

Talking Vehicles: Cooperative Driving via Natural Language

Jiaxon Cui¹, Chen Tang¹, Jarrett Holtz³, Janice Nguyen²
Alessandro G Allievi³, Hang Qiu², Peter Stone^{1, 4}

¹The University of Texas at Austin

²University of California, Riverside

³Robert Bosch LLC

⁴Sony AI

cuijiaxon@utexas.edu

Abstract

Past work has demonstrated that autonomous vehicles can drive more safely if they communicate with one another than if they do not. However, their communication has often not been human-understandable. 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. In this work, we propose a suite of traffic tasks in 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 high-level command 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 and demo videos are available at <https://anonymous.4open.science/r/talking-vehicles>.

Introduction

Driving is inherently a multi-agent problem (Dinneweth et al. 2022) – each driver makes independent decisions based on their own perceptions while interacting with others on the road. As we transition towards (semi-)autonomous vehicles, centralized control (Antonio and Maria-Dolores 2022) of all cars may appear efficient, but it is impractical and unlikely to be widely accepted. On the other hand, cooperative driving through communication channels is more practical and can still offer significant benefits even when implemented in a limited capacity. Past research has demonstrated the advantages of cooperative perception among autonomous cars (Wang et al. 2020; Cui et al. 2022; Xu et al. 2022b). However, these benefits are restricted to vehicles that use the same encoder, limiting broader participation from those with different encoders and leaving human drivers reliant solely on their local perceptions without being privy to the collaboration efforts.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

As the use of visual-language models in state-of-the-art driver-assistant systems grows, enabling natural language reasoning for traffic situations, we are motivated by the potential for human drivers and semi-autonomous vehicle controllers to benefit from cooperative driving. To this end, we have developed a cooperative driving system that leverages natural language for communication and coordination. Prior works have explored training driving agents to make and explain driving decisions in natural language (Wayve 2023; Ma et al. 2023) or to coordinate with human drivers within a vehicle (Deruyttere et al. 2022), leveraging extensive datasets (Kim et al. 2018, 2019; Qian et al. 2023; Sima et al. 2023). However, there is a lack of datasets featuring **inter-vehicle** communication in natural language alongside corresponding driving actions to facilitate cooperation.

Recent advances in Large Language Models (LLMs) present new opportunities for agents to speak and understand natural language messages in cooperative driving scenarios. In this work, we explore the potential of LLM agents to collaborate using natural language and optimize communication and collaboration strategies over **self-play** interactions, avoiding the need to collect real-world data. For this purpose, we develop TalkingVehiclesGym, a multi-agent simulation framework that realistically models vehicle-to-vehicle communication in a set of accident-prone scenarios. Building on an LLM agent framework, we proposed a multi-agent learning method for LLM agents, LLM+DEBRIEF, which enables LLM agents to engage in centralized discussions post-interaction to refine their cooperation strategies, which are later incorporated into decentralized agent execution. Our experimental results in simulation demonstrate that when LLM agents struggle to collaborate effectively, our proposed learning method helps them learn what to communicate and how to respond to messages through interactions.

While this work has not gotten to the point of fully human-usable communication — e.g., by enforcing short, real-time messaging — this paper takes a crucial step in that direction by restricting all messages to be in natural language. Specifically, we demonstrate that natural language communication can enable cooperative perception and negotiation among autonomous vehicles. Steps remain to accelerate the system for real-time deployment, integrate real human behaviors and messages, and guarantee the safety and reliability of the communication system.

Problem Definition

In this paper, we focus on the subset of agents that are actively participating in the cooperation. We assume that these cooperative vehicles implicitly aim to help each other, treating all other (referred to as "background") vehicles as uncontrollable elements of the environment. Therefore, we frame the problem of *Talking Vehicles* as a partially observable stochastic game (POSG), focusing on optimizing the social welfare of a *focal population* (\mathcal{F}) (Agapiou et al. 2022) defined as the joint reward of all participating agents as the primary objective. The reward functions associated with each agent’s individual tasks may or may not fully align, necessitating coordination among agents to achieve high joint rewards. 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 their own partial observations and received messages from other agents. In this problem, each agent’s action space comprises two main components: **(1) generating messages** and **(2) controlling the vehicle**. In this work, the message generation space is a natural language (English).

A POSG is defined by the tuple $\langle \mathcal{I}, \mathcal{S}, \{\mathcal{O}_i\}, \{\mathcal{A}_i\}, \mathcal{P}, \{\mathcal{R}_i\} \rangle$, where $\mathcal{I} = \{1, 2, \dots, N\}$ refers to the identities of all agents in a scenario; \mathcal{S} is the state space comprehensively describing the environment; \mathcal{O}_i is the observation space describing agent i ’s view of the 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 .

The focal group of agents is denoted by $\mathcal{F} \subseteq \mathcal{I}$, representing a subset of all agents \mathcal{I} . The goal for each agent $i \in \mathcal{F}$ is to optimize a policy π_i to maximize the expected cumulative task returns of all the agents in \mathcal{F} , given background agent policies outside the focal group:

$$\max_{\{\pi_i\}_{i \in \mathcal{F}}} \mathbb{E} \left[\sum_{i \in \mathcal{F}} \sum_{t=0}^{t=\infty} R_i(s_t, \mathbf{a}_t) \mid \{\pi_j\}_{j \notin \mathcal{F}, j \in \mathcal{I}} \right],$$

where s_t is the state at time t , and $\mathbf{a}_t = (a_1^t, a_2^t, \dots, a_N^t)$ is the joint action of all agents at time t .

The policy is structured to have both control and communication capabilities. Specifically, $\pi_i(O_i, \{M_j\}_{j \in \mathcal{F}}) \rightarrow \mathcal{A}_i$ maps the observation of agent i and received messages $\{M_j\}_{j \in \mathcal{F}}$ to its action space $\mathcal{A}_i = \langle \mathcal{M}_i, \mathcal{C}_i \rangle$, where \mathcal{M}_i represents the message generation space, which is constrained to human-understandable language, and \mathcal{C}_i denotes the vehicle control space with dimensions for throttle, brake, and steering inputs. At time step t , the message M_i generated by agent i is broadcast to all other connected agents, which are those within a specific communication radius, at the next time step $t + 1$.

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 or potentially negotiate about motion plans in natural languages;
2. How can learned agents **comprehend** the received natural language messages and **incorporate** them into high-level cooperative driving decisions?

Assumptions. In this work, we assume that agents intend to communicate truthfully, accurately conveying their intentions and following through on their stated decisions. While all focal agents are designed to act cooperatively, avoiding deceptive messages or aggressive actions that could disrupt others’ driving objectives, we assume that background agents in our framework are not explicitly optimized for cooperation or adversarial behavior. Additionally, we assume that agents communicate in a manner that ensures the syntax and semantics of all messages are human-understandable, such that a human operator could comprehend the exchanged information if controlling one of the vehicles. Exploring scenarios where these assumptions are explicitly lifted could be an interesting direction for future work.

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 the closed-loop evaluation of urban driving policies. TalkingVehiclesGym supports a flexible configuration of multi-agent scenarios, incorporating heterogeneous agents such as language agents, sensory agents, human agents, heuristic behavior agents, etc. It also enables **in-episode** communication between agents using a realistically simulated communication protocol MQTT. The simulation dynamics are built on CARLA (Dosovitskiy et al. 2017), a high-fidelity urban driving simulator.

Scenarios (\mathcal{P}) and Rewards (\mathcal{R}). TalkingVehiclesGym has been set up with several accident-prone scenarios where multi-agent communication could be advantageous. Furthermore, TalkingVehiclesGym demonstrates how to tackle different collaboration dynamics, as shown in Figure 1. Scenarios labeled with **Cooperative Perception** represent opportunities for agents to benefit from shared information about areas outside their immediate line of sight and scenarios tagged with **Negotiation** are designed to demonstrate the advantages of agents discussing and reconciling conflicts in their plans. In each scenario, a focal group (\mathcal{F}) of agents is defined and operates alongside background agents governed by pre-configured behaviors. Each focal agent is assigned a task described in natural language, with success determined by reaching target locations within a specified time limit without collisions. An agent who successfully completes the task earns a reward of +1, while any agent involved in a collision incurs a penalty of -1. Remaining stagnant at any point until timeout results in a reward of 0, because conservative policies are safe, even if they are not ideal. Agents without motion targets, such as a stationary truck in cooperative perception tasks, do not earn rewards directly for themselves. However, the optimization objective encourages these agents to send messages that assist others in achieving their tasks. For detailed scenario descriptions and rewards setup, please refer to Appendix .

Observation Space (\mathcal{O}). Our environment integrates a diverse range of sensor and simulator inputs inherited from CARLA. To emphasize reasoning and multi-agent learning, we simplify environmental perception for **text-based agents** by introducing a rule-based, **partially observable captioneer**.



Figure 1: Overview of Scenarios and Agent Roles. **Green circles:** Focal agents, agents aim at establishing coordination through communication; **Red circles:** Potential colliders; **Blue circles:** Background agents.

This module abstracts away the perception task, that would otherwise require object detection or vision-language models, by directly converting scenario information such as the motion states of the ego vehicle and others, lane details, and road conditions into natural language descriptions that convey **factual** information while maintaining the partial observability imposed by the agent’s line-of-sight sensors. For agents equipped with a transmitter/receiver device (**transceiver**), real-time communication is enabled during episodes. These agents can broadcast and receive structured messages containing key information such as the content, timestamp, sender ID, and sender location by subscribing to topic-specific communication channels. Received messages ($\{M_j\}_{j \in \mathcal{F}}$) are stored in a buffer, with recent message exchanges incorporated into the agent’s observations to support decision-making. An example of a text-based observation is provided in Appendix .

Action Space (\mathcal{A}). The action space for each agent encompasses both vehicle control and communication. The control space \mathcal{C} includes dimensions for throttle, brake, and steering. To reduce the decision-making frequency, agents execute high-level vehicle motion commands represented as temporal sequences of low-level vehicle controls ($C_t, C_{t+1}, \dots, C_{t+k}$), where each command spans k time steps. These high-level commands include atomic actions such as go (maintain a target speed), stop, slow down, speed up, change to the left lane, and change to the right lane. The message generation space \mathcal{M} is restricted to natural language tokens in this work but is flexible enough to support other communication modes, such as neural network-based representations. In this work, messages are generated alongside the high-level commands at a frequency of every 0.5 seconds ($k = 10$ simulation steps).

Method

The core technical challenge of the *talking vehicles* problem is to enable agents to communicate in natural language in order to facilitate cooperation and act correspondingly. To establish an initial solution, we adopt an **LLM agent framework** (Figure 2) that prompts LLMs as a foundational prior

for autonomous agents to engage in human-like communication, structuring the message within natural language space, allowing agents to interpret messages and make informed driving decisions. A key challenge of using LLMs lies in that they are not specifically trained for driving tasks. To overcome this limitation, we introduce **LLM+DEBRIEF** (Algorithm 1), a **novel multi-agent learning methods for LLM agents** built upon feedback loops that allow LLM agents to iteratively refine their communication and motion control policies through trial-and-error interactions with confederate agents. Inspired by how humans reflect and debrief after a Hanabi game, we enable agents to discuss cooperative strategies after each interaction episode.

Agent Policy

An agent acts according to an LLM policy $\pi_i(O_i, \{M_j\}_{j \in \mathcal{F}}) \rightarrow \langle \mathcal{M}_i, \mathcal{C}_i \rangle$, where the distribution over actions follows the LLM used by the agent. Here, O_i represents a comprehensive text observation encompassing task and goal descriptions, environment details, and common traffic rules, expressed as a sequence of tokens $\{t_i^o\}$. A received message $M_j = \{t_j^m\}$ and a message to send $M_i = \{t_i^m\}$ are also token sequences generated by language agents. $\mathcal{C}_i = \{t_i^c\}$ represents a token sequence for high-level commands. The joint probability of selecting a command and generating a message is expressed as $P_i(\{t_i^m\}; \{t_i^c\} | \{t_i^o\}; \{\{t_j^m\}\}_{j \in \mathcal{F}})$ where ";" indicates token concatenation and the language model determines the probabilities.

In-Context Knowledge. Instead of fine-tuning the weights of LLMs via gradient-based methods, we adapt the policy by modifying input tokens (prompts). We define $\mathbf{K}_i = \{t_i^k\}$ as agent i ’s accumulated knowledge and $\mathbf{S}_i = \{t_i^s\}$ as its cooperative strategy. The joint probability of generating commands and messages is then influenced by these additional prompt tokens: $P_i(\{t_i^m\}; \{t_i^c\} | \{t_i^o\}; \{\mathbf{K}_i\}; \{\mathbf{S}_i\}; \{\{t_j^m\}\}_{j \in \mathcal{F}})$.

Chain-of-Thought (CoT) Reasoning. Research has shown that LLMs make decisions better when provided with sufficient context. To leverage this, we prompt LLMs to reason

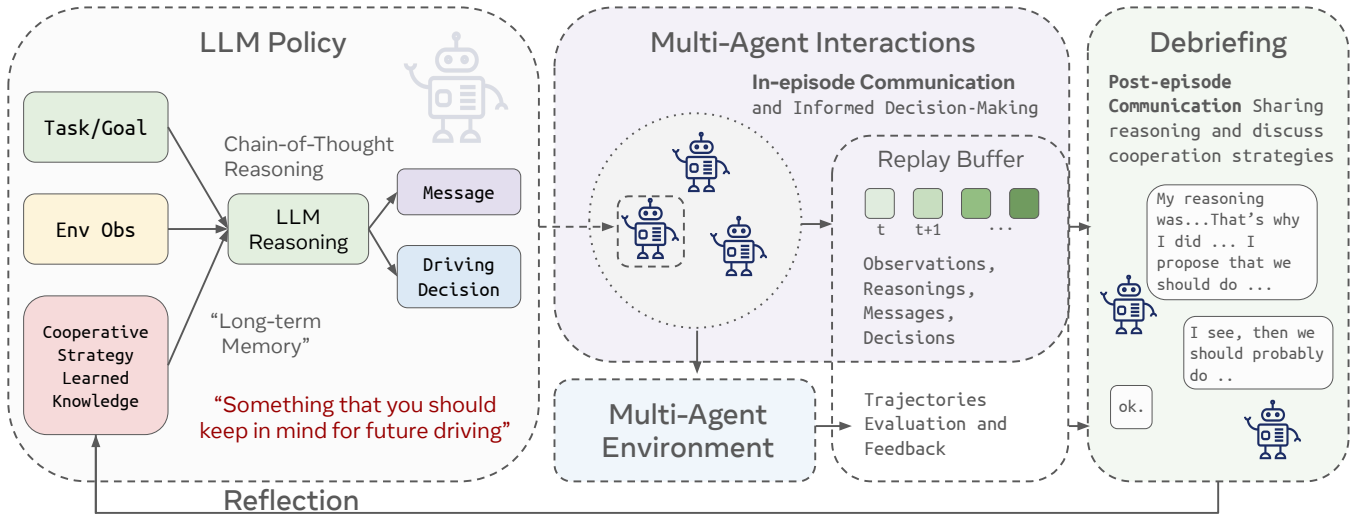


Figure 2: LLM+DEBRIEF Agent Framework. A LLM+DEBRIEF agent is provided a task to complete for each scenario, and the environment will provide the text observation of the surroundings and message dialogues. Along with the previous learned knowledge, the policy first performs chain-of-thought reasoning 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 make the decisions based on 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, commands, and others’ reactions}, 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 knowledge for future decision-making.

step-by-step about the environment, incorporating observations, received messages, and in-context knowledge. The reasoning process generates an output token sequence $R_i = \{t_i^r\}$. Following this reasoning, the LLM agent generates structured action tokens by combining the reasoning with the inputs: $P_i(\{t_i^m\}; \{t_i^c\}|\{t_i^k\}; \{t_i^s\}; \{t_i^o\}; \{\{t_j^m\}_{j \in \mathcal{F}}\}; \{t_i^r\})$. The final output is in a JSON format with keys: "command" and "message".

Agent Learning: Post-Episode Debriefing

The learning process is depicted in Figure 2. Initially, the LLM agents interact with each other in the scenarios, accumulating experience, which is stored in a replay buffer. Following the interaction phase, the agents engage in a debriefing session where they utilize past experiences as context to collaboratively refine a cooperative strategy. The outcomes of these discussions are distilled into two critical components: knowledge ($K_i = \{t_i^k\}$) and cooperative strategies ($S_i = \{t_i^s\}$). These components are subsequently integrated as in-context knowledge for future interactions, playing a pivotal role in shaping and improving the policy.

Replay Buffer. We store transition data $T_i = \langle o_{i,t}, a_{i,t}, o_{i,t+1} \rangle$, which includes current and next observations, commands, messages, and reasoning in a **replay buffer**, serving a repository for further learning and iterative refinement. When an episode concludes, the environment evaluates each agent’s performance and provides scalar rewards along with **verbal feedback**, such as Vehicle 109 collided with Vehicle 110. or Vehicle 111 stagnated too long to complete its task. Each transition in the replay buffer is subsequently **retrospectively**

labeled with enriched metadata, including responses from other agents, collision details (e.g., time to collision), stagnation specifics, and final rewards.

Batch Context Sampling. Before engaging in the post-episode discussion (debriefing), each learning agent reflects on past experiences by sampling a **batch** of learning data from its replay buffer. While analyzing the entire trajectory would provide a comprehensive understanding of failure cases, computational constraints necessitate sampling a subset of key frames. To prioritize relevant data, the sampling process heuristically assigns higher probabilities to transitions that occur immediately before collisions, involve actions contributing to collisions, or lead to stagnation due to agents slowing down. Additionally, transitions that feature more intensive multi-agent interactions are given more weight. These selected samples serve as the context for subsequent analysis and strategy formulation, allowing the agent to focus on critical scenarios for improving performance.

Debriefing. A debriefing session begins when an episode concludes in a failure and is conducted in a **turn-based** manner over N rounds, with a focus on improving cooperation in future interactions. The speaking order is determined randomly for each session, and agents take turns speaking in a round-robin format. The agent chosen to speak first is responsible for proposing a **joint** cooperative strategy ($S_1, S_2, \dots, S_{i \in \mathcal{F}}$) for everyone participating in the debriefing (the focal group). This agent begins by reasoning through its transition data batch, analyzing the consequences and influence on other agents of its actions, and formulating a proposed strategy. Subsequently, the other agents take

turns sharing their perspectives, providing feedback, or offering alternative insights based on their analysis of their own batches of experience. After the discussion, each agent summarizes the discussion to develop **individual** cooperative strategies (\mathcal{S}_i) and knowledge (\mathcal{K}_i). These outcomes serve as in-context guidelines for future driving tasks. This joint discussion for future individual decision-making structure mirrors the principles of the Centralized Training Decentralized Execution (CTDE) framework (Bernstein et al. 2002), a widely utilized approach in multi-agent learning.

Implementation Details

We utilize gpt-4o-mini with a temperature of 0.2 for the agent policy, making decisions and collecting experiences every 0.5 seconds (10 simulation frames). Received message dialogs are maintained within a 2-second window based on message age during each episode. The debriefing process is conducted after each episode for a total of 30 episodes, comprising $N = 1$ round of discussion among agents followed by a final round of individual reflection to summarize and consolidate the discussion outcomes. To enable stronger reasoning and summarization capabilities, gpt-4o is employed for the debriefing sessions and reflection. The transition data are sampled from the trajectory with a batch size of 4. Further details, including decision latencies and additional configurations, are available in Appendix .

Experiments

In this section, we aim to empirically evaluate the following hypotheses:

- (1) LLM agents can establish collaboration without prior interactions;
- (2) Natural language communication enhances the performance and coordination of LLM agents compared to those without communication;
- (3) Decentralized reflection enables LLM agents to improve their collaborative ability over interactions;
- (4) Centralized discussion among LLM agents provides additional improvements in collaboration and communication compared to decentralized reflection.

Metrics. Evaluation metrics are established based on the outcomes of agents who can incur reward (reward-eligible) for their tasks in the focal group, which is scenario-specific. For a scenario with N reward-eligible agents in the focal group, evaluated over M episodes, we utilize three key metrics: 1. the **average total reward (R)**, normalized by the group size, is given by $\frac{1}{N} \frac{1}{M} \sum_{m=1}^M \sum_{i \in \mathcal{F}} \sum_{t=0}^{t=\infty} R_i(s_t, \mathbf{a}_t)$; 2. the **average collision rate (CR)**, normalized by the group size, is $\frac{1}{N} \frac{1}{M} \sum_{m=1}^M \sum_{i \in \mathcal{F}} \mathbf{1}(\text{agent } i \text{ involved in a collision})$, where collisions may involve both focal and background agents; 3. the **average success rate (SR)**, also normalized by the group size, is $\frac{1}{N} \frac{1}{M} \sum_{m=1}^M \sum_{i \in \mathcal{F}} \mathbf{1}(\text{agent } i \text{ succeeded})$. Here, $\mathbf{1}$ is the indicator function, equal to 1 if the event occurs and 0 otherwise.

Baselines. We established several baselines and scenarios to evaluate our hypotheses:

- (1) an LLM agent using Chain-of-Thought (CoT) reasoning only (**Zero-shot**),
- (2) an LLM agent with CoT reasoning contextualized with knowledge from decentralized reflection (**Reflection**),
- (3) an LLM agent that corrects past actions via decentralized reflection, storing these corrections in a vector-based, retrievable memory and uses few-shot retrieved example augmented generation (**Correction+RAG (Retrieval Augmented Generation)**), The retrieval augmented method without communication (**Correction+RAG (Silent)**) 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 **Correction+RAG (Comm)** method, but they do not actively optimize the messages. For a fair comparison across baseline LLM agents, we do not initialize the knowledge with human data, nor is there human involvement during the learning process.

We apply the same batch context sampling method for reflection or correction for all LLM agent baselines as our method. Additionally, we include **Coopernaut** (Cui et al. 2022), a LiDAR-based cooperative driving method, as a reference for cooperative perception. Note that since Coopernaut has the advantage of using intermediate sensor data representations rather than natural language communication, we expect it to perform better than any of our evaluated methods. However, we include it as an aspirational reference point for possible future methods.

Experimental Setup. For each baseline¹, we consider two settings labeled as “Silent” and “Comm”, respectively. In the “Silent” setting, the method operates without communication; policies focus solely on controlling the vehicle without generating messages. In contrast, the “Comm” setting allows a method to either generate messages alone or both messages and driving commands. For each LLM-based learning method, we allow agents to interact 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 metrics listed above.

Quantitative Results

Tables 1 and 2 present the quantitative evaluation of all methods across tasks. Notably, none of the LLM methods in this work operate in real-time, with our approach requiring approximately 12 seconds per decision step using gpt-4o-mini. However, the inference latency of LLM agents primarily depends on context length and generation speed, both of which are expected to improve with advancements in LLM architectures and inference techniques. On average, the natural language message bandwidth remains below 50

¹Except for LLM+Debrief, which is only tested under the “Comm” setting since it is particularly designed for improving multi-agent communication.

words per decision step, requiring less than 0.01 Mbps. Table 3 in Appendix provides detailed latency measurements and message size statistics.

Communication does not guarantee collaboration in zero-shot interactions. Our zero-shot experiments, conducted in both silent and communication modes, reveal that LLM agents (gpt-4o-mini) without prior coordination cannot reliably establish collaboration solely through independent decision-making. While communication in zero-shot settings generally improves the task success rate compared to silent mode, the enhancement remains marginal. This result suggests that LLM-based reasoning alone is insufficient for fostering effective coordination without prior training or explicit strategies. Our preliminary experiments with Llama3-8B-Instruct and gpt-4o follow a similar pattern.

Decentralized reflection and correction with RAG memory reduce collisions. The decentralized learning methods, Reflection and Correction+RAG, show significant promise in reducing collision rates across tasks. Reflection allows agents to independently analyze their observations, actions, and outcomes to generate individual policies that lead to safer decision-making than Zero-shot. However, this self-reflection process often results in overly conservative behavior (e.g. in the perception-red-light scenario), resulting in frequent timeouts and limiting improvements in overall success rates despite a lower collision rate. In contrast, the Correction+RAG method demonstrates more robust performance than Reflection under decentralized learning. By fully recording successful episodes and refining commands and messages at key frames selected through heuristic batch sampling, it effectively corrects past behaviors while preserving successful coordination patterns. However, although the method improves the control strategy, we found it qualitatively does not always generate consistent messages with the actions, possibly due to the open-loop revisions. Future work should assess its generalization capability in environments with diverse dynamics and varying environmental descriptions.

Centralized debriefing enhances coordination more than decentralized reflection. The debriefing method, which focuses on generating explicit cooperation strategies, enables LLM agents to achieve more stable collaboration compared to decentralized reflection or zero-shot approaches, evidenced by higher success rates across tasks. However, debriefing remains less effective than sensor- and representation-based methods (Coopernaut) in certain scenarios.

The potential of natural language communication for multi-agent coordination. Natural language communication demonstrates strong potential for enhancing multi-agent coordination. Across tasks, methods operating in communication settings generally outperform their silent counterparts. Agents trained with communication also show greater adaptability and improve performance over interactions. Conversely, in silent modes, agents tend to adopt more cautious behaviors, but their performance remains suboptimal even after reflection or correction. These results suggest that natural language communication can play a pivotal role in enabling agents to understand and handle complex scenarios, offering a promising avenue for advancing multi-agent coordination in domains like autonomous driving.

Qualitative Analysis

This section uses the negotiation-highway-merge scenario as an example scenario to analyze policy behaviors and qualitatively examines the learned knowledge and cooperation strategies. For detailed insights into the learned knowledge, see Appendix , and refer to the supplementary videos for comprehensive demonstrations of policy behaviors.

First, we present a recorded communication exchange between agents in a demonstration video:

Vehicle 121 (on the highway): Vehicle 120, I am slowing down to create a gap for your merge. Please proceed safely.

Vehicle 120 (merging) replied: Thank you, Vehicle 121, I will speed up to merge into the gap you create. Please maintain your speed to facilitate my merge.

This form of communication is highly human-interpretable, paving the way for future human participation in multi-agent collaboration. In contrast, the $(x, y, z, feature)$ latent representation generated by Coopernaut lacks interpretability for humans and requires all vehicles in the system to share the same encoder, limiting its flexibility in mixed-autonomy settings. While in this work we do not enforce that the communication be suitable for humans to participate in the collaboration directly, the results suggest that it may be possible to move in that direction in the future by enforcing short, real-time messages.

Second, the in-context knowledge developed through the debriefing process demonstrates a **clear and coherent cooperation strategy**, defining each agent’s role and their coordination mechanisms (Appendix), in contrast to the purely reactive policies formed through self-reflection without explicit discussion of cooperation strategies (Appendix).

Third, **agents behave according to their learned knowledge and cooperation strategy.** In the negotiation-highway-merge scenario, the debriefing-based policy’s behavior follows the developed structured cooperation strategy: when the merging vehicle requests to enter the highway, highway vehicles explicitly slow down to create a gap, enabling a smooth and coordinated merge. In contrast, under the Correction+RAG (Silent) mode, the lack of clear cooperation strategy leads to uncertainty. Both the merging and highway vehicles struggle to determine the right of way, often resulting in either a collision or a prolonged indecisive interaction at the junction. We encourage readers to watch the supplementary videos accompanying this paper for a deeper understanding of the qualitative differences between policies.

Related Work

Vehicle-to-vehicle Communication. Vehicle-to-Vehicle (V2V) or Vehicle-to-everything (V2X) communication offers the potential to effectively facilitate multi-vehicle cooperation, improving the safety and reliability of autonomous vehicles in urban driving scenarios. Existing research predominantly concentrates on cooperative perception data sets (Yu et al. 2022; Xu et al. 2022b; Li et al. 2022) and tasks like cooperative detection and prediction (Wang et al. 2020; Chen

Table 1: Experiment Results for **Cooperative Perception** Scenarios.

Scenario		Overtake (Perception)			Red Light			Left Turn			
		Method	LLM	Comm	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow
Name	LLM	Comm	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow	CR \downarrow	SR \uparrow
Zero-shot (Silent)	Yes	No	-0.87	93.3	6.7	-0.87	93.3	6.7	-0.93	96.7	3.3
+Reflection (Silent)	Yes	No	-0.26	36.7	10.0	-0.87	93.3	6.7	-0.27	63.3	36.7
+Correction+RAG (Silent)	Yes	No	0.07	33.3	40.0	-0.73	86.7	13.3	0.20	40.0	60.0
Zero-shot (Comm)	Yes	Yes	-0.46	73.3	26.7	-0.33	66.7	33.3	-1.00	100.0	0.0
+Reflection (Comm)	Yes	Yes	0.40	30.0	70.0	0.07	10.0	26.7	0.60	20.0	80.0
+Correction+RAG (Comm)	Yes	Yes	0.70	6.7	76.7	0.73	13.3	87.7	-0.60	76.7	16.7
+Debrief (Comm)	Yes	Yes	0.63	16.7	80.0	1.00	0.0	100.0	0.60	20.0	80.0
Coopernaut (Comm)	No	Yes	1.00	0.0	100.0	0.97	0.0	96.7	1.00	0.0	100.0

Table 2: Experiment Results for **Negotiation** Scenarios.

Scenario		Overtake (Negotiation)			Highway Merge			Highway Exit			
		Method	LLM	Comm	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow
Name	LLM	Comm	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow	CR \downarrow	SR \uparrow	R \uparrow	CR \downarrow	SR \uparrow
Zero-shot (Silent)	Yes	No	-0.13	55.0	41.7	-0.87	93.3	6.7	-0.53	63.3	36.5
+Reflection (Silent)	Yes	No	0.80	3.0	83.3	-0.37	68.3	31.7	0.20	40.0	60.0
+Correction+RAG (Silent)	Yes	No	0.00	50.0	50.0	0.03	48.3	51.7	-0.16	58.3	41.7
Zero-shot (Comm)	Yes	Yes	0.53	23.3	76.5	-1.00	100.0	0.0	-0.60	65.0	35.0
+Reflection (Comm)	Yes	Yes	0.73	11.7	85.0	0.53	23.3	76.7	0.32	33.3	65.0
+Correction+RAG (Comm)	Yes	Yes	0.83	6.7	90.0	-0.07	53.3	46.7	-0.16	58.3	41.7
+Debrief (Comm)	Yes	Yes	1.00	0.0	100.0	1.00	0.0	100.0	0.63	10.0	73.3

et al. 2019; Xu et al. 2022a), leveraging sensor data from cameras, LiDAR, and other resources. The message aggregation strategies include early fusion (Qiu et al. 2022), late fusion, and intermediate fusion (Wang et al. 2020). Considering the limited V2V bandwidth, efforts were made to reduce the message size (Hu et al. 2022). Aoki, Higuchi, and Altintas (2020) developed a reinforcement learning method for selecting what information to be transmitted in cooperative perception. Although Cui et al. (2022) developed end-to-end driving policies and derived the critical information from expert supervision, the cooperation is still completed at the perception level. Notably, the focus of past approaches (do Nascimento and de Lucena 2017) on communication efficiency has prevented human understandability: the messages have all been in an inscrutable latent representation, with all participating vehicles needing to be informed explicitly about the syntax and semantics of the representation. In contrast, our work focuses on the message in the natural language space to enhance both participation by all vehicles, eventually including those under human control, and communication understandability.

Multi-Agent Social Interactions with LLMs. Large Language Models (LLMs) show promising communication and reasoning capabilities, suggesting their potential in multi-agent interaction scenarios. For instance, Generative Agents (Park et al. 2023) represents an early attempt at employing

LLM agents for free-form chatting, demonstrating the believable behaviors of LLM agents in spreading information. However, this study did not evaluate the LLMs’ capabilities in planning or solving multi-agent tasks. Cicero (Bakhtin et al. 2022), on the other hand, finetunes a language model to imitate human behaviors from a dataset to generate truthful messages in the game of Diplomacy, which is mixed-motive and requires communication in natural language. They train reinforcement learning policies to analyze the dialogue select actions and generate deceptive messages through value filtering. Recent 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 some 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 var-

ious 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.

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 vehi-*

cles problem, this research opens up several exciting future research areas for further exploration and development. First, the current LLM+DEBRIEF framework takes text descriptions 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. We refer readers to Appendix for a full discussion of limitations and future work.

Acknowledgements

This work has taken place in the Learning Agents Research Group (LARG) at the Artificial Intelligence Laboratory, The University of Texas at Austin. LARG research is supported in part by the National Science Foundation (FAIN-2019844, NRT-2125858), the Office of Naval Research (N00014-18-2243), Army Research Office (W911NF-23-2-0004, W911NF-17-2-0181), DARPA (Cooperative Agreement HR00112520004 on Ad Hoc Teamwork), Lockheed Martin, and Good Systems, a research grand challenge at the University of Texas at Austin. The views and conclusions contained in this document are those of the authors alone. Peter Stone serves as the Executive Director of Sony AI America and receives financial compensation for this work. The terms of this arrangement have been reviewed and approved by the University of Texas at Austin in accordance with its policy on objectivity in research.

References

- Agapiou, J. P.; Vezhnevets, A. S.; Duñez-Guzmán, E. A.; Matyas, J.; Mao, Y.; Sunehag, P.; Köster, R.; Madhushani, U.; Kopparapu, K.; Comanescu, R.; et al. 2022. Melting Pot 2.0. *arXiv preprint arXiv:2211.13746*.
- Antonio, G.-P.; and Maria-Dolores, C. 2022. Multi-Agent Deep Reinforcement Learning to Manage Connected Autonomous Vehicles at Tomorrow’s Intersections. *IEEE Transactions on Vehicular Technology*, 71(7): 7033–7043.
- Aoki, S.; Higuchi, T.; and Altintas, O. 2020. Cooperative perception with deep reinforcement learning for connected vehicles. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, 328–334. IEEE.
- Bakhtin, A.; Brown, N.; Dinan, E.; Farina, G.; Flaherty, C.; Fried, D.; Goff, A.; Gray, J.; Hu, H.; et al. 2022. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science*, 378(6624): 1067–1074.
- Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of operations research*, 27(4): 819–840.
- Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Chen, X.; Choromanski, K.; Ding, T.; Driess, D.; Dubey, A.; Finn, C.; et al. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*.
- Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.
- Chen, Q.; Tang, S.; Yang, Q.; and Fu, S. 2019. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 514–524. IEEE.
- Cui, C.; Ma, Y.; Cao, X.; Ye, W.; and Wang, Z. 2023. Receive, reason, and react: Drive as you say with large language models in autonomous vehicles. *arXiv preprint arXiv:2310.08034*.
- Cui, J.; Qiu, H.; Chen, D.; Stone, P.; and Zhu, Y. 2022. Coopernaut: end-to-end driving with cooperative perception for networked vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17252–17262.
- Deruyttere, T.; Grujicic, D.; Blaschko, M. B.; and Moens, M.-F. 2022. Talk2Car: Predicting physical trajectories for natural language commands. *Ieee Access*, 10: 123809–123834.
- Ding, X.; Han, J.; Xu, H.; Zhang, W.; and Li, X. 2023. HiLM-D: Towards High-Resolution Understanding in Multimodal Large Language Models for Autonomous Driving. *arXiv preprint arXiv:2309.05186*.
- Dinneweth, J.; Boubezoul, A.; Mandiau, R.; and Espié, S. 2022. Multi-agent reinforcement learning for autonomous vehicles: A survey. *Autonomous Intelligent Systems*, 2(1): 27.
- do Nascimento, N. M.; and de Lucena, C. J. P. 2017. FIoT: An agent-based framework for self-adaptive and self-organizing applications based on the Internet of Things. *Information Sciences*, 378: 161–176.
- Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; and Koltun, V. 2017. CARLA: An open urban driving simulator. In *Conference on robot learning*, 1–16. PMLR.
- Driess, D.; Xia, F.; Sajjadi, M. S. M.; Lynch, C.; Chowdhery, A.; Ichter, B.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; Huang, W.; Chebotar, Y.; Sermanet, P.; Duckworth, D.; Levine, S.; Vanhoucke, V.; Hausman, K.; Toussaint, M.; Greff, K.; Zeng, A.; Mordatch, I.; and Florence, P. 2023. PaLM-E: An Embodied Multimodal Language Model. In *arXiv preprint arXiv:2303.03378*.
- Hu, S.; Fang, Z.; Fang, Z.; Chen, X.; and Fang, Y. 2024. AgentsCoDriver: Large Language Model Empowered Collaborative Driving with Lifelong Learning. *arXiv preprint arXiv:2404.06345*.
- Hu, Y.; Fang, S.; Lei, Z.; Zhong, Y.; and Chen, S. 2022. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 35: 4874–4886.
- Jin, Y.; Shen, X.; Peng, H.; Liu, X.; Qin, J.; Li, J.; Xie, J.; Gao, P.; Zhou, G.; and Gong, J. 2023. Surrealdriver: Designing generative driver agent simulation framework in urban contexts based on large language model. *arXiv preprint arXiv:2309.13193*.
- Kim, J.; Misu, T.; Chen, Y.-T.; Tawari, A.; and Canny, J. 2019. Grounding Human-To-Vehicle Advice for Self-Driving Vehicles. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Kim, J.; Rohrbach, A.; Darrell, T.; Canny, J.; and Akata, Z. 2018. Textual Explanations for Self-Driving Vehicles. *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Li, Y.; Ma, D.; An, Z.; Wang, Z.; Zhong, Y.; Chen, S.; and Feng, C. 2022. V2X-Sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. *IEEE Robotics and Automation Letters*, 7(4): 10914–10921.
- Light, J.; Cai, M.; Shen, S.; and Hu, Z. 2023. AvalonBench: Evaluating LLMs Playing the Game of Avalon. *arXiv e-prints*, arXiv–2310.
- Ma, Y.; Cao, Y.; Sun, J.; Pavone, M.; and Xiao, C. 2023. Dolphins: Multimodal Language Model for Driving. arXiv:2312.00438.
- Malla, S.; Choi, C.; Dwivedi, I.; Choi, J. H.; and Li, J. 2023. DRAMA: Joint Risk Localization and Captioning in Driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1043–1052.
- Mao, J.; Niu, M.; Jiang, C.; Liang, H.; Chen, J.; Liang, X.; Li, Y.; Ye, C.; Zhang, W.; Li, Z.; et al. 2021. One million scenes for autonomous driving: Once dataset. *arXiv preprint arXiv:2106.11037*.
- Mao, J.; Qian, Y.; Zhao, H.; and Wang, Y. 2023a. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*.

- Mao, J.; Ye, J.; Qian, Y.; Pavone, M.; and Wang, Y. 2023b. A Language Agent for Autonomous Driving.
- Nie, M.; Peng, R.; Wang, C.; Cai, X.; Han, J.; Xu, H.; and Zhang, L. 2023. Reason2Drive: Towards Interpretable and Chain-based Reasoning for Autonomous Driving. *arXiv preprint arXiv:2312.03661*.
- Park, J. S.; O'Brien, J.; Cai, C. J.; Morris, M. R.; Liang, P.; and Bernstein, M. S. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 1–22.
- Qian, T.; Chen, J.; Zhuo, L.; Jiao, Y.; and Jiang, Y.-G. 2023. NuScenes-QA: A Multi-modal Visual Question Answering Benchmark for Autonomous Driving Scenario. *arXiv preprint arXiv:2305.14836*.
- Qiu, H.; Huang, P.; Asavisanu, N.; Liu, X.; Psounis, K.; and Govindan, R. 2022. AutoCast: Scalable Infrastructure-less Cooperative Perception for Distributed Collaborative Driving. In *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '22*.
- Sha, H.; Mu, Y.; Jiang, Y.; Chen, L.; Xu, C.; Luo, P.; Li, S. E.; Tomizuka, M.; Zhan, W.; and Ding, M. 2023. LanguageMPC: Large language models as decision makers for autonomous driving. *arXiv preprint arXiv:2310.03026*.
- Shao, H.; Hu, Y.; Wang, L.; Waslander, S. L.; Liu, Y.; and Li, H. 2023. LMDrive: Closed-Loop End-to-End Driving with Large Language Models. *arXiv preprint arXiv:2312.07488*.
- Shi, Z.; Fang, M.; Zheng, S.; Deng, S.; Chen, L.; and Du, Y. 2023. Cooperation on the Fly: Exploring Language Agents for Ad Hoc Teamwork in the Avalon Game. *arXiv preprint arXiv:2312.17515*.
- Sima, C.; Renz, K.; Chitta, K.; Chen, L.; Zhang, H.; Xie, C.; Luo, P.; Geiger, A.; and Li, H. 2023. DriveLM: Driving with Graph Visual Question Answering. *arXiv preprint arXiv:2312.14150*.
- Sun, P.; Kretschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. 2020. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2446–2454.
- Vezhnevets, A. S.; Agapiou, J. P.; Aharon, A.; Ziv, R.; Matyas, J.; Duéñez-Guzmán, E. A.; Cunningham, W. A.; Osindero, S.; Karmon, D.; and Leibo, J. Z. 2023. Generative agent-based modeling with actions grounded in physical, social, or digital space using Concordia. *arXiv preprint arXiv:2312.03664*.
- Wang, T.-H.; Manivasagam, S.; Liang, M.; Yang, B.; Zeng, W.; and Urtasun, R. 2020. V2vnet: Vehicle-to-vehicle communication for joint perception and prediction. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, 605–621. Springer.
- Wayve. 2023. LINGO-1: Exploring Natural Language for Autonomous Driving.
- Wen, L.; Fu, D.; Li, X.; Cai, X.; Ma, T.; Cai, P.; Dou, M.; Shi, B.; He, L.; and Qiao, Y. 2023a. Dilu: A knowledge-driven approach to autonomous driving with large language models. *arXiv preprint arXiv:2309.16292*.
- Wen, L.; Yang, X.; Fu, D.; Wang, X.; Cai, P.; Li, X.; Ma, T.; Li, Y.; Xu, L.; Shang, D.; et al. 2023b. On the road with GPT-4V (ision): Early explorations of visual-language model on autonomous driving. *arXiv preprint arXiv:2311.05332*.
- Xu, R.; Xiang, H.; Tu, Z.; Xia, X.; Yang, M.-H.; and Ma, J. 2022a. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *European conference on computer vision*, 107–124. Springer.
- Xu, R.; Xiang, H.; Xia, X.; Han, X.; Li, J.; and Ma, J. 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)*, 2583–2589. IEEE.
- Xu, Y.; Wang, S.; Li, P.; Luo, F.; Wang, X.; Liu, W.; and Liu, Y. 2023a. Exploring large language models for communication games: An empirical study on werewolf. *arXiv preprint arXiv:2309.04658*.
- Xu, Z.; Zhang, Y.; Xie, E.; Zhao, Z.; Guo, Y.; Wong, K. K.; Li, Z.; and Zhao, H. 2023b. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *arXiv preprint arXiv:2310.01412*.
- Yang, S.; Liu, J.; Zhang, R.; Pan, M.; Guo, Z.; Li, X.; Chen, Z.; Gao, P.; Guo, Y.; and Zhang, S. 2023. Lidar-llm: Exploring the potential of large language models for 3d lidar understanding. *arXiv preprint arXiv:2312.14074*.
- Yu, H.; Luo, Y.; Shu, M.; Huo, Y.; Yang, Z.; Shi, Y.; Guo, Z.; Li, H.; Hu, X.; Yuan, J.; and Nie, Z. 2022. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21361–21370.

Method

The Algorithm 1 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 gpt-4o-mini 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.

Table 3 summarizes the average latencies and message sizes for each scenario under the communication setting, evaluated using gpt-4o-mini on Nvidia A100 GPUs and Intel Gen 10 CPUs. The metrics include partial observable captioner latency (in seconds), reasoning latency (in seconds), decision latency (in seconds, excluding reasoning latency), and message size (in Mb). Data is aggregated over 10 episodes at each LLM decision step. Scenarios without communication exhibit slightly lower reasoning and decision latencies compared to those with communication within the same order of magnitude.

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 Replay-Buffer  
for  $j=1, 2, 3 \dots$  // Training epoch do  
   $\{obs_i\} = env.reset()$   
  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  
       $reasoning_i \leftarrow agents.reason(obs_i, K_i)$   
      // Get decisions for each agent based on observation and knowledge  
       $message_i, control_i \leftarrow agents.act(obs, K_i, reasoning_i)$   
    end for  
    // Step the environment with actions  
     $\{next\_obs_i\} \leftarrow env.step(\{message_i, control_i\})$   
    // Store experience to the replay buffer  
    ReplayBuffer.add(obs, next_obs, reasonings, messages)  
    // Message Dialog becomes part of the observation  
     $\{obs_i\} \leftarrow \{next\_obs_i\} \cup \{message_i\}$   
  end while  
  // Get episode feedback from the environment  
   $feedback \leftarrow env.evaluate()$   
  // 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 strategy=None then  
      cooperation_strategy = agentr.propose()  
    else  
      cooperation_strategy = agentr.revise()  
    end if  
  end for  
  //Summarize the dialogue and use it for future learning  
   $\{K_i\} \leftarrow agent.reflect()(\{K_i\})$   
end for  
last  $\{\pi_{i,j}\}$  during the last iteration of self-play
```

Table 3: Captioning, Reasoning, Decision Latency, Message Size using gpt-4o-mini LLM Policy

Latencies \ Scenario	Scenario			Scenario		
	Overtake	Left Turn	Red Light	Overtake	Highway Merge	Highway Exit
Captioner Latency (s)	0.022	0.023	0.025	0.022	0.017	0.016
Reasoning Latency (s)	10.32	10.89	9.93	9.57	12.10	10.55
Decision Latency (s)	1.06	1.25	1.37	0.86	1.05	1.27
Message Size (Mb)	0.0016	0.0013	0.0014	0.0014	0.0005	0.0005

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.

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.

Talking Vehicles Gym

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

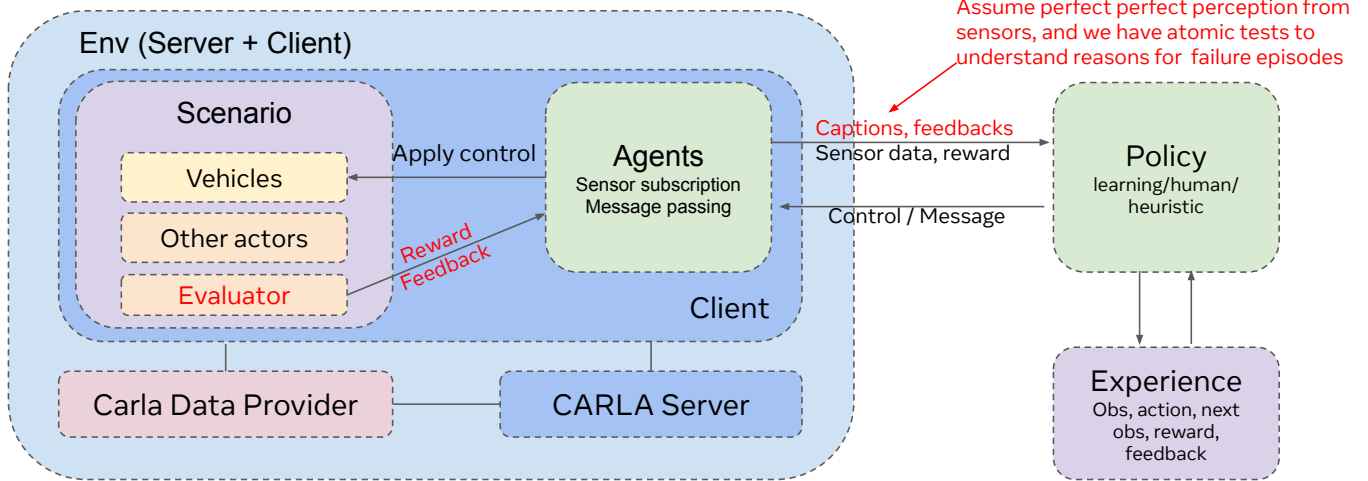


Figure 3: TalkingVehiclesGym

Table 4: 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.

Limitations and Future Work

Although we demonstrate some initial success of LLM+DEBRIEF in the *talking vehicles* problem, this research opens up several areas that require further exploration and develop-

ment.

Inference Time. The response time of large language models (LLMs) can be prohibitive, particularly when rapid decision-making is critical. Currently, it takes several seconds

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.

The 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"}
```

for LLMs to process and respond to prompts, which is too slow for real-time applications. Future efforts could explore model distillation techniques to create smaller, more efficient models that retain the capabilities of their larger counterparts but operate at a faster pace.

Human Evaluation. Designing an intuitive and user-friendly interface for human interaction with autonomous vehicles is essential. Although our framework opens up the potential to cooperate with human drivers, the complexity of effective communication interfaces is substantial. Comprehensive human-centered evaluations using human-friendly interfaces are deferred to future studies.

Communication Challenges. Real-world vehicle-to-vehicle communication faces numerous challenges, including time delays that result in outdated information and the potential for adversarial attacks with deceptive messages. Future research should focus on developing methods to mitigate these risks, ensuring timely and secure exchanges of information. Techniques such as real-time data verification and robust communication protocols will be critical for enhancing the reliability and safety of V2V systems.

Multi-modal Perception. Our experiments primarily rely on text-based inputs, assuming an idealized perception system, due to the strong performance of existing LLMs and the nascent state of multi-modal pre-trained models. The TalkingVehiclesGym allows for the integration of various sensors, presenting an opportunity to experiment with and refine multi-modal foundation models. Future work should leverage these capabilities to address the complex dynamics of real-world environments, enhancing the applicability of LLMs across different sensory inputs.

Scalability and Generalization While this paper reports on a successful proof-of-concept, the scalability of LLM+DEBRIEF to handle diverse traffic scenarios and different environmental conditions has not been extensively tested. Future research should investigate the generalization capabilities of the model across various settings and under different and denser traffic conditions to ensure its robustness and reliability in a wider range of real-world applications.

Knowledge Preservation and Batch Sampling The details of the learning framework LLM+DEBRIEF could be further improved. For example, selecting what data to learn from and how to sample the critical data and assign credits to each step, especially in the multi-step tasks. Another issue is that the LLMs could lose context during learning if we ask them to revise the existing knowledge, causing the knowledge “unclean” issue and instability in the learning if we integrate the knowledge revision too much.

Example Learned Knowledge and Cooperative Strategies

The following knowledge are fully generated by LLMs without human modification.

Red Light (Perception)

Car1 (Proceeding Forward with Green Lights)

knowledge: As a vehicle tasked with passing the intersection safely and timely when the traffic light is green, I must remain vigilant and responsive to any warnings about potential threats, such as vehicles approaching against a red light. If a threat is identified, I should prepare to stop immediately, prioritizing safety over timeliness, and only proceed once the threat has passed and the intersection is confirmed clear. I should rely on real-time updates from vehicles with a clear view of the intersection, as they provide critical information on potential hazards. By preparing to stop when alerted to a threat and proceeding only when safe, I can ensure a safe and efficient passage through the intersection without causing unnecessary stagnation.

cooperative strategy: In our refined cooperative strategy, the vehicle with the best view of the intersection, typically a stationary large vehicle in the left-turn lane, will continuously monitor the traffic situation and communicate any potential threats, such as vehicles approaching against a red light. Upon identifying a threat, this vehicle will immediately alert all others to prepare to stop. Once the threat has passed, it will promptly communicate the all-clear signal. The vehicle closest to the intersection with a green light should proceed first, followed by others in a staggered manner to maintain flow and avoid congestion. As a vehicle approaching the intersection with a green light, I will remain alert and ready to move once the all-clear is given, ensuring both safety and efficiency in passing through the intersection. This approach ensures that vehicles do not slow down unnecessarily, preventing stagnation while maintaining safety.

Truck (Stopped at Intersection Waiting for Turning)

knowledge: As the stationary vehicle with the best view of the intersection, my role is to continuously monitor the traffic situation and communicate critical information to ensure other vehicles pass safely and efficiently. I will immediately alert all vehicles of any potential threats, such as vehicles approaching against a red light, and instruct those with a green light to prepare to stop. Once the threat has passed, I will promptly communicate the all-clear signal. The vehicle closest to the intersection with a green light should proceed first, followed by others in a staggered manner to maintain flow and avoid congestion. My communication will be clear, timely, and assertive, ensuring safety by preventing collisions while maintaining traffic flow to avoid stagnation. Vehicles should remain alert and ready to move once the all-clear is given, ensuring a coordinated and efficient passage through the intersection.

cooperative strategy: In our refined cooperative strategy, the stationary vehicle with the best view of the intersection, typically a large truck in the left-turn lane, will continuously monitor the traffic situation and communicate any potential threats, such as vehicles approaching against a red light. Upon

identifying a threat, this vehicle will immediately alert all others to prepare to stop. Once the threat has passed, it will promptly communicate the all-clear signal. The vehicle closest to the intersection with a green light should proceed first, followed by others in a staggered manner to maintain flow and avoid congestion. Vehicles approaching the intersection should remain alert and ready to move once the all-clear is given, ensuring both safety and efficiency in passing through the intersection. This approach ensures that vehicles do not slow down unnecessarily, preventing stagnation while maintaining safety.

Left Turn (Perception)

Car1 (Left-Turning) knowledge: To effectively execute a northbound left turn at the intersection, I should: Use my sensors to actively monitor oncoming traffic from lane -2 and ensure I yield when necessary. Rely on real-time updates from the stationary observer vehicle for additional context, but prioritize visually confirming that approaching vehicles have visibly slowed down or stopped to yield before proceeding. Communicate my intention to turn left directly to approaching vehicles, ensuring they are aware and can adjust their speed or yield. If the approaching vehicles do not confirm yielding or if there is any uncertainty, I should pause and wait for a larger gap before proceeding. Prioritize safety by making decisions based on both communicated information and real-time observations, ensuring a smooth and efficient traffic flow through the intersection without causing collisions or unnecessary stagnation.

cooperative knowledge: In our refined cooperative strategy, the stationary observer vehicle facing south will continuously monitor the intersection and provide real-time updates on the speed and distance of all approaching vehicles, particularly those from the northbound lane. The observer will send clear and urgent messages to approaching vehicles, instructing them to yield and adjust their speed to create a safe gap for the left-turning vehicle. As the left-turning vehicle, I will reduce my speed as I approach the intersection, allowing more time for observation and decision-making. I will proceed with the left turn only after receiving direct confirmation from the approaching vehicle that it is yielding and has visibly slowed down. If the approaching vehicle does not confirm yielding or is traveling at a high speed, I will pause and wait for a larger gap before proceeding. This strategy ensures that all vehicles are aware of each other's intentions, reducing the risk of collisions while maintaining smooth traffic flow without unnecessary stopping or stagnation.

Truck

knowledge: As the stationary observer facing south, my role is to continuously monitor the intersection and provide real-time updates on the speed and distance of all approaching vehicles, especially those from the northbound lane. I will send clear and timely messages to approaching vehicles, instructing them to yield and adjust their speed to create a safe gap for the left-turning vehicle. The left-turning vehicle should reduce its speed as it approaches the intersection and proceed only after receiving direct confirmation from the approaching vehicle that it is yielding and has visibly slowed down or stopped. If an approaching vehicle does not confirm

yielding, I will advise the left-turning vehicle to pause and wait for a larger gap. I will ensure precise communication and coordination among all vehicles to prevent collisions and maintain smooth traffic flow, while remaining vigilant for any occluded vehicles and adapting the strategy as needed.

cooperative knowledge: In our refined cooperative strategy, as the stationary observer facing south, I will continuously monitor the intersection and provide real-time updates on the speed and distance of all approaching vehicles, particularly those from the northbound lane. I will send repeated and urgent messages to approaching vehicles, instructing them to yield and adjust their speed to create a safe gap for the left-turning vehicle. The left-turning vehicle will reduce its speed as it approaches the intersection, allowing more time for observation and decision-making. It will proceed only after receiving explicit confirmation from me that the intersection is clear and that approaching vehicles have visibly slowed down or stopped. If the approaching vehicles do not yield, the left-turning vehicle will pause and wait for a larger gap before proceeding. This strategy ensures that the left-turning vehicle can safely and efficiently complete the turn without causing collisions or unnecessary stagnation, while I facilitate communication and coordination among all vehicles involved.

Overtake (Perception)

Car1 (Overtaking Car)

knowledge: To safely and efficiently overtake a stopped truck in lane 1 by moving into the opposite lane, I will use my advanced sensors, such as radar and LIDAR, to continuously assess lane -1 for oncoming traffic, including detecting occluded vehicles. I will rely on real-time updates from the stationary vehicle, which will monitor lane -1 and provide updates on any visible traffic. Before initiating the overtaking maneuver, I must ensure that my sensor data aligns with the stationary vehicle's observations and infrastructure-based sensors confirm lane conditions. If any oncoming traffic is detected, I will maintain my current speed and position in lane 1 until it is safe to overtake. Immediate communication and verification of lane conditions are crucial before proceeding with the maneuver. This approach ensures that overtaking is only attempted when truly safe, preventing collisions and maintaining smooth traffic flow without causing stagnation..

cooperative strategy: In our cooperative strategy, the stationary vehicle in lane 1 (the truck) will continuously monitor lane -1 for visible oncoming traffic and provide real-time updates on any detected vehicles' speed and distance. Simultaneously, the overtaking vehicle (myself, originally in lane 1) will use advanced sensors to independently verify lane conditions, ensuring no occluded vehicles are present. Before initiating the overtaking maneuver, both vehicles must confirm that their observations align, utilizing infrastructure-based sensors as a third layer of verification for lane conditions. If any oncoming traffic is detected, the stationary vehicle will immediately communicate this, prompting the overtaking vehicle to maintain its position and speed in lane 1. Once all parties confirm that lane -1 is clear, the overtaking vehicle can proceed with the maneuver. This strategy ensures a smooth and efficient overtaking process, minimizing the

risk of collisions and avoiding stagnation by maintaining a steady flow of traffic.

Truck (Stopped Truck in Lane 1)

knowledge: As the stationary truck in lane 1, my role is to continuously monitor lane -1 for visible oncoming traffic and provide real-time updates to the overtaking vehicle behind me, including the speed and distance of detected vehicles. I must ensure my updates include immediate alerts about any detected traffic and a clear confirmation signal when lane -1 is safe for overtaking. The overtaking vehicle should independently verify lane conditions using its sensors and only proceed when both my updates and its sensor data confirm that lane -1 is clear. If oncoming traffic is detected, I will immediately communicate that overtaking is unsafe, prompting the overtaking vehicle to maintain its position and speed in lane 1. Before the overtaking maneuver, the overtaking vehicle must confirm receipt of my updates and alignment with its sensor data. Infrastructure-based sensors should be utilized for additional cross-verification of lane conditions. This approach ensures a safe and efficient overtaking maneuver, preventing collisions and maintaining smooth traffic flow.,

cooperative strategy: In our cooperative strategy, as the stationary truck in lane 1, my role is to continuously monitor lane -1 for any visible oncoming traffic and provide real-time updates to the overtaking vehicle behind me. I will communicate the speed and distance of any detected vehicles to ensure clarity. The overtaking vehicle, positioned behind me in lane 1, will use its advanced sensors to independently verify lane conditions, ensuring no occluded vehicles are present. Before initiating the overtaking maneuver, both vehicles must confirm that their observations align, utilizing infrastructure-based sensors as a third layer of verification for lane conditions. If any oncoming traffic is detected, I will immediately communicate this, prompting the overtaking vehicle to maintain its position and speed in lane 1. Once all parties confirm that lane -1 is clear, the overtaking vehicle can proceed with the maneuver. This strategy ensures a smooth and efficient overtaking process, minimizing the risk of collisions and avoiding stagnation by maintaining a steady flow of traffic.

Overtake (Negotiation)

Car1 (Overtaking Car)

knowledge: Updated Knowledge for Future Driving1. **Advance Signaling**: Always signal intentions to merge into the opposite lane well in advance, including current speed and planned changes.2. **Thorough Assessment**: Carefully evaluate the speed and position of vehicles in the opposite lane, accounting for potential occluded vehicles.3. **Maintain Safe Distance**: Keep a minimum safe distance of at least 10 meters from any vehicle in the opposite lane during merging to prevent rapid closure and collisions.4. **Synchronized Speed Adjustments**: Coordinate speed changes with vehicles in the opposite lane to facilitate safe merging without disrupting traffic flow.5. **Proactive Adaptation**: Be prepared to adjust speeds based on the actions of other vehicles to maintain safe distances and avoid conflicts.6. **Prioritize Safety**: If a safe gap can-

not be established, prioritize safety by slowing down rather than accelerating.7. **Continuous Monitoring**: Maintain situational awareness of surrounding vehicles' speeds and positions to make informed decisions during maneuvers.8. **Anticipate Conflicts**: Recognize that other vehicles may not always respond as expected; plan for alternative actions if a merging attempt becomes unsafe.9. **Real-Time Communication**: Continuously communicate intentions and actions to vehicles in the opposite lane, ensuring both can proceed safely and efficiently.10. **Dynamic Response**: Be ready to slightly decelerate if encountering oncoming traffic or if the distance to the vehicle in the opposite lane becomes too close.11. **Clear Coordination**: Establish clear communication protocols regarding maximum speeds and safe distances to prevent misunderstandings during overtaking maneuvers.12. **Cooperative Maneuvering**: Work collaboratively with other vehicles to ensure that both parties can fulfill their tasks efficiently and safely, preventing collisions and stagnation.13. **Gradual Speed Increase**: When overtaking, gradually increase speed while monitoring the distance to the vehicle in the opposite lane to avoid sudden closures.14. **Acknowledge Cooperation**: Appreciate the cooperation of other vehicles in facilitating safe merges, enhancing overall traffic flow and safety.15. **Continuous Feedback Loop**: Implement a feedback mechanism to assess the effectiveness of maneuvers and adjust strategies based on real-time interactions with other vehicles.,

cooperative strategy: In our cooperative strategy for safe overtaking, when the overtaking vehicle intends to pass a stationary vehicle in the opposite lane, it will signal its intention well in advance and gradually increase its speed while ensuring it maintains a safe distance of at least 10 meters behind the vehicle traveling in the same lane. The vehicle in the opposite lane should maintain its current speed but be prepared to slightly accelerate if the overtaking vehicle approaches too closely, ensuring that the distance does not close too quickly. If the overtaking vehicle detects oncoming traffic or finds itself too close to the vehicle in the opposite lane, it will slightly decelerate to create a safe gap without causing stagnation. Continuous communication regarding speeds and distances will be essential, allowing both vehicles to adapt their actions in real-time, facilitating a smooth overtaking maneuver while ensuring safety on the road and preventing collisions. This coordinated approach will enable both vehicles to complete their tasks efficiently and safely, maintaining smooth traffic flow.

Car2 (Opposite Car)

knowledge: To effectively execute the task of going forward and keeping in lane -1 while in a hurry, I should remember the following key points: 1. **Clear Communication**: Signal intentions to nearby vehicles, especially when another vehicle in the opposite lane plans to merge into lane -1. 2. **Maintain Safe Distance**: Keep a safe distance of at least 10 meters from any merging vehicle to facilitate safe lane changes and avoid collisions. 3. **Dynamic Speed Adjustment**: Be prepared to adjust my speed based on the actions of merging vehicles, ensuring a safe buffer zone without causing stagnation. 4. **Gradual Speed Changes**: Both vehicles should gradually increase or decrease speed to prevent abrupt

changes that could lead to collisions. 5. **Prioritize Safety**: If a merging vehicle accelerates too quickly or if the situation becomes uncertain, adjust my speed to maintain safety while keeping forward momentum. 6. **Situational Awareness**: Stay vigilant about the flow of traffic, particularly regarding vehicles traveling against the normal flow, to prevent dangerous situations. 7. **Cooperative Maneuvering**: Work collaboratively with merging vehicles to ensure safe transitions into lane -1 while maintaining progress towards my destination. 8. **Anticipate Actions**: Proactively anticipate the merging vehicle's actions and adjust my speed accordingly to maintain safety and avoid collisions. 9. **Real-Time Adaptation**: Continuously communicate and adapt to the real-time actions of other vehicles to enhance safety and efficiency in maneuvers. 10. **Avoid Conflicts**: Recognize when my task conflicts with another vehicle's intention and be ready to adjust my actions to prevent collisions. 11. **Monitor Distance Closely**: Regularly assess the distance to merging vehicles and adjust speed to maintain a safe gap, especially during overtaking maneuvers. 12. **Prepare for Unforeseen Situations**: Be ready to decelerate or stop if necessary to avoid collisions, especially when encountering vehicles traveling against the normal flow. 13. **Enhanced Coordination**: Ensure that both vehicles involved in overtaking maneuvers maintain clear communication about their speeds and distances to facilitate safe and efficient transitions. 14. **Adapt to Overtaking Dynamics**: When another vehicle intends to overtake, maintain speed but be prepared to slightly accelerate or decelerate based on its approach to ensure a safe gap is maintained. By adhering to these updated principles, I can improve safety and efficiency in future driving scenarios.,

cooperative strategy: In our cooperative strategy, when the overtaking vehicle intends to pass a stationary vehicle in the opposite lane, it will signal its intention well in advance and gradually increase its speed while maintaining a safe distance of at least 10 meters behind the vehicle traveling in lane -1. The vehicle in lane -1, tasked with going forward and maintaining momentum, will keep its current speed but be prepared to slightly accelerate if the overtaking vehicle approaches too closely, ensuring that the distance does not close too quickly. If the overtaking vehicle detects oncoming traffic or finds itself too close to the vehicle in lane -1, it will slightly decelerate to create a safe gap without causing stagnation. Continuous communication regarding speeds and distances will be essential, allowing both vehicles to adapt their actions in real-time, facilitating a smooth overtaking maneuver while ensuring safety on the road and preventing collisions. This coordinated approach will enable both vehicles to complete their tasks efficiently and safely, maintaining forward movement without causing delays.

Highway Merge (Negotiation)

Car1 (Merging Vehicle)

knowledge: To effectively merge onto the highway when in a hurry, I should initiate communication by clearly stating my intention to merge, including my current speed and position. As I approach the merge point, I will gradually accelerate to match the highway speed. The vehicle in the

rightmost lane should proactively create a gap by slightly reducing its speed earlier, allowing me to merge smoothly. Continuous communication is essential to coordinate actions and avoid collisions. I must maintain a safe merging distance and speed, avoiding abrupt maneuvers, and ensure my actions do not cause stagnation or disrupt traffic flow. I should also be prepared to adjust my speed dynamically based on real-time feedback from the highway vehicle.,

cooperative strategy: In our cooperative strategy for merging onto the highway, the merging vehicle (myself) will initiate the process by clearly communicating its intention to merge, including its current speed and position. As the merging vehicle approaches, it will gradually accelerate to match the highway speed, ensuring a smooth transition. The vehicle in the rightmost lane of the highway will proactively create a gap by slightly reducing its speed earlier, allowing the merging vehicle to enter the lane without needing to match speeds precisely. Throughout the process, both vehicles will maintain clear communication, with the merging vehicle leading in signaling its actions and the highway vehicle responding by adjusting speed to facilitate a safe merge. This approach prevents collisions and avoids traffic stagnation by ensuring a coordinated and efficient merging process.

Car2 (Highway Vehicle)

knowledge: To effectively execute my task of keeping on the original highway lane and moving forward while in a hurry, I should proactively create a gap by slightly reducing speed earlier when a merging vehicle is detected, allowing it to merge smoothly. The merging vehicle should communicate its intention and adjust its speed to match the highway flow. I must maintain open communication and responsiveness to the merging vehicle's actions, ensuring a safe and efficient merge without causing collisions or traffic stagnation. Once the merge is complete, I can gradually accelerate to my desired speed, maintaining a steady flow in my lane.,

cooperative strategy: In our cooperative strategy for highway merging, the vehicle on the highway (myself) will proactively create a gap by slightly reducing speed earlier when an approaching merging vehicle is detected, allowing it to enter the lane without needing to precisely match speeds. The merging vehicle will initiate the process by clearly communicating its intention to merge, along with its current speed and position, and will gradually accelerate to match the highway flow, ensuring a smooth transition. Throughout the process, both vehicles will maintain open communication, with the merging vehicle leading in signaling its actions and the highway vehicle responding by adjusting speed to facilitate a safe merge. This approach ensures that the merging vehicle can complete its task efficiently and safely, while the highway vehicle continues forward without causing traffic stagnation or collisions.

Highway Exit (Negotiation)

Car1 (Exiting Highway)

knowledge: To exit the highway via the leftmost lane, initiate communication to signal the intention to merge from the high-speed lane at least 20 meters before the exit. Maintain a reasonable speed while seeking a safe gap to merge, ideally ahead of the left traffic flow. Upon receiving confirmation

from the vehicle in the leftmost lane that it will create a gap, visually confirm the gap size and adjust speed to align with the traffic flow in the target lane. Ensure continuous communication to dynamically adjust actions as needed, ensuring a safe and efficient merge without causing collisions or traffic stagnation. If the vehicle in the leftmost lane cannot create a gap, be prepared to continue in the current lane and take the next available exit. Always maintain awareness of surrounding vehicles and adjust strategies as necessary to ensure safety. Prioritize early communication, visual confirmation of the gap, and flexibility in speed adjustments to facilitate a smooth merge.

cooperative knowledge: In our cooperative strategy, the vehicle intending to merge from the high-speed lane (myself) should initiate communication at least 20 meters before the exit, clearly signaling the intention to merge into the leftmost lane. Upon receiving this communication, the vehicle already in the leftmost lane should promptly assess the traffic conditions and, if safe, slightly adjust its speed by either maintaining or slightly accelerating to create a gap without causing stagnation. The merging vehicle should then align its speed with the traffic flow in the target lane, ensuring a smooth and timely merge. Continuous communication is crucial, with both vehicles updating each other on their actions and intentions to dynamically adjust as needed. If a safe merge is not feasible due to proximity or timing, the merging vehicle should be prepared to continue in its current lane and take the next available exit, ensuring all tasks are completed without collisions or traffic stagnation. This approach ensures that the vehicle in the leftmost lane can continue its task of proceeding forward safely while facilitating the merging vehicle's transition.

Car2 (Staying on the Highway)

knowledge: To effectively execute the task of staying in the leftmost lane and proceeding forward on the highway while prioritizing safety and urgency, I should maintain a speed close to the speed limit to ensure smooth traffic flow. When a vehicle in an adjacent lane signals an intention to merge into my lane, I should promptly acknowledge this and assess traffic conditions. If traffic flow allows, I should slightly slow down to create a gap, ensuring the merging vehicle can align with the traffic flow without causing stagnation. Continuous communication is crucial, with updates on actions and intentions to dynamically adjust as needed. I must remain vigilant, adapting my actions based on the traffic environment, always prioritizing safety and efficient lane occupancy, and ensuring sufficient space for merging vehicles. If a safe merge is not feasible, the merging vehicle should continue in its current lane and take the next available exit.

cooperative knowledge: In our cooperative strategy, when a vehicle in an adjacent lane intends to merge into the leftmost lane to exit the highway, it should communicate its intention at least 20 meters before the exit, signaling its plan to merge. Upon receiving this communication, the vehicle already in the leftmost lane (myself) should promptly assess traffic conditions and, if safe, slightly slow down to create a gap for the merging vehicle, ensuring no stagnation occurs. The merging vehicle should align its speed with mine be-

fore reaching the critical merging point, facilitating a smooth transition. Continuous communication is crucial, with both vehicles updating each other on their actions and intentions to dynamically adjust as needed. If a safe merge is not feasible due to proximity or timing, the merging vehicle should be prepared to continue in its current lane and take the next available exit, ensuring all tasks are completed without collisions or traffic stagnation. This approach allows me to stay in the leftmost lane and proceed forward safely while facilitating the merging vehicle's transition.

Highway Merge (Negotiation) Silent Reflection

Car1 (Merging Vehicle) knowledge: Updated Knowledge for Merging onto the Highway:1. ****Continuous Monitoring:**** Always be aware of vehicles behind, directly in front, or to the side, as they pose immediate collision risks.2. ****Early Gap Identification:**** Identify potential merging gaps early and adjust speed in advance to align with these gaps, considering both vehicles ahead and behind.3. ****Dynamic Speed Adjustment:**** Accelerate only when a clear and safe gap is confirmed. Be ready to slow down if a vehicle behind is approaching quickly or if a vehicle ahead is close.4. ****Maintain Safe Distance:**** Prioritize keeping a safe distance from vehicles directly ahead and behind. If a vehicle is too close, adjust speed to increase the gap before merging.5. ****Safety Over Speed:**** Prioritize safe merging over speed. Avoid aggressive maneuvers that could lead to collisions, even if it means a slight delay.6. ****Flexible Strategy:**** Adapt strategies based on real-time traffic conditions. Reassess and choose a safer alternative if a planned action seems unsafe.7. ****Immediate Threat Focus:**** Pay special attention to vehicles approaching from behind in your intended merging path. Adjust your strategy to create a safe gap with these vehicles before merging.8. ****Proximity Awareness:**** If a vehicle is within a critical distance behind, prioritize adjusting speed to ensure a safe merging gap.9. ****Cautious Acceleration:**** When a vehicle is directly in front and moving slower, maintain speed or slow down to allow it to move further away before attempting to merge. By applying these strategies, you can merge onto the highway more effectively and safely, even when in a hurry.

Car2 (Highway Vehicle) knowledge: Updated Knowledge for Navigating Highway Merge Junctions 1. ****Early Detection and Assessment:**** Identify merging vehicles early, focusing on their speed and proximity. If they are close, prepare to adjust your speed promptly to facilitate safe merging.2. ****Prioritize Safety Over Speed:**** Always prioritize avoiding collisions over maintaining speed. Adjust your speed to ensure safe distances from merging vehicles, even if it causes a slight delay.3. ****Dynamic Speed Adjustment:**** Be ready to slow down significantly if a merging vehicle is very close. Avoid abrupt speed increases that could reduce merging space and lead to collisions.4. ****Continuous Monitoring:**** Maintain awareness of the speed and position of nearby vehicles, especially those merging. Be vigilant of vehicles approaching from behind and to the side.5. ****Proactive Space Creation:**** Act promptly to create space for merging vehicles. Adjust your speed early to prevent conflicts and maintain a smooth flow.6. ****Anticipate Merging Intentions:**** If a vehicle is

close and in a merging lane, anticipate its intention to merge and adjust your speed or position accordingly to prevent collisions.7. ****Balance Urgency with Caution****: While in a hurry, balance the need for speed with safety. Ensure that any speed adjustments do not compromise the safety of merging vehicles.8. ****Immediate Response to Close Proximity****: When a merging vehicle is extremely close, prioritize immediate action to slow down or create space, even if it means temporarily reducing speed significantly.9. ****Evaluate Lane Change Options****: If safe and necessary, consider a temporary lane change to allow merging vehicles to enter your lane smoothly, while maintaining your original route.10. ****Avoid Stagnation****: Ensure that actions do not cause stagnation or impede traffic flow. Adjust speed dynamically to maintain progress while ensuring safety.