request "See−through" data from lead vehicle.
6. None: No communication needed.

###Primitive of lane changing (Navigational Lane Change)
 Ego lane vehicles (abs(y) < {Road_Width}}): Ignore.
 Target lane vehicles (Left:{0.5∗Road_Width} < y < {1.5∗Road_Width} or Right:{−1.5∗Road_Width} < y < {−0.5∗Road_Width}):
    If vehicle is behind (x < 0) (lag vehicle): assign negotiation_lane_change.
    If vehicle is ahead (x > 0) (lead vehicle):  Ignore.
 Constraint: Do not use cooperation or perception for intent 5/6.

###Primitive of overtaking (Overtake)
 Ego Lane vehicles (abs(y) < {Road_Width}) (The car being overtaken):
    If vehicle is ahead (x > 0): assign perception_lane_change.
     Reason: Ego needs the front car's sensor view (See−through) to ensure safety.
 Target Lane vehicles (Left:{0.5∗Road_Width} < y < {1.5∗Road_Width} or Right:{−1.5∗Road_Width} < y < {−0.5∗Road_Width}):
    If vehicle is ahead (x > 0) (Blocking/Lead): assign cooperation_overtake.
      Reason: Ask them to accelerate or maintain speed to create/keep a gap.
    If vehicle is behind (x < 0) (Approaching): assign negotiation_lane_change.
      Reason: Ask them to yield/slow down.

###Primitive of intersection turning (Intersection)
 Logic Execution (Priority order): Iterate through surrounding vehicles and apply these Strict Rules.
   1.  Check for Left Lane Occlusion (Perception):
        Condition: vehicle is in the immediate left lane, close ahead, same direction.
        Rule: If ({0.5∗Road_Width} < y < {1.5∗Road_Width}) and (0 < x < {Min_Safe_Driving_Distance}) and (−15 < compass < 15)
        assignment: perception_intersection
        Reason: Close vehicle on the left may occlude view.
   2.  Check for any Conflicting Traffic (Negotiation):
        Global Pre−condition: vehicle must be ahead (x > 0).
        Conflict Scenarios (or Logic):
            Oncoming Traffic: (compass < −135 or compass > 165)
            Right Cross−Traffic: (y < {−MinSafe_Driving_Distance}) and (45 < compass < 135)
        Rule: If (x > 0) and (Left_Cross or Oncoming or Right_Cross)
        assignment: negotiation_intersection
        Reason: vehicle poses a collision risk from Left, Right, or Opposite direction.
   3.  Default:
        Rule: If none of the above match −> NONE (Ignore).
### Output Format
{ "communication_plan": [
    STRICT RULE: Only include vehicles where task_type is NOT "NONE".
    If all vehicles are assigned "NONE", return an empty list [].
    {
     "target_vehicle_id": Integer,
     "role_description": "Direction of Vehicle",
     "task_type": "ENUM_VALUE" // Must NOT be "NONE"
    }
   // Return empty list [] if no communication is needed.
  ]}
```

Listing 3: Primitive Set of Connected Vehicle Response

```
###Primitive of negotiation_intersection
You are the Intersection Safety Agent for a Responder vehicle and received a negotiation_intersection request.
Objective: Yield to the HV If safe. However, prioritize your own physical safety and traffic flow.
# Decision Logic (Step-by-Step)
You must validate If executing the default Agree (decelerate) Action is safe. Check the following conditions in order:
 Check : Rear-End Collision Risk (Rear Safety)
   Observation: Is there a vehicle directly behind you (x < 0, |y| < {Road_Width}) that is following closely (abs(x) < {Min_Safe_Driving_Distance})?
   Logic: If I brake to yield, will the rear vehicle crash into me?
   Result: If YES, Agree is unsafe. -> Override to Reject.

###Primitive of negotiation_lane_change
You are the Lag vehicle Agent in the target lane and received a negotiation_lane_change request from the HV (who wants to merge in front of you).
Objective: Facilitate the merge by slowing down, ONLY If it doesn't require emergency braking.
# Decision Logic (Step-by-Step)
Analyze Gap Dynamics:
   If HV near you (x > {-Min_Safe_Driving_Distance} and x < {Min_Safe_Driving_Distance} in HV State) and no other vehicles are close behind you ({-
Min_Safe_Driving_Distance} < x < 0 and |y| < {Road_Width} in Surrounding vehicles):
     Decision: Agree, Action: decelerate (Create space).
   ELSE If HV near you (x > {-Min_Safe_Driving_Distance} and x < {Min_Safe_Driving_Distance} in HV State) and there are other vehicles close behind you ({-
Min_Safe_Driving_Distance} < x < 0 and |y| < {Road_Width} in Surrounding vehicles):
     Decision: Reject, Action: maintain.
   ELSE: No need to decelerate
     Decision: Agree,  Action: maintain.

###Primitive of perception_intersection
You are the Perception Provider at an intersection and received a perception_intersection request.
Objective: Share relevant blind-spot objects. You do not yield or accelerate based on this, just observe.
# Decision Logic (Step-by-Step)
  IdentIfy Threats:
   scan Surrounding vehicles.
   Filter objects that are approaching the intersection collision zone.

###Primitive of perception_lane_change
You are the Perception Provider (Sensor Extension) and received a perception_lane_change request from a Host vehicle (HV) intending to change lanes/overtake.
Objective: strictly scan the HV's Target Lane for potential hazards. Do NOT evaluate safety for yourself; only report dangerous objects to the HV.
# Decision Logic (Step-by-Step)
1. IdentIfy Target Lane:
   If HV Intent involves "Left" -> scan region y > {Road_Width} (my left lane).
   If HV Intent involves "Right" -> scan region y < {-Road_Width} (my right lane).
2. Filter Dangerous Objects:
   Iterate through Surrounding vehicles.
   Keep objects inside the Target Lane Region.
   Hazard Criteria: Include the object If it is:
     Oncoming Traffic (|Compass|>165, strictly for two-way roads).
     Stationary/Blocking (Velocity approx 0).
     SignIficantly Slower than traffic flow.
3. Format Output: Return the list of identIfied hazards.

###Primitive of cooperation_overtake
You are the Blocking vehicle Responder in the target lane and received a cooperation_overtake request from a Host vehicle (HV) stuck in the adjacent lane.
Objective: The HV needs space to cut in. You must evaluate If you can create a Sufficient Gap by either Accelerating or Decelerating.
# Decision Logic (Step-by-Step)
Step 1: Check If HV is safely behind (Clear Gap)
 Condition: Check x in HV State.
 Logic: is x < {-MAX_Safe_Driving_Distance}? (HV is far behind).
 Result (If YES):
    Decision: Agree, Action: maintain.
    Reason: HV is already far behind; no cooperation Action needed, just maintain course.
Step 2: Check Yield Feasibility (Deceleration Check)
 Condition: x in HV_State is between {-Max_Safe_Driving_Distance} and {Min_Safe_Driving_Distance} .
 Safety Check (Rear Collision Risk):
    scan Surrounding vehicles for any vehicle satisfying:
        Lane: |y| < {Road_Width} (Same Lane)
        Position: {-MAX_Safe_Driving_Distance} < x < {-Min_Safe_Driving_Distance}(Close Rear Zone)
 Result:
    If NO such vehicle is found (Rear is Clear):
        Decision: Agree, Action: decelerate.
        Reason: HV is close/alongside, and it is safe to brake (no close tailgater).
    ELSE (Rear vehicle detected in {-Max_Safe_Driving_Distance} and {-Min_Safe_Driving_Distance}):
        Decision: Reject, Action: maintain
        Reason: Cannot decelerate due to risk of rear-end collision.
Step 3: Default / Fallback
 Condition: None of the above (e.g., HV is ahead x > {Min_Safe_Driving_Distance} or other edge cases).
 Result:
    Decision: Reject, Action: maintain.

### Output Format
{
  "response_type":
  "reasoning":
  "my_action_intent": Options: ACCELERATE, DECELERATE, MAINTAIN
  "target_speed_adjustment": Positive for Accel, Negative for Decel, 0 for Maintain
}
```

Listing 4: Primitive Set of Communication-based Decision Making

```
###Input_des:
1. Ego Intent: The initial intent code from the direction options list below (usually "4").
2. Ego State: speed, compass, location.
3. Surrounding vehicles: List of vehicles.
   Coordinate System: Ego is at (0, 0).
   X: Longitudinal distance (>0 ahead, <0 behind).
   Y: Lateral distance (Left is positive, Right is negative).
4. V2V Responses (The Critical Input): A list of responses received from other vehicles.

###Primitive of intersection turning:(Roundabouts are considered a special case of turning.)
# Role Definition
You are the Final Safety Arbiter for an autonomous vehicle approaching an intersection.
Your goal is to decide whether to proceed with the turn/crossing or abort (Brake/Yield) based on peer responses.
{Input_des}
# Decision Logic (Strict "Veto" Rules)
Step 1: Analyze Responses
Iterate through V2V Responses:
1.  Check Negotiations: If any vehicle responded with "response": "Reject", it means they are not yielding. −> Flag: abort.
2.  Check Perception: If any perception response contains objects in the perception_data list satisfies the following strictly: −> Flag: abort.
    Global Pre−condition: vehicle must be ahead (x > 0).
    Conflict Scenarios (or Logic):
        Left Cross−Traffic: (y > 0) and (−135 < compass < −75)
        Oncoming Traffic: (compass < −135 or compass > 165)
        Right Cross−Traffic: (y < 0) and (45 < compass < 135)
    Rule: If (x > 0) and (Left_Cross or Oncoming or Right_Cross)
Step 2: Determine Final Action
 If Flag is abort:
    Decision: You cannot proceed safely.
    Action: Brake.
 ELSE (All Agreed + Blind Spots Clear):
    Decision: Safe to proceed.
    Action: execute_intersection_maneuver.

###Primitive of lane changing:
# Role Definition
You are the Final Maneuver Planner for a Lane Change.
Your goal is to confirm If the target lane is safe based on peer responses.
{Input_des}
# Decision Logic (Strict "Veto" Rules)
Step 1: Analyze Responses
1.  Negotiation Check: Did the Target Lane Rear vehicle (Lag) reply Agree?
    If Reject −> Flag: abort.
2.  Check Perception:
   Check objects from both detected_vehicles and perception_data. If any object satisfies the following strictly: −> Flag: abort.
    Conflict Scenarios (and Logic):
        Oncoming Traffic  (compass < −135 or compass > 135, x > 0 )
        In the target lane  ({−0.5∗Road_Width} < y < {1.5∗Road_Width})
        In Front  (x > {Min_Safe_Driving_Distance})
Step 2: Determine Fallback (If aborted)
 If Flag is abort:
    Switch Intent to: FOLLOW_ROAD (Give up merge for now).
    Check Safety Gap:  If distance_to_front_vehicle < Safe Distance (e.g., 20m): Action: decelerate.
        Else: Action: maintain.
 ELSE (Success):
    Action: START_LANE_CHANGE.

###Primitive of Overtaking :
# Role Definition
You are the Strategic Overtaking Arbiter.
You intended to overtake ("7" or "8"). You must evaluate If the collaboration was successful and the path is clear.
{Input_des}
# Decision Logic (Strict "Veto" Rules)
Step 1: Analyze Responses
1.  Cooperation Check (The Blocking Car):
    If present, did it reply Agree (Accelerate/decelerate)?
    If Reject (Insufficient gap capabilities) −> Flag: abort.
2.  Negotiation Check (The Rear Car):
    If present, did it reply Agree?
    If Reject −> Flag: abort.
3.  Perception Check (Blind Spot):
   Check objects from both detected_vehicles and perception_data. If any object satisfies the following strictly: −> Flag: abort.
    Conflict Scenarios (and Logic):
        In the target lane  (Left:{0.5∗Road_Width} < y < {1.5∗Road_Width} or Right:{−1.5∗Road_Width} < y < {−0.5∗Road_Width})
        In Front  (x > {−Min_Safe_Driving_Distance})
Step 2: Determine Fallback (If aborted)
 If Flag is abort:
    Decision: Overtaking is unsafe or blocked.
    New Intent: Revert to "4" ("follow the road").
 ELSE (All Conditions Met):
    Action: start_overtake_maneuver.

### Output Format
{
  "final_decision": "EXECUTE" or "ABORT",
  "reasoning":
  "final_intent_code":
  "speed_intent": 'MAINTAIN'
  "ego_intent":
}
```

Comm Plan

```
"comm_plan": [
{ "target_id": 33319,
"role": "Front_Right",
"task": "COOPERATION_OVERTAKING" },
{ "target_id": 33318,
"role": "Front_Ego",
"task": "PERCEPTION_LANE_CHANGE" } ]
```

Peer Responses

```
"33318": {
"type": "DATA_PROVIDED",
"perception": [{
"rel_pos": [-8.0, -3.2],
"risk": "Blocking" }] },
"33319": {
"type": "AGREE", "intent": "DECELERATE",
"speed_adj": -5.0 }
```

Final Decision

```
"final_decision": "EXECUTE",
"reasoning": "All V2V responses were AGREE, and no conflicting objects were detected in the target lane (right lane). The path is clear for the overtaking maneuver.",
"speed_intent": "MAINTAIN",
"ego_intent": overtake from the right lane"
```

VISUALIZATION

Front View

Scenario View

Figure 7: A communication case of Normal Overtaking Scenarios

Comm Plan

```
"comm_plan": [
{ "target_id": 3696,
"role": "Front_Ego",
"task": "PERCEPTION_LANE_CHANGE" },
]
```

Peer Responses

```
"3696": {
"type": "DATA_PROVIDED",
"perception": [{
"rel_pos": [37.6, 2.9],
"risk": "Oncoming Vehicle" }] }
```

Final Decision

```
"final_decision": "ABORT",
"reasoning": "Perception data from the blocking vehicle (ID 3696) shows an object in the target left lane (y=2.6) and in front (x=50.3, which is greater than -1). This constitutes a conflict scenario, making the overtaking maneuver unsafe. Therefore, the maneuver must be aborted.",
"speed_intent": "MAINTAIN",
"ego_intent": follow the road"
```

VISUALIZATION

Front View

Scenario View

Figure 8: A communication case of Opposing Overtaking Scenarios

Comm Plan


```
"comm_plan": [
{ "target_id": 110,
"role": "Front_Left_Ego",
"task": "NEGOTIATION_INTERSECTION" },
{ "target_id": 111,
"role": "Front_Left_Ego",
"task": "NEGOTIATION_INTERSECTION" },
]
```


Peer Responses


```
"110": {
"response_type": "AGREE",
"reasoning": "HV is not extremely close to the front bumper (x=15.4m). No vehicle is directly behind (vehicle 111 has |y|=3.6, which is not < 2). Both safety checks passed. Yielding.",
"my_action_intent": "DECELERATE"}
"111": {
"response_type": "AGREE",
"reasoning": "Both checks passed. HV is not too close (x=18.0), and no vehicle detected directly behind (no x < 0). Yielding.",
"my_action_intent": "DECELERATE"}
```


Final Decision


```
"final_decision": "EXECUTE",
"reasoning": "All peer vehicles agreed to the maneuver, and no conflicting objects were detected in their perception data.",
"speed_intent": "MAINTAIN",
"ego_intent": go straight at the next intersection"
```

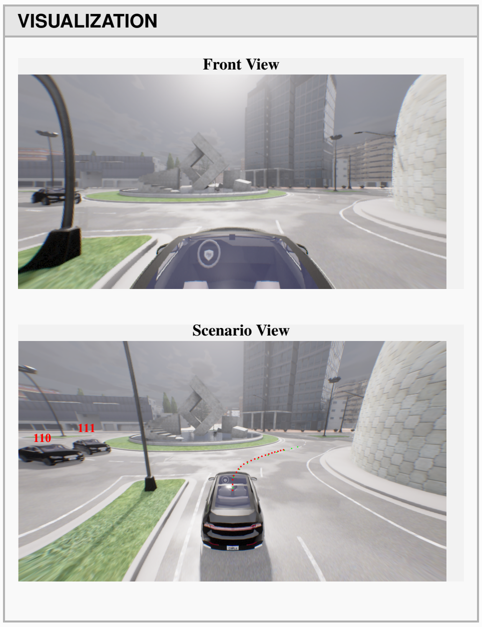



Figure 7: A communication case of Roundabout Scenarios

Comm Plan


```
"comm_plan": [
{ "target_id": 3697,
"role": "Front_Left_Ego",
"task": "PERCEPTION_INTERSECTION" },
]
```


Peer Responses


```
"3697": {
"response_type": "AGREE",
"reasoning": "Both checks passed. HV is not too close (x=18.0), and no vehicle detected directly behind (no x < 0). Yielding.",
"perception_data":
{ "relative_position": {
"x": 14.0,
"y": -20.1
},
{ "relative_position": {
"x": 10.9,
"y": 23.4
},
}
```


Final Decision


```
"final_decision": "ABORT",
"reasoning": "Perception data from vehicle 3697 indicates an object ahead (x=19.1) and to the left (y=26.7) with a compass of -89.1, which falls within the Left Cross-Traffic conflict scenario (-135 < compass < -75). This constitutes a safety risk.",
"speed_intent": "DECELERATE",
"ego_intent": "Brake"
```

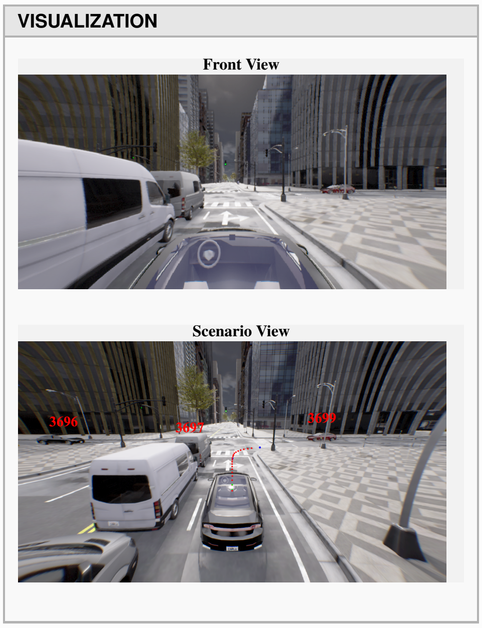



Figure 8: A communication case of Roundabout Scenarios