TÉCNICO LISBOA

Using autonomous vehicles to improve traffic conditions

Sofia Afonso Campos de Carvalho

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Francisco António Chaves Saraiva de Melo
Prof. Ana Maria Severino de Almeida e Paiva

Examination Committee

Chairperson: Prof. Pedro Tiago Gonçalves Monteiro
Supervisor: Prof. Francisco António Chaves Saraiva de Melo
Members of the Committee: Prof. Rui Miguel Carrasqueiro Henriques

November 2022

This work was created using LaTeX typesetting language
in the Overleaf environment (www.overleaf.com).

Acknowledgments

First of all, I would like to thank my parents for all their love and support, for always giving me the opportunity to learn about the subjects that interest me, and for their guidance and patience when I needed the most. I would also like to thank my dear grandparents, stepparents, aunts, and uncles for always believing in my capabilities and keeping high ambitions for my future, filling me with love and determination. Additionally, I want to thank my brother Rodrigo for always caring after me and for clearing the path in front of me, inspiring me with his own courage and successes.

I would like to thank my dissertation supervisor Prof. Francisco Melo, for guiding me through this project and for helping me overcome the difficulties that came up, ultimately making this project possible.

Furthermore, I want to thank my dearest friends Rita, Marta A., Marta G, Afonso, and Guilherme, for always cheering me up, for celebrating my achievements with me, and for helping me grow from my mistakes. For helping me face the challenge of starting this project when I was away from my family and friends, I also want to thank my Erasmus friends Osman and Lior, who showed me a home away from home.

Finally, I would like to thank my boyfriend Duarte for going through this journey with me, for sharing my difficulties and giving me the strength to face them, for always making the time to help me, and, most importantly, for always believing in me. For all your love, thank you.

This work would not be possible without each and every one of you—as such, I sincerely thank all of you.

Abstract

This work focuses on demonstrating the possible advantages of designing autonomous vehicles (AV) to improve traffic conditions. Given the constant improvement of the capabilities of autonomous vehicles and the lack of effective methods to counter traffic congestion, AVs have become a potential asset for finely managing traffic. We document and briefly describe the different existing approaches on this topic, from which we decide to address the traffic-critical setting of an open multi-lane highway with a merging single road. We thus present a deep reinforcement learning strategy that trains the system of AVs on that road to minimize the congestion generated by the merge. This solution is tested in a traffic micro-simulator and proven to effectively improve the network's outflow and the vehicles' average speed at a 10% AV penetration rate. Replays of the simulations can be seen in SimulationReplays[1].

Keywords

Autonomous vehicles; Traffic congestion; Deep reinforcement learning

[1] https://drive.google.com/drive/folders/1sa5y2FAWpInOHvYFTHaO56dgzNgIEE-j?usp=sharing

Resumo

Dado o contínuo desenvolvimento das capacidades dos carros autónomos e a atual falta de métodos eficientes para combater a congestão do trânsito, este trabalho realça as possíveis vantagens da utilização dos carros autónomos como ferramenta para a melhoria das condições de trânsito. Começando pela documentação e comparação de algumas das estratégias já desenvolvidas para este tema, apresentamos em seguida uma solução que beneficia o trânsito numa estrada aberta com múltiplas vias e com uma secção de união com outra estrada (esta tendo apenas uma via). Utilizando deep reinforcement learning para treinar os carros autónomos na estrada principal, esta solução leva o sistema a desenvolver comportamentos que minimizam a congestão inevitavelmente gerada pela união das duas estradas. Finalmente, testamos a nossa solução num micro-simulador de trânsito. Com 10% do tráfego autónomo, provamos desta forma que esta estratégia leva efetivamente a um aumento da velocidade média dos carros, bem como a um aumento do fluxo médio de carros na estrada. Disponibilizamos vídeos das simulações em SimulationReplays[2].

Palavras Chave

Carros autónomos; Congestão de trânsito; Deep reinforcement learning

[2] https://drive.google.com/drive/folders/1sa5y2FAWpInOHvYFTHaO56dgzNgIEE-j?usp=sharing

Contents

List of Figures

List of Tables

Acronyms

AV	Autonomous Vehicle
HDV	human-driven vehicle
IDM	Intelligent Driver Model
MDP	Markov Decision Process
ML	Machine Learning
MLP	Multi-Layer Perceptron
POMDP	Partially Observable Markov Decision Process
POSG	Partially Observable Stochastic Game
PPO	Proximal Policy Optimization
RL	Reinforcement Learning
SAE	Society of Automotive Engineers
TRPO	Trust-Region Policy Optimization

1

Introduction

Contents

Traffic congestion is a serious problem in big cities around the world. Fig. 1.1 demonstrates the great amounts of time that people lose every day in traffic, showing that the current methods to counter the problem are lacking. These methods (for instance, constructing better roads) are often costly and time-consuming, proving overall not appropriate for solving a worldwide problem. Therefore, there is an urgent need for alternative ways of countering traffic congestion.

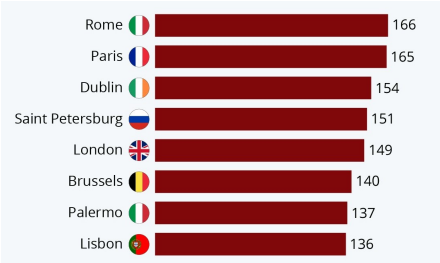

Figure 1.1: Average hours lost to congestion per driver in major European cities in 2019. Taken from https://www.statista.com/

On another note, autonomous vehicles (AVs), or self-driving cars, have been turning into valuable assets. For one, the capabilities of AVs have been continuously improving (as is reported in the next section). Additionally, the amount of AVs on the road has been quickly growing, and this growth is expected to continue in the next years (see fig. 1.2). AVs are currently simply designed as a means of transport, with the goal of getting the owner from one place to another. However, this approach might not be considering their full potential—in effect, people are sharing the roads with robots and could use them, for example, to benefit traffic conditions.

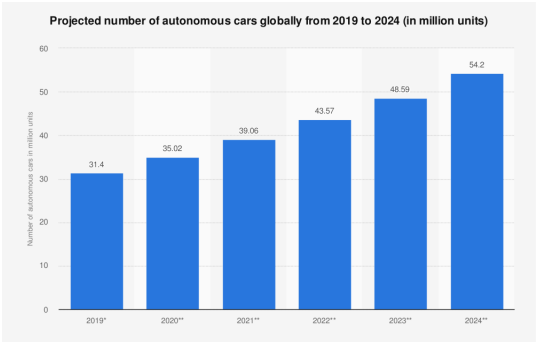

Figure 1.2: Projected number of AVs globally from 2019 to 2024. Taken from https://www.statista.com/

That said, this work is based on the premise that, with the right knowledge about the current state of traffic, there are optimal driving behaviors that can be followed to avoid traffic congestion. In traffic-critical scenarios like intersections or roads with merges, the drivers should be able to reduce or even avoid congestion when following a calculated optimal behavior. Moreover, the drivers may even be able to surpass the need for common non-optimal control mechanisms, like traffic lights and ramp meters. That said, human drivers do not usually have access to this required information, and even if they did,

they may favor their own goal (quickly reaching their destination) over improving traffic. AVs, on the other hand, can simply be programmed to follow the optimal behavior when in situations that commonly lead to congestion, and can easily assess the state of traffic by communicating with other AVs on the road. Additionally, the ability to communicate with other AVs allows them to create new complex traffic-beneficial strategies that require the cooperation of multiple vehicles.

1.1 Problem Description

It follows naturally that we could exploit AVs to improve traffic. To this end, we first need to identify which situations commonly lead to congestion, to then document which practices can be beneficial in those situations, and finally develop the AV control algorithms that implement those behaviors.

After exploring the existing works on this topic, we noticed that there were no presented solutions for improving traffic on the critical scenario of an on-merge section in an open road with multiple lanes. Additionally, we found that the deep reinforcement learning (deep RL) approaches that trained the AVs to learn an optimal traffic-beneficial behavior were often very successful. Therefore, our work focuses on developing a deep RL-based AV control strategy that prevents traffic congestion in a multi-lane highway with a merge section. Finally, note that, since one AV cannot impact the traffic of an entire road, our strategy inevitably involves coordinating with other AVs and influencing the human-driven vehicles so that they all work towards the same goal.

In summary, this work addresses the following research question: how can we design autonomous vehicles to improve traffic conditions?

1.2 Contributions

We highlight the contributions of this work:

- The documentation and comparison of existing solutions for autonomous vehicle control algorithms that prevent traffic congestion;
- The development of an autonomous vehicle control strategy that improves traffic conditions on an open multi-lane highway with a single-lane on-merge section;
- The implementation of this solution in a traffic simulation that considers the existence of both autonomous and human-driven vehicles, and the evaluation of its effectiveness.

1.3 Organization of the Document

The remainder of this thesis is organized as follows: In chapter 2, we explore the history of autonomous vehicles and clarify some concepts that will be used throughout the document. In chapter 3 we document some of the existing works on using AVs to improve traffic conditions. We then describe our envisioned strategy and present an overview of how our project works in Chapter 4, and in Chapter 5 we explain in detail each of the performed experiments. In chapter 6, we present the results of the experiments and comment on the effectiveness of our solution. Finally, in Chapter 7 we reflect on the overall contributions and limitations of our project.

2

Background

Contents

Since our project focuses on the topic of autonomous vehicles, we found it interesting to explore how they first originated and how they were improved over the years. As such, we document their history in the following section. We then further contextualize our project by clarifying some less trivial concepts that are used throughout this document.

2.1 The evolution of autonomous vehicles

A CAV is a connected and autonomous vehicle. An autonomous vehicle, also known as a self-driving car, is a vehicle that is able to operate with little to no human assistance, making use of a variety of sensors, like GPS and radar. In 2014, the Society of Automotive Engineers (SAE) defined six levels of automation for these cars, which are described in fig. 2.1. A connected vehicle is one that, using WIFI and radio, is able to communicate with other vehicles. An important note is that, currently, the term AV is commonly used to refer to CAVs, being implicit that an AV is also connected.

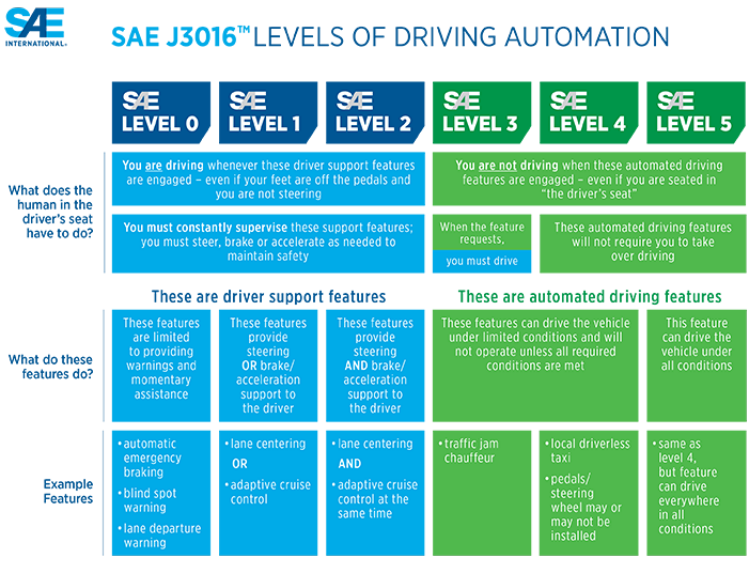

Figure 2.1: Levels of automation, as defined by SAE

The first records of autonomous vehicles date from the 1920s—in 1925, Houdina Radio Control developed a radio-controlled car, which received control signals from a normal car in front of it. In the 1950s, RCA Labs took a different approach, burying circuits on the pavement of a highway and instead using them to control the vehicle. In the next decades, the advantages of these systems were researched and their adoption was estimated to result in a 50% increase in road capacity and a 40% reduction in road accidents. 1989 then marked the base of contemporary control strategies, corresponding to the first use of neural networks in the control of AVs, by the Carnegie Mellon University. Particularly, Google began developing its driverless car project in 2009 and experimented with its technology in 2012. In 2014, Tesla announced its first AutoPilot, capable of lane control with autonomous steering and braking.

Distinctively, Tesla's system was able to receive updates and improve over time. In 2020, the first regulations on autonomous features were defined, and multiple autonomous public transport vehicles were made available. As of now, the adoption of AVs is becoming more common, and AVs are expected to continue improving rapidly.

Throughout these years, one of the most complex concerns was regarding the relationship between humans and driverless cars. In particular, measuring the humans' sensibility to AVs is critical for the assurance of safety when designing AV control algorithms, if not for the general decision of whether or not to include autonomy in traffic. Other problems, like how driverless vehicles should perceive humans around them, as well as how humans and robots should interact and communicate have raised ethical problems and have thus been thoroughly discussed [7–9].

These questions prove especially important since fully autonomous traffic is not expected to be a reality in the near future. Consequently, control strategies are necessarily developed on the scope of mixed-autonomy traffic and have to envision solutions for interacting with humans in human-driven vehicles. That said, when developing an AV control strategy, the biggest concern should forcibly be assuring safety guarantees for the owner and for the humans around it, only then followed by optimally achieving the owner's goal. Furthermore, when striving to develop a solution for AVs to improve traffic, the already existent control strategies should be outperformed without compromising their current operation or guarantees on the owner's safety and goal.

Even so, it should be noted that any AV control strategy that aims to improve traffic will be forced to favor the most beneficial behavior for traffic over the most efficient behavior for the owner, requiring the owner to sacrifice their own time to benefit others—naturally, the question comes up, will the AV owners be willing to collaborate and adopt these traffic-beneficial solutions, even if the solutions prove slightly prejudicial to them?

2.2 Terminology

Throughout this work, we frequently make use of concepts on the scope of machine learning. As such, these concepts are defined below.

Reinforcement Learning is one of the 4 basic categories of Machine Learning. Machine Learning (ML) is a type of artificial intelligence where a system uses previous data to "learn" and more accurately predict new outcomes, without being explicitly programmed to do so. Reinforcement Learning (RL) distinguishes itself by working through trial and error—an agent following an RL algorithm has a defined goal and a set of actions that it can perform to achieve the goal, and is rewarded or punished for performing an action that is, respectively, beneficial or prejudicial towards the goal. The agent then uses this feedback to update an internal policy. In the context of ML, a policy is a probability distribution over

the possible actions (depending on the state), ultimately guiding how the agent should act. Therefore, an agent will attempt to define an optimal policy that maximizes its expected reward. On another note, Deep Learning is a family of ML strategies that use deep artificial neural networks (neural networks with multiple layers) to repeatedly process the input data into defining features. Deep RL is thus a family of techniques that combine deep learning with reinforcement learning, using neural networks to process the gathered information into actions, and ultimately representing the RL-learned behavior as neural networks. Finally, Transfer Learning works by storing the knowledge gained while training in one task (for example, storing the parameters of the resulting optimal policy) and then applying it to improve training on a different but related task.

There are two main types of RL algorithms: model-based and model-free. In model-based RL, the agent constructs an internal model of the problem given its experience to the moment, and then constructs its policy according to this model. In model-free RL, the agent uses its experience to directly learn its policy (or an action-value function, which informs how good each action is at a given state), without the use of a world model. A Markov Decision Process (MDP) is a mathematical framework often used in model-based RL problems, defined through a set of elements: the states an agent can be in; the actions it can take; for each action, the probability of transitioning from one state to another (called probability functions); the reward function for the agent's actions. Following this model, the agent's policy is first initialized to some value. Then, at each timestep, the agent is at some state, chooses an action according to its current policy, transitions to another state according to the probability functions, receives a reward, and updates its policy function according to what it did and the reward it received from doing so. This way, the agent's policy is eventually able to accurately guide it to the goal. A Partially Observable Markov Decision Process (POMDP) is a commonly used variation of an MDP that extends it by including information on what an agent senses at each time step (the observations), assuming that it is unable to know its own exact state. In a POMDP, the policy is calculated according to what the agent can observe, instead of the actual states it is in.

That said, a policy is updated according to a policy optimization algorithm, that strives to maximize the expected reward. Following the Trust-Region Policy Optimization (TRPO) algorithm, each policy update is bounded by a defined maximum difference between the new and old policies, expressed in terms of KL-divergence. The update thus corresponds to the largest possible improvement of the system's performance that satisfies this constraint. The Proximal Policy Optimization (PPO) algorithm is a simplification of the TRPO algorithm, using essentially first-order methods to keep the new policy close to the old, while the TRPO algorithm requires complex second-order methods.

In the scope of this work, a centralized policy is one that is calculated by a centralized controller after joining the information sensed by the set of agents that are within its reach. In contrast, a distributed policy is calculated by each agent, using only the knowledge obtained by itself.

Game theory allows to model a game and obtain optimal strategic decision-making. A game is any situation that involves and is dependent on the actions of two or more rational agents, the players. Each player has a given information set at each time step and gets a given payoff from arriving at an outcome of the game. A Nash Equilibrium is an outcome from which no player has the incentive to deviate since no player is able to increase their payoff by changing their decision unilaterally.

3

Related Work

Contents

In this chapter, we explore and discuss relevant works that have been developed on how to improve traffic conditions using autonomous vehicles. The works are organized into subsections depending on their main focus, so that different works that share the same general goal can be compared.

3.1 Robots designed to influence humans

In the context of using AVs to improve traffic, the considered scenario is usually that of mixed autonomy (where only a percentage of the vehicles are autonomous while the others are human-driven), rather than that of fully autonomous traffic. Therefore, the need for humans to cooperate frequently arises, as the AVs' actions alone are not sufficient to significantly change traffic. Consequently, the used strategy in most experiments involves purposely influencing humans into acting towards the desired goal.

For that reason, before diving into the studies and experiments designed specifically in the context of autonomous vehicles and traffic improvement, we first go through a few more general works that focus on designing robots to influence the humans they are interacting with. The main idea is to model the human's reactions to the robot's actions – by first gathering knowledge on the human's goals and beliefs on the robot and the environment – and then exploit this model, making the robot choose the action that will trigger the desired human reaction.

In [10], the authors explore the idea that, when a robot (R) is aware that a human (H) trusts that R's actions are rational, R may behave in an unexpected way—a way that exploits H's trust to more efficiently reach R's goal.

The used setting is a game where H does not know R's objective, but R knows both. R will thus exploit H's uncertainty about R's goal. In the most relevant examples, R and H shared the same goal. The authors experimented by varying the way R models H. They compared four solutions.

In the first solution, R is conservative, assuming H knows R's goal. R is unable to exploit the uncertainty, as it assumes there is none. In the second one, R is optimistic, assuming it can make H believe that R has an arbitrary objective. Although this solution would be most profitable for R, it is unrealistic. In the third one, R assumes H is rational, implying that H knows that R is aware of H's uncertainty. However, realistically, H does not act as completely rational. H trusts that R will follow a simple goal.

Therefore, the fourth solution is a more accurate simplification of the rational one, where R models H as trusting. Here, R assumes that H thinks R is acting conservatively. Using this solution, R ends up purposely moving away from the shared goal, to make H think that R has a different objective. H is thus forced to counter R's actions and ends up getting more involved in the game.

This work, through the use of the trusting model, brings the interesting starting point that it is possible for robots and humans to share communicative actions that eventually help the robots reach their objective.

In the next work, we take a step further and focus on how to efficiently influence a whole team of humans, rather than focusing on the individual influencing process. The authors of [1] argue that there are necessary hierarchies between a team (leaders and followers) and that the robots can exploit them— if they can identify a leader of a group, influencing this leader will result in changing the actions of all its followers, bringing the same results as influencing each human individually, while being significantly more efficient. The used setting involves a robot, a team of human agents, and a set of goals. A preferred goal is chosen, however, it is unknown to the humans, and thus the humans can choose to target any goal from the set. For the robots, the objective is to influence the whole team into targeting the preferred goal.

To this end, the robot is designed to first understand and model the team's hierarchy, i.e. which humans are following others' actions and which humans are leading. Note that there may be multiple groups in the team that do not share a common leader and thus have no connection with each other. Then, the robot finds the most influential leader, which corresponds to the human agent that has the most followers and that is not already targeting the preferred goal (otherwise it does not need to be influenced). Finally, the robot influences this most influential leader into the preferred goal, effectively correcting them and their followers with one move. The robot then repeats the procedure, until the whole team is targeting the preferred goal.

Fig. 3.1 exemplifies this strategy, illustrating the hierarchy models for two possible scenarios. The green circles are goals, the orange triangles are human agents and the black triangles are robot agents. An arrow between, for example, agent 1 (a1) and agent 2 (a2), signifies that a1 is following a2. An arrow between an agent and a goal shows that the agent is targeting that goal. That said, if the team's hierarchy corresponds to the graph shown in fig. 3.1a, then there is only one team, and all the agents follow the same leader, a2. In this case, a2 is the most influential leader and the robot focuses on making a2 target the preferred goal. If the hierarchy is as shown in fig. 3.1b, then there are 2 teams, where a1 is the leader of a team that is not targeting the preferred goal, and a2 is the leader of a team that is targeting the preferred goal. In this situation, a1 is the most influential leader.

The findings in this paper are scalable to different team sizes and applicable to various tasks, therefore also proving relevant to the context of our work— given the scale of the problem of improving traffic conditions, influencing each human individually is unrealistic, and a valuable alternative would be to take advantage of the hierarchies between humans to develop more efficient strategies.

The conclusions found in the previous works lead to a new and valuable technique for AV control mechanisms in mixed-autonomy scenarios: the fact that humans can be influenced by robots suggests that human drivers can be influenced by autonomous vehicles as well. This knowledge allows the

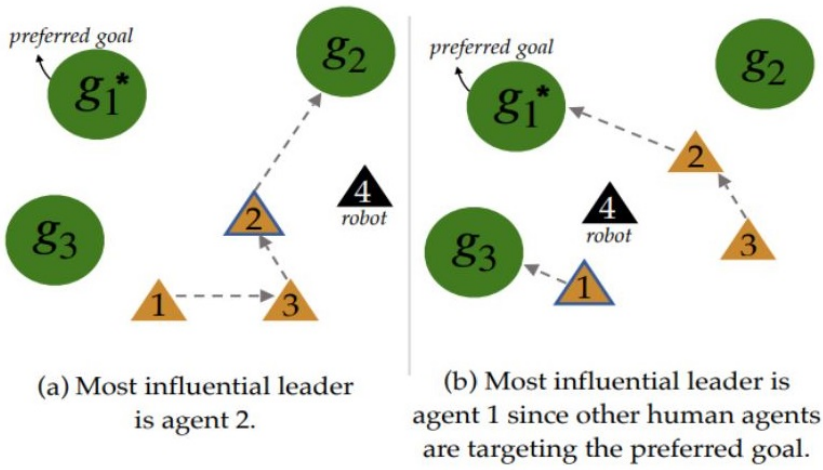

Figure 3.1: Examples of graphs for possible team hierarchies and respective most influential leaders. Taken from [1] (Fig. 3)

development of more efficient and complex solutions to improve traffic conditions that count on the cooperation (even if unintentional) of not only AVs, but also the humans around them.

3.2 Autonomous vehicles designed to maximize road utility

Moving into our context of traffic and autonomous vehicles, and given that our solution focuses on the context of multi-lane roads, we studied two works that explored the idea of reducing congestion by balancing the lane usage between the vehicles on a road, to maximize the roads' capacity and utility. The idea is that, instead of blindly building new roads to allocate more space, we can investigate how to make the best use of the roads that are already available and whose capacity might not be fully utilized currently.

The authors of [11] presented the first step towards countering this problem: they present a model for the lane-change dynamics of a vehicle traveling on a multi-lane road. Distinctively, this work accounted for the significant differences between lanes in a multi-lane road - like its free-flow speed, wave speed, and jam density - which were usually ignored. The proposed lane-change model is very complex, taking into consideration details such as the density variations between the lanes, the human's interest in maintaining their route, or the human's interest in keeping to the right.

The authors extensively tested the model using real-world data to conclude that their model made accurate estimations of the lane flow distribution. This makes the proposed model a very useful resource for balancing lane usage—for a controller to achieve balanced lane usage on a road, it will inevitably have to reorganize the vehicles on that road, making it fundamental to first predict how these

vehicles would naturally be organized, as well as their usual lane-change behavior, to then either make use of their natural lane-changes in the reorganization or to be aware of when the vehicles would have the incentive to move away from the planned organization. While we do not include this model explicitly in our solution, complex lane-change models like this are built into the frameworks that we used to simulate vehicle behavior, showing that lane-change dynamics are indispensable when striving to accurately represent traffic.

The work [2] takes the next step, effectively proposing a solution for balancing lane usage using autonomous vehicles. The goal of this work is to develop a centralized AV controller that, after finding the maximum affordable increase to a road's capacity, calculates the optimal vehicle configuration, and rearranges the vehicles according to this configuration.

To calculate the road's capacity (the number of vehicles that can simultaneously travel on the road) and the optimal configuration, the controller uses a helpful property of autonomous vehicles known as platooning. Platooning is defined as the ability of 2 sequential AVs to safely keep a smaller headway than the one that would be required if a human-driven vehicle (HDV) was involved. It follows that the optimal configuration is one where, in each lane, there will optimally be either only AVs or only HDVs. If a mixed lane exists, then the AVs will rearrange themselves so that no HDV is between them.

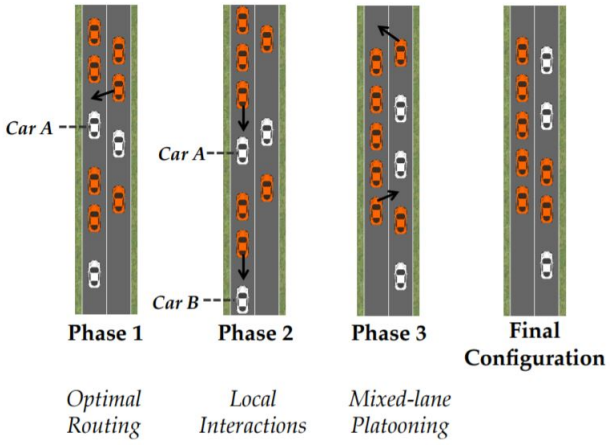

Figure 3.2: Phase 1: the AVs follow optimal lane assignment. Phase 2: the AVs influence the HDVs to follow optimal lane assignment. The acting AVs pair with the HDVs A and B. Phase 3: the AVs platoon in the mixed lane. Taken from [2] (Fig. 2)

For the rearranging process, the controller makes use of the property that was concluded on section 3.1, that robots (in this case AVs) are able to influence humans (in this case, the HDVs) into taking a desirable action. The process is distributed to each AV, which individually interacts with the HDVs around it (making them change lanes, slow down, or speed up) to reach the configuration that is intended by the controller. This process is illustrated in fig. 3.2. Unfortunately, the authors highlight the

solution's complexity and consequent difficulty to compute.

Besides the benefits that these mechanisms bring by themselves, the techniques for lane balancing can also be combined with other approaches to make them more efficient—the constant maximum usage of the road's capacity provides an additional efficiency gain in every scenario that involves a multi-lane road. Therefore, our solution takes inspiration from the algorithms in this section, recognizing the multiple lanes as a resource to reorganize the vehicles in a way that is optimal for our goal.

3.3 Autonomous vehicles designed to dissipate stop-and-go waves

Ideally, every vehicle would travel at its optimal performance, showing constant speed and spacing. However, this is not the case in natural highway traffic, as small disturbances inevitably form and propagate backward, eventually expanding and forming jams. This effect is known as stop-and-go waves and is commonly the cause of traffic congestion. Therefore, many experiments focus specifically on programming autonomous vehicles to mitigate these waves, usually considering scenarios that involve lane drops or on-ramp sections to generate the initial disturbance. Since our project shares this goal, we go through some of the works that focus on dissipating stop-and-go waves.

In the paper [12], the authors take a theoretical look at the problem of stop-and-go waves. Considering a single-lane ring road with periodic disturbances, they define the waves as coming from a property known as string-instability—the amplification of a disturbance as it passes down a singly connected chain. As such, they calculate the AVs' ability to string-stabilize the road, that is, to successfully avoid the formation of a wave after a disturbance has occurred, ideally allowing the vehicles to travel optimally at all times, despite the periodic disturbances. They arrived to the important conclusion that, if at least 6% of the considered vehicles are autonomous (and uniformly distributed between the other vehicles), then an optimal controller is able to efficiently string-stabilize the lane.

The problem was solved with unrealistic conditions such as the vehicles needing no safety headway between them, and thus, the exact presented results only stand in the context that is considered in this work. However, this work brought the important notion that stop-and-go waves can be avoided with only a relatively small percentage of autonomous vehicles on the network (a small AV penetration rate), contradicting the general first guess that traffic would need to be mostly autonomous for these strategies to have a significant impact. This way, it served as starting point and motivation for other researchers to develop more realistic methods with the goal of countering stop-and-go waves.

The work [13] distinguishes itself by considering the more complex scenario of open roads. This

work presents a traffic control strategy for a group of AVs to dissipate the effects of stop-and-go waves on an open single-lane highway with a merge section to generate the perturbations. Once again, the context is that of mixed-autonomy traffic, and the AVs are assumed to be uniformly distributed.

The authors define a centralized RL controller with access to the AVs' observations and actions. The plan is that, after training the controller, the AVs that are at the beginning of the road learn to slow down or even stop in the event of a wave near the on-ramp, making the vehicles behind them also slow down prematurely, and this way smoothing the impact of the wave.

Note that, following this strategy, a higher AV penetration rate translates to needing a less drastic deceleration per AV, and since the vehicles can speed up again once the jam is cleared, a higher number of AVs will inevitably provide better results. Therefore, the authors test their solution for increasing AV penetration rates (the defined percentage of autonomous vehicles on the road).

That said, the AVs are first trained in a closed ring road with similar characteristics to the open road. Then, by applying transfer learning to the strategies that they learned in the closed network, they are put on the open road and retrained there. The AVs were found to perform better with this warm start than when using a random initial state or a human-driver-like initial state, which proves that the results of closed network experiments can in fact be applied to open network problems by using transfer learning.

Surprisingly, the experiments show that a 2.5% AV penetration rate was sufficient to contribute greatly to dissipating the waves. Furthermore, at 10% the group of AVs was able to roughly dissipate the waves completely, with the vehicles moving twice as fast and with a 13% improvement in throughput. Fig. 3.3 illustrates the network when using a 0% penetration rate and a 10% penetration rate. The authors note that this strategy is a better alternative to ramp-metering since, providing the same results, it is applicable to waves formed everywhere in the controlled area, instead of just the ramp.

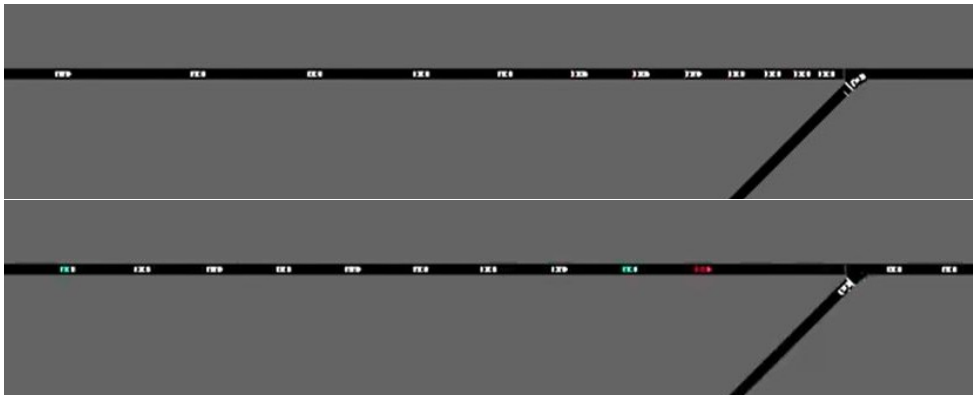

Figure 3.3: Merging of a vehicle in a setting with 0% AV penetration rate (top) and with 10% AV penetration rate (bottom). Red vehicles are AVs, while blue vehicles are observed HDVs and white vehicles are unobserved HDVs. Taken from the provided videos of the experiments at https://sites.google.com/view/itsc-dissipating-waves.

This work showed multiple interesting conclusions. For one, the knowledge gained in closed road

experiments can be transferred to the context of open roads, facilitating the development of traffic improvement strategies on open networks. Additionally, it suggested that AV control algorithms may successfully replace the current less effective and more expensive ramp meters. Finally, it presented an AV control strategy for single-lane open roads with an on-ramp that is effective while not being too complex, which is commonly the issue with other solutions. For that reason, the solution allows itself to be expandable to more realistic conditions—our project takes great inspiration from this work, striving to expand it to the context of multi-lane roads.

The experiments in the paper [3] consider a more detailed version of the same setting. Using the same scenario (single-lane open highway with a merge section), this work adds the possibility for AVs on the new lane to cooperate with AVs on the main road to plan a smoother merge. Furthermore, the controller is designed to improve not only throughput, but also fuel consumption and emission. To that effect, the authors define a cooperative control zone that involves a part of each lane, and propose a hierarchical framework to prepare for the merge.

When a new vehicle enters this zone, the method of cooperative merging is then chosen between 4 options, depending on the autonomy of the vehicles that surround it, as well as its own: if it is an AV and the surrounding vehicles are AVs as well, then full coordination is done; if it is an AV but some of its surrounding vehicles are not AVs, then partial coordination is done; if it is an AV but none of its surrounding vehicles are AVs, then adaptive following is done; if it is not an AV, then no coordination can be done. A scheme of each scenario is shown in fig. 3.4, for a better understanding. In the partial coordination mode, the authors plan that the new AV will act according to the HDV that surrounds it, merging before or after it, depending on the HDV's heading. In adaptive following, the AV is not able to know the HDV's heading, and so it will try to estimate when the HDV will arrive at the merge section and adapt to it, while safely and efficiently following the HDVs in front of it.

For the purpose of energy efficiency, the AVs have two modes of driving: free driving at an optimal acceleration that minimizes fuel consumption, and adaptive cruising (when they are sufficiently close to an HDV) at an optimal acceleration that minimizes fuel consumption while maintaining a safe distance.

This framework was able to achieve, at a 30% AV penetration rate, an improvement in throughput of 4.9% and fuel consumption of 34.7%. Distinctively, this paper's findings prove the various advantages of incorporating AVs into current traffic systems, going beyond the previously studied improvement in road throughput. However, although this work presents a refreshing and more challenging take on the problem of dissipating stop-and-go waves, the described algorithm is extremely complex, needing to be simplified to be applicable to more realistic situations such as our multi-lane scenario.

In another outlook at designing AVs to mitigate stop-and-go waves, the work [14] designs the con-

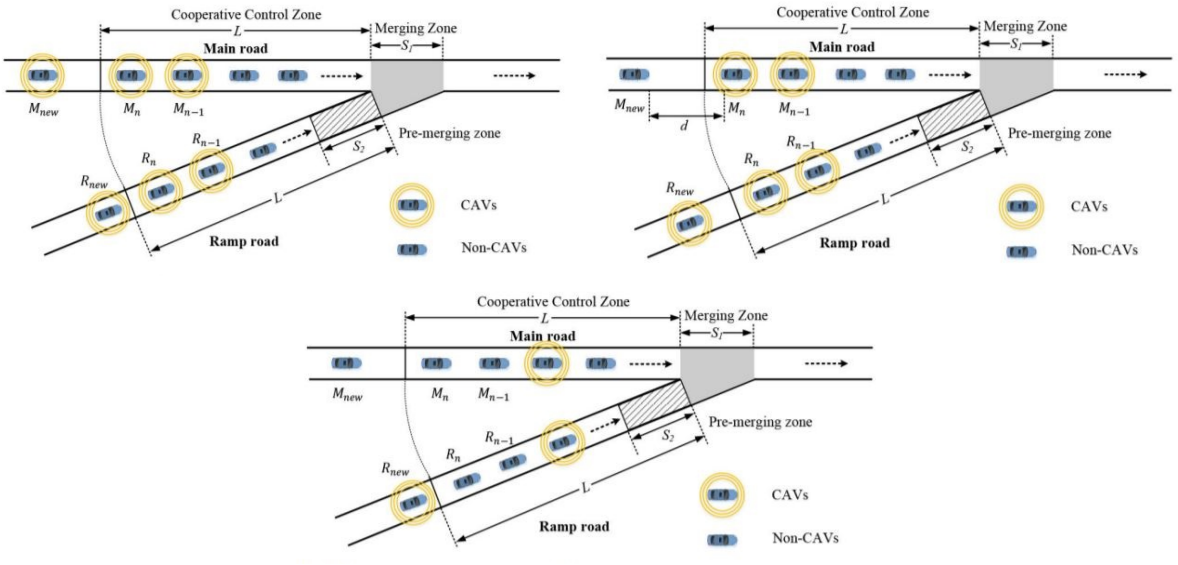

Figure 3.4: Diagrams of the full coordination mode (top left), partial coordination mode (top right) and adaptive following mode (bottom). Taken from [3] (Fig. 3, Fig. 4 and Fig. 6)

troller to perform differently according to the density of the road. The considered network is a multi-lane highway with a lane-drop section as a bottleneck to create the perturbation. The general method involves using reinforcement learning to train the AVs to control the speed of the lanes that are estimated to be causing the highest variations in speed. Then, two simple modes are distinguished – one for moderately congested scenarios and another for severely congested scenarios. In the first one, the AVs are expected to purposely create gaps to reduce queuing in the bottleneck. In the second one, much like in [13], the AVs travel at lower speeds upstream to control the inflow in the congested area. This approach was shown to work efficiently at a 10% AV penetration rate, assuring higher mobility and safety, and thus presenting an effective algorithm that distinctively takes into consideration and adapts to density variations in traffic.

Another important take from this study is that the authors proved that an RL-based solution performed better than a rule-based one, by including a rule-based controller which they compared against the developed RL-based controller. This conclusion further motivates the usage of reinforcement learning in our solution.

3.4 Autonomous vehicles designed to model human behavior

As previously stated, for traffic conditions to improve in a mixed-autonomy setting, it is necessary that human-driven vehicles comply with the envisioned strategies. The best way to assure the human drivers' compliance is often by influencing them into choosing the desired action. However, as every

human is different, so is their behavior on the road and their reactions to the AVs—and these differences can be significant. Thus, the policies used by the AVs to influence humans should avoid joining all human drivers in one model. In this section, we go through some approaches that focus on modeling human behavior in traffic to design more accurate strategies to influence them. These strategies were unfortunately not included in our approach. Nevertheless, we find it essential to mention these works, since the accurate modeling of human behavior is an indispensable part of developing a realistic traffic improvement mechanism.

The authors of [15] model human-AV interaction as a 2-agent game. Specifically, it is a Partially Observable Stochastic Game (POSG) with simplifications: the physical states are fully observable, the model is deterministic and the reward parameters do not change over time. It is partially observable since the agents do not know each other's rewards.

In this game, the uniqueness of the human's model is expressed through their reward, which is written as a combination of various features that make them avoid road boundaries, stay inside the lanes, and avoid other vehicles. Then, the AV is designed to keep an estimated model for the human's reward and use it to influence them toward the AV's goal. The authors define two modes for their experiments: offline and online estimation.

In the offline case, the AV assumes a fixed simulated human driver. The AV's reward function is defined as a combination of the necessary terms for speed and collision avoidance, with an additional term for the impact that it has on the human. This last term depends on the desired goal and on the human's behavior (if it effectively does what was intended). The results for the offline mode showed that this setting allows the AV to automatically find its own optimized methods of influencing the human: if the goal was to make the human slow down, the AV would brake to force the human behind it to do so as well, or if the goal was to make the human change lanes, the AV would block two lanes to force the human to move to the remaining lane. When the solution was tested on real users, the authors found that, although the estimated model for each human was not perfect, the AV could still influence them, proving the approach's efficacy. These findings are very promising since the AV's behavior did not have to be explicitly coded for each scenario, and instead appeared naturally.

In the online case, the AV is designed to estimate the human's reward parameters on the road. A belief function is a function that shows the current estimate of an agent on a value that it does not have access to. It is initialized to some value and then updated over time, according to the information that the agent obtains. That said, the AV keeps a belief function for the human's reward. Then, throughout the experiment, the AV has to balance an explore-exploit trade-off, as it has to constantly choose between 'exploring' actions—which allow it to update its belief on the human model—and 'exploiting' actions—which make use of its current belief to work on influencing the human into the desired goal. The AV's

reward function is thus modified from the offline case to also encourage 'exploring' actions.

The results of the online experiments confirmed that the AV, when actively choosing to take 'exploring' actions (actively updating the belief function), achieved a more accurate belief of the reward function than when simply choosing to avoid collisions (passively updating the belief function). Another interesting finding is that, using this setting, the AV ends up naturally showing safe behaviors, adapting its actions to the human's personality—if the person was distracted, the AV would refrain from influencing it to avoid possible collisions. The online mode was also tested with real users, and the results supported the conclusions of the simulations. Despite the various interesting findings of this work, the authors noted that the described game model is very complex, even in discrete state-action space, and that the involvement of multiple human drivers would be both a modeling and a computational challenge.

In another perspective, the authors of the paper [16] consider an AV that has access to the human's goal, and explore the effectiveness of designing an AV to be socially aware, experimenting with different AV behaviors. In this work, social awareness is defined as the ability of the AV to infer which actions are favorable to a human driver, while realizing the impact that the AV itself has on the human's intent. The authors experiment in a simulated intersection, where two vehicles, an AV and a reactive HDV, intend to continue straight ahead. In this situation, as there is an interaction zone and the vehicles do not share the same goal, they are forced to cooperate to choose who passes first.

The authors show that, for the simple case of a reactive AV, both vehicles become stuck in a deadlock (as both are afraid of crashing) unless one of the vehicles has a significantly higher speed than the other. However, when the socially aware AV is proactive, it tricks the human into thinking the AV is aggressive, this way managing to pass first. Then, if the human's speed is set to be significantly higher, the proactive AV realizes that it should wait for the human to pass first.

Finally, the authors experiment with a graceful AV that, in reaching its own goal, makes an effort to choose actions that are favorable to the human. They conclude that a socially aware and graceful AV purposely accelerates to reach the intersection slightly before the human, gently making the human retreat to allow the AV to pass first. In the case where the human's speed is significantly higher, the AV purposely reaches the intersection after the human, maintaining its gracefulness and allowing the human to pass first.

The authors conclude that designing the autonomous vehicle as socially aware allows the AV to learn more sophisticated strategies for influencing the human driver, such as passive-aggressive behaviors that persuade the human to yield. Furthermore, this work shows that a socially graceful AV is able to develop safer strategies when interacting with human drivers—which is an important finding since the methods for influencing human drivers usually sacrifice safety, requiring the AVs to follow dangerous behaviors.

As a final example, we look at the paper [17]. The Intelligent Driver Model (IDM) is a commonly used microscopic car-following model for standard human behavior. In this work, the authors describe a method that models a given human's driving behavior on a stochastic extension of the IDM, where the parameters of the model are set to the result of a particle filtering performed on the observed behavior of that human until the moment. Comparing the predicted maneuvers to real-world data sets of human trajectories, the authors test the model's accuracy and safety, confirming that it performs better than rule-based models and black-box driving models, and thus presenting an effective alternative to the online estimation of individual human driving behaviors.

3.5 Other interesting works

In this last section, we go through some works that, while not included in our solution, allow us to consider additional advantages and discover alternative capabilities of designing AVs to improve traffic.

Some of these studies focus on specific critical scenarios that commonly originate traffic congestion and on the improvement of traffic conditions in that specific setting, rather than attempting to develop a strategy that is suitable to every scenario. Given the great number of different situations that may generate traffic congestion, finding a solution that solves every problem can be overwhelming, if not impossible without first developing individual solutions for each scenario, thus making experiments like these very relevant in the context of traffic improvement.

3.5.1 Improving traffic near non-signalized intersections

The paper [4] demonstrates the advantages of having leading AVs at a non-signalized 4-way intersection, presenting a method that, in that setting, is able to dissipate the stop-and-go waves generated by the intersection. The authors consider the four possible cases for the leading vehicles illustrated in fig. 3.5: mixed-autonomy traffic but leading AVs at the intersection; full-autonomy traffic; all HDVs; mixed-autonomy traffic but leading HDVs at the intersection. The vehicles are assumed to drive straight ahead, not changing directions. Following the proposed method, the AVs, when in the front at an intersection, will coordinate with the other leading AVs (through a centralized controller) and either slow down or accelerate through the intersection.

The experiment that considered full autonomy traffic led to the best performance, followed by the leading AV experiment, and then by the leading HDV experiments. In the first two cases, the authors report that traffic congestion was partially cleared and the traffic flow got smoother, effectively showing that the AVs are capable of minimizing the consequences of the intersection, and may avoid the need for including physical signalization methods. Finally, we highlight that, although the full autonomy case

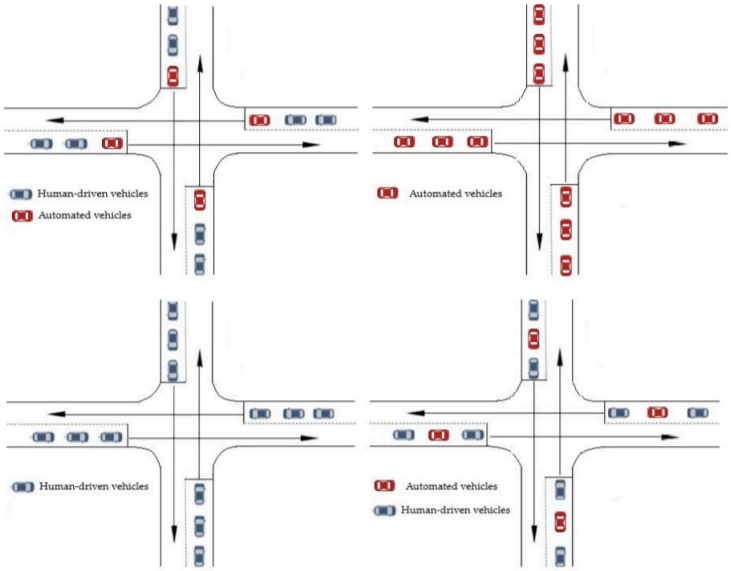

Figure 3.5: Different cases for the experiments: mixed-autonomy traffic with leading AV (top left); full-autonomy traffic (top right); all human-driven traffic (bottom left); mixed-autonomy traffic with leading human-driven vehicle (bottom right). Taken from [4] (Fig. 5 and 6)

outperformed the mixed-autonomy one, the mixed-autonomy case still allowed for significant benefits, once again highlighting that full autonomy is not required for autonomous vehicles to improve traffic.

3.5.2 Improving traffic near pedestrian crossings

The work [18] proposes using AVs to develop a more efficient alternative for managing pedestrian crossings (both for the drivers and for the pedestrians) than the currently used traffic lights. The proposed strategy is simple: the AVs are designed to communicate with each other so that, nearing a pedestrian crossing, two consecutive AVs create a comfortable space between them for the pedestrian to cross. Then, when it is safe, they signal the human to cross. Using this approach, the AVs were shown to leave the conflict section 30% sooner and, without needing to fully stop, they quickly recovered their initial speed. This method proves especially useful at industrial sites, where conflict often arises between cars and pedestrians and thus where the cars would benefit greatly from not stopping as often. In addition, this study included various tests on the responses of humans to the AVs' signals and behavior, in order to help future studies on the AV-human communication alternatives.

3.5.3 Improving the passing of emergency vehicles

In [5], a game theory approach is described for designing autonomous emergency vehicles to interact with and influence HDVs. The envisioned method avoids treating every human as self-interested and thus, similarly to the studies described in section 3.4, involves the individual modeling of the hu-

man drivers. However, in this study, the humans' behaviors are modeled through their social value orientation—their willingness to cooperate with other agents. Fig. 3.6 illustrates the possible social value orientations, as defined in psychology, and their regularity. Like so, the controller for the emergency vehicle is able to measure the influence that it can have on each driver's trajectory.

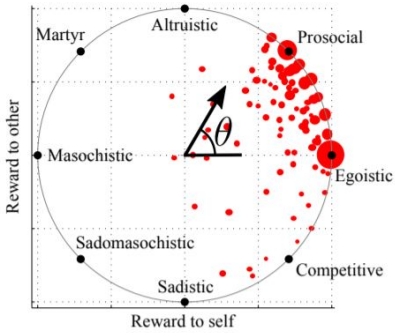

Figure 3.6: Social Value Ring. The size of a red circle corresponds to the proportion of human subjects. Taken from [5] (Fig. 2)

Following the envisioned method, the controller sets a semi-cooperative utility function for each vehicle (HDVs and emergency vehicle), including their own reward and the other agents' rewards (with varying weight). Then, according to these functions and to the social value orientation of each HDV, the controller calculates the system's nash equilibrium. Note that, by definition, no agent has the incentive to deviate from the nash equilibrium—thus, if the estimations are accurate, this equilibrium defines both the trajectories that the HDVs will take, and the optimal trajectory that the controller should command the emergency vehicle to take.

This strategy was shown to lead to the generation of different cooperative maneuvers between the HDVs and the emergency AV, allowing a more efficient and safer passing for the emergency vehicle. This work's findings thus prove that mixed-autonomy traffic could also improve the handling of emergency situations, since it shows that emergency vehicles could perform more efficiently and safely if they were autonomous, regardless of the level of autonomy of the vehicles around them.

3.5.4 Accounting for safety

In the studies described in [6], the goal is to design AVs to assure higher safety and efficiency in high-risk situations where a potential accident is likely. The paper proposes a hierarchical reinforcement and imitation learning approach (referred to as H-ReIL), that uses a high-level policy that switches between different low-level policies. The high-level policy is learned through reinforcement learning, while the low-level policies are learned with imitation learning. The authors argue that imitation and reinforcement learning alone are unable to model the presented problem—they cannot model rapid phase transitions (required for handling near-accident scenarios) or cover all the possible states in traffic—, thus motivating

the need to define a new combined approach.

The authors performed experiments to compare five policies, which switched between two modes (the low-level policies), chosen to model a timid and an aggressive driver. These modes were learned through a large amount of driver behavior demonstrations. The compared policies were: the always timid driver, which favors safety, choosing medium speed to avoid all potential accidents; the always aggressive driver, which favors efficiency, choosing fast speed to reach its destination as soon as possible; the random policy, which randomly switches between the first two policies; the imitation learning policy, which switches between the first two policies according to driver behavior demonstrations; the proposed H-ReIL policy. Then, these policies were tested using an autonomous vehicle in typical near-accident scenarios, like crossing at an intersection with low visibility or having a car traveling in the wrong direction cut into the AV's lane.

Fig. 3.7 compares the performance of the different policies. Although for low time limits the aggressive policy achieved better results, this policy was unable to improve with higher time limits, as it often resulted in collisions. Thus, the H-ReIL is shown to provide the best trade-off between safety and efficiency. Additionally, fig. 3.8 shows some of the performed simulations using the H-ReIL policy, with notes on the behavior of the AV. We can see that, following this policy, the AV generally opts for an aggressive behavior but quickly switches to a timid behavior when a near-accident situation occurs. Therefore, the H-ReIL policy is shown to allow the AV to simultaneously maximize its efficiency when there are no potential threats, and act appropriately and safely when a threat is present.

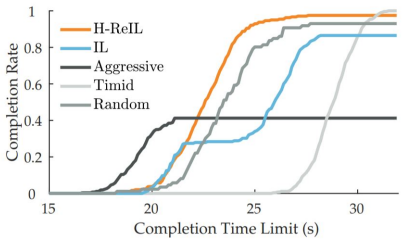

Figure 3.7: The completion rate for each policy with varying time limits. The completion rate is the proportion of the trajectories in which the AV safely reaches the destination within the time limit. Taken from [6] (Fig. 7)

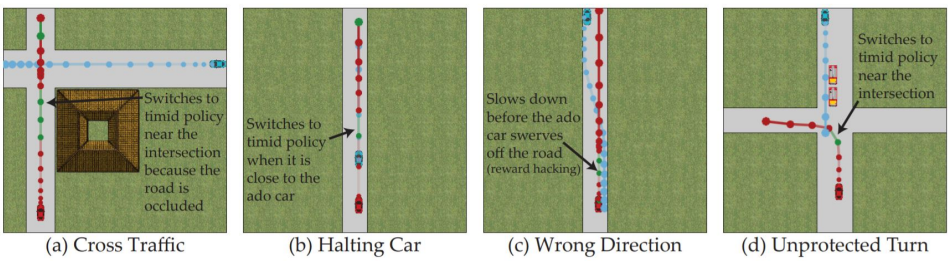

Figure 3.8: Simulations done in CARLO simulator. The blue color corresponds to the trajectory of the HDV. The red and green colors correspond to the AV taking the aggressive and the timid policies, respectively. Taken from [6] (Fig. 5)

3.5.5 Scaling existing approaches

The study described in [19] proposes a method to scale already existing strategies on using AVs to improve traffic conditions. The authors of this paper suggest taking the behaviors learned from experiments performed in smaller networks, and using transfer learning to apply this knowledge to bigger-scaled scenarios. Specifically, they use a modular approach to transfer learning, where the transferred methods are only applied to a section of the new scenario.

The authors tested this method and evaluated the resulting policy in the large-scale scenario, confirming that the learned policy allowed for an improvement in the network's outflow (compared to zero-autonomy traffic), and further showing that it performed better than a regular policy trained from scratch. Moreover, they noted that the modular transfer learning approach required much less training time (80% reduction in setup time), as it avoided collecting samples from the entire network. Therefore, this study proved that using modular transfer learning to scale existing approaches is not only possible but very promising.

As a side note, the authors of this work argue that the commonly used average speed metric is insufficient for evaluating open network traffic problems. They explain that this metric can be manipulated by the evaluated policy, since it can choose to prevent new vehicles from entering the road until the road is clear for the vehicles to go through it at the desired speed, compromising the inflow and outflow of the network while achieving a higher average speed. Clearly, this policy is not functional in a real-world setting, as the vehicles that must wait outside will always take an undetermined amount of time to get to the other side. Therefore, they propose that the used evaluation metric should instead be the network's outflow.

Another distinctive characteristic of this study is that it proposes an effective distributed algorithm for the same purpose of traffic improvement. The developed distributed solution consists of a multi-agent reinforcement learning policy that relies only on the knowledge sensed by each vehicle. Note that using this distributed policy is more realistic in an open network than using a centralized one, as the latter relies on having access to information from all the AVs. However, contrasting with the remaining of this paper, the proposed distributed solution is only applicable to a small open network and the authors propose scaling it as future work.

4

Using reinforcement learning to train autonomous vehicles

Contents

Between the various solutions that have been developed in the last years on the topic of using autonomous vehicles to prevent traffic congestion, we highlight the studies that focus on the regulation of traffic when nearing on-merge and lane-drop sections. The papers on this topic usually considered settings where there was either only one lane per road or where lane changes were not allowed. In fact, very few of the mentioned works explored the possible advantages of lane-changing, which accentuates the work in [2]. This paper introduces the idea of using AVs to reorganize the vehicles traveling on a multi-lane road into a more favorable distribution. In effect, this strategy paints multi-lane roads as an additional means to improve traffic and can be joined with other works to expand their solutions.

4.1 Project overview

Given the lack of solutions provided for these situations, we address the problem of developing an autonomous vehicle control strategy that prevents traffic congestion near a bottleneck section on a road with multiple lanes, and that takes into consideration the possibility (and advantages) of lane-changing.

Specifically, we consider the scenario of a multi-lane highway with a merge section. In this scenario, the network is open, meaning that there is a constant inflow of vehicles entering the road (as well as leaving). Additionally, the traffic has mixed autonomy, meaning that there are both autonomous and conventional vehicles on the road. We assume the AVs to be uniformly distributed, which means that, at an AV penetration rate of P%, every $\frac{100}{P}$ vehicle that enters the network is autonomous.

To train the AVs in this network, we define a centralized agent that has access to each AV's observations and can act through each AV's actions. Note that the agent does not have full access to the state of the network at each step, and instead only gets partial information from what the AVs can observe of their own surroundings. As such, the problem is modeled using a POMDP. We then use deep reinforcement learning to train this agent to learn an optimal policy that maximizes the network's outflow and the vehicles' average speed, since those are the best indicators of clear traffic.

Fig. 4.1 presents an overview of the project setup. The network and vehicles are represented in a traffic micro-simulator. At each step, the agent obtains the AVs' observations from the simulator and sends the AVs' next actions, based on the observations that it got and on an internal policy that it keeps. The agent then receives a reward depending on the state of the system and updates its policy according to that reward.

4.2 Strategy

Our plan is to develop a controller that dissipates the waves formed by the merge section, while also optimally reorganizing the vehicles on the main road, this way taking advantage of the multiple lanes. This problem can be modeled as the combination of two simpler problems:

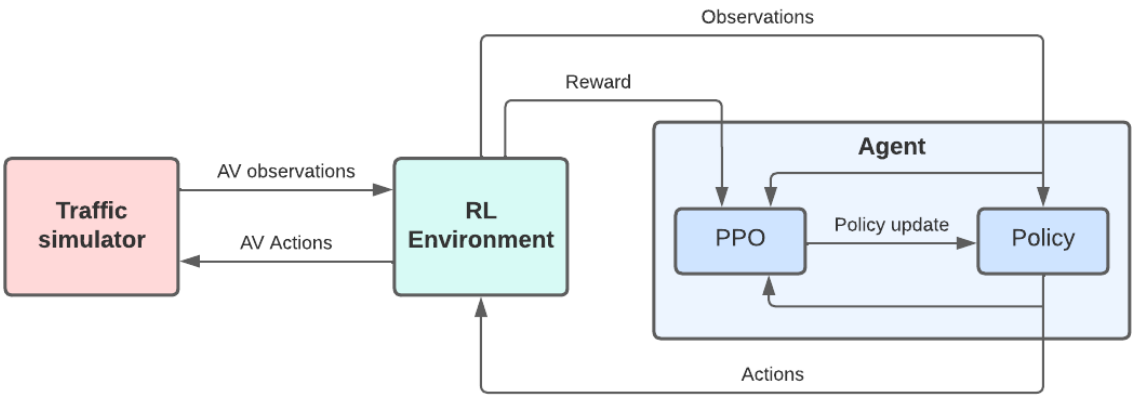

Figure 4.1: Overview of the project setup

1. a single-lane highway with a merge section, where the controller learns to maximize the network's outflow and dissipates stop-and-go waves;

2. a multi-lane highway without a merge section, where the controller learns to rearrange the AVs and the vehicles around them before the merge section into a distribution that would be optimal for handling the merge.

We decided to first experiment with training the AVs in these two sub-problems and then join the experiments to build the more complex scenario, to conclude whether these two behaviors can be joined to reach our initial goal.

That said, in scenario 1 we follow the approach described in [13], with the exception of some minor necessary changes to update the solution.

For scenario 2, inspired by [2], the idea is to take advantage of having multiple lanes on the main road instead of just working around the added complexity that they bring. From what we concluded in [13], a higher AV penetration rate allows for an overall smoother merge, as the AVs can detect the upcoming waves and cooperate to dissipate them. It follows that the optimal organization for the vehicles on the main road is one where the AVs are concentrated on the lane that suffers the merge. Therefore, the goal is to make the AVs learn to reorganize that lane into having the largest possible AV rate.

For the final scenario, we join these two strategies, expecting the AVs to manage between reorganizing the lane that suffers the merge and dissipating the waves to allow for a smooth merge.

5

Implementation

Contents

Having outlined our problem and strategy, in this chapter we describe our experiments in detail.

5.1 Preliminary experiments

5.1.1 Single-lane highway with merge section

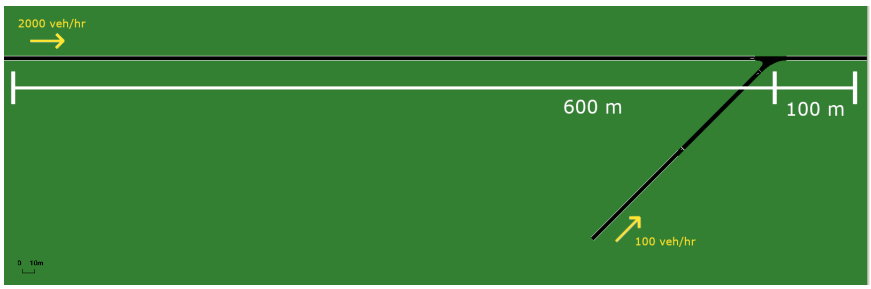

Figure 5.1: Network configuration of the first preliminary experiment: single-lane highway with merge section

Fig. 5.1 illustrates the network configuration for the first scenario. The main road has a single lane of length $L_{hw} = 700$ m, with pre-merge length $L_{hw\_pre} = 600$ m and post-merge length $L_{hw\_post} = 100$ m. The merging road has length $L_m = 100$ m. We define the inflows of the main and merging roads to be, respectively, $f_{hw} = 2000$ veh/hr and $f_m = 100$ veh/hr, and the AV penetration rate to be $P_{AV} = 10\%$.

The observation space of the agent corresponds to what each AV can observe at each step. In this scenario, the observations are the AV's speed v_i, the speeds $v_{i,lead}$ and $v_{i,fol}$ of the vehicles directly in front and behind it, and the time headways $h_{i,lead}$ and $h_{i,fol}$ between the AV and those same vehicles. The action space is the bounded acceleration a_i of each AV.

The reward function is a weighted sum defined in eqs. 5.1 to 5.4. The first term R_v rewards the proximity of the system's speed $v(t)$ to the desired speed v_{des}, while the second R_{out} rewards high network outflows $out(t)$. We add a cost C_h for small AV space and time headways since these are indicators of congested traffic. In eq. 5.4, $h_{min,t}$ and $h_{min,s}$ are, respectively, the minimum desirable time and space headways, while $h_{i,t}(t)$ and $h_{i,s}(t)$ are the corresponding headways of AV i at time step t. For the experiments, we used $v_{des} = 25$ m/s, $out_{des} = 2100$ veh/hr, $h_{min,t} = 1$ s, $h_{min,s} = 7$ m, $a_1 = 0.1$ and $b_1 = 0.1$.

$$R_1 = a_1 \times R_v + (1 - a_1) \times R_{out} - b_1 \times \sum_{i \in AV} C_h \tag{5.1}$$

$$R_v = \|v_{des}\| - \|v_{des} - v(t)\| \tag{5.2}$$

$$R_{out} = \min[out(t)/out_{des}, 1] \tag{5.3}$$

$$C_h = \min[h_{i,t}(t) - h_{min,t}, h_{i,s}(t) - h_{min,s}, 0] \tag{5.4}$$

Note that in the original solution presented in [13], only the average system speed was considered—however, we found that, in that case, the AVs ended up exploiting the reward function by stopping at the beginning of the main road (blocking the inflow) until the road was clear and then speeding through the road, this way managing to output high average speeds at the cost of lowering the network outflow. Therefore, we followed the approach suggested in [19] as a workaround to this problem, adding the term that rewards the network's outflow. Additionally, the original headway cost only accounted for time headways. We decided to account for space headways as well since the time headway can be uninformative at very low speeds.

5.1.2 Multi-lane highway without merge section

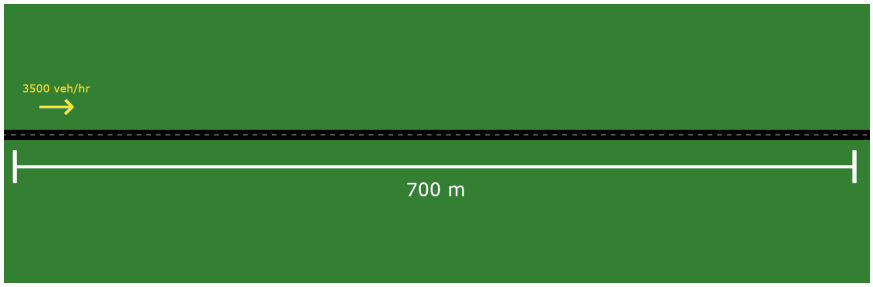

Figure 5.2: Network configuration of the second preliminary experiment: multi-lane highway

For the second experiment, the network is as shown in fig. 5.2, where the main road has two lanes instead of one, and there is no merging lane. The main road length and the AV penetration rate are the same as in the first experiment, $L_{hw} = 700$ m and $P_{AV} = 10\%$. Given the extra lane, the inflow of the main road is increased to $f_{hw} = 3500$ veh/hr. Notice that we purposefully choose a high inflow for this scenario, since we want to test the controller's ability to learn the intended behavior in high-density traffic. This is because this experiment should serve as a preliminary experiment to the one in section 5.2, where the traffic is expected to have high density caused by the bottleneck. The departing lane is chosen randomly for each vehicle. Additionally, we assume the lane that suffers the merge to be the right lane (also called lane 0).

The observation space includes some of the observations already described in 5.1.1 for each AV—v_i, $v_{i,fol}$ and $h_{i,fol}$—, with the addition of the AV's lane l_i and the type fol_i (AV or HDV) of the vehicle directly behind it. The action space consists of each AV's bounded acceleration a_i and their direction d_i.

The reward function is the weighted sum defined in eqs. 5.5 to 5.7 and 5.2. The first term R_v, defined in the previous section, rewards proximity to the desired speed. The second term R_p rewards the proximity of the AV rate $p(t)$ on the right lane to the desired rate p_{des}. The final term B_{push} is a bonus

that rewards AVs for influencing an HDV to change out of the right lane. For this scenario, we used the constants $v_{des} = 25$ m/s, $p_{des} = 0.25$, $a_2 = 0.75$ and $b_2 = 0.1$.

$$R_2 = a_2 \times R_v + (1 - a_2) \times R_p + b_2 \times \sum_{j \in HDV} B_{push} \tag{5.5}$$

$$R_p = \min[p(t)/p_{des}, 1] \tag{5.6}$$

$$B_{push} = \begin{cases} 1, & \text{if } lane_j(t-1) = 0 \wedge lane_j(t) \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{5.7}$$

5.2 Main experiment: Multi-lane highway with merge section

Figure 5.3: Network configuration of the main experiment: multi-lane highway with merge section

Our final experiment considers the open network scenario of a 2-lane highway of length $L_{hw} = 700$ m, with a merge section where the merging road is single-lane and has length $L_{hw} = 100$ m. As in 5.1.1, the main road has pre-merge length $L_{hw\_pre} = 600$ m and post-merge length $L_{hw\_post} = 100$ m. The inflows on the main and merging road are set to $f_{hw} = 3500$ veh/hr and $f_m = 200$ veh/hr, and the AV penetration rate remains at $P_{AV} = 10\%$. As in the previous section, the departing lane for each vehicle is chosen randomly. Fig. 5.3 illustrates the network.

The observation space for this problem combines the ones used in the preliminary experiments, including, for each AV: the AV's speed v_i and lane l_i, the speeds $v_{i,lead}$, $v_{i,fol}$ and headways $h_{i,lead}$, $h_{i,fol}$ of the vehicles directly in front and behind the AV, and the type fol_i of the vehicle directly behind it.

The action space is the same as in 5.1.2, corresponding to each AV's bounded acceleration a_i and their direction d_i. Note that the direction corresponds to the intent or not of moving to the lane on the right/left of the current lane and that the AVs cannot be in between two lanes. Additionally, in this environment, any action chosen by the agent that would lead to immediate collisions (for example, changing lanes when there is no space on the target lane) is overwritten by the simulator.

Since the network is open, the number of AVs (and consequently, the number of observations and actions) varies throughout the experiment. Therefore, we define a fixed maximum size N_{AV} for the set of controlled AVs at each step and use zero padding for the observation and action space when there are fewer than N_{AV} AVs in the network.

The reward function used to train the vehicles is a weighted sum of the reward functions used in the previous section 5.1, defined as follows:

$$R = \alpha \times R_1 + (1 - \alpha) \times R_2 \tag{5.8}$$

where R_1 is defined in eq. 5.1, R_2 is defined in eq. 5.5 and $\alpha \in [0, 1]$. R_1 rewards behaviors that lead to the dissipation of the waves that may form near the merge, while R_2 rewards behaviors that lead to an increase of the AV ratio on the right lane. We experiment with a set of different α values to conclude which proportion between these two terms allows for the best results. Regarding the first term, we set $v_{des} = 25$ m/s, $out_{des} = 3700$ veh/hr, $h_{min,t} = 1$ s, $h_{min,s} = 7$ m, $a_1 = 0.1$ and $b_1 = 0.1$. Regarding the second term, we set $p_{des} = 0.25$, $a_2 = 0.75$ and $b_2 = 0.1$.

We point out that we were unfortunately forced to limit the number of lanes to two due to the complexity of the experiments—the training duration proved proportional to the number of vehicles currently in the network since the environment required a running simulation of the network while training. This originated a significant difference in the training times when we added a second lane on the main road and ultimately rendered adding a third lane unfeasible.

5.3 Simulations

The experiments are implemented in Flow [20], an open-source framework developed by the authors of the work [13] to perform reinforcement learning experiments in traffic micro-simulators. Flow allows for the creation of various traffic-oriented RL tasks with the goal of developing control strategies for autonomous vehicles. Additionally, we use SUMO [21] for the execution of the simulations. SUMO is a renowned open-source traffic micro-simulator. Flow and SUMO are frequently mentioned throughout the studied literature—given that our work strives to build on the already existing strategies, we decided for simplicity to use the same frameworks. Finally, the human-driven vehicles' behavior and dynamics are modeled using the IDM [22], a microscopic car-following model built into SUMO.

The simulations are executed with time steps of 0.2 s and a total duration of 3600 s. The RL agent receives observations and chooses new actions every 5 simulation time steps, repeating its actions in the meantime.

The agent is trained using the PPO algorithm, with discount factor $\gamma = 0.999$ and a learning rate of 0.001. Furthermore, we use an Multi-Layer Perceptron (MLP) actor-critic policy with hidden layers (32, 32, 32) and relu activation function. We also performed some experiences using the TRPO algorithm

as was suggested in [13] but concluded that the small difference in the results did not justify the added complexity (and consequent additional training time) when compared to the PPO algorithm.

6

Results

Contents

In the following sections, we present the numerical results for each of the experimental settings described in Chapter 5 and reflect on the behavior of the trained controllers.

Each simulation had a total duration of 3600s, and the results were averaged over 50 simulations to account for stochasticity between simulations. The considered metrics throughout the experiments are the average of the system's speed at each step ($speed$), the total outflow over the duration of the simulation ($outflow$), the average number of vehicles on the highway at each step ($\#vehicles$), and the average percentage of AVs on the right lane at each step ($P_{AV,lane0}$).

We also present videos of the simulations performed both with and without the trained RL-controller. These videos are available at SimulationReplays. The title of each video is informative of the experiment setting, the level of autonomy, and, in the case of the final experiments, the used value for α. Although some videos may be longer due to simulator processing delays, each video shows a total of 1200 simulation seconds. The current simulation time can be assessed in the top left corner. Regarding the colors of the vehicles, red vehicles correspond to AVs, white vehicles to HDVs, and blue vehicles to the HDVs that the AVs can currently observe. Finally, we recall that only a limited group of AVs can be controlled at each step—as such, in the event of a jam, there can be some uncontrolled AVs in the network that will act as HDVs.

6.1 Single-lane highway with merge section

$P_{AV,total}$ (%)	speed (m/s)	outflow (veh/hr)	# vehicles	$P_{AV,lane0}$ (%)
0	7.08	1440	49	-
10	14.25	1560	24	-

Table 6.1: Numerical results of the first preliminary experiment (5.1.1)

Table 6.1 shows the obtained results for the single-lane highway with merge section experiment (described in 5.1.1), comparing the zero-autonomy simulations to the mixed-autonomy simulations performed using the trained controller. The final metric is not relevant in a single-lane scenario and is thus not included.

In the mixed-autonomy simulations, we notice an 8% increase in the outflow of the network, along with a 50% increase in the average speed, with the vehicles effectively traveling at double the speed. The number of vehicles on the network at each step also decreased to half in the case of mixed autonomy, showing a clear improvement in the road's efficiency.

From the simulation replay of the mixed autonomy setting[1], we observe that, as expected, the AVs at the beginning of the highway slow down in the event of a wave, forcing the line of vehicles behind them to slow down as well, and successfully dissipating the waves.

[1] Video titled *SingleLaneMerge_MixedAutonomy*

6.2 Multi-lane highway without merge section

$P_{AV,total}$ (%)	speed (m/s)	outflow (veh/hr)	# vehicles	$P_{AV,lane0}$ (%)
0	20.2	3460	38	10*
10	16.4	3436	51	16

Table 6.2: Numerical results of the second preliminary experiment (5.1.2)

In table 6.2, we assess the results of the multi-lane highway experiment. To be able to evaluate the $P_{AV,lane0}$ evaluation metric (which represents the main goal of this experiment) on the zero-autonomy simulations, we perform the simulations using puppet AVs which, without a trained RL controller, simply act as HDVs while carrying the label of being an AV. That said, in the zero-autonomy simulations, this value corresponds to the set AV penetration rate of 10%. This is expected since the departure lane for each vehicle is random.

In the mixed autonomy setting, we see that the controller was able to increase this percentage to 16%, but there is a consequent decrease of 19% in the average speed and also a slight decrease in the outflow.

From the replay of the simulation[2], we see that the AVs move to the right lane when possible, and sometimes slow down in front of an HDV, eventually influencing it to change to the left lane, as intended. However, since the inflow is high, the traffic is inevitably very dense, and thus the repeated lane changes in the mixed-autonomy scenario end up generating jams (as seen toward the end of the video), whereas in the zero-autonomy scenario there are no disturbances and the vehicles travel at their free-flow speed. Additionally, we note that, after these jams are formed, the attempts of the AVs to change to the right lane are overwritten given the lack of space on that lane. This justifies the decrease in the average speed and why the AV percentage increase on the right lane was not larger.

In any case, our insight is that, in the multi-lane merge scenario where both solutions are joined, the obtained increase in the AV penetration rate on the right lane should compensate for its consequences, since the controller in that environment has more resources to dissipate the waves.

6.3 Multi-lane highway with merge section

For the final scenario, we show in table 6.3 the results of the experiments for different values of the α scalar in eq. 5.8. We compare the experiments for $\alpha = \{1, 0.75, 0.50, 0.25, 0\}$. To understand these results, we start by looking at the experiment where $\alpha = 0.50$, which is the one that achieved the best performance.

[2] Video titled *MultiLaneHighway_MixedAutonomy*

α	speed (m/s)	outflow (veh/hr)	# vehicles	$P_{AV,lane0}$ (%)
1	4.10	1955	113	11
0.75	6.43	2614	99	9
0.50	13.1	3045	53	5
0.25	8.08	2801	87	10
0	7.82	2823	89	11

Table 6.3: Numerical results of the main experiments (5.2), for different α values

6.3.1 Best case strategy

Surprisingly, in the experiment where $\alpha = 0.50$, the $P_{AV,lane0}$ was actually lower than the original AV penetration rate on the inflow of 10%. We turned to the replay of the simulation[3] to better understand the learned behavior, where we noticed that the controller had naturally developed a different strategy for dissipating the waves in this multi-lane scenario. In the replay, the AVs end up staying mostly on the left lane, where they behave similarly to the first preliminary experiment, slowing down periodically to open space in the lane. Then, in the event of a wave near the merge, they quickly alternate between the two lanes, surprising the vehicles and thus forcing them to brake.

Although this behavior effectively resulted in decreasing the impact of the wave on both lanes, we did not plan or anticipate it. Therefore, we briefly reflect on how this behavior was learned from our implementation.

We first note that, in the event of a wave in a multi-lane scenario, the vehicles in the lane that suffers the merge are more likely to change into the adjacent lane to avoid stopping, rather than staying in their current lane. Thus, the wave on the right lane only aggravates once there is no more space on the adjacent lane for the vehicles to change into, that is after another wave has formed on the adjacent lane. The video of the zero-autonomy simulation for this setting[4] shows this behavior. This effectively translates to a second parallel merge of the right lane into the left lane—and this merge is much more alarming than the original since the number of vehicles on the right lane is almost 10 times the number of vehicles on the actual merging road. With this in mind, we understand why the left lane should be the most closely monitored, and consequently the one with the higher AV percentage. Additionally, we note that the HDVs have a slight resistance to overtaking on the right, meaning that when an AV brakes to prepare the left lane for the merge, it indirectly manages to slightly affect the right lane as well.

Finally, we realize that, for increasing values of α in our reward function, the term that rewards high AV percentages on the right lane (originally defined in eq. 5.6) ends up amounting to very small values. Additionally, since there is a constant inflow and outflow of vehicles, and since this term is recalculated in every time step, it frequently varies regardless of the AVs' actions. Therefore, we believe the controller

[3] Video titled *MultiLaneMerge_MixedAutonomy_alpha050*
[4] Video titled *MultiLaneMerge_ZeroAutonomy*

ultimately interprets this term as an incentive for changing into the right lane, rather than as a continuous reward for staying in that lane, justifying the repeated changes between lanes.

6.3.2 Remaining results

For every other value of α, we get significantly worse results. Note that, when α is 0, the reward function becomes equal to $R2$ (eq. 5.5), and the experiment is similar to the one described in section 5.1.2, however applied to the more complex scenario of an on-merge section. Accordingly, in the simulation video[5] the AVs change to the right lane as soon as possible. However, since they lack the incentive to dissipate the waves, they do not slow down to prepare their lane for the merge, and allow the waves to quickly expand.

Likewise, when α is 1, the reward function is $R1$ (eq. 5.1) and the controller is rewarded solely for behaviors that lead to the dissipation of eventual waves, as in the experiment of section 5.1.1. From the simulation [6], we can see that the AVs rarely change lanes, and simply focus on braking extensively in their current lane, a strategy that ends up generating its own jams. We believe that, without the incentive to change to the right lane, the controller is unable to learn the working strategy of alternating lanes, and instead chooses a defensive strategy that is ineffective.

In the replay of the experiments where α is 0.25[7] and 0.75[8], we recognize some efforts to follow the same strategy as in the $\alpha = 0.50$ experiments—we see that the AVs change lanes repeatedly, trying to control both lanes, and also brake in their lane, managing to create small gaps. However, the controller is unable to balance these two behaviors and the waves inevitably propagate. That said, when $alpha = 0.75$, the strategy works slightly better in the beginning than when $alpha = 0.25$, as the AVs are able to dissipate the first waves. Nevertheless, when a wave is finally able to expand, the efforts of the controller to mitigate the wave are counterproductive and end up generating additional jams behind the AVs, whereas when $alpha = 0.25$ the waves simply propagate as in the zero-autonomy setting.

This detail, along with the overall worse results in table 6.3 for $alpha < 0.5$ than for $alpha > 0.5$, leads us to conclude that the strategy used to mitigate stop-and-go waves in the preliminary experiment 5.1.1 is only effective in the multi-lane scenario if it is joined with the incentive to change lanes, and is otherwise actually harmful to traffic.

6.3.3 Achieved traffic improvement

Finally, we evaluate the performance of the best-case experiment, $\alpha = 0.50$, by comparing it against the zero-autonomy setting. Table 6.4 shows the results for both simulations. We notice that, using the

[5] Video titled *MultiLaneMerge_MixedAutonomy_alpha0*
[6] Video titled *MultiLaneMerge_MixedAutonomy_alpha1*
[7] Video titled *MultiLaneMerge_MixedAutonomy_alpha025*
[8] Video titled *MultiLaneMerge_MixedAutonomy_alpha075*

$P_{AV,total}$ (%)	speed (m/s)	outflow (veh/hr)	# vehicles	$P_{AV,lane0}$ (%)
0	10.2	2902	94	10*
10	13.1	3045	53	5

Table 6.4: Performance of the best main experiment (5.2, $\alpha = 0.50$) against a zero-autonomy traffic simulation

RL-controller, there is a 28% increase in the average speed of the vehicles, along with a 5% increase in the network's outflow and a 44% decrease in the density of traffic.

We conclude that the trained AV controller, although not following the strategy we envisioned, effectively improved traffic conditions on a two-lane highway with a merge section.

7

Conclusion

Contents

7.1 Conclusions

This project highlights the benefits of autonomous vehicles and their possible advantages to traffic. We document some of the existing works on this topic and propose a deep RL-based autonomous vehicle control mechanism that improves traffic conditions near a merge section on an open two-lane highway.

Since the envisioned strategy is the composition of two sub-strategies, we additionally present implementations for each—we update an existing AV control strategy that dissipates stop-and-go waves near a merge section in a single-lane highway, and develop a strategy that reorganizes the vehicles on a multi-lane highway—and prove their effectiveness in a mixed-autonomy traffic micro-simulation. We then study the proportion between the two behaviors that allows for the best performance in the joined scenario, noticing the AVs' best-learned behavior to be different from our envisioned strategy. As such, we document this alternative behavior and how it developed from our implementation, so that in the future it may be reproduced and possibly optimized.

From the simulations, this autonomous vehicle control solution, at a 10% AV penetration rate, is shown to allow for a 28% increase in the average vehicle speed and a 5% increase in the network's outflow, with the AVs effectively minimizing the consequences of the merge section and utilizing both lanes to better prepare the vehicles for the merge.

Therefore, we conclude that this work effectively presents an autonomous vehicle control approach that minimizes traffic congestion on a multi-lane merge, proving the potential benefits of designing autonomous vehicles to improve traffic conditions.

7.2 System Limitations and Future Work

Our solution is unfortunately limited from only being tested and proved to work on a two-lane scenario, and therefore we see it as future work to expand it to the context of more lanes.

Furthermore, it does not account for every human being unique and ultimately assumes that every human's behavior in traffic will be similar, which leaves a significant gap between our solution's effectiveness in the simulator and its true effectiveness in real-life traffic.

Knowing that the optimal learned strategy was actually very different from our envisioned one, we additionally suggest re-implementing the solution to reward those optimal behaviors, simplifying it and possibly even achieving better results.

Bibliography

[1] M. Kwon, M. Li, A. Bucquet, and D. Sadigh, "Influencing leading and following in human-robot teams." in *Robotics: Science and Systems*, 2019.

[2] D. A. Lazar, R. Pedarsani, K. Chandrasekher, and D. Sadigh, "Maximizing road capacity using cars that influence people," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, p. 8.

[3] J. Ding, H. Peng, Y. Zhang, and L. Li, "Penetration effect of connected and automated vehicles on cooperative on-ramp merging," *IET Intelligent Transport Systems*, vol. 14, no. 1, p. 10, 2019.

[4] D. Quang Tran and S.-H. Bae, "Proximal policy optimization through a deep reinforcement learning framework for multiple autonomous vehicles at a non-signalized intersection," *Applied Sciences*, vol. 10, no. 16, p. 19, 2020.

[5] N. Buckman, W. Schwarting, S. Karaman, and D. Rus, "Semi-cooperative control for autonomous emergency vehicles," 2021.

[6] Z. Cao, E. Bıyık, W. Z. Wang, A. Raventos, A. Gaidon, G. Rosman, and D. Sadigh, "Reinforcement learning based control of imitative policies for near-accident driving," *arXiv preprint arXiv:2007.00178*, 2020.

[7] J. K. Choi and Y. G. Ji, "Investigating the importance of trust on adopting an autonomous vehicle," *International Journal of Human-Computer Interaction*, vol. 31, no. 10, pp. 692–702, 2015.

[8] D. Rothenbücher, J. Li, D. Sirkin, B. Mok, and W. Ju, "Ghost driver: A field study investigating the interaction between pedestrians and driverless vehicles," in *2016 25th IEEE international symposium on robot and human interactive communication (RO-MAN)*. IEEE, 2016, p. 8.

[9] S. Nyholm and J. Smids, "The ethics of accident-algorithms for self-driving cars: An applied trolley problem?" *Ethical theory and moral practice*, vol. 19, no. 5, p. 15, 2016.

[10] D. P. Losey and D. Sadigh, "Robots that take advantage of human trust," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, p. 8.

[11] H. H. S. Nagalur Subraveti, V. L. Knoop, and B. van Arem, "First order multi-lane traffic flow model–an incentive based macroscopic model to represent lane change dynamics," *Transportmetrica B: Transport Dynamics*, vol. 7, no. 1, p. 23, 2019.

[12] C. Wu, A. M. Bayen, and A. Mehta, "Stabilizing traffic with autonomous vehicles," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, p. 7.

[13] A. R. Kreidieh, C. Wu, and A. M. Bayen, "Dissipating stop-and-go waves in closed and open networks via deep reinforcement learning," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, p. 6.

[14] P. Y. J. Ha, S. Chen, J. Dong, R. Du, Y. Li, and S. Labi, "Leveraging the capabilities of connected and autonomous vehicles and multi-agent reinforcement learning to mitigate highway bottleneck congestion," *arXiv preprint arXiv:2010.05436*, 2020.

[15] D. Sadigh, N. Landolfi, S. S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for cars that coordinate with people: leveraging effects on human actions for planning and active information gathering over human internal state," *Autonomous Robots*, vol. 42, no. 7, p. 22, 2018.

[16] Y. Ren, S. Elliott, Y. Wang, Y. Yang, and W. Zhang, "How shall i drive? interaction modeling and motion planning towards empathetic and socially-graceful driving," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, p. 7.

[17] R. P. Bhattacharyya, R. Senanayake, K. Brown, and M. J. Kochenderfer, "Online parameter estimation for human driver behavior prediction," in *2020 American Control Conference (ACC)*. IEEE, 2020, p. 6.

[18] M. Zhang, A. Abbas-Turki, A. Lombard, A. Koukam, and K.-H. Jo, "Autonomous vehicle with communicative driving for pedestrian crossing: Trajectory optimization," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020, p. 6.

[19] J. Cui, W. Macke, H. Yedidsion, A. Goyal, D. Urielli, and P. Stone, "Scalable multiagent driving policies for reducing traffic congestion," *arXiv preprint arXiv:2103.00058*, 2021.

[20] C. Wu, A. R. Kreidieh, K. Parvate, E. Vinitsky, and A. M. Bayen, "Flow: A modular learning framework for mixed autonomy traffic," *IEEE Transactions on Robotics*, 2021.

[21] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of sumo-simulation of urban mobility," *International journal on advances in systems and measurements*, vol. 5, no. 3&4, 2012.

[22] M. Treiber, A. Hennecke, and D. Helbing, “Congested traffic states in empirical observations and microscopic simulations,” *Physical review E*, vol. 62, no. 2, p. 1805, 2000.