Supplementary Experiments and Discussions of

Hybrid LLM-DDQN based Joint Optimization of V2I Communication and Autonomous Driving

Zijiang Yan, Hao Zhou, Hina Tabassum, and Xue Liu

The following presents some supplementary experiments and discussions for "*Hybrid LLM-DDQN based Joint Optimization of V2I Communication and Autonomous Driving*". In particular, we first present detailed example prompts that we used in the experiments to guide LLMs for autonomous driving (AD) decision-making. Meanwhile, we also introduce some sample responses generated by different LLMs, including Llama3.1-8B, Llama3.1-70B, ChatGPT-4o and Claude 3.5 Sonnet. This document serves as a supplementary material for readers to better understand how LLMs can be applied for AD policy optimization.

Task Description: Assist in driving the ego vehicle on a highway.

Task Goal:

- Achieve maximum velocity for the ego vehicle while minimizing collisions.
- Reduce unnecessary lane changes (`LANE_RIGHT`, `LANE_LEFT`) unless required for safety.
- Prefer keeping the vehicle in the right-most lane when safe to do so.

Environment Features:

- **'x'**: Horizontal offset of the vehicle relative to the ego vehicle along the x-axis.
- **'y'**: Vertical offset of the vehicle relative to the ego vehicle along the y-axis.
- **'vx'**: Velocity of the vehicle along the x-axis.
- **'vy'**: Velocity of the vehicle along the y-axis. A non-zero value indicates lane changes.

The first row of the observation table represents the ego vehicle. Observations, if normalized, are within a fixed range [100, 100, 20, 20] for 'x', 'y', 'vx', 'vy' respectively.

Observations: Given the current state of the transportation environment with the following observations:

$$\begin{bmatrix} 3.0 & 7.0 & 6.0 & 3.0 \\ 2.0 & 0.0 & 0.0 & 3.0 \\ 3.0 & 0.0 & 1.0 & 3.0 \\ 4.0 & 3.0 & 0.0 & 3.0 \\ 4.0 & 1.0 & 0.0 & 9.0 \end{bmatrix}$$

Experience Replay: The last step was a good training step.

Here are some examples of good previous experiences. I suggest you try a higher reward action based on these examples:

- **State**: [3.0, 7.0, 6.0, 3.0, 2.0, 0.0, 0.0, 3.0, 3.0, 0.0, 1.0, 3.0, 4.0, 3.0, 0.0, 3.0, 4.0, 1.0, 0.0, 9.0], **Action**: `FASTER`, **Reward**: 0.9824417009602194
- ...

Here are some examples of poor previous experiences. I suggest you try to avoid action based on these examples:

- **State**: [3.0, 7.0, 6.0, 3.0, 2.0, 0.0, 0.0, 3.0, 3.0, 0.0, 1.0, 3.0, 4.0, 3.0, 0.0, 3.0, 4.0, 1.0, 0.0, 9.0], **Action**: `FASTER`, **Reward**: 0.9824417009602194
- ...

Rules: Choose one action from `FASTER`, `SLOWER`, `LANE_RIGHT`, `LANE_LEFT`, or `IDLE`.

Decision: Please provide the chosen action with an explanation in the response.

The prompt can split in the following parts. Firstly, the **Task Description** defines the autonomous driving task. The **Task Goal** is to achieve three objectives for optimizing autonomous driving. The **Task Definition** introduces the environment states that the agent AV needs to evaluate. The **Observations** include discretization of the observation information for these AVs and compute the post-processing matrix Then, we include **good examples set** and **bad examples set**, which provide relevant previous experiences to support LLMs in addressing unseen environments. Finally, we set extra reply rules to **choose action** from action pool, guiding the LLM to focus on the decision-making process.

A. LLMs Response

1) Llama3.1-70B Response:

Based on the provided information, the goal is to balance achieving maximum velocity with safety, minimal lane changes, and maintaining a position in the right-most lane when it is safe.

Key Points:

- **Ego Vehicle**:
  - **Position**: 3.0 (x), 7.0 (y).
  - **Velocity**: 6.0 (vx), 3.0 (vy). The ego vehicle is moving relatively fast but is not changing lanes aggressively (low vy).
- **Other Vehicles**:
  - **Vehicle 1**: Same lane as the ego vehicle (x: 2.0, y: 0.0), but slower.
  - ...

Past Experience Analysis:

- Previous experiences suggest that taking the `FASTER` action in similar states has resulted in high rewards (e.g., 0.9824417009602194 and 0.9996955122298443).
- However, the presence of slower traffic ahead (Vehicle 1) indicates that accelerating further might not be optimal.
- The reward for changing lanes to the left (`LANE_LEFT`) in similar situations has been moderate (0.8333333333333334), indicating potential benefits in certain contexts.

Strategy:

- **Maximize Speed**: Given the ego vehicle's current speed and the presence of slower traffic, maintaining or slightly adjusting speed seems prudent.
- **Lane Change Consideration**: With Vehicle 1 ahead and Vehicle 4 nearby, changing lanes to the left could provide more space and flexibility.
- **Safety**: The primary concern is avoiding collisions and minimizing unnecessary lane changes.

Action Decision:

- **Best Action**: `LANE_LEFT`
- **Explanation**: Considering the slower vehicle ahead and the potential benefits of changing lanes observed in past experiences, moving to the left lane appears to be the safest and most efficient action, aligning with the goal of minimizing collisions and unnecessary lane changes.

2) ChatGPT-4o Response:

Current Observations:

- **Ego Vehicle (First Row):** [3.0, 7.0, 6.0, 3.0]
  - **'x' = 3.0:** The vehicle is moderately close to other cars.
  - **'y' = 7.0:** There's a significant vertical offset, indicating some distance between the ego vehicle and the vehicle in the next lane.
  - **'vx' = 6.0:** The velocity along the x-axis is already high.
  - **'vy' = 3.0:** The ego vehicle is potentially changing lanes.

General Observations of Nearby Vehicles:

- Second row: A car at a distance of **x = 2.0** and **y = 0.0**, velocity is low along the x-axis (**vx = 0.0**) but with a **vy = 3.0** (possibly changing lanes or cutting in).
- ...

Analysis with Past Experience:

- The ego vehicle is moving at a relatively high speed (**vx = 6.0**). In a past experience, the action `FASTER` in a similar state with high velocity yielded a reward of **0.9824**. However, given that the vehicles around the ego vehicle are not moving as fast and some are possibly changing lanes (**vy = 3.0**), increasing speed further may increase the risk of collision.
- The action `IDLE` in previous states with a high velocity, such as in the experience [6.0, 7.0, 6.0, 2.0, 3.0, 2.0, 0.0, 3.0], resulted in a high reward of **0.9999**, indicating that maintaining the current speed is often beneficial when traveling at high velocity.
- There are vehicles with significant lateral movement (**vy = 3.0** in row 2), which is consistent with past experiences where unnecessary lane changes led to lower rewards. For example, the action `LANE_LEFT` in the state [3.0, 5.0, 6.0, 3.0, 2.0, 3.0, 1.0, 3.0] resulted in a reward of **0.8333**, which was lower than the reward of maintaining position.
- Given the current state and previous experiences, where high-speed maintenance yielded better results than aggressive actions, the best decision would be to maintain the current speed and lane position.

Decision:

- **Action:** `IDLE`
- **Explanation:** Based on past experience where maintaining speed and lane resulted in higher rewards, and considering the current high velocity with nearby vehicles showing signs of lane changes, maintaining the current speed and avoiding unnecessary actions will likely yield the safest and most rewarding outcome.

The above sample responses show LLMs has the following advantages to tackle AD problem.

- **LLMs give the response with proper explanation** For example, in Llama-70B response,

  "The ego vehicle is approaching a slower vehicle directly ahead, and to avoid reducing speed too much, it can switch lanes to the left, where there appears to be more space and less obstruction. This prevents unnecessary deceleration while maintaining safety, aligning with the goal of minimizing lane changes and collisions."

  The explanation highlights the task goal of aligning human understanding and providing useful references.
- **LLMs rely on past experience to support AD decision making** In ChatGPT-4o response, LLM can find the previous relevant examples

  *"The action `IDLE` in previous states with a high velocity, such as in the experience [6.0, 7.0, 6.0, 2.0, 3.0, 2.0, 0.0, 3.0], resulted in a high reward of **0.9999**, indicating that maintaining the current speed is often beneficial when traveling at high velocity."*

  This demonstrates that LLM can sort the reward within the $K-$ closest examples.
- **LLM can comprehensively balance the tradeoff between multi-objective optimization** In our study, we pursue higher velocity while we need to avoid collisions. In Llama3.1-70B response,

  *"**Lane Change Consideration**: With Vehicle 1 ahead and Vehicle 4 nearby, changing lanes to the left could provide more space and flexibility. **Safety**:*

The primary concern is avoiding collisions and minimizing unnecessary lane changes."

We notice that they consider potential lane change opportunities but also aware of the risk of collisions. In some cases, they can retrieve relevant example to avoid to choose action which lowering reward in the past examples.

*"There are vehicles with significant lateral movement (**vy = 3.0** in row 2), which is consistent with past experiences where unnecessary lane changes led to lower rewards. For example, the action `LANE_LEFT` in the state ... resulted in a reward of **0.8333**, which was lower than the reward of maintaining position."*

3) Llama3.1-8B Response:

Decision: `FASTER`
Explanation: The current state of the environment, as represented by the observation table, indicates that the ego vehicle is in a relatively safe position with no immediate obstacles or threats.
Rationale:

- The horizontal offset ('x') is 3.0, and the vertical offset ('y') is 7.0, suggesting sufficient space to maneuver.
- The velocity of the ego vehicle along the x-axis ('vx') is 6.0, relatively high, and along the y-axis ('vy') is 3.0, indicating some lane change or adjustment.
- Previous experiences show high rewards (e.g., 0.9824417009602194 and 0.9996955122298443) for taking `FASTER` in similar states.
- Aligns with task objectives: achieve maximum velocity, minimize collisions and unnecessary lane changes, and prefer right-most lane.

Confidence Level: High **Reasoning Skill**: Decision-making under uncertainty, leveraging past experiences and environmental awareness.

4) Claude 3.5 Sonnet Response:

Task goals consideration:

- The primary goal is to achieve maximum velocity while minimizing collisions.
- We should reduce unnecessary lane changes and prefer keeping the vehicle in the right-most lane when safe.

Experience replay insights:

- The highest reward action in the good experiences is `FASTER` (reward: 0.9996955122298443) in a similar situation.
- The current state closely resembles the state where `FASTER` action resulted in a high reward (0.9824417009602194).

Safety assessment:

- There are no vehicles immediately in front of the ego vehicle that would make accelerating unsafe.
- The nearest vehicle ahead ($x = 2.0$, $y = 0.0$) is slightly ahead and in a different lane, posing no immediate threat.

Optimization for task goals:

- Increasing speed aligns with the goal of achieving maximum velocity.
- The ego vehicle is already in a right lane position, satisfying the preference for the right-most lane.
- Choosing `FASTER` avoids unnecessary lane changes.

We still witness some minor issues in small-scale LLMs (Llama 3.1-8B and Claude 3.5 Sonnet).

- **Some small-scale LLMs compute result without sufficient evidence support:** In Llama 3.1-8B response we can see

  "The velocity of the ego vehicle along the y-axis ('vy') is 3.0, indicating some lane change or adjustment."

  The vertical velocity ('vy') being 3.0 might indicate a lane change, but without specific lane position or nearby vehicle context, this assumption is weak. The model seems to be inferring behavior from incomplete data, making this a potential hallucination. In Claude 3.5 Sonnet response, we observe

  "The nearest vehicle ahead ($x = 2.0$, $y = 0.0$) is slightly ahead and in a different lane, posing no immediate threat."

  While it is true that the vehicle in another lane poses no immediate threat, this assessment doesn't consider the potential for lane changes or future behavior. This simplification could mislead the decision, suggesting incomplete safety analysis.

- **Ambiguous reference to task goals:** In Claude 3.5 Sonnet Response, we see

  "The ego vehicle is already in a right lane position, satisfying the preference for the right-most lane."

  This implies that simply being in the right-most lane satisfies task goals, but the task might prioritize safety or velocity over lane position in some scenarios. Overemphasizing lane positioning might not align with the broader task goals, which could lead to suboptimal decisions.
- **Small Scale LLMs still Lack of Comprehensive Trade-off Analysis:** Neither response effectively balances multiple task objectives such as maximizing speed, avoiding collisions, and minimizing lane changes. While they touch on individual objectives, the trade-offs between these objectives are not comprehensively discussed. Llama-3.1-8B and Claude 3.5 Sonnet responses make use of exact numerical values (e.g., reward values, vehicle positions) without fully explaining how they apply to the current decision-making process. These values may be hallucinated or overly generalized.

B. Novelty Comparison with Prior LLM enabled AD works

To address this comment, we summarized the unique aspects of our study compared to prior works in **Table I** of the response document. Moreover, we would like to emphasize the following unique aspects.

- **Limited Studies on Joint AD and V2I Communication Optimization Policies:** Previous frameworks, such as DiLu [1], introduced reasoning and reflection modules for autonomous driving (AD) decision-making, but did not integrate vehicular communication into their optimization. Similarly, "Drive Like a Human" [2] identified key abilities, such as reasoning and interpretation, for AD systems. However, it did not explore joint AD optimization with V2I communication.

  It is important to note that jointly optimizing AV motion dynamics and wireless connectivity is critical to balancing the trade-offs between communication handovers and speed of AVs in order to achieve the following [3], [4], **(1)** receive real-time traffic information from the wireless network [5], **(2)** access timely navigation data [6], **(3)** communicate effectively with surrounding vehicles [7], [8], **(4)** predict dangerous situations and make informed decisions [9].
- **Handling High Complexity:** AD and V2I observation space has at least $M_1 \times (N_{AD} + N_{V2I})$ entries at each time step, where N_{AD}, N_{V2I} represent the number of features in AD and V2I, respectively. Given this, the agent should decide on both transport and telecommunication actions for each AV which is challenging.
- **Distinct Prompt Engineering Approach:** To enhance token efficiency, we discretize transportation states into arrays representing AV observations and group target and surrounding AVs into matrices. This approach compresses transportation information within a token-efficient structure, enabling efficient decision-making without sacrificing environmental complexity.
- **Introduction of Experience Replay Pools:** Unlike prior works [1], [2], our framework generates distinct *good* and *bad* experience pools, storing non-collision and collision examples, respectively. These pools enable the system to leverage high-reward examples for decision-making, improving learning efficiency and robustness.

C. Novelty Comparison with Prior Joint V2I-AD optimization MORL approaches

Also, in prior studies, such as [3], [4], a joint Markov Decision Process (MDP) was designed for V2I communications and AD. The state space in these works included detailed kinematics-related features [10] for multiple AVs, represented as a $M_1 \times F$ array, where $F \rightarrow \{x_j, y_j, v_j, \psi_j, n_R^j, n_T^j\}$. This framework, while effective, posed challenges in terms of computational complexity and data processing.

Our proposed hybrid **LLM-DDQN** framework introduces several methodological advancements:

- We integrate LLM with DDQN to sequentially optimize AD and V2I decisions. The transportation decisions derived from LLMs (action $\mathbf{a}^{j,\mathrm{AD}}$) are iteratively fed as inputs into the DDQN-based V2I optimization, reducing the state space dimensions and enhancing computational efficiency.
- Unlike prior works that required six features per AV, the proposed V2I state space is defined as:
  $$\mathcal{S}^{\mathrm{V2I}} = [\mathbf{n}_R^j, \mathbf{n}_T^j, \mathbf{a}^{j,\mathrm{AD}}],$$
  where $\mathbf{n}_R^j$ and $\mathbf{n}_T^j$ represent the numbers of reachable RBSs and TBSs for AV j, and $\mathbf{a}^{j,\mathrm{AD}}$ is the transportation action produced by LLMs. This reduction from six features to three significantly eases computational burdens while retaining decision-making effectiveness.
- By leveraging LLMs' reasoning capabilities for transportation decisions, our framework offloads a significant portion of the decision-making process, allowing DDQN to focus on optimizing network connectivity and handoff control.

D. Training Complexity Analysis

In AD, timely decision-making is critical, especially as task complexity and vehicle speed increases. Below, we outline how our hybrid LLM-DDQN framework addresses these challenges:

- LLMs are deployed via **Ollama** [?] on edge servers, close to AVs, to reduce network latency. This eliminates delays caused by remote server interactions, ensuring that response times meet real-time requirements for AD.
- To reduce the computational burden, we provide LLMs with only the most relevant examples instead of all past examples. Specifically, we supply the top 5 closest past good and bad examples based on Euclidean distance, as shown in the *good/bad* example mechanism rather than all examples. This focused input ensures efficient processing while retaining high-quality decision-making, shown in the provided examples, the experience replay mechanism enables LLMs to learn and refine their decision-making efficiently.
- LLMs generate responses token-by-token. As soon as a decisive keyword—such as `FASTER`, `SLOWER`, `LANE_RIGHT`, `LANE_LEFT`, or `IDLE`—is detected, the action is applied immediately, bypassing the need to process the entire output. This significantly accelerates the decision-making pipeline.
- The highway simulation operates at 15 Hz [11], meaning the system processes 15 actions per second for each AV. This aligns with the decision latency constraints of AD systems, which require sub-100ms response times.

See sample videos on *RF-THz-Highway-Env* at https://patrickyanz.github.io/envelope-videos.html

1) Total Training Complexity:

To evaluate the impact of increasing task complexity, we conducted experiments to compare the training times of the proposed **Hybrid-LLM-DDQN** framework with benchmark algorithms [3], [4].

2) Test Setup:

We evaluated all methods across five test instances, varying key parameters: the desired minimum and maximum longitudinal velocities ($v_{\min}, v_{\max}$), the number of TBSs (n_T), and

Aspect	Dilu (Wen et al., 2024) [1]	Drive Like a Human (Fu et al., 2024) [2]	Proposed Approach
Scenario Description	Describes the **ego vehicle's lane**, **speed**, and the **state of surrounding vehicles** (e.g., positions, speeds).	Provides an **ego vehicle's state**, including lane, speed, and a step-by-step **thought process** for evaluating actions based on vehicle safety and environment.	Provides normalized **observations** of ego vehicle and surrounding vehicles (x, y, v_x, v_y), emphasizing **task-specific goals** and environment dynamics.
Reasoning Process	Linear: Checks **acceleration**, then **IDLE**, then **deceleration**. Evaluates distances and relative speeds.	Multi-step reasoning: Thought-based process with **action evaluations**, **conflict checks**, and **comparison of multiple options** (e.g., lane change safety).	**Experience Replay**: Learns from past actions to maximize rewards, minimize collisions, and avoid unnecessary lane changes. Leverages **training data** for better decision-making.
Task Goals	Maintain current speed if safe and avoid collisions.	Minimize collisions, maximize safety, and **choose the safest action** (e.g., IDLE, lane change).	**Achieve maximum velocity** while minimizing collisions. Reduce unnecessary lane changes. **Prefer the right-most lane** when safe to do so.
Action Choices	`FASTER`, `SLOWER`, `IDLE`.	`FASTER`, `SLOWER`, `IDLE`, `LANE_LEFT`, `LANE_RIGHT`.	`FASTER`, `SLOWER`, `IDLE`, `LANE_LEFT`, `LANE_RIGHT`.
Environment Features	Basic: Lane position, speed, acceleration.	Detailed: Evaluates actions based on **conflicts** (e.g., lane safety, acceleration feasibility).	**Normalized features**: Horizontal and vertical offsets (x, y), velocities (v_x, v_y). Accounts for task-specific rules like preferred lane and collision avoidance.
Strengths	Simple and interpretable.	**Comprehensive safety checks** ensure robustness and adaptability in complex scenarios.	Combines **reward optimization** with real-world goals (e.g., speed, safety, and lane preference). **Experience replay** improves decision-making over time.
Decision-Making Style	Reactive: Responds to immediate distances and speeds.	Thought-driven: Evaluates all possible actions step-by-step before choosing the safest option.	Data-driven: Uses **reinforcement learning techniques** (e.g., experience replay) for continuous improvement.

Table I: Comparison of Approaches for AD Decision-Making[Please take care of the column width in the table, e.g., the first column width should be reduced.]

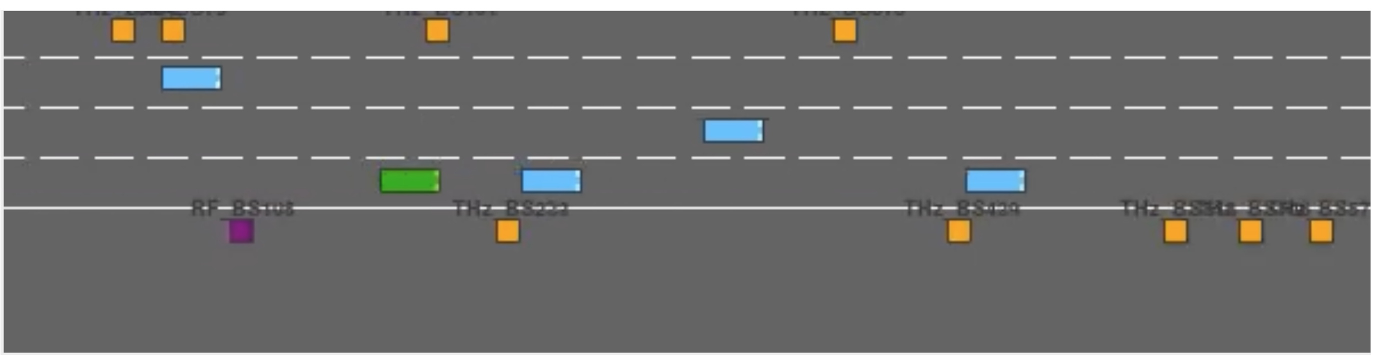

Figure 1: Sample simulation visualized simulation *RF-Thz-Highway Env*. The green rectangle box and blue rectangle boxes are target ego AV and The yellow and purple blocks represent the RF BSs and Thz BSs, respectively.

the number of AVs (M). These combinations are detailed in Table II.

Instance	$v_{\min}$	$v_{\max}$	n_T	M
I-(20,30,20,20)	20 m/s	30 m/s	20	20
I-(25,35,20,20)	25 m/s	35 m/s	20	20
I-(20,30,10,20)	20 m/s	30 m/s	10	20
I-(20,30,20,50)	20 m/s	30 m/s	20	50
I-(30,40,20,20)	30 m/s	40 m/s	20	20

Table II: Test Instances

Training Time Results:

We trained each algorithm for 4000 episodes until convergence on these the aforementioned test instances (presented in Table-II) and recorded the total training time, as summarized in Table III. The results demonstrate that the proposed **Hybrid-LLM-DDQN** achieves competitive training times, comparable to state-of-the-art V2I-AD multi-objective optimization benchmarks, even as task complexity increases.

Instance	MO-Q [3]	MO-DQN [3]	MO-Dueling-DDQN [?]	MO-PPO [?]	MO-DDQN-Envelope [4]	Hybrid-LLM-DDQN
I-(20,30,20,20)	30' 36"	31' 10"	30' 45"	31' 50"	30' 15"	**28' 59"**
I-(25,35,20,20)	27' 24"	27' 22"	28' 59"	28' 41"	**26' 25"**	26' 43"
I-(20,30,10,20)	30' 11"	30' 53"	30' 32"	31' 22"	**30' 06"**	30' 18"
I-(20,30,20,50)	32' 00"	32' 20"	31' 50"	33' 00"	31' 45"	**31' 22"**
I-(30,40,20,20)	21' 45"	22' 08"	21' 33"	22' 52"	21' 17"	**21' 02"**

Table III: Comparison of Training Times for Different Methods

From Table III, we observe

- Training time remains manageable even with higher vehicle speeds and greater task complexity, ensuring the framework's applicability to real-time systems.

- The proposed Hybrid-LLM-DDQN demonstrates efficiency while maintaining robust decision-making capabilities in high-speed environments where safety-critical, swift decisions are required.

E. Performance Comparison on different LLMs

Desire Velocity (m/s)	Model	Number of AVs M			
		M=10	M=20	M=30	M=40
15	ChatGPT-3.5	24.72/11.25/35.97	24.28/10.41/34.69	22.26/8.43/30.69	19.42/7.47/26.89
	Llama3.1-8B	25.38/13.66/39.04	24.99/10.45/35.44	22.17/9.09/31.26	20.29/8.07/28.36
	Llama3.1-70B	**26.12/15.21/41.33**	**25.41/12.26/37.67**	**23.49/10.28/33.77**	**21.71/8.22/29.93**
20	ChatGPT-3.5	21.18/9.42/30.60	20.44/7.08/27.52	19.91/7.55/27.46	18.33/4.89/23.22
	Llama3.1-8B	21.76/11.99/33.75	21.78/9.37/31.15	20.36/9.21/29.57	**20.81/6.93/27.74**
	Llama3.1-70B	**22.88/13.78/36.66**	**21.99/11.36/33.35**	**21.02/10.15/31.17**	20.04/6.72/26.76
25	ChatGPT-3.5	17.72/8.32/26.04	16.42/6.41/22.83	15.83/5.89/21.72	14.22/5.12/19.34
	Llama3.1-8B	**18.89/8.70/27.59**	16.59/6.38/22.97	16.22/5.94/22.16	15.47/5.39/20.86
	Llama3.1-70B	18.34/8.45/26.79	**16.78/6.46/23.24**	**16.38/6.03/22.41**	**16.05/5.81/21.86**
30	ChatGPT-3.5	16.51/7.31/23.82	16.44/5.77/22.21	15.38/5.71/21.09	14.78/5.32/20.10
	Llama3.1-8B	16.93/7.65/24.58	16.81/5.92/22.73	**16.51/5.82/22.33**	15.11/5.46/20.57
	Llama3.1-70B	**17.42/7.92/25.34**	**17.04/6.12/23.16**	15.42/5.81/21.23	**16.22/5.43/21.65**

Table IV: Evaluation performance on various LLMs.

We tested multiple state-of-the-art LLMs, including ChatGPT-3.5 [12], Llama3.1-8B [13], Llama3.1-70B [14], and Claude 3.5 [15]. As highlighted in the supplementary material, the performances of these models varied. In the main paper, we focused on Llama3.1-8B and Llama3.1-70B because they demonstrated superior performance in our simulation and evaluation.

We conducted experiments in the *RF-THz-Highway-Env* simulation environment [4], [11]. The experiments varied the number of AVs and desired velocities to simulate diverse highway scenarios. The evaluation metrics included AD rewards, V2I rewards, and total rewards, as defined in [3]. Table IV summarizes the comparative performance of ChatGPT-3.5, Llama3.1-8B, and Llama3.1-70B. Table IV highlights the superior evaluation performance of Llama3.1-70B and Llama3.1-8B compared to ChatGPT-3.5. These models consistently exhibit higher rewards across all metrics, showing their effectiveness in optimizing AD and V2I performance.

References

[1] L. Wen, D. Fu, X. Li, X. Cai, M. Tao, P. Cai, M. Dou, B. Shi, L. He, and Y. Qiao, "Dilu: A knowledge-driven approach to autonomous driving with large language models," in *12th International Conference on Learning Representations (ICLR 2024)*, 2024.

[2] D. Fu, X. Li, L. Wen, M. Dou, P. Cai, B. Shi, and Y. Qiao, "Drive like a human: Rethinking autonomous driving with large language models," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 910–919.

[3] Z. Yan and H. Tabassum, "Reinforcement learning for joint v2i network selection and autonomous driving policies," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 1241–1246.

[4] ——, "Generalized multi-objective reinforcement learning with envelope updates in urllc-enabled vehicular networks," *arXiv preprint arXiv:2405.11331*, 2024.

[5] F. Zhang, J. Xi, and R. Langari, "Real-time energy management strategy based on velocity forecasts using v2v and v2i communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 2, pp. 416–430, 2016.

[6] J. Rezgui, É. Gagné, G. Blain, O. St-Pierre, and M. Harvey, "Platooning of autonomous vehicles with artificial intelligence v2i communications and navigation algorithm," in *2020 global information infrastructure and networking symposium (giis)*. IEEE, 2020, pp. 1–6.

[7] Z. Zhang, Q. Wu, P. Fan, N. Cheng, W. Chen, and K. B. Letaief, "Drl-based optimization for aoi and energy consumption in c-v2x enabled iov," *arXiv preprint arXiv:2411.13104*, 2024.

[8] J. Shi, J. Du, Y. Shen, J. Wang, J. Yuan, and Z. Han, "Drl-based v2v computation offloading for blockchain-enabled vehicular networks," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 3882–3897, 2022.

[9] M. Malinverno, G. Avino, C. Casetti, C.-F. Chiasserini, F. Malandrino, and S. Scarpina, "Performance analysis of c-v2i-based automotive collision avoidance," in *2018 IEEE 19th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE, 2018, pp. 1–9.

[10] P. Polack, F. Altché, B. d'Andréa Novel, and A. de La Fortelle, "The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?" in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 812–818.

[11] E. Leurent, "An environment for autonomous driving decision-making," https://github.com/eleurent/highway-env, 2018, gitHub repository.

[12] OpenAI, "ChatGPT," 2024, version: GPT-3.5, Model: ChatGPT. [Online]. Available: https://openai.com

[13] Meta AI, "Llama3 7B," 2024, version: Llama3.1-7B. [Online]. Available: https://ai.meta.com

[14] ——, "Llama3 70B," 2024, version: Llama3.1-70B. [Online]. Available: https://ai.meta.com

[15] Anthropic, "Claude 3.5 Sonnet," 2024, model: Claude 3.5 Sonnet. [Online]. Available: https://www.anthropic.com