

---

# TALKING VEHICLES: COOPERATIVE DRIVING VIA NATURAL LANGUAGE

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

Using natural language as a vehicle-to-vehicle (V2V) communication protocol offers the potential for autonomous vehicles to drive cooperatively not only with each other but also with human drivers. Simple and effective messages for sharing critical observations or negotiating plans to achieve coordination could improve traffic safety and efficiency compared to methods without communication. In this work, we propose a suite of traffic tasks in vehicle-to-vehicle autonomous driving where vehicles in a traffic scenario need to communicate in natural language to facilitate coordination in order to avoid an imminent collision and/or support efficient traffic flow, which we model as a general-sum partially observable stochastic game. To this end, this paper introduces a novel method, LLM+DEBRIEF, to learn a message generation and control policy for autonomous vehicles through multi-agent discussion. To evaluate our method, we developed a gym-like simulation environment that contains a range of accident-prone driving scenarios that could be alleviated by communication. Our experimental results demonstrate that our method is more effective at generating meaningful and human-understandable natural language messages to facilitate cooperation and coordination than untrained LLMs. Our anonymous code is available in supplementary materials.

## 1 INTRODUCTION

State-of-the-art autonomous driving policies are commonly designed from the perspective of a single agent’s sensors. Therefore, to enhance safety, it is essential to account for multi-agent interactions. Among the strategies employed is incorporating motion prediction or intention inference of other traffic participants in decision-making. Some models independently forecast the future movements of these traffic participants (Wu et al., 2023), whereas others account for mutual interactions among autonomous vehicles and other traffic agents (Seff et al., 2023). However, the challenge lies in the substantial uncertainty in predicting other drivers’ intentions, often resulting in excessively cautious driving strategies (Rhinehart et al., 2021).

Given its cooperative nature, the safe driving problem could be largely simplified by enabling vehicles to communicate their intentions and observations with each other. The concepts of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication have recently emerged as a promising approach for multi-vehicle cooperation, garnering considerable research interest (Wang et al., 2020; Cui et al., 2022; Xu et al., 2022a;b). While the focus of V2V communication has predominantly been on cooperative perception rather than cooperative control, it often employs modalities (latent representations, LiDAR points, locations of objects, etc.) that are not intuitively nor easily understood by humans, thus requiring that all the participating vehicles be autonomous and share the same protocol. Natural language, as a refined and highly adaptable form of human communication, offers the potential for human drivers to also participate in this cooperation. If vehicles could “speak” in human language, it would pave the way for developing technologies that facilitate communication between autonomous vehicles and human drivers, enhancing cooperation and understanding in mixed-autonomy traffic environments. Even in fully autonomous settings, a natural language interface could offer flexibility for agents to explain their decisions, talk

054  
055  
056  
057  
058  
059  
060  
061  
062  
063  
064  
065  
066  
067  
068  
069  
070  
071  
072  
073  
074  
075  
076  
077  
078  
079  
080  
081  
082  
083  
084  
085  
086  
087  
088  
089  
090  
091  
092  
093  
094  
095  
096  
097  
098  
099  
100  
101  
102  
103  
104  
105  
106  
107

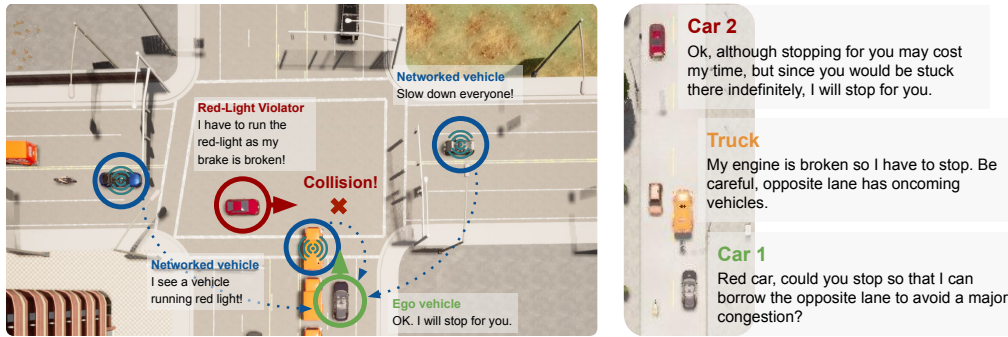


Figure 1: **Left:** A red-light violation scenario where there is a car running the red light. **Right:** An overtake scenario where a truck is broken and stopped on a two-lane two-way road.

to other autonomous vehicles with proper translation, quickly adapt to different traffic regulations, and negotiate plans.

As a concrete example of the power of using language in V2V communication, consider the following scenario, as illustrated in Figure 1. A truck has broken down and stopped on a two-way road divided by a yellow dashed line. The vehicles trailing the truck cannot see whether the lane going in the opposite direction is free from oncoming cars, which could allow them to use that lane to safely move around the truck to circumvent the growing traffic jam. If the truck is capable of communication, it could relay a message such as, “The opposite lane is clear, you may safely perform a lane change to pass me.” Alternatively, in busier times, the queued vehicles could initiate a dialogue with oncoming traffic, proposing, “Can we take turns using the lane to avoid major congestion?” Through simple but effective communication about their intentions or observations, a mixture of human-driven and autonomous vehicles could significantly enhance traffic flow and overall efficiency.

In this work, we first introduce the problem of *talking vehicles*. In this problem setting, a scenario in autonomous driving is formulated as a multi-agent partially observable and general-sum game wherein each traffic participant is pursuing a cooperative goal modulated by individual preferences<sup>1</sup>. When conflicts arise, or unexpected events occur, vehicles have the opportunity to generate and broadcast messages that contain observations of abnormal events or negotiations to cooperatively reach their goals. The vehicles can then make decisions according to the received messages.

Enabling vehicles to “talk” with intentions and convey helpful information presents significant challenges. There are studies like Dolphins (Ma et al., 2023) and LINGO-1 (Wayve, 2023) that have trained Visual-Language-Action models to both make driving decisions and articulate their reasoning to humans. However, training such models requires extensive data. At the time of writing this paper, only a limited number of datasets exist that provide language commentary data for single-agent driving scenarios (Kim et al., 2018; 2019; Qian et al., 2023; Sima et al., 2023). To the best of our knowledge, datasets featuring natural language data for inter-vehicle communication are not yet available.

On the other hand, multi-agent “selfplay” learning in a high-fidelity simulator requires no real-world data and allows for closed-loop learning and evaluation. With recent advances in large language models (LLMs), there has been a surge of interest in applying LLMs to multi-agent games (Bakhtin et al., 2022; Xu et al., 2023a; Light et al., 2023). However, the synergy between LLMs and multi-agent games remains unexplored in the context of autonomous driving. Existing efforts in autonomous driving mainly focus on leveraging LLMs for decision-making in a *single-agent* setting (Mao et al., 2023a; Shao et al., 2023; Ma et al., 2023).

Thus, we propose a novel method, LLM+DEBRIEF, as an initial attempt to use LLMs to help solve the cooperative driving problem in a V2V communication framework. This method

<sup>1</sup>Specifically, we assume all cars seek a smooth traffic flow. Meanwhile, each car prefers to reach its destination as quickly as possible.

---

108 optimizes the in-context knowledge of a language model as the message generation policy  
109 through a turn-based post-episode discussion and leverages LLMs to analyze the dialogue  
110 history among neighboring vehicles to generate general driving commands and messages to  
111 send. Finally, an atomic controller executes the control according to the updated instructions.  
112 To test our method and provide a research test bed for the community, we build a simulation  
113 framework containing an array of interesting multi-agent driving scenarios that support  
114 communications in natural languages. Our experimental results show an improvement in  
115 driving safety and efficiency of our method compared to the methods without communication.

116 In summary, this paper makes the following contributions:

- 117 1. We introduce the task of socially interactive driving using natural language, aiming  
118 to revolutionize how autonomous vehicles communicate and interact in a traffic  
119 environment.
- 120 2. We develop a novel method enabling autonomous vehicles to dynamically generate  
121 critical observation messages and collaboratively decide driving plans, enhancing  
122 safety and efficiency in traffic navigation.
- 123 3. We present a simulation framework designed to realistically model vehicle-to-vehicle  
124 communication through natural language, featuring diverse scenarios that capture a  
125 range of traffic conditions and interactive challenges.
- 126 4. We evaluate our methodology within this simulation framework, and find that  
127 compared to alternative approaches, our method is able to generate meaningful  
128 messages and improve traffic safety as well as efficiency.

129 We will open-source our code upon the publication of this paper. An example code repository  
130 is provided in the supplementary materials.

## 131 2 RELATED WORK

132 **Vehicle-to-vehicle Communication.** Vehicle-to-Vehicle (V2V) or Vehicle-to-everything  
133 (V2X) communication offers the potential to effectively facilitate multi-vehicle cooperation,  
134 improving the safety and reliability of autonomous vehicles in urban driving scenarios.  
135 Existing research predominantly concentrates on cooperative perception data sets (Yu et al.,  
136 2022; Xu et al., 2022b; Li et al., 2022) and tasks like cooperative detection and prediction  
137 (Wang et al., 2020; Chen et al., 2019; Xu et al., 2022a), leveraging sensor data from cameras,  
138 LiDAR, and other resources. The message aggregation strategies include early fusion (Qiu  
139 et al., 2022), late fusion, and intermediate fusion (Wang et al., 2020). Considering the  
140 limited V2V bandwidth, efforts were made to reduce the message size (Hu et al., 2022). Aoki  
141 et al. (2020) developed a reinforcement learning method for selecting what information to  
142 be transmitted in cooperative perception. Although Cui et al. (2022) developed end-to-end  
143 driving policies and derived the critical information from expert supervision, the cooperation  
144 is still completed at the perception level. In contrast, our work focuses on the message  
145 in the natural language space to enhance both cooperative perception and formulation  
146 of cooperative driving strategies. Real-world communication often suffers from caveats  
147 in the communication mechanisms, including packet loss, latencies (Lei et al., 2022), and  
148 localization errors. Although some works consider adversarial attacks (Tu et al., 2021) in  
149 V2V communication, we assume all vehicles are cooperative in this work.

150 **Multi-Agent Social Interactions with LLMs.** Large Language Models (LLMs) show  
151 promising communication and reasoning capabilities, suggesting their potential in multi-agent  
152 interaction scenarios. For instance, Generative Agents (Park et al., 2023) represents an  
153 early attempt at employing LLM agents for free-form chatting, demonstrating the believable  
154 behaviors of LLM agents in spreading information. However, this study did not evaluate the  
155 LLMs’ capabilities in planning or solving multi-agent tasks. Cicero (Bakhtin et al., 2022),  
156 on the other hand, finetunes a language model to imitate human behaviors from a dataset  
157 to generate truthful messages in the game of Diplomacy, which is mixed-motive and requires  
158 communication in natural language. They train reinforcement learning policies to analyze  
159 the dialogue select actions and generate deceptive messages through value filtering. Recent  
160  
161

---

works on communication games Werewolf (Xu et al., 2023a) and Avalon (Light et al., 2023) demonstrate that LLMs can achieve impressive results in multi-agent settings. This success is particularly notable when LLMs are combined with reinforcement learning or fine-tuning. Shi et al. (2023) find that LLMs can be applied to achieve Ad Hoc teamwork in the Avalon game. Recently, a generative agent-based social simulator (Vezhnevets et al., 2023) has been proposed to serve as a test bed for multi-agent LLM interactions as well.

**LLMs for Autonomous Driving.** LLMs have shown great potential in solving various autonomous driving tasks. In particular, they are promising in tackling corner cases (Wen et al., 2023b) due to their reasoning ability and the common-sense knowledge embedded, yielding a more generalizable autonomous driving stack. Recent studies have explored various approaches to tailor state-of-the-art LLMs for driving. Similar to other embodied tasks (Driess et al., 2023; Brohan et al., 2023), a foundational challenge lies in grounding LLMs in the real world—the LLMs need to perceive and understand the traffic scenarios. A straightforward approach is to obtain the observations from oracle perception models (Mao et al., 2023b) and convert them to textual descriptions (Mao et al., 2023a; Sha et al., 2023; Jin et al., 2023; Cui et al., 2023). Some other studies tackled this challenge by introducing Visual Language Models (VLMs), which are adapted to driving domains through in-context instruction tuning (Ma et al., 2023) or fine-tuning (Wayve, 2023; Xu et al., 2023b; Ding et al., 2023; Yang et al., 2023). To enhance LLMs’ reasoning ability, prior works have investigated incorporating handcrafted guidance and examples in the prompts (Sha et al., 2023; Jin et al., 2023; Cui et al., 2023), structuring the reasoning procedure (Mao et al., 2023b; Sima et al., 2023), and fine-tuning the models on driving datasets. Notably, fine-tuning LLMs and VLMs requires an extensive amount of driving data with language labels. While a limited number of such datasets are available (Kim et al., 2018; 2019; Malla et al., 2023), they were mostly created in the pre-LLM era and, thus, are not designed for LLM fine-tuning. While several works have attempted to adapt existing language-driving datasets for LLM fine-tuning (Ding et al., 2023; Xu et al., 2023b; Ma et al., 2023), growing attention has been drawn to directly augment large-scale multimodal driving datasets, such as nuScenes (Caesar et al., 2020), Waymo (Sun et al., 2020), and ONCE (Mao et al., 2021), with language labels (Qian et al., 2023; Shao et al., 2023; Sima et al., 2023; Nie et al., 2023). Note that existing models were predominantly evaluated in *open-loop* fashions, except for Shao et al. (2023); Sha et al. (2023); Jin et al. (2023). The open-loop evaluation results may not effectively imply the models’ closed-loop performance after deployment. In contrast, similar to Surrealdriver (Jin et al., 2023), we conduct closed-loop tests of the proposed multi-agent communication and control framework in CARLA (Dosovitskiy et al., 2017). More importantly, none of the existing works have explored LLMs in a multi-agent setting with V2V communication as we did. LanguageMPC (Sha et al., 2023) was demonstrated in a multi-agent scenario, yet its controller is centralized.

### 3 PROBLEM DEFINITION

We frame the problem of *Talking Vehicles* as a general-sum partially observable stochastic game (POSG), focusing on optimizing the social welfare of a *focal population* ( $\mathcal{F}$ ) (Agapiou et al., 2022) — defined as the cumulative reward of all agents of interest — as the primary objective. This problem is a general-sum game because it includes scenarios that are not entirely cooperative; conflicts of interest may exist and some agents may have to sacrifice their individual interests for the overall benefits. Each agent’s observation space is limited to a partial view of the full state, and agents make decisions in a decentralized manner based on ego partial observations and received messages from other agents. Within this problem, the action space for each agent has two main components: 1. the generation of messages, and 2. the control of the vehicle. In this work, the message generation space is scoped within natural language.

A POSG can be described by the tuple  $\langle \mathcal{I}, \mathcal{S}, \{\mathcal{O}_i\}, \{\mathcal{A}_i\}, \mathcal{P}, \{\mathcal{R}_i\}, \gamma \rangle$ , where  $\mathcal{I} = \{1, 2, \dots, N\}$  refers to the identities of actionable agents;  $\mathcal{S}$  is the state space comprehensively describing the environment;  $\mathcal{O}_i$  is the observation space of agent  $i$ ’s state;  $\mathcal{A}_i$  is the action space of agent  $i$ ;  $\mathcal{P}$  represents the state transition function  $\mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times \dots \times \mathcal{A}_N \rightarrow \mathcal{S}$ ;  $\mathcal{R}_i$  is the reward function of agent  $i$ ; Finally,  $\gamma$  is the discount factor.

In this problem, the goal for each agent  $i \in \mathcal{I}$  is to optimize a policy  $\pi_i$  to maximize the expected sum of all the agents’ returns in the focal population  $\mathcal{F} \subseteq \mathcal{I}$ :

$$\max_{\pi_1, \pi_2, \dots, \pi_N} \mathbb{E}_{\pi_1, \pi_2, \dots, \pi_N} \left[ \sum_{i \in \mathcal{F}} R_i \right] \quad (1)$$

The policy  $\pi_i(O_i, \{M_j\}_{j \in \mathcal{I}}) \rightarrow \mathcal{A}_i$  maps the observation of agent  $i$  and received messages  $\{M_j\}_{j \in \mathcal{I}}$  to its action space  $\mathcal{A}_i = \langle \mathcal{M}_i, \mathcal{C}_i \rangle$ , where  $\mathcal{M}_i$  is the message generation space, and  $\mathcal{C}_i$  is the control space which includes dimensions for throttle, brake, and steering inputs. The generated message  $M_i$  by agent  $i$  at time step  $t$  is broadcast to all other connected agents within a specific communication range (*Radius*) at the next time step  $t + 1$ .

In summary, the essence of the *talking vehicles* problem is to enable each agent to derive effective control-communication strategies from its observations and the messages it receives. These strategies, coupled with appropriate vehicle control actions, aim to achieve coordinated driving behavior.

This problem presents the following technical challenges to the machine-learning community:

1. How can learned agents understand the situation and generate meaningful messages to help others perceive the environment and potentially negotiate about plans;
2. How can learned agents comprehend and analyze the received messages and generate reasonable high-level driving plans as well as low-level controls optimizing the social welfare;

In this work, we assume that the agents could communicate truthfully in real-time, meaning they accurately convey their real intentions in a timely fashion and follow through on their stated decisions. However, since they act simultaneously, other agents will only be able to process the received message in the next decision step. Additionally, we assume that all agents aim to cooperate with the focal population and do not send deceptive messages or act aggressively to intentionally sabotage the driving goals of others. Exploring scenarios where these assumptions are lifted could be an interesting direction for future work.

## 4 METHOD

The core technical problem that we address is to let agents communicate purposely to facilitate cooperation and act correspondingly using human language as the medium. Training language policies to perform particular communication tasks by self-play is known to generate non-human comprehensible languages. While Bakhtin et al. (2022) imitates human communication strategy through a large dataset and filters the message through values, they require a planning or reinforcement learning agent specifically trained for this game. To establish an initial solution to the *talking vehicles* problem, we start with large language models (LLMs) as a foundational prior for autonomous agents to engage in human-like communication, regularizing the message within natural language space, allowing agents to interpret messages, and making driving decisions accordingly. A key challenge of using LLMs lies in that they are not specifically trained for driving tasks. To address this, we propose a method, LLM+DEBRIEF, which introduces feedback loops that enable LLMs to improve multi-agent coordination in driving scenarios interactively with confederate agents through trial and error. In particular, just like humans debrief after a *Werewolf* game, we enable language models to debrief to reinforce their strategies after each interaction episode. As demonstrated by Figure 2, our method consists of three main components: **In-episode communication**, **Chain-of-Thought Reasoning**, and **Post-Episode Debriefing**. Please refer to Appendix A for a detailed algorithmic description of the method (see Algorithm 1).

**In-episode Communication.** Each driving agent is equipped with a transceiver module that enables real-time communication during episodes. Agents broadcast and receive structured messages by subscribing to topic-specific communication channels. Each message follows a predefined format containing key details such as content, timestamp, sender ID, and sender location. This structured approach ensures that messages are contextually relevant



270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323

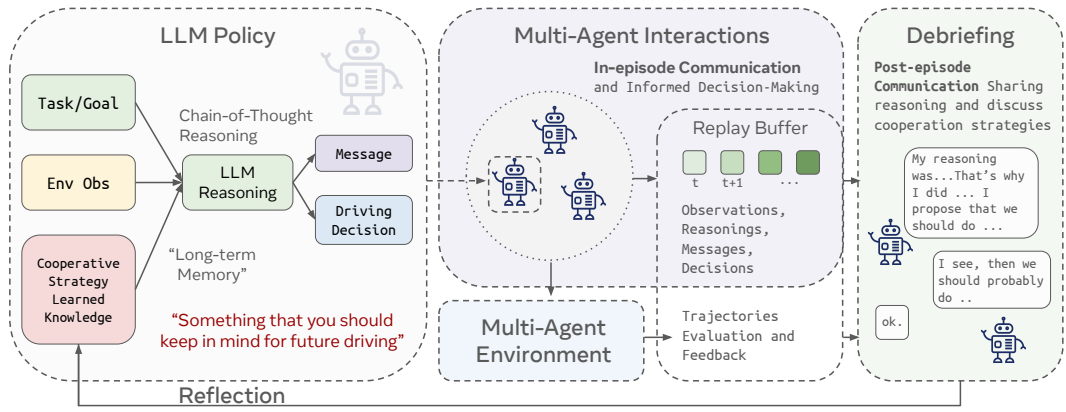


Figure 2: Method. A LLM+DEBRIEF policy is provided a task to complete for each scenario, and the environment will provide the text observation of the surroundings and message dialogues. Along with the previous experience, the policy first performs chain-of-thought reasoning about all the inputs, generates messages to others, and drives decisions based on observation and in-episode communication. Then the agents within the multi-agent environment will perform the decisions made by their corresponding policies. After each episode, agents receive feedback from the environment evaluation containing information like timeout, success, or collision information. This feedback, along with the {observation, reasoning, message, and commands}, are stored in a replay buffer for future learning. During the debriefing phase, agents are able to revise their strategy and knowledge about decision-making by learning others’ reasoning during and after episodes. Such knowledge is then stored as a long-term experience for future decision-making.

and easily interpretable within multi-agent communication. Received messages are stored in a buffer, and recent message dialogs are incorporated into the agent’s observations for decision-making.

**Chain-of-Thought Reasoning.** Ye et al. (2024) observed that current LLMs can make irreversible mistakes when computing variables without sufficient context. To address this, we prompt the LLM to first reason about the environment based on its task, observations, received messages, cooperative strategy, and accumulated knowledge before making decisions. After reasoning, the LLM generates actions in a structured JSON format with keys: "reasoning", "command", and "message". These outputs are then translated into vehicle controls and communication messages to publish. All observations, commands, messages, and reasoning are stored in a replay buffer for further learning and refinement.

**Post-Episode Debriefing.** After each episode, the learning agents replay their experiences, similar to how humans debrief after a game to adapt their strategies. During this process, agents share how they have interpreted messages, made driving decisions, and assess the effectiveness of their actions. This structured post-episode analysis mirrors the Centralized Training Decentralized Execution (CTDE) framework commonly used in multi-agent learning.

When an episode concludes, the environment evaluates the performance of the agents in the focal group and provides rich semantic feedback, such as “Vehicle 109 collided with Vehicle 110” or “Time out because you failed to help Vehicle 109 complete its task.” Each learning data point is retrospectively labeled with other agents’ responses and the episode’s outcomes for further learning. Each learning data point is retrospectively labeled with other agents’ responses and the episode’s outcomes for further learning.

The post-episode discussion is conducted in a turn-based manner over  $N$  rounds. In each debrief, one agent samples a batch of its learning data and proposes a cooperation strategy, while the other agents either refine or adjust this strategy based on their own experiences.

After debriefing, agents gain a clearer understanding of one another and summarize the discussion to form cooperative strategies and individual knowledge for future driving tasks. Chain-of-thought reasoning is applied during debriefing to reinforce decision-making processes.

#### 4.1 IMPLEMENTATION DETAILS

The complete algorithm is presented in Algorithm 1. We use Llama-3-8B-Instruct (Dubey et al., 2024) at a temperature of 0.2, collecting experiences every 0.5 seconds (10 simulation frames) and continuously updating the knowledge base for prompting. The received message dialog is maintained within 2 seconds of message age. The debriefing process spans 2 rounds, followed by a final individual reflection step to summarize the discussion. A batch size of 2 is used to sample transition data from the trajectory. In some scenarios, initial frames are skipped to speed up the simulation. The experiments were conducted on two Nvidia A40 40GB GPUs to run the LLM policies and manage the server. Additional details can be found in Appendix A and the anonymous code repository.

### 5 ENVIRONMENT

To provide concrete and typical driving scenarios that expose the *talking vehicles* challenge, we have developed a simulation environment, **TalkingVehiclesGym**, which is a multi-agent gymnasium environment for closed-loop evaluation of urban driving policies. TalkingVehiclesGym provides a flexible configuration of heterogeneous agents (such as language agents, sensory agents, human agents, behavior agents, etc.) and policies in the environment. This framework also enables **in-episode** communication capabilities of agents using a realistically simulated communication protocol **MQTT** and the dynamic simulation is built upon CARLA (Dosovitskiy et al., 2017), a high-fidelity urban driving simulator. A wide array of sensors are also available for models that handle corresponding modalities. To simplify environmental perception for language-only models, TalkingVehiclesGym is equipped with a **partially-observable** captioner that translates the observation into text for an agent but preserves the partial observability of line-of-sight sensors.



Figure 3: *Overview of Test Scenarios.* Agent roles are marked with circles with different colors. **Red:** Potential Colliders; **Green:** Focal Agents, agents having both driving control and communication capabilities; **Blue:** Other Cooperative Agents, agents that can communicate to help the focal agents. Detailed descriptions of environment dynamics are in Table 3, Appendix B.

TalkingVehiclesGym has been set up with several accident-prone scenarios where multi-agent communication could be advantageous (Figure 3). Scenarios labeled with **Cooperative Perception** represent opportunities for agents to benefit from shared information about areas outside their immediate line of sight. On the other hand, scenarios tagged with **Negotiation** are designed to demonstrate the advantages of agents discussing and reconciling their plans when conflicts arise. For a detailed description of the design structure of TalkingVehiclesGym, and scenario descriptions (Table 3), please refer to Appendix B.

**Atomic Actions.** Since LLMs take considerable time to generate reasoning and decisions, and are not highly accurate with numerical tasks, it is currently impractical to use them for controlling low-level vehicular motions. Instead, this paper focuses on high-level decision-making with natural language communications. Our multi-agent communication and simulation framework allows models to consider only high-level plans, abstracting trajectories into a sequence of atomic actions such as Go, Stop, Slow Down, Speed Up, Change to Left Lane, Change to Right Lane, Turn Left, Turn Right.

---

## 6 EXPERIMENTS

In this section, we aim to verify the following hypotheses with empirical experiments:

1. LLM agents can perform better with communication than without; LLM agents can effectively respond to natural language messages without training and cooperate with each other through communication to improve their performance;
2. LLM agents can autonomously improve themselves in the collaboration tasks through decentralized reflection without human intervention;
3. Debriefing can further enhance LLM learning from interactions more effectively than decentralized reflection alone;

**Metrics.** In each scenario, we define a focal population whose movements we are interested in and can control to accomplish a specific task within a certain time limit. Evaluation metrics are then established based on the performance of this **focal** population over 30 evaluation episodes. We utilize three key metrics: the average total reward (**R**) accrued by the focal population, the average number of collisions per episode (**CR**) of the population, and the average episodic success rate (**SR**) normalized by the size of the population. Success is defined as reaching target locations within a designated time frame without collision. An agent who successfully completes the task earns a reward of +1. Conversely, collision incurs a penalty of  $-1$  for each agent involved in the collision while remaining stagnant at any point until timeout results in a reward of 0 because, although not ideal, conservative policies are at least safe.

**Baselines.** We established several baselines and scenarios to evaluate our hypothesis. These baselines include: (1) an untrained LLM, (2) an LLM trained with decentralized reflection that updates in-context knowledge (LLM+Reflection), (3) an LLM that corrects past actions via decentralized reflection, storing these corrections in a vector-based, retrievable memory and uses few-shot retrieval augmented generation (LLM+Reflection+RAG), and (4) an LLM trained with debrief discussions as outlined in Section 4 (LLM+Debrief). The retrieval augmented method without communication adapts DiLU (Wen et al., 2023a), a non-communicating single-agent LLM-based approach that drives via reflection, to our environment. The multi-agent communication extension of DiLU, AgentsCoDriver (Hu et al., 2024), resembles the Reflection+RAG (Comm) method, but they do not actively optimize the messages. For a fair comparison across DiLU, AgentsCoDriver, and other baselines, we do not initialize the knowledge with human data, nor is there human involvement during the learning process. Additionally, we include Coopernaut (Cui et al., 2022), a LiDAR-based cooperative driving method, as a reference for cooperative perception. Note that since Coopernaut relies on intermediate sensor data representations rather than natural language communication, its results are not directly comparable to the other methods being compared.

**Experiment Setup.** For each baseline<sup>2</sup>, we consider two settings labeled as “Silent” and “Comm”, respectively. In the “Silent” setting, the method operates without communication, where policies focus solely on controlling the vehicle without generating messages. In contrast, the “Comm” setting allows the method to either generate messages alone or both messages and driving commands. For each LLM-based learning method, we train the models for up to 30 episodes per scenario, with early stopping if the scenario is solved, indicated by 10 consecutive successful episodes. After training, we evaluate each method over 30 episodes and report the average performance across these evaluations.

### 6.1 QUANTITATIVE RESULT

**LLMs can facilitate cooperation through language communication in zero-shot.** Table 1 presents our evaluation of LLM agents without training across all scenarios under both the “Comm” and “Silent” settings. We observe that, even without learning, LLMs are able

---

<sup>2</sup>Except for LLM+Debrief, which is only tested under the “Comm” setting since it is particularly designed for improving multi-agent communication.



Table 1: Experiment Results for Communication vs Silent Agents. We evaluate on adversarial cases where being aggressive or conservative will result in failure.

Scenario			Overtake (Perception)			Red Light			Left Turn		
Method	LLM	Comm	R ↑	CR ↓	SR ↑	R ↑	CR ↓	SR ↑	R ↑	CR ↓	SR ↑
Untrained (Silent)	Yes	No	0.00	0.00	0.0	-0.60	0.80	20.0	0.20	0.33	53.3
Untrained (Comm)	Yes	Yes	-0.63	0.66	33.3	0.80	0.07	86.7	0.20	0.33	56.7
Coopernaut	No	Yes	1.00	0.00	100.0	0.97	0.00	96.7	0.93	0.03	96.7

Scenario			Overtake (Negotiation)			Highway Exit			Highway Merge		
Method	LLM	Comm	R ↑	CR ↓	SR ↑	R ↑	CR ↓	SR ↑	R ↑	CR ↓	SR ↑
Untrained (Silent)	Yes	No	0.50	0.67	3.3	-0.93	1.5	20.0	-1.26	1.63	18.5
Untrained (Comm)	Yes	Yes	0.50	0.70	53.3	-0.73	0.6	33.3	-0.10	1.03	45.5

to leverage communication to foster some levels of cooperation in most scenarios, indicated by higher success rates when addressing conflict or partial observation challenges. However, there remains substantial room for improvement. Interestingly, while communication enables cooperation, it also tends to increase the frequency of collisions compared to the Silent setting. We hypothesize that it is because LLMs become overly confident in their perceptions or behave more aggressively in driving tasks. In contrast, without communication, LLMs often adopt overly conservative policies, particularly in scenarios like Overtake (Perception). The typical message length generated by LLMs ranges from 0 to 50 words, requiring less than 0.01 Mbps, a stark contrast to the 5.1 Mbps reported in Coopernaut Cui et al. (2022), highlighting the efficiency of using natural language as the communication protocol, especially in negotiation tasks.

Table 2: Experiment Results for Improvement Methods.

Scenario			Red Light			Highway Merge		
Method	LLM	Comm	R ↑	CR ↓	SR ↑	R ↑	CR ↓	SR ↑
Untrained	Yes	No	-0.6	0.80	20.0	-1.26	1.63	18.5
+Reflection	Yes	No	-0.73	0.86	13.3	-0.86	1.43	28.5
+Reflection+RAG	Yes	No	-1.00	1.00	0.00	-2.00	2.00	0.0
Untrained	Yes	Yes	0.80	0.07	86.7	-0.10	1.03	45.5
+Reflection	Yes	Yes	0.70	0.13	83.3	0.20	0.87	50.0
+Reflection+RAG	Yes	Yes	-0.93	0.96	3.3	-2.00	2.00	0.0
+Debrief	Yes	Yes	0.80	0.07	<b>90.0</b>	<b>0.40</b>	<b>0.57</b>	<b>51.5</b>

**LLMs can be further improved through reflection and debriefing.** Table 2 evaluates different training methods in Red Light (Perception) and Highway Merge (Negotiation) scenarios. We found that LLMs’ performance in negotiation tasks improves with reflection, but incorporating in-context knowledge updates and revising them with new experiences proves more reliable than LLMs’ self-correcting actions without human oversight.

We hypothesize that the failure of Retrieval Augmented Generation (RAG) methods in our environment stems from the complexity of generating accurate messages—the search space is too large for language models to easily correct without additional message validation. Improvements from reflection were less pronounced in perception tasks but significantly enhanced performance in negotiation tasks, where reflection benefited both silent and communication settings. The LLM+Debrief method achieved the best performance overall, underscoring the potential of collective discussion in improving cooperation.

## 6.2 QUALITATIVE ANALYSIS

While the main cooperation mode in perception tasks is sharing critical or abnormal traffic information, the cooperation mode in negotiation mainly lies in the argument on road priority. Interestingly, we found that LLMs can form a convention through in-episode communication and demonstrate diverse conventions across episodes. For example, in the evaluation of the LLM+Debrief model, we found that LLMs could give way to vehicles on the main highway,

486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539

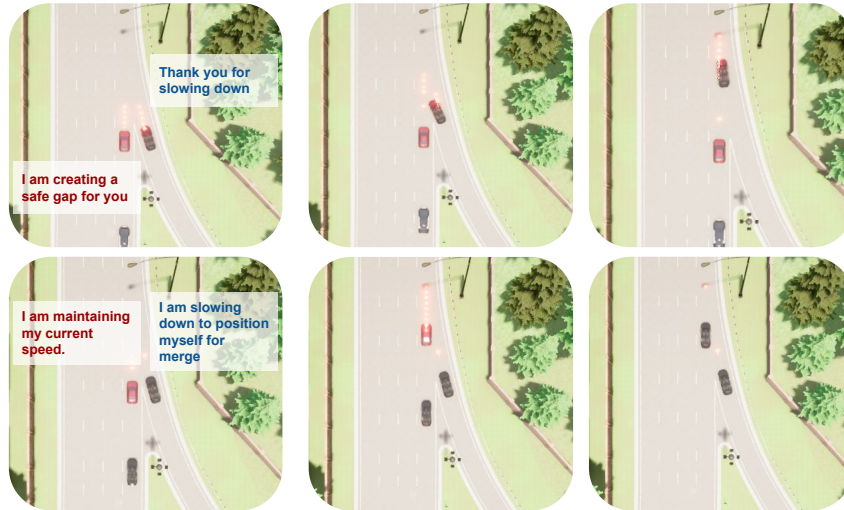


Figure 4: Diverse cooperation modes in the Highway Merge negotiation scenario. In this scenario, both vehicles aim to merge quickly as they are in a hurry. **Top:** The merging vehicle successfully negotiates for the highway vehicles to yield. **Bottom:** The merging vehicle opts to wait for the highway vehicle to pass before merging.

while sometimes successfully convince the highway cars to slow down for them, demonstrated in Figure 4. We provided videos demonstrating messages and reasoning process along with the paper in the supplementary materials.

In Red Light Violation scenario, we compare the generated cooperative strategy and knowledge for future driving from reflection and debriefing. We found that the debriefed knowledge and cooperative strategies are more comprehensive and proactive than the decentralized reflection knowledge. Details in Appendix D.

## 7 CONCLUSION AND FUTURE WORK

In summary, we identify a novel application domain for large language models in multi-agent learning systems, *talking vehicles*, where agents are required to send natural language messages and understand natural language messages to incorporate them into driving plans. Solutions to the *talking vehicles* problem have the potential to enable autonomous agents to facilitate cooperative perception and negotiation with human drivers. As a first attempt to solve the *talking vehicles* challenge, we propose a new method, LLM+DEBRIEF, for generating messages and comprehending received messages. Our experiments show the effectiveness of LLM+DEBRIEF quantitatively and qualitatively.

**Limitations and Future Work.** While we provide initial evidence of LLM+DEBRIEF’s potential in the *talking vehicles* problem, this research opens up several exciting future research areas for further exploration and development. First, the current LLM+DEBRIEF framework takes text description as observations, which relies on an idealized perception system. The TalkingVehiclesGym environment is able to provide multi-modal sensor observations. In future work, we are interested in developing a multi-modal extension of LLM+DEBRIEF, which allows end-to-end perception and reasoning over the rich context information embedded in multi-modal observations. Second, this paper reports on a successful proof-of-concept, we are interested in scaling the evaluation benchmark and solution to more diverse traffic scenarios and operation conditions resembling real-world V2V communication, e.g., subject to time delays, adversarial attacks, and limited bandwidths. Last, our framework opens up the exciting potential to create a cooperative driving system for mixed-autonomy traffic scenarios. We are interested in realizing this potential, by studying the *talking vehicles* problem with human-in-the-loop experiments and exploring framework design to enable efficient communication between autonomous vehicles and human drivers.

540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593

---

## REFERENCES

- John P Agapiou, Alexander Sasha Vezhnevets, Edgar A Duéñez-Guzmán, Jayd Matyas, Yiran Mao, Peter Sunehag, Raphael Köster, Udari Madhushani, Kavya Kopparapu, Ramona Comanescu, et al. Melting pot 2.0. *arXiv preprint arXiv:2211.13746*, 2022.
- Shunsuke Aoki, Takamasa Higuchi, and Onur Altintas. Cooperative perception with deep reinforcement learning for connected vehicles. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pp. 328–334. IEEE, 2020.
- Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, et al. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624): 1067–1074, 2022.
- Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.
- Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.
- Qi Chen, Sihai Tang, Qing Yang, and Song Fu. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 514–524. IEEE, 2019.
- Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, and Ziran Wang. Receive, reason, and react: Drive as you say with large language models in autonomous vehicles. *arXiv preprint arXiv:2310.08034*, 2023.
- Jiaxun Cui, Hang Qiu, Dian Chen, Peter Stone, and Yuke Zhu. Coopernaut: end-to-end driving with cooperative perception for networked vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17252–17262, 2022.
- Xinpeng Ding, Jianhua Han, Hang Xu, Wei Zhang, and Xiaomeng Li. Hilm-d: Towards high-resolution understanding in multimodal large language models for autonomous driving. *arXiv preprint arXiv:2309.05186*, 2023.
- Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pp. 1–16. PMLR, 2017.
- Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In *arXiv preprint arXiv:2303.03378*, 2023.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- Senkang Hu, Zhengru Fang, Zihan Fang, Xianhao Chen, and Yuguang Fang. Agentscodriver: Large language model empowered collaborative driving with lifelong learning. *arXiv preprint arXiv:2404.06345*, 2024.
- Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 35:4874–4886, 2022.

- 
- 594 Ye Jin, Xiaoxi Shen, Huiling Peng, Xiaohan Liu, Jingli Qin, Jiayang Li, Jintao Xie, Peizhong  
595 Gao, Guyue Zhou, and Jiangtao Gong. Surrealdriver: Designing generative driver agent  
596 simulation framework in urban contexts based on large language model. *arXiv preprint*  
597 *arXiv:2309.13193*, 2023.
- 598 Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. Textual  
599 explanations for self-driving vehicles. *Proceedings of the European Conference on Computer*  
600 *Vision (ECCV)*, 2018.
- 601 Jinkyu Kim, Teruhisa Misu, Yi-Ting Chen, Ashish Tawari, and John Canny. Grounding  
602 human-to-vehicle advice for self-driving vehicles. In *The IEEE Conference on Computer*  
603 *Vision and Pattern Recognition (CVPR)*, 2019.
- 604 Zixing Lei, Shunli Ren, Yue Hu, Wenjun Zhang, and Siheng Chen. Latency-aware collabora-  
605 tive perception. In *European Conference on Computer Vision*, pp. 316–332. Springer,  
606 2022.
- 607 Yiming Li, Dekun Ma, Ziyang An, Zixun Wang, Yiqi Zhong, Siheng Chen, and Chen Feng.  
608 V2x-sim: Multi-agent collaborative perception dataset and benchmark for autonomous  
609 driving. *IEEE Robotics and Automation Letters*, 7(4):10914–10921, 2022.
- 610 Jonathan Light, Min Cai, Sheng Shen, and Ziniu Hu. Avalonbench: Evaluating llms playing  
611 the game of avalon. *arXiv e-prints*, pp. arXiv–2310, 2023.
- 612 Yingzi Ma, Yulong Cao, Jiachen Sun, Marco Pavone, and Chaowei Xiao. Dolphins: Multi-  
613 modal language model for driving, 2023.
- 614 Srikanth Malla, Chiho Choi, Isht Dwivedi, Joon Hee Choi, and Jiachen Li. Drama: Joint  
615 risk localization and captioning in driving. In *Proceedings of the IEEE/CVF Winter*  
616 *Conference on Applications of Computer Vision*, pp. 1043–1052, 2023.
- 617 Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang,  
618 Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, et al. One million scenes for autonomous  
619 driving: Once dataset. *arXiv preprint arXiv:2106.11037*, 2021.
- 620 Jiageng Mao, Yuxi Qian, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with  
621 gpt. *arXiv preprint arXiv:2310.01415*, 2023a.
- 622 Jiageng Mao, Junjie Ye, Yuxi Qian, Marco Pavone, and Yue Wang. A language agent for  
623 autonomous driving. 2023b.
- 624 Ming Nie, Renyuan Peng, Chunwei Wang, Xinyue Cai, Jianhua Han, Hang Xu, and Li Zhang.  
625 Reason2drive: Towards interpretable and chain-based reasoning for autonomous driving.  
626 *arXiv preprint arXiv:2312.03661*, 2023.
- 627 Joon Sung Park, Joseph O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang,  
628 and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior.  
629 In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and*  
630 *Technology*, pp. 1–22, 2023.
- 631 Tianwen Qian, Jingjing Chen, Linhai Zhuo, Yang Jiao, and Yu-Gang Jiang. Nuscenes-qa: A  
632 multi-modal visual question answering benchmark for autonomous driving scenario. *arXiv*  
633 *preprint arXiv:2305.14836*, 2023.
- 634 Hang Qiu, Pohan Huang, Namjoon Asavisanu, Xiaochen Liu, Konstantinos Psounis, and Ramesh  
635 Govindan. Autocast: Scalable infrastructure-less cooperative perception for distributed  
636 collaborative driving. In *Proceedings of the 20th Annual International Conference on*  
637 *Mobile Systems, Applications, and Services, MobiSys ’22*, 2022.
- 638 Nicholas Rhinehart, Jeff He, Charles Packer, Matthew A Wright, Rowan McAllister, Joseph E  
639 Gonzalez, and Sergey Levine. Contingencies from observations: Tractable contingency  
640 planning with learned behavior models. In *2021 IEEE International Conference on Robotics*  
641 *and Automation (ICRA)*, pp. 13663–13669. IEEE, 2021.

---

648 Ari Seff, Brian Cera, Dian Chen, Mason Ng, Aurick Zhou, Nigamaa Nayakanti, Khaled S  
649 Refaat, Rami Al-Rfou, and Benjamin Sapp. Motionlm: Multi-agent motion forecasting  
650 as language modeling. In *Proceedings of the IEEE/CVF International Conference on*  
651 *Computer Vision*, pp. 8579–8590, 2023.

652 Hao Sha, Yao Mu, Yuxuan Jiang, Li Chen, Chenfeng Xu, Ping Luo, Shengbo Eben Li,  
653 Masayoshi Tomizuka, Wei Zhan, and Mingyu Ding. Languagempc: Large language models  
654 as decision makers for autonomous driving. *arXiv preprint arXiv:2310.03026*, 2023.

655 Hao Shao, Yuxuan Hu, Letian Wang, Steven L Waslander, Yu Liu, and Hongsheng Li.  
656 Lmdrive: Closed-loop end-to-end driving with large language models. *arXiv preprint*  
657 *arXiv:2312.07488*, 2023.

659 Zijing Shi, Meng Fang, Shunfeng Zheng, Shilong Deng, Ling Chen, and Yali Du. Cooperation  
660 on the fly: Exploring language agents for ad hoc teamwork in the avalon game. *arXiv*  
661 *preprint arXiv:2312.17515*, 2023.

662 Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Ping  
663 Luo, Andreas Geiger, and Hongyang Li. Drivelm: Driving with graph visual question  
664 answering. *arXiv preprint arXiv:2312.14150*, 2023.

665 Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul  
666 Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception  
667 for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference*  
668 *on computer vision and pattern recognition*, pp. 2446–2454, 2020.

670 James Tu, Tsunhsuan Wang, Jingkang Wang, Sivabalan Manivasagam, Mengye Ren, and  
671 Raquel Urtasun. Adversarial attacks on multi-agent communication. In *Proceedings of the*  
672 *IEEE/CVF International Conference on Computer Vision*, pp. 7768–7777, 2021.

673 Alexander Sasha Vezhnevets, John P Agapiou, Avia Aharon, Ron Ziv, Jayd Matyas, Edgar A  
674 Duéñez-Guzmán, William A Cunningham, Simon Osindero, Danny Karmon, and Joel Z  
675 Leibo. Generative agent-based modeling with actions grounded in physical, social, or  
676 digital space using concordia. *arXiv preprint arXiv:2312.03664*, 2023.

677 Tsun-Hsuan Wang, Sivabalan Manivasagam, Ming Liang, Bin Yang, Wenyuan Zeng, and  
678 Raquel Urtasun. V2vnet: Vehicle-to-vehicle communication for joint perception and  
679 prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK,*  
680 *August 23–28, 2020, Proceedings, Part II 16*, pp. 605–621. Springer, 2020.

681 Wayve. Lingo-1: Exploring natural language for autonomous driving. 2023. URL <https://wayve.ai/thinking/lingo-natural-language-autonomous-driving/>.

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

685 Licheng Wen, Xuemeng Yang, Daocheng Fu, Xiaofeng Wang, Pinlong Cai, Xin Li, Tao  
686 Ma, Yingxuan Li, Linran Xu, Dengke Shang, et al. On the road with gpt-4v (ision):  
687 Early explorations of visual-language model on autonomous driving. *arXiv preprint*  
688 *arXiv:2311.05332*, 2023b.

689 Xiyang Wu, Rohan Chandra, Tianrui Guan, Amrit Bedi, and Dinesh Manocha. Intent-aware  
690 planning in heterogeneous traffic via distributed multi-agent reinforcement learning. In  
691 *7th Annual Conference on Robot Learning*, 2023.

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

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



---

702 Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang  
703 Liu. Exploring large language models for communication games: An empirical study on  
704 werewolf. *arXiv preprint arXiv:2309.04658*, 2023a.

705  
706 Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kenneth KY Wong, Zhenguo  
707 Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via  
708 large language model. *arXiv preprint arXiv:2310.01412*, 2023b.

709 Senqiao Yang, Jiaming Liu, Ray Zhang, Mingjie Pan, Zoey Guo, Xiaoqi Li, Zehui Chen,  
710 Peng Gao, Yandong Guo, and Shanghang Zhang. Lidar-llm: Exploring the potential of  
711 large language models for 3d lidar understanding. *arXiv preprint arXiv:2312.14074*, 2023.

712  
713 Tian Ye, Zicheng Xu, Yuanzhi Li, and Zeyuan Allen-Zhu. Physics of language models: Part  
714 2.1, grade-school math and the hidden reasoning process. *arXiv preprint arXiv:2407.20311*,  
715 2024.

716 Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo,  
717 Hanyu Li, Xing Hu, Jirui Yuan, and Zaiqing Nie. Dair-v2x: A large-scale dataset for  
718 vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF*  
719 *Conference on Computer Vision and Pattern Recognition*, pp. 21361–21370, 2022.

720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755

756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809

---

## A METHOD

The algorithm implements LLM+DEBRIEF, a centralized multi-agent learning framework that leverages communication and reflection using large language models (LLMs) to enhance coordination between agents in a simulated environment.

Messages exchanged between agents during the simulation are incorporated into their observations, allowing the agents to adapt their strategies continuously. After completing each episode, the environment provides feedback, which is used to label and process the experiences in the replay buffer for further learning. This feedback helps the agents refine their knowledge and improve their decision-making in subsequent episodes.

The debriefing process plays a crucial role in this algorithm. After each episode, agents engage in multiple rounds of debriefing, where they propose or revise cooperative strategies based on their experiences and interactions. The order of debriefing is randomized to simulate natural dialogues, enhancing the realism of the communication. Once the debriefing rounds are complete, the agents reflect individually, summarizing the discussions and updating their knowledge bases. This reflection step is critical for improving future performance, enabling agents to learn from successes and failures.

At the conclusion of the training process, the agents' knowledge and policies are updated, with the final policies from the last self-play iteration being used for further evaluations. The entire process is designed to improve the agents' ability to communicate effectively and make informed decisions in a multi-agent setting.

The agents use `Llama-3-8B-Instruct` to generate and interpret messages, with a temperature setting of 0.2 to ensure more deterministic outputs. The environment updates every 0.5 seconds (equivalent to 10 simulation frames), and the agents' messages are considered relevant for up to 2 seconds, ensuring timely and efficient communication. The experiments were conducted on two Nvidia A40 40GB GPUs, which were used to manage both the LLM-based policies and the simulation environment. This setup allowed the agents to run their LLM-based decision-making processes in parallel, enhancing the scalability of the system and enabling more efficient training.

810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863

---

**Algorithm 1** Multi-Agent Centralized Debrief Reflection with Communication

---

**Input:** Multi-agent Simulation Environment `env`, LLM agents  $\{\pi_{i \in \mathcal{I}}\}$ , Debriefing round  $R$ .  
**Initialize:** Knowledge  $\{K_{i \in \mathcal{I}}\}$ , Replay Buffer `ReplayBuffer`  
**for**  $j=1, 2, 3, \dots$  // Training epoch **do**  
   $\{\text{obs}_i\} = \text{env.reset}()$   
  **while**  $t < T$  // Time step **do**  
    **for**  $i=1, \dots, N$  //Per agent, but execute in parallel **do**  
      // Get CoT reasoning for each agent based on observation and knowledge  
       $\text{reasoning}_i \leftarrow \text{agents.reason}(\text{obs}_i, K_i)$   
      // Get decisions for each agent based on observation and knowledge  
       $\text{message}_i, \text{control}_i \leftarrow \text{agents.act}(\text{obs}, K_i, \text{reasoning}_i)$   
    **end for**  
    // Step the environment with actions  
     $\{\text{next\_obs}_i\} \leftarrow \text{env.step}(\{\text{message}_i, \text{control}_i\})$   
    // Store experience to the replay buffer  
    `ReplayBuffer.add`( $\text{obs}$ ,  $\text{next\_obs}$ ,  $\text{reasonings}$ ,  $\text{messages}$ )  
    // Message Dialog becomes part of the observation  
     $\{\text{obs}_i\} \leftarrow \{\text{next\_obs}_i\} \cup \{\text{message}_i\}$   
  **end while**  
  // Get episode feedback from the environment  
   $\text{feedback} \leftarrow \text{env.evaluate}()$   
  // Label all the transition data in hindsight  
  `data_post_processing`(`ReplayBuffer`)  
  // Debriefing and learning from feedback, update knowledge  
  // Randomly decide debrief order  
  **for**  $r=1, \dots, R$  **do**  
    **if**  $\text{strategy} = \text{None}$  **then**  
       $\text{cooperation\_strategy} = \text{agent}_r.\text{propose}()$   
    **else**  
       $\text{cooperation\_strategy} = \text{agent}_r.\text{revise}()$   
    **end if**  
  **end for**  
  Summarize the dialogue and use it for future learning  
   $\{K_i\} \leftarrow \text{agent.reflect}()(\{K_i\},$   
**end for**  
last  $\{\pi_{i,j}\}$  during the last iteration of self-play

---

## B ENVIRONMENT

The environment follows gymnasium and PettingZoo API, assuming a parallel-acting environment because we leverage this feature to parallel language model inference. We made significant changes to support multi-agent communication and heterogeneous agent configuration in CARLA. TalkingVehiclesGym wraps around the CARLA server and the client to set up agents as a bridge between the simulator and learning agent policies that are able to learn from replay buffers. Language Communication Agents can the MQTT-based transceiver we implemented, and the communication is among Agents instead of going through server.

### Talking Vehicles Gym

A multi-agent, gym-like (pettingzoo), high-fidelity, communication-supporting, scenario-based environment

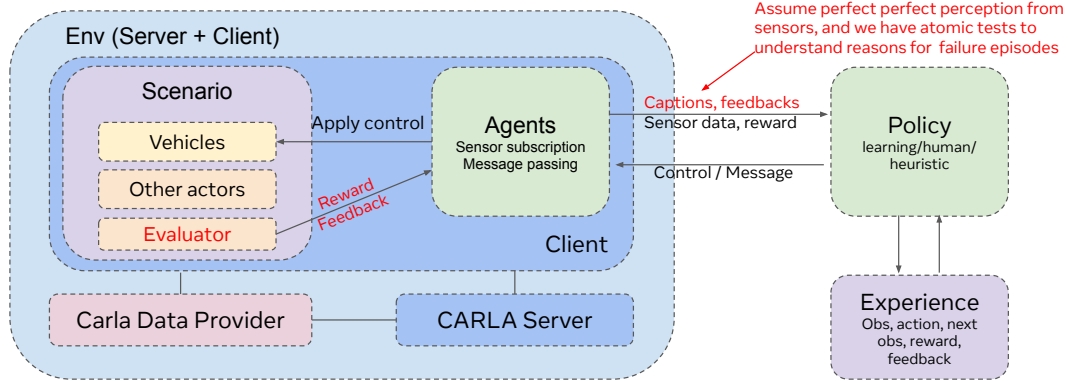


Figure 5: TalkingVehiclesGym

Table 3: Example Scenarios. Here we describe the fundamental composition of each accident-prone scenario, where the background agents can be configured in terms of density, controlling policies, and communication capabilities.

Interaction Type	Scenario Name	Description
Cooperative Perception	Overtake	A vehicle plans to overtake a broken and stopped truck by moving into the opposite lane. The truck can still communicate but the opposite-going car can not.
	Left Turn	A vehicle tries to turn left on a left-turn yield light when a truck is blocking the view of the opposite lane. The truck is able to communicate.
	Red Light Violation	A vehicle is crossing the intersection when there is another vehicle running the red light. Lidar fails to sense the other vehicle because of the lined-up vehicles waiting for a left turn, one of those cars being able to communicate.
Negotiation	Overtake	A vehicle is going to borrow the opposite lane to overtake a stopped truck. The truck is not able to connect, but an opposite-going car is able to communicate.
	Highway Merge	A vehicle is going to merge onto the highway but the target lane has continuous traffic flows. A vehicle on that lane is able to communicate and alter plans.
	Highway Exit	A vehicle is going to exit the highway but it needs to cross lanes where there is a traffic flow. A vehicle in the flow is able to communicate and alter plans.

918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971

## C PROMPTS

This figure serves as a demonstration of the prompts; the prompts that we use are more complex as they are structured in the code.

### System

You are driving a car, and your goal is to accomplish a given <task>. You can coordinate with any other vehicles to avoid collisions and or reduce wait time. I will give you description of the driving situation from your LiDAR perception, but note that it may be partially observable.\n\nThe observations are formatted as:  
Observation: <observation>  
Possible actions are: 1. stop (stop the vehicle) 2. go (follow the planned route)

1. Sign of the Lane ID indicate the direction of the lane.
2. Your received messages are from other vehicles. Be sure to analyze the messages and how that message will affect you.
3. You decide an action every 10 frames (0.5 seconds).
4. Do not switch your action unless it is necessary.

### User

'Observation: You are driving the Vehicle 465, and it is a vehicle.lincoln.mkz\_2020. Your current speed is: 4.26 m/s, and the speed limit is: 8.33 m/s. The traffic light is currently: green. You are at an intersection. Around you, there are other vehicles:  
(1)Vehicle 467 is a vehicle.carlamotors.carlacola, and is stationary. It is at an intersection. It is 3.83 meters behind you and 3.66 meters to your left.  
(2)Vehicle 466 is a vehicle.audi.tt, traveling at speed: 8.11 m/s. It is moving closer to you. It is at an intersection. It is 3.95 meters ahead of you and 2.58 meters to your left.  
Your task is: pass the intersection if the traffic lights are green.  
Analyze the other vehicles in the situation, received messages from other vehicles and how to react or negotiate to accomplish your task safely.

### Agent

Based on the observation, I can analyze the situation as follows:\n\n1. Vehicle 467 (Carlacola) is stationary and is at an intersection, which means it is not a threat to my movement.  
2. Vehicle 466 (Audi TT) is traveling towards me and is at an intersection. It is moving at a speed of 8.11 m/s, which is slightly above the speed limit. This vehicle may pose a threat to my movement if it does not slow down or yield to me.  
Considering the task is to pass the intersection if the traffic lights are green, I should prioritize my safety and the safety of other vehicles.  
Action: 2. go (follow the planned route)  
I will continue to follow the planned route and pass the intersection, but I will also keep an eye on Vehicle 466 and be prepared to adjust my speed or action if necessary to avoid a collision.

### User

You MUST respond with an action, formatted as a JSON object with the following structure:  
command: <command>  
where you replace <command> with your actual command, choosing from: go, stop  
Do nothing else but return the action.

### Agent

```
{"command": "go"}
```



---

## D EXAMPLE LEARNED KNOWLEDGE AND COOPERATIVE STRATEGIES

**Red Light - Debrief - Truck. “knowledge”:** Here is the revised knowledge for myself to keep in mind for future driving: When approaching an intersection, I will prioritize continuous information sharing and mutual awareness with other cooperative vehicles. I will share my intentions and speed with other vehicles, and request information about their status and intentions. I will provide detailed information about approaching vehicles, including their speed and distance, as well as any obstacles or hazards on the road. I will also request and receive similar information from other vehicles, creating a shared understanding of the situation. As a big truck, I will continuously monitor the surrounding vehicles and adjust my actions accordingly, considering the distance and speed of other vehicles, including those that are stationary or slowing down. I will be aware of the potential for occluded vehicles and take steps to mitigate any potential hazards. I will also be prepared to adjust my speed and trajectory in response to changing circumstances, such as a vehicle slowing down or stopping unexpectedly. I will maintain a watchful eye on the intersection, identifying potential hazards or situations that may cause stagnation or inefficiency, and sending messages to other vehicles to address these issues. I will also be mindful of the importance of slowing down and communicating intentions when approaching an intersection, and be prepared to adjust my speed and trajectory in response to changing circumstances. I will also consider the following key principles: \* Prioritize continuous information sharing and mutual awareness with other cooperative vehicles \* Share intentions and speed with other vehicles, and request information about their status and intentions \* Provide detailed information about approaching vehicles, including their speed and distance, as well as any obstacles or hazards on the road \* Request and receive similar information from other vehicles, creating a shared understanding of the situation \* Continuously monitor the surrounding vehicles and adjust actions accordingly \* Be aware of the potential for occluded vehicles and take steps to mitigate any potential hazards \* Be prepared to adjust speed and trajectory in response to changing circumstances \* Identify potential hazards or situations that may cause stagnation or inefficiency, and send messages to other vehicles to address these issues \* Be mindful of the importance of slowing down and communicating intentions when approaching an intersection, and be prepared to adjust speed and trajectory in response to changing circumstances. By following this strategy, I can help ensure a safe and efficient journey for all cooperative vehicles, and adapt to changing circumstances to reduce the risk of accidents and improve overall traffic flow.

**“cooperative knowledge”:** The revised cooperative strategy for all vehicles is as follows: We will focus on sharing critical traffic information to ensure a safe and efficient passage through the intersection. Each vehicle will monitor the surrounding environment and share updates with other cooperative vehicles, including stationary vehicles on adjacent roads and approaching vehicles. We will request information from other vehicles to gain a better understanding of the situation and provide warnings about potential hazards. By sharing information and coordinating our actions, we can work together to avoid accidents and optimize traffic flow. We will send messages to cooperative vehicles indicating the presence of stationary vehicles, approaching vehicles, and potential hazards, and request updates on their speed and direction. Additionally, we will prioritize caution and adjust our speed and trajectory accordingly to ensure a safe passage through the intersection. This strategy will enable us to make informed decisions and adapt to changing situations, ultimately leading to a safer and more efficient intersection experience.

**Red Light - Debrief - Car1. “knowledge”:** Here is the revised knowledge summary for myself to keep in mind for future driving: \* When approaching an intersection, prioritize caution and coordination by continuously monitoring the situation and adjusting speed and actions accordingly to ensure a safe and efficient passage. \* Foster a culture of open communication and mutual awareness by actively listening to and responding to messages from other vehicles, and be prepared to adjust actions accordingly. \* Consider the speed and distance of approaching vehicles and adjust my own speed and actions accordingly to avoid potential conflicts and ensure a safe passage. \* Prioritize coordination and communication when approaching an intersection with multiple vehicles, and be prepared to adjust speed and actions based on the responses of other vehicles. \* When receiving messages from other vehicles, carefully consider the information and adjust actions accordingly to ensure a

---

1026 safe and efficient passage.\* Be prepared to adjust speed and actions based on the traffic  
1027 light status and the presence of other vehicles, and maintain a safe distance to avoid  
1028 potential collisions.\* Prioritize mutual awareness and cooperation by actively monitoring the  
1029 surroundings, including any occluded vehicles, and adjusting speed and trajectory accordingly  
1030 to ensure a safe and efficient passage.\* Continuously monitor the situation and adjust speed  
1031 and actions accordingly to ensure a safe and efficient passage, even when encountering  
1032 unexpected situations or changes in the traffic environment.\* Prioritize flexibility and  
1033 adaptability when navigating intersections, and be prepared to adjust speed and actions  
1034 in response to changing circumstances.\* Consider the potential impact of my actions on  
1035 other vehicles and the intersection as a whole, and adjust speed and actions accordingly  
1036 to ensure a safe and efficient passage.\* Use clear and concise communication to convey my  
1037 intentions and status to other vehicles, and be prepared to adjust my actions based on their  
1038 responses.\* Continuously share information about my current speed, position, and trajectory  
1039 with other cooperative vehicles, and adjust my actions accordingly based on their responses.\*  
1040 Prioritize coordination and communication with other vehicles to ensure a smooth and safe  
1041 passage through intersections, and be prepared to adjust speed and actions based on the  
1042 responses of other vehicles.I will keep these principles in mind for future driving, focusing on  
1043 communication, coordination, and mutual awareness to ensure a safe and efficient passage  
1044 through intersections.

1044 **“cooperative knowledge”**: The revised cooperative strategy for all vehicles is as follows:  
1045 We will focus on sharing critical traffic information to ensure a safe and efficient passage  
1046 through the intersection. Each vehicle will monitor the surrounding environment and share  
1047 updates with other cooperative vehicles, including stationary vehicles on adjacent roads  
1048 and approaching vehicles. We will request information from other vehicles to gain a better  
1049 understanding of the situation and provide warnings about potential hazards. By sharing  
1050 information and coordinating our actions, we can work together to avoid accidents and  
1051 optimize traffic flow. We will send messages to cooperative vehicles indicating the presence  
1052 of stationary vehicles, approaching vehicles, and potential hazards, and request updates on  
1053 their speed and direction. Additionally, we will prioritize caution and adjust our speed and  
1054 trajectory accordingly to ensure a safe passage through the intersection. This strategy will  
1055 enable us to make informed decisions and adapt to changing situations, ultimately leading  
1056 to a safer and more efficient intersection experience.

1057 **Red Light - Reflection - Truck. “knowledge”**: Here is the revised knowledge I can  
1058 keep in mind for future driving: \* When approaching an intersection, be aware of the  
1059 vehicles around you, including stationary and moving vehicles, and consider their speed and  
1060 direction.\* When receiving messages from other vehicles, acknowledge and respond to them  
1061 to maintain a cooperative and safe environment.\* When slowing down or stopping, make  
1062 sure to communicate your actions to other vehicles around you to avoid potential collisions  
1063 or misunderstandings.\* When coordinating with other vehicles, consider their speed and  
1064 direction and suggest slowing down or adjusting course to ensure a safe passage.\* When  
1065 encountering a potential threat or accident-prone situation, prioritize safety and communicate  
1066 with other vehicles to take necessary precautions.\* As a stationary vehicle, focus on sharing  
1067 critical traffic information with other vehicles to help them navigate the intersection safely,  
1068 and prioritize clear and concise communication to avoid misunderstandings.\* Be aware of  
1069 possible occlusions and use sensors to detect occluded vehicles, adjusting actions accordingly.\*  
1070 Prioritize safety and communicate with other vehicles to take necessary precautions in  
1071 potential threat or accident-prone situations.\* When providing guidance to other vehicles,  
1072 consider the intersection layout and suggest a safe path, taking into account the speed and  
1073 direction of other vehicles.\* When receiving guidance from other vehicles, acknowledge and  
1074 follow their instructions to ensure a safe passage.\* As a stationary vehicle, be prepared to  
1075 provide guidance and support to other vehicles, especially those approaching the intersection,  
1076 and prioritize clear and concise communication to avoid misunderstandings. I removed the  
1077 mention of Vehicle ID and revised the knowledge to focus on general principles and strategies  
1078 for safe and cooperative driving.

1078 **Red Light - Reflection - Car1. “knowledge”**: Based on my analysis and proposed  
1079 strategy, I have revised my knowledge summary for future driving as follows:\* When

---

1080 approaching an intersection, prioritize caution and consider the possibility of other vehicles  
1081 approaching or changing their speed.\* Be aware of stationary vehicles around you and adjust  
1082 your speed accordingly to ensure a safe passage.\* When receiving messages from other  
1083 vehicles, take them into consideration and adjust your actions accordingly to ensure a safe  
1084 and smooth passage.\* Always be vigilant and consider potential occlusions that may hide  
1085 other vehicles from your view.\* When following a planned route, be prepared to adjust your  
1086 speed or actions if necessary to ensure a safe and efficient passage.\* When approaching  
1087 a traffic light, be prepared to stop or slow down if necessary, as the light could change  
1088 to red at any moment.\* When driving, always prioritize caution and awareness of your  
1089 surroundings, and be prepared to adjust your actions accordingly to ensure a safe and  
1090 efficient completion of your tasks.\* Consider the possibility of other vehicles changing their  
1091 speed or direction, and adjust your actions accordingly to maintain a safe distance and avoid  
1092 potential collisions.\* Be prepared to adjust your speed or actions in response to changing  
1093 road conditions, such as construction or inclement weather.\* Always maintain a safe distance  
1094 from other vehicles and be prepared to stop or slow down if necessary to avoid a collision.\*  
1095 When approaching a stationary vehicle, be prepared to adjust your speed and trajectory to  
1096 ensure a safe passage, taking into account the vehicle's position and any potential hazards.\*  
1097 When receiving a message from a stationary vehicle, take it into consideration and adjust  
1098 your actions accordingly to ensure a safe and smooth passage.\* When approaching a potential  
1099 collision or hazard, prioritize caution and take immediate action to avoid the situation, such  
1100 as slowing down or changing direction.\* Always keep in mind the potential consequences of  
1101 your actions and be prepared to adapt to changing situations.I have revised the knowledge  
1102 summary to focus on general principles and strategies for safe and efficient driving, without  
1103 referencing specific Vehicle IDs. This will allow me to adapt to changing situations and  
1104 vehicles without relying on specific IDs.

1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133