Reinforcement Learning Project - Task 4

Abdellah Oumida

abdellah.oumida@student-cs.fr

GitHub Repository

Other Members:
Idriss Mortadi, Abdeaziz Guelfane & Aymane Lotfi

April 2025

Abstract

This study investigates the impact of varying lane preference rewards on the behavior of reinforcement learning (RL) agents in a highway environment. Specifically, we analyze how different levels of the `right_lane_reward` parameter influence the agent's lane occupancy, collision rate, and average speed. The hypothesis posits that stronger lane preference rewards will lead to agents prioritizing the rightmost lane, potentially sacrificing speed due to congestion but improving safety by reducing the need for lane changes. Conversely, weaker or penalized rewards will encourage more lane changes, potentially increasing speed but raising collision risk. Our experiment involved training and evaluating multiple agents with varying reward structures, and the results demonstrate a clear trade-off between lane discipline (safety) and traffic navigation efficiency (speed).

Contents

1 Task 4: Experimental Study

1.1 Motivation and Hypothesis

In this experiment, we explore how varying the lane preference reward affects the agent's behavior in a highway environment. The original reward structure incentivizes staying in the rightmost lane (+0.5 reward). We hypothesize that:

- **Stronger lane preference rewards** will cause the agent to prioritize staying in the rightmost lane, even at the cost of reduced speed (due to traffic congestion), leading to fewer lane changes and lower collision rates.
- **Weaker or penalized lane preference rewards** will encourage lane changes, as the agent seeks to overtake slower vehicles in other lanes, increasing speed but also increasing collision risk due to more aggressive lane changes.

The reward function influencing the agent's behavior can be expressed mathematically as:

$$R_{\text{total}} = R_{\text{lane}} + R_{\text{collision}} + R_{\text{speed}} + R_{\text{other}}$$

Where:

- R_{lane} is the reward for lane occupancy, with positive values for the rightmost lane and negative values for lane changes.
- $R_{\text{collision}}$ is the penalty for collisions.
- R_{speed} is the reward for maintaining higher speeds.
- R_{other} are other rewards that we are not going to investigate in this study.

The impact of R_{lane} on the agent's behavior is the main focus of this experiment.

1.2 Experimental Design

1.2.1 Variables

The independent variable in this experiment is the lane preference reward (`right_lane_reward`), which is varied as follows:

$$\text{right\_lane\_reward} \in \{-0.5, 0, 0.5, 1.0\}$$

This variable penalizes or rewards the agent for staying in the rightmost lane.

The dependent variables are:

- Lane occupancy (percentage of time spent in each lane).
- Collision rate (percentage of episodes where the agent crashes).
- Average speed (average speed of the agent during evaluation episodes).

1.2.2 Methodology

1. **Training:** Four DQN agents are trained using different values for the `right_lane_reward` parameter $(-0.5, 0, 0.5, 1.0)$. The training consists of 15,000 steps, with hyperparameters set as follows: discount factor $\gamma = 0.95$, learning rate $\alpha = 1 \times 10^{-4}$, batch size of 32, buffer size of 1,000,000, and target network update interval of 500 steps. The neural network architecture uses two hidden layers with 128 units each.

2. **Evaluation:** After training, each agent is evaluated over 30 episodes to measure lane occupancy, collision rate, and average speed. The evaluation process tracks the lane index, speed, and collision events at each step.

1.3 Results and Discussion

1.3.1 Lane Occupancy Analysis

The lane occupancy is analyzed by calculating the percentage of time the agent spends in each lane. The formula for lane occupancy in lane L_i (where $i \in \{0, 1, 2, 3\}$ represents the lanes) is given by:

$$\text{Lane Occupancy in Lane } L_i(\%) = \frac{\text{Time spent in lane } L_i}{\text{Total time}} \times 100$$

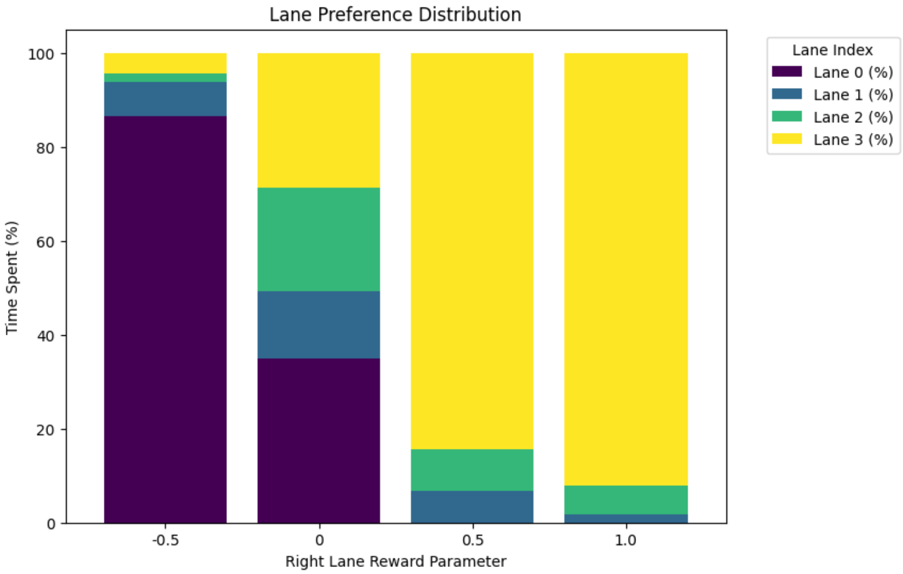


Figure 1: Lane preference distribution for agents with different right lane reward values.

As shown in Figure 1, agents with higher right lane rewards spend more time in the rightmost lane (Lane 3), with the trend scaling proportionally to the reward value. A neutral reward results in a more balanced distribution across lanes, while a negative reward shifts occupancy toward the leftmost lane (Lane 0).

1.3.2 Collision Rate vs. Lane Reward

The collision rate is calculated as the number of collisions per total steps in the evaluation episodes:

$$C = \frac{\text{Number of collisions}}{\text{Number of episodes}}$$

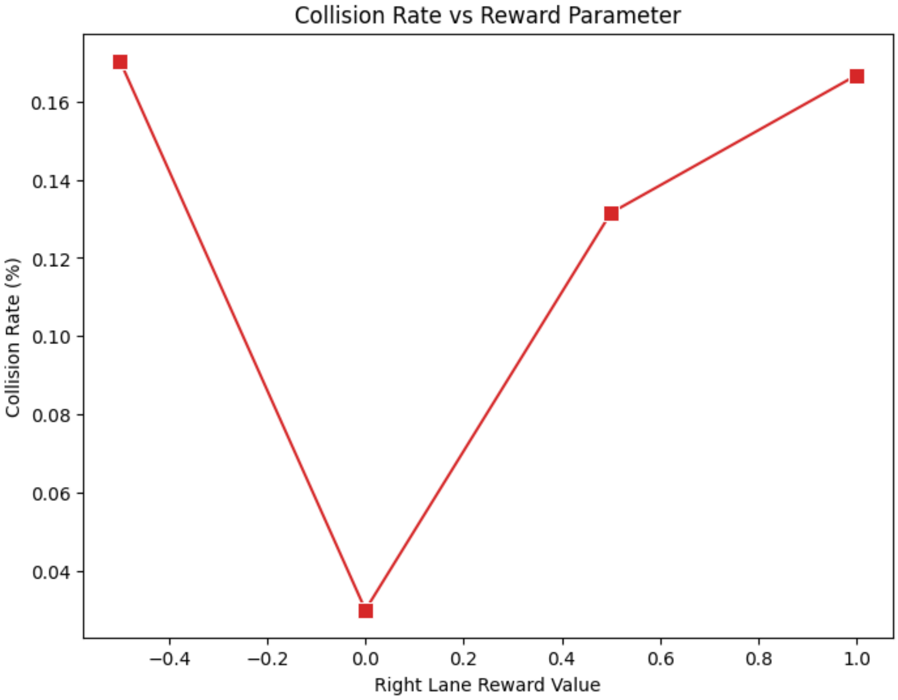


Figure 2: Collision rate distribution for agents with different right lane reward values.

Figure 2 illustrates a U-shaped trend in collision rates. Negative rewards result in higher collision rates due to frequent lane changes, whereas a neutral reward minimizes collisions by prioritizing survival. The less steep slope observed for positive rewards suggests that consistently driving in the rightmost lane reduces the likelihood of collisions, although not as effectively as the neutral reward.

1.3.3 Speed vs. Lane Reward

The average speed v of the agent is calculated as:

$$v = \frac{\sum_{t=1}^{T} v_t}{T}$$

where T is the total number of evaluation steps, and v_t is the speed at time step t.

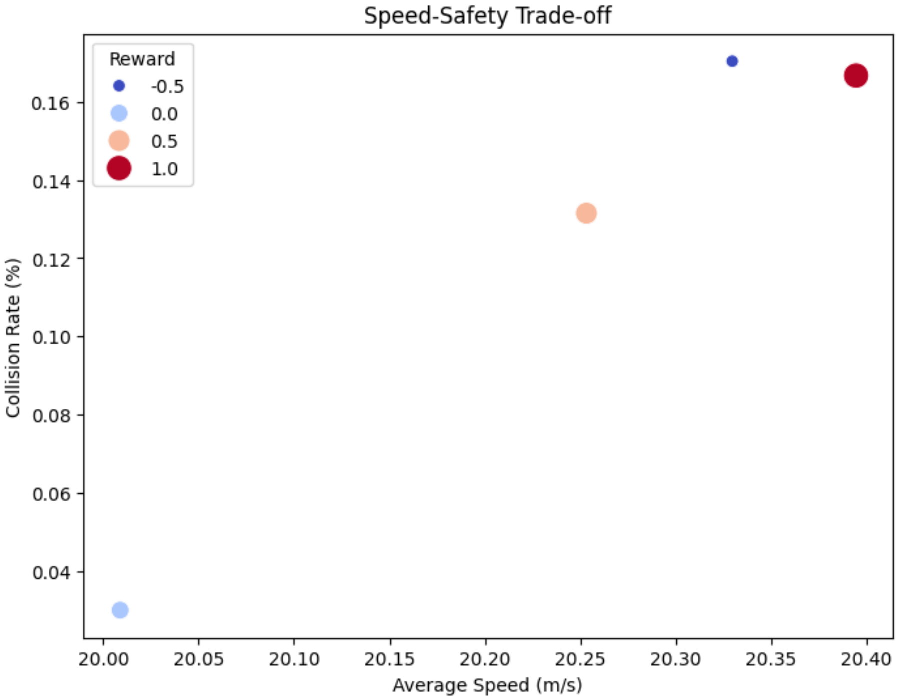
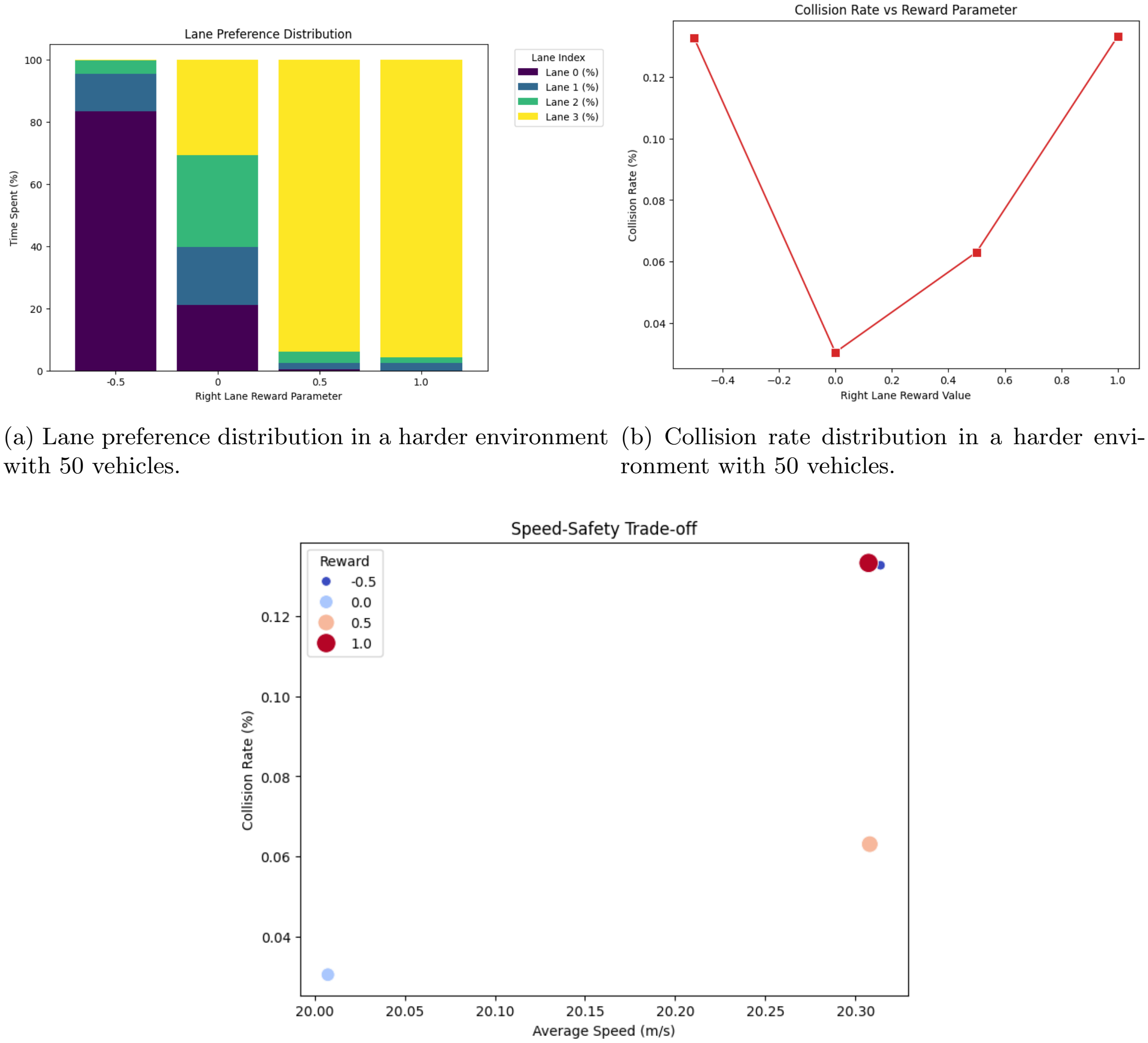


Figure 3: Speed and collision rate distribution for agents with different right lane reward values.

Figure 3 illustrates the relationship between speed and collision rates. A neutral reward yields moderate speeds and the lowest collision rates, emphasizing safety. Positive rewards increase speed but slightly elevate collision risk, while negative rewards result in cautious driving with lower speeds. We can also see a slight correlation between the collision rate and the average speed.

1.3.4 Evaluation on a Harder Environment

An additional evaluation was conducted in a more challenging environment with 50 vehicles, compared to the initial 15 vehicles, to test the agent's robustness. The results, shown in Figures 4a, 4b, and 4c, confirm the findings from the initial evaluation. A neutral reward (right_lane_reward = 0) continues to achieve the lowest collision rates, highlighting its effectiveness in prioritizing safety even with increased traffic density. Speed differences across reward values remain minor, likely due to the low high-speed reward parameter (set at 0.1), which does not strongly encourage efficiency over safety. The collision rate differences between reward variants are also not significant, suggesting that the agent adapts well and avoids collisions effectively across different reward settings.



(a) Lane preference distribution in a harder environment with 50 vehicles.

(b) Collision rate distribution in a harder environment with 50 vehicles.

(c) Speed and collision rate distribution in a harder environment with 50 vehicles.

Figure 4: Performance metrics in a harder environment with 50 vehicles.

1.4 Conclusion

The evaluations in both standard (15 vehicles) and harder (50 vehicles) environments reveal that a neutral right lane reward consistently minimizes collision rates, prioritizing safety. Positive rewards enhance lane discipline and speed but slightly compromise safety, while negative rewards increase collision risk due to aggressive lane changes. The minor differences in collision rates across reward values indicate that the agent adapts effectively to avoid collisions, regardless of the reward structure. However, the limited variation in speed suggests that the current reward parameters prioritize safety over efficiency, as the high-speed reward is insufficient to incentivize faster driving. These findings underscore the importance of reward shaping in autonomous driving systems to balance safety and efficiency, particularly in denser traffic conditions.

Bibliography

- Highway Env GitHub: `https://github.com/eleurent/highway-env`
- Stable Baselines3 Documentation: `https://stable-baselines3.readthedocs.io/`
- Deep Q-Network (DQN) Paper: Mnih et al., ”Human-level control through deep reinforcement learning,” Nature, 2015.