Autonomous Highway Overtaking

Project Proposal

Faculdade de Ciências da Universidade do Porto, Rua do Campo Alegre s/n, 4169-007 Porto

Introduction to Intelligent Robotics (CC3046)

Guilherme Magalhães (202205505)

Miguel Amaro (202106985)

João Baptista (202207629)

1 Project Goals

This project aims to develop and compare two strategies for an autonomous emergency vehicle overtaking maneuver in a highway scenario.

Regarding the scenario, there will be a lead car (car_1) maintaining constant speed, and an autonomous vehicle (car_2) performing the overtaking maneuver. The objective is to train the autonomous vehicle so that he can perform an overtaking maneuver, while obeying the rules mentioned in the following chapter.

The primary objective is to implement both a Reinforcement Learning (RL) approach and a rule-based system to materialize this scenario in a robot simulator. Once the initial objective is attained, we will then move on to the performance evaluation, where the simulation will be performed under both ideal conditions (clean LiDAR, constant speed) and more challenging scenarios (noisy LiDAR, variable speeds).

2 Proposed Approach

In order to start developing our work, we first need to establish the rules our simulation will follow. Thus, none of these rules shall be broken: both cars are on a 3-lane highway; both cars start in the middle lane; car_2 starts behind car_1; car2_must not establish contact with any object other than the road itself; car_2 must only overtake using the leftmost lane; car_2 must not return to the middle lane.

We are now capable of materializing our idea in the Webots[1] simulator, using Python[2]. We will use a pre-existing Webots world ("Highway Overtake")[3] that consists of an obstacle-free straight highway. We will also make use of 2 models from "Webots for automobiles PROTO nodes"[4],CitroenCZero/CitroenCZeroSimple for car_1 and MercedesBenzSprinter/MercedesBenzSprinterSimple for car_2.

For our automated car, we will incorporate a LiDAR sensor with 360 rays and a field of view of 180º. This will allow the robot to detect nearby objects and, as such, know when he is able to start/stop the overtaking maneuver.

Regarding the training of the automated car, we will use two separate approaches, whose performance we will compare in a later step. The first approach relies on Reinforcement Learning, where we will use the PPO algorithm, since it works with continuous and discrete actions (suitable for steering, acceleration, and brake control) and it performs good on noisy environments (important for the non-optimal scenarios). Furthermore, its stability and sample efficiency make it the strongest contender. On the other hand, our second approach will not use any type of learning. and will instead base its actions on explicitly discretized rules (rule-based system). This allows for a baseline comparison, and a safer and more reliable simulation.

3 Proposed Experiments for Evaluation

As mentioned in the first chapter, we intend to test our simulation with varying conditions. We will divide our experiments in two different scenarios:

Baseline Scenario

For our baseline performance test, we will run the models on an optimal-condition scenario, where the LiDAR is noise-free, and the lead car is at constant car speed.

Non-optimal Scenario

We will then repeat the simulation in a scenario with challenging conditions, where the LiDAR is subject to noise and the lead car changes its velocity. In order to analyze the robustness of our models, the noise added will have 2 different levels and the changes in velocity will be randomized.

Both these scenarios will be repeated for one hundred trials each, where we will be measuring the following performance metrics: success rate of the overtakes (percentage); maneuver time (seconds); minimum inter-vehicle distance (meters). With these metrics we plan to build a table that allows us to compare each model in each scenario.

4 Milestones Outline

Week 1 – April 3rd

- Setup environment and robots

Week 2 – April 10th

- Development of rule-based system controller

Week 3 – April 17th

- Development of rule-based system controller (continuation)
- Run baseline experiment using rule-based controller to collect metrics

Week 4 – April 24th

- Research the RL algorithm and start first implementation

Week 5 – May 1st

- Implementation of RL model (continuation)
- Run baseline experiment using RL model to collect metrics

Week 6 – May 8th

- Development of the non-optimal scenarios (LiDAR noise and velocity changes)

Week 7 – May 15th (checkpoint)

- Run both models on the non-optimal scenarios to collect metrics

Week 8 – May 22nd

- Final evaluation of the models
- Analysis and comparison of final results

Week 9 – May 29th

- Scientific paper
- Final revision of the code, simulation and results
- Preparation for presentation

5 Checkpoint Goals

For the May 15th checkpoint we intend to deliver the complete implementation of the baseline scenario with a functional Webots environment and the functional rule-based system. We will also include the initial RL framework, along with comparative metrics between approaches, methodology documentation, and preliminary results analysis.

6 Referências

[1] https://cyberbotics.com/

[2] https://www.python.org/

[3] https://cyberbotics.com/doc/automobile/highway-overtake

[4] https://cyberbotics.com/doc/automobile/proto-nodes?version=R2025a