Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Scalability of existing MARL Frameworks for Multi-Lane On-Ramp Merging of CAVs in Mixed Traffic Scenarios

Sai Bala Subrahmanya Lakshmi Kanth Rayanapati, MCS

A Dissertation

Presented to the University of Dublin, Trinity College

in partial fulfilment of the requirements for the degree of

Masters in Computer Science

Supervisor: Dr. Melanie Bouroche

April 2024

Scalability of existing MARL Frameworks for Multi-Lane On-Ramp Merging of CAVs in Mixed Traffic Scenarios

Sai Bala Subrahmanya Lakshmi Kanth Rayanapati, Masters in Computer Science
University of Dublin, Trinity College, 2024

Supervisor: Dr. Melanie Bouroche

Recent improvements in autonomous driving have the potential to revolutionise transportation systems by improving traffic safety and efficiency and reducing traffic congestion. However, even with the current advancements, the seamless integration of Connected Autonomous Vehicles (CAVs) into complex mixed traffic scenarios like highway on-ramp merging still remains a substantial challenge.

Existing approaches to highway on-ramp merging are predominantly focused on single-lane highway on-ramp merging scenarios and often overlook the scenarios where multi-lane on-ramps exist, leaving the behaviour of the CAVs highly unknown in such scenarios. So, to address this crucial gap, this dissertation explores the scalability of existing Multi-Agent Reinforcement Learning (MARL) frameworks to a multi-lane highway on-ramp merging scenario of CAVs in mixed traffic. This dissertation extends the "highway-env" merge simulation environment to include an additional lane on the on-ramp and tests the scalability of the MAPPO, MADQN, and MAACKTR algorithms.

The results show that the MAPPO algorithm is highly efficient and scalable to the modified (multi-lane on-ramp) environment. In contrast, MAACKTR and MADQN algorithms show inconsistent performance and are not scalable to the multi-lane on-ramp environment.

Acknowledgments

I sincerely thank my supervisor, Dr. Melanie Bouroche, and her PhD student, Mr. Bharathkumar Hegde, for their invaluable guidance and support throughout my dissertation journey. Their insightful feedback and constant encouragement were crucial for the completion of my dissertation, and I am grateful for their mentorship.

I also owe a heartfelt thank you to my amma, nanna, and sister for their continuous support and trust in me throughout my integrated master's course. This dissertation would not have been possible without their love and sacrifices over the last five years.

Sai Bala Subrahmanya Lakshmi Kanth Rayanapati

University of Dublin, Trinity College
April 2024

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter introduces the work by first discussing the background (Section 1.1), then highlights the motivation behind the work (Section 1.2) before presenting the research question addressed (Section 1.3). It then summarises the contributions of this work (Section 1.4) before concluding with the report roadmap (Section 1.5).

1.1 Background

Over the past few years, there has been a surge of enthusiasm surrounding self-driving Autonomous Vehicles (AVs), particularly Connected Autonomous Vehicles (CAVs) and their integration into the current world. Rapid advancements in the fields of Artificial Intelligence (AI), electronics, information and communication technologies have played a significant role in the growth of autonomous driving technologies Rosique et al. [2023]. This enthusiasm is led by the potential of AVs to revolutionise transportation systems by increasing safety and efficiency Pendleton et al. [2017]. The autonomous vehicle market is estimated to grow with an average annual growth rate of approximately 20.75% Statista [2023], and by 2030, AVs are estimated to be 76% less likely to be involved in traffic accidents than human-driven vehicles (HDVs) Statista [2024].

Over the years, many automotive companies have heavily invested in autonomous driving technologies to capture the growing demand for AVs Rauniyar et al. [2018]. Waymo (formerly the Google self-driving car project) Waymo LLC [2024], Tesla Tesla, Inc. [2024], and General Motors (GM) General Motors [2024] are the industry leaders in developing and deploying autonomous driving technologies Nikitas et al. [2017]. Other major companies, such as Volvo, Toyota, and Ford, are actively researching and have announced plans to launch fully AVs in the near future, indicating a significant shift in the automotive industry towards autonomous driving Rauniyar et al. [2018].

Recognising the potential of AVs in enhancing safety and efficiency, governments worldwide are actively supporting the development of autonomous driving technologies. Recently, the government of UK has funded £150 million to boost the development of autonomous driving technologies Centre for Connected and Autonomous Vehicles [2023]. In November 2016, the European Commission adopted a Cooperative Intelligent Transport Systems (C-ITS) strategy to converge the investments and regulatory frameworks across the EU to develop and deploy mature C-ITS European Commission [2024]. Further, there is an ongoing effort to establish a standardised framework following international standards for ensuring the reliability and safety of AV systems Takács et al. [2018].

Although AV technologies offer various benefits, in their current stage of development, there are numerous safety and ethical challenges in integrating AVs and CAVs into public traffic networks Martens and van den Beukel [2013]. While companies are actively researching into these safety and cybersecurity concerns, accidents involving AVs cannot be eliminated entirelyAndreia Martinho and Chorus [2021]. Studies show that the number of accidents caused by AVs has increased with the increase in AVs on public road networks Wong et al. [2022]. Notable incidents involving AVs include the 2016 Tesla accident Shepardson [2017] and the 2018 Uber accident CNN [2023]Mayer et al. [2023]. These incidents highlight the challenges that AVs and CAVs face in adapting to the conventional traffic infrastructure and navigating complex traffic situations, such as lane changing, traffic congestion, and on-ramp merging Lengyel et al. [2020].

The automotive industry is actively exploring the potential of multiple-vehicle cooperation to enhance the safety and efficiency of AVs Muzahid et al. [2023]. While AVs operate independently using internal sensors, CAVs communicate with their surrounding vehicles, infrastructure, and other entities to reduce accidents Tang and He [2020]. CAVs improve their decision-making strategies by facilitating real-time information exchange between the vehicles and their environment Susilawati et al. [2023]. They have the potential to address various complex traffic challenges like lane changing, traffic congestion, and on-ramp merging safely and efficiently. However, this integration is not straightforward as the CAVs must not only act as individual vehicles but also have to interact with the surrounding vehicles and the environment to perform safe and efficient actions.

In conclusion, CAVs represent a significant advancement in the automotive industry, potentially enhancing safety and efficiency. While they offer numerous benefits, a few challenges must be addressed to ensure their successful integration into the public traffic network. This sets the stage for exploring multi-lane merging strategies to enable safe and effective on-ramp merging of CAVs in mixed traffic conditions.

1.2 Motivation

The rapid advancements in the field of Connected Autonomous vehicles (CAVs) have got the potential to revolutionise the transportation system by improving traffic safety, efficiency, and reducing traffic congestion.

In ideal scenarios where CAVs operate in a CAV-only environment, implementing CAV technologies can significantly improve traffic safety and overall traffic management as the behaviours of CAVs are predictable and uniform. However, with current CAV technologies, this transition from a world with Human Driven Vehicles (HDVs) to a world of CAVs will not happen in a day, and HDVs will continue to share the roads with CAVs for the foreseeable future. So, CAVs must adapt to these scenarios and learn to co-exist with HDVs that exhibit a wide variety of unpredictable driving styles.

Despite the extensive research in this area, seamless integration of CAVs into complex mixed traffic scenarios remains a substantial challenge as the CAVs should not only react to any potential hazards on the road but also have to factor in the behaviours of the HDVs sharing the road. CAVs must be able to communicate and adapt to the diverse behaviours exhibited by both CAVs and HDVs. This diversity in driving behaviours introduces significant uncertainties that CAVs must navigate to execute safe and efficient actions. Even with vast amounts of research being done on various aspects of the CAVs, these solutions need to be optimised for various scenarios.

One such scenario that requires more research and optimised solutions is the highway on-ramp merging of CAVs in mixed traffic. In such complex scenarios, the CAVs must not only consider the immediate actions of the adjacent vehicles (including both CAVs and HDVs) but also react according to the overall traffic flow and behaviour patterns across multiple lanes to ensure safe and efficient actions.

Most existing approaches to highway on-ramp merging of CAVs are mainly focused on scenarios with a single-lane on-ramp (Figure 1.1). While effective to a certain degree in such settings, these approaches are not proven to scale effectively to tackle the complexities posed by multi-lane on-ramps scenarios (Figure 1.2). Multi-lane on-ramps are increasingly common in urban and suburban highway systems. They are designed to reduce congestion and improve overall traffic flow. However, the presence of multiple lanes for merging further complicates the merging process for the CAVs due to the presence of additional variables and interactions between the CAVs and HDVs.

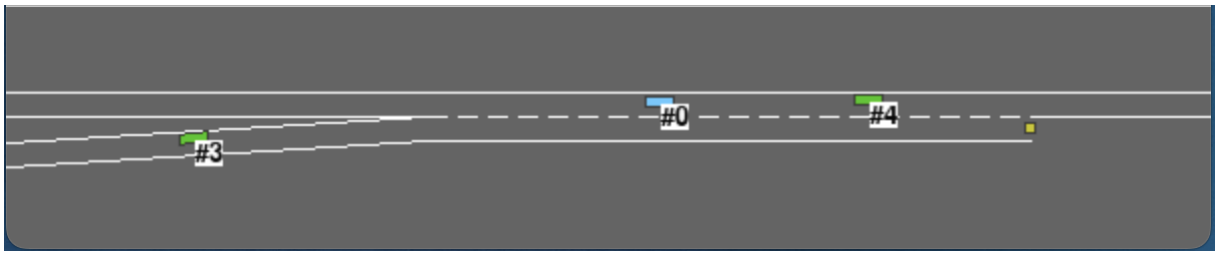

Figure 1.1: Single-lane highway on-ramp Chen et al. [2022].

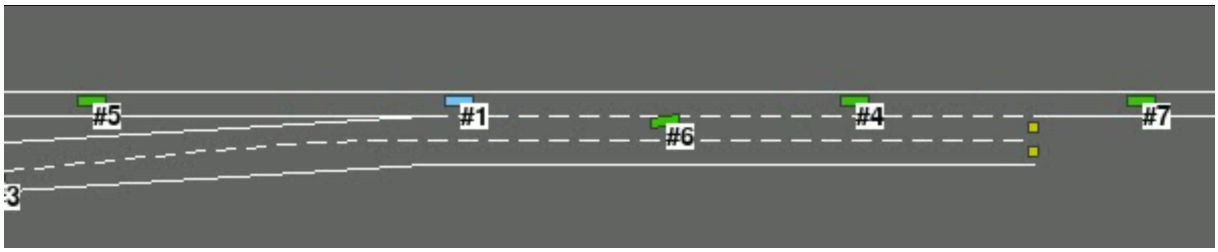

Figure 1.2: Multi-lane highway on-ramp

Therefore, in this paper, I aim to address the gap in the scalability of the existing MARL frameworks in accommodating the increased complexity of multi-lane merging scenarios. In doing so, I seek to extend the existing state-of-the-art frameworks for on-ramp merging to include multi-lane on-ramp scenarios. Then, I plan to explore the adaptability and scalability of these existing frameworks in multi-lane merging scenarios.

This research is motivated by the need to develop scalable, efficient, and safe on-ramp merging strategies for CAVs in mixed traffic conditions, with a particular focus on the primarily underexplored area of multi-lane highway merge ramps. By focusing on multi-lane merging scenarios, I aim to make a significant contribution to this field. Proving the successful scalability of these existing frameworks to multi-lane merging scenarios would be a pivotal step towards the integration of CAVs that promote safe and efficient actions.

1.3 Research Question

The aim of this research is to answer the question :
"Can the existing MARL frameworks for on-ramp merging of Connected Autonomous Vehicles (CAVs) in mixed traffic conditions be adapted effectively to handle multi-lane merging scenarios?"

1.4 Contributions

The main contribution of my work are as follows:

- Investigating multiple state-of-the-art MARL frameworks for highway on-ramp merging.
- Designing and implementing a simulation platform by extending the merge environment from "highway-env", to support the evaluation of MARL algorithms over multi-lane merging scenarios.
- Evaluating the existing frameworks' scalability and performance in multi-lane merging scenarios.

1.5 Structure of the Report

This section discusses the structure of my dissertation.

1. **Chapter 1** begins with a background on Connected Autonomous Vehicles (CAVs), with a focus on the current landscape. Then, we delve into the motivation for the dissertation before presenting the research question.
2. **Chapter 2** introduces the essential concepts used in this dissertation and further discusses the related work in this domain.
3. **Chapter 3** highlights the simulation environment choice and provides a detailed analysis of the changes made to the simulation environment to make it suited to answer our research question. Further, it highlights the challenges and solutions developed.
4. **Chapter 4** offers a detailed evaluation of the results of the simulations conducted. It explains the evaluation metrics, design, and environmental parameters used in detail. It provides a thorough discussion of the results.
5. **Chapter 5** summarises the dissertation by presenting the key findings and drawing conclusions about the study. Finally, it proposes directions for possible future work.

Chapter 2

Literature Review

This chapter initially discusses the background of Autonomous Vehicles (Section 2.1.1), Connected Autonomous Vehicles (Section 2.1.2), Reinforcement Learning (Section 2.1.3), Multi-Agent Reinforcement Learning (Section 2.1.4), highway-env simulator (Section 2.1.5), and OpenAI Gym (Section 2.1.6). Further it discusses the relevant work done in this domain (Section 2.2). Finally, identifies the gap in the research and indicates the direction for this research (Section 2.3)

2.1 Background

2.1.1 Autonomous Vehicles (AVs)

Autonomous vehicles (AVs), or self-driving cars, are a significant innovation in transportation technology, allowing vehicles to operate without human intervention. The primary functions of an autonomous vehicle revolve around the ability to perceive the environment, make informed decisions, and execute control without human input Zanchin et al. [2017]. According to their current classifications, notable from SAE International, autonomous driving is described into five levels based on the extent of human driver necessity and the sophistication of automated systems. Level 0 implies no automation, while level 5 represents full automation, where no Human intervention is required under any circumstances Ribbens [2017].

AVs are equipped with various sensors like cameras, RADAR, and LIDAR, which work with Artificial Intelligence (AI) and aid in navigating and understanding complex traffic scenarios. The transition from human-operated to fully autonomous vehicles involves integrating these technologies to handle all types of driving traditionally managed by humans Zanchin et al. [2017].

A true autonomous system operates independently without the need for external in-

puts or communications; however, when vehicles rely on external communications with infrastructure or other vehicles for information gathering or navigating, they are part of a "cooperative" system rather than being fully "autonomous" Connelly et al. [2006]. Despite their potential, AVs must be capable of understanding and predicting human behaviour accurately to safely co-exist with human-operated vehicles and pedestrians Rendong Bai and Liu [2019].

2.1.2 Connected Autonomous Vehicles (CAVs)

Connected Autonomous vehicles represent a transformative progression within the autonomous sector by combining autonomous driving technologies with advanced communication systems Umberto Montanaro and Mouzakitis [2019]. These vehicles can operate autonomously for extended periods of time without the need for human involvement, significantly enhancing vehicle functionality and transportation efficiency Talebpour and Mahmassani [2016a]Umberto Montanaro and Mouzakitis [2019]. The capability of CAVs to revolutionise highway traffic stabilisation and performance, along with safety, due to their potential to completely transform modes of transportation, has sparked enormous recognition.

CAVs encompass the concepts of autonomous vehicles (AVs) and Vehicle to Vehicle (V2V) Communication. AVs are vehicles where human decision-making is either supplemented or entirely replaced by autonomous systems. On the other hand, V2V communication facilitates wireless connectivity between autonomous vehicles and tower vehicles within a wireless communication range Ali Alheeti and McDonald-Maier [2017]. Unlike traditional vehicles, CAVs utilise cooperative capabilities such as a combination of sensors, Artificial Intelligence, and Machine Learning Algorithms. These advanced technologies, facilitated by the networked communication between the vehicles and the surrounding infrastructure, enables CAVs to perceive the environment, make decisions and navigate safely Talebpour and Mahmassani [2016a].

The core technology of CAVs includes complex systems for path planning, vehicle management, and environment identification. An open platform strategy that makes use of traditional vehicles and sensors to speed up the development and testing of these innovations is considered essential for improving the algorithms that allow CAVs to carry out challenging autonomous navigating operations Kato et al. [2015].The advancements of CAVs have inspired a wide array of research, focussing on their interaction with pedestrians and cyclists, policy implications, and enhancements in traffic management Stanciu et al. [2018]He et al. [2022b]Fagnant and Kockelman [2015].

A primary benefit of CAVs is their ability to improve traffic flow and traffic capacity

on highways while lowering fuel consumption and environmental impacts by employing safe and efficient driving practices Luettel et al. [2012]. These vehicles might not only significantly lower human error rates that result in collisions but also investigations into the safety impacts of CAVs on highways and areas prone to accidents reveal that they would decrease the likelihood of collisions and improve driver awareness of their surroundings Luettel et al. [2012]Papadoulis et al. [2019a]Zhang et al. [2021]. The concept of CAVs is built upon ultra-reliable, low-latency communication theories, empowering vehicles to interact with their environment, transfer crucial data and make data-driven decisions in real time Yamazato [2017]Khan et al. [2021]. CAVs can constantly change their behaviour according to the kind of vehicle initiating the communication Martin-Gasulla et al. [2019]. Furthermore, employing game theory to model lane-changing behaviours in connected environments highlights the crucial overlap between human decision-making processes and telecommunications in the concepts of CAVs Talebpour et al. [2015].

The rise of CAVs calls for substantial changes to existing road infrastructure to support the mixed traffic environments of autonomous and manual-operated vehicles He et al. [2022b]. While acknowledging the potential limitations in CAVs' ability to perceive information about roads and nearby vehicles, it is essential to recognise that their decision-making processes may not always be perfect Yao et al. [2021].

The communication framework essential for CAVs involves a range of communication modes, including vehicle-to-vehicle (V2V), vehicle-to-roadside unit (V2R), vehicle-to-infrastructure (V2I), vehicle-to-personal device (V2P), and vehicle-to-sensor (V2S) communication. This wide array of communication types underscores the convoluted network architecture that underpins the seamless operation of CAVs Wang [2023].

2.1.3 Reinforcement Learning (RL)

Reinforcement learning (RL) is a machine learning method that trains agents to make decisions that maximize the numerical rewardAWS [2023]Sutton and Barto [2020]. Unlike supervised learning, where the systems are trained on labelled examples, and unsupervised learning, which tries to find the hidden structures in unlabeled data, the agent in Reinforcement learning is not guided through the actions to be done. Instead, it learns the optimal decision-making strategies through "trial-and-error" exploration of the environment. Reinforcement learning also considers a "delayed-reward" as the cumulative reward attained for the learner's actions not only depends on the current reward but also on all subsequent rewards in the sequence Sutton and Barto [2020]. For example, in a backgammon game, a reward of 1 might be linked to a state of having moved all one's pieces off the board, representing a winning state. While a reward of 0 can be associated

with all the states leading up to a win. So, the agent's objective would be to maximize the long-term reward instead of just focusing on the immediate gains Sutton and Barto [2020].

In reinforcement learning, the agent actively learns through its own experience by interacting with its environment and receiving rewards for its actions. These rewards can be both favourable and unfavourable based on the outcomes of the actions taken by the agent. A positive reward can encourage the agent to repeat its actions, which gradually leads to the achievement of the goal, and a negative reward discourages the agent from entering certain undesirable or dangerous situations. Negative rewards act as penalties that promote the agent to learn and avoid harmful actions Fuchida et al. [2010]. Learning from these rewards and penalties, the agent gains insights into the actions to take in the environment to achieve the maximum reward.

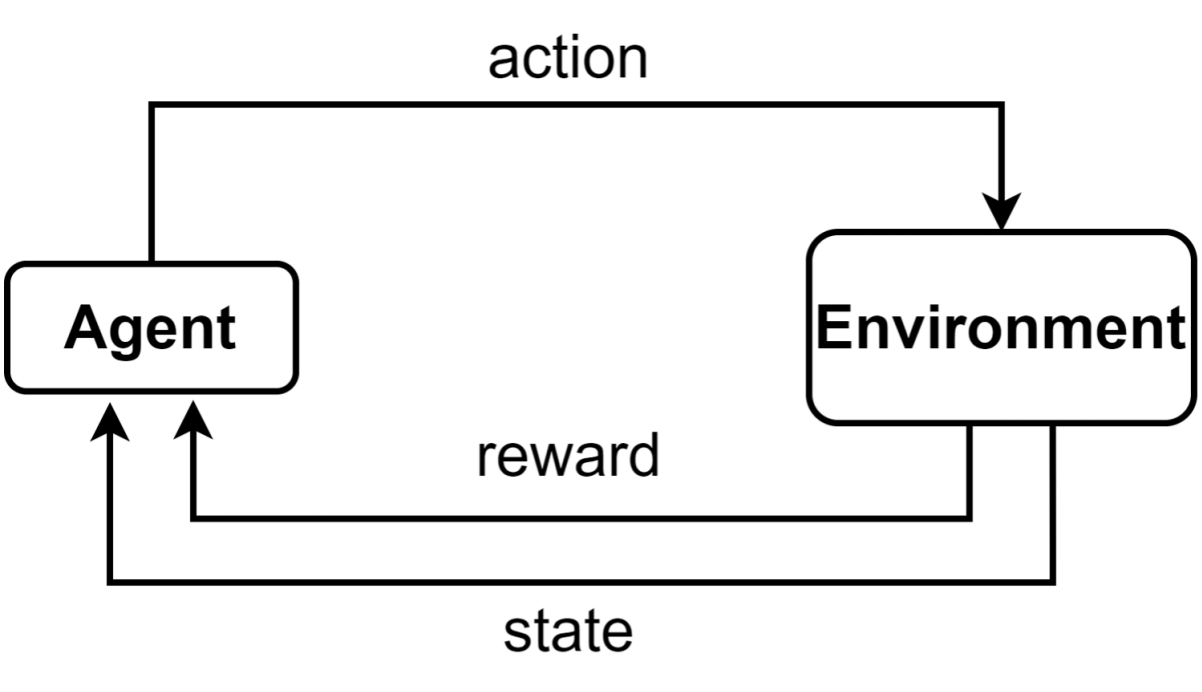

Figure 2.1: Basic Reinforcement Learning (RL) Framework Wang et al. [2023]

One of the critical challenges in reinforcement learning is finding the optimal exploration-exploitation trade-off. On the one hand, the reinforcement learning agent needs to select actions it has not selected before to learn and explore the environment to discover actions that yield the maximum reward. On the other hand, to obtain a high reward, the agent must also choose actions that have been successful in the past to acquire the maximum reward. This balance between exploration and exploitation is necessary for yielding the highest reward. So, a reinforcement learning agent must try a variety of actions, continuously learning and adapting from strategies that give the maximum cumulative reward

Sutton and Barto [2020].

In Reinforcement Learning (RL), at each time step t, the transition of the agent from the current state $s_t \in S \subseteq \mathbb{R}^n$ to the next state $s_{t+1} \in S \subseteq \mathbb{R}^n$ by taking the action $a_t \in A \subseteq \mathbb{R}^m$ will result in a reward $r_t \in \mathbb{R}$ based on the reward function R. The transition of the agent from state s_t to s_{t+1} is called an iteration, and the sequence of states that lead to a terminal state is called an episode Doe and Smith [2018] Chen et al. [2022].

Markov Decision Processes (MDPs)

Most Reinforcement Learning (RL) problems are modelled as Markov Decision Processes (MDPs) because this framework allows for structured representation and solution of sequential decision-making problems with limited feedback. MDPs are mathematical formulations that define the interactions between an agent and its environment through states, actions, and rewards, helping RL algorithms learn optimal behaviours van Otterlo and Wiering [2012] Doe and Smith [2018] Hu et al. [2018].

In Partially Observable Markov Decision Processes (POMDPs), where the agent can only observe part of the state, the MDP is defined by a tuple (S, A, R, P, γ) Doe and Smith [2018] Hu et al. [2018]. Here:

- S is a finite set of states, i.e., the state space.
- A is a finite set of actions, i.e., the action space.
- $R : S \times A \times S \to \mathbb{R}$ is the reward function.
- $P : S \times A \times S \to [0, 1]$ is the state transition probability matrix.
- $\gamma \in [0, 1]$ is the discount factor.

Policy

A policy π in reinforcement learning (RL) is a strategy or a distribution over actions given the current state Cai et al. [2021] OpenAI [2018]. It is expressed as:

$$\pi(a \mid s) = P(a_t = a \mid s_t = s).$$

Policies can be of two types:

- **Deterministic policies:** The actions are determined solely by the current state Cai et al. [2021] OpenAI [2018].

- **Stochastic policies:** A random noise is added to the action chosen by the policy Cai et al. [2021] OpenAI [2018].

Return

Reinforcement Learning (RL) aims to choose actions that maximise the expected return value over time Tangkaratt et al. [2018]. The return, G_t, is the total discounted reward from time step t that guides the agent to make optimal decisions Nguyen et al. [2020]:

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

Discounted future rewards help interpret the current value of future rewards. The discount factor γ influences the agent's behaviour by valuing either immediate or delayed rewards; a γ value close to 0 leads to short-sighted evaluations, and a γ value close to 1 leads to far-sighted evaluations Xie et al. [2020].

Value Function

In Reinforcement Learning (RL), value functions evaluate the desirability of states or state-action pairs based on the expected return. These functions are central to determining the best policies an agent can follow. There are two primary types of value functions:

State Value Function The state value function, denoted as $V(s)$, takes a state s as input and calculates the agent's expected return, or cumulative reward, from following policy π Winder.AI [N.d.] Karunakaran [2021] Li et al. [2004]. The mathematical expression is:

$$V^{\pi}(s) \doteq \mathbb{E}^{\pi}[G \mid s] = \mathbb{E}^{\pi}\left[\sum_{k=0}^{T} \gamma^k r_k \mid s\right]$$

where:

- G is the return,
- s is the state,
- γ is the discount factor,
- r is the reward.

State-Action Value Function Commonly denoted by $Q(s, a)$, the state-action value function takes a state s and an action a as inputs. It calculates the expected return of taking action a in state s, under policy π Karunakaran [2021] Xie et al. [2020]:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^{k} r_{t+k+1} \mid s_t = s, a_t = a\right]$$

where:

- G is the return,
- s is the state,
- a is the action,
- γ is the discount factor,
- r is the reward.

Bellman Equation

The idea behind the Bellman equation is that the value of the current state is the sum of the expected value of the immediate reward and the discounted value of the next state.

For a given policy π, the state value function $V^{\pi}(s)$ can be expressed by the Bellman equation as OpenAI [2018] Grosse et al. [2020] face [N.d.]:

$$V^{\pi}(s) = \mathbb{E}_{\pi}[R_{t+1} + \gamma V^{\pi}(S_{t+1}) \mid S_t = s]$$

where:

- $\mathbb{E}_{\pi}$ denotes the expected value under policy π,
- R_{t+1} is the reward at the next time step,
- γ is the discount factor,
- S_{t+1} is the state at the next time step.

Similarly, the Bellman equation for the state-action value function $Q^{\pi}(s, a)$ is given by OpenAI [2018] Grosse et al. [2020] face [N.d.]:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a]$$

The Bellman equation helps in iteratively updating the value functions.

Optimal Value Functions

The **optimal state value function** $V^*(s)$ is defined as the maximum state value function over all policies:

$$V^*(s) = \max_{\pi} V^{\pi}(s)$$

where $V^{\pi}(s)$ represents the value function under policy π for state s OpenAI [2018].

Similarly, the **optimal state-action value function** $Q^*(s, a)$ is the maximum state-action value function over all policies:

$$Q^*(s, a) = \max_{\pi} Q^{\pi}(s, a)$$

where $Q^{\pi}(s, a)$ represents the state-action value function under policy π for state s and action a OpenAI [2018].

Optimal Policy

The **optimal policy** π^* is defined as being better than or equal to all other policies OpenAI [2018]. This is expressed as:

$$\pi^* \geq \pi, \quad \text{for all } \pi$$

This means that for the optimal policy π^*, the following statements hold for all states s and actions a:

$$V^{\pi *}(s) \geq V^{\pi}(s) \quad \text{and} \quad Q^{\pi *}(s, a) \geq Q^{\pi}(s, a)$$

A Reinforcement Learning (RL) agent aims to learn the optimal policy

$$\pi^* : S \to A$$

that maximises the reward

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

Chen et al. [2022] where:

- r_{t+k} is the reward at time step $t + k$,
- γ, the discount factor, $\gamma \in (0, 1]$.

Advantage Function

In Reinforcement Learning, we sometimes do not describe an action as being better in terms of absolute values. Instead, we can define it as, on average, how much better it is than the others (relative advantage of the action). This advantage function can be mathematically represented by OpenAI [2018]:

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s)$$

where:

- $A_{\pi}(s,a)$ denotes the advantage of taking action a in state s under policy π,
- $Q_{\pi}(s,a)$ is the state-action value function,
- $V_{\pi}(s)$ is the state value function.

Policy Gradient

Policy gradient methods in Reinforcement Learning (RL) focus on directly optimising the policy by estimating the gradient of the agent's policy with respect to its parameters Li et al. [2021]. They use gradient ascent to find weights and iteratively improve the expected returns. The policy is updated using the following equation,

$$\nabla J(\theta) = \mathbb{E}[\nabla \log \pi_{\theta}(s,a) A^{\pi_{\theta}}(s,a)]$$

Miller [2023] where:

- $\nabla J(\theta)$: The gradient of the objective function with parameters θ,
- $\nabla \log \pi_{\theta}(s,a)$: The gradient of the logarithm of the policy π,
- $A^{\pi_{\theta}}(s,a)$: The advantage function of taking action a from state s.

2.1.4 Multi-Agent Reinforcement Learning (MARL)

Multi-Agent reinforcement learning (MARL) is a field of study that is an extension of Reinforcement Learning (RL), where multiple autonomous agents interact within a shared environment, working towards a common goal to maximise the sum of received rewards. Each agent receives rewards based on the actions chosen. In contrast to single-agent reinforcement learning, where the learner interacts with the environment to maximise their own reward, MARL agents must learn and coordinate their actions with other agents

in the same environment to achieve common goals or compete against each other MAR Busoniu et al. [2008].

Similar to Reinforcement Learning (RL), MARL algorithms learn optimal policies via trial-and-error to maximise the agents' cumulative rewards and returns. A set of "n" individual agent actions, referred to as joint action, can change the dynamics of the environment based on the individual rewards of the agents obtained as a result of this change in the environment.

MARL is a rapidly growing field that showcases its adaptability across a broad spectrum of fields, including robotics, game theory, complex systems, distributed control, and resource management. Multiple agents must interact and coordinate in these fields to adapt to the environment and yield maximum rewards. MARL techniques are versatile and can be applied to various real-world scenarios like autonomous driving that involve cooperative, competitive, or mixed behaviours of multiple agents. In autonomous driving, MARL can be utilised to model intelligent agents that exhibit cooperative behaviours to make collective decisions in complex scenarios like highway on-ramp merging Zhu et al. [2024].

MARL techniques provide a robust and adaptable framework for tackling various complex real-world scenarios involving multiple agents.

2.1.5 Highway-env

Highway-env is an open-source, lightweight simulation platform that includes various environments, such as Highway, Merge, Roundabout, Parking, Intersection, and many more. It has been developed and maintained by Eduard Leurent Leurent [2018b] since 2018 and is used for simulating decision-making scenarios in autonomous driving tasks Leurent [2018c]. Highway-env is lightweight and highly computationally efficient compared to other open-source simulators like CARLA and SUMO Sun et al. [2021].

Some Key features of highway-env are:

Realistic Simulation: Highway-env models dynamic vehicle behaviors such as acceleration, deceleration, and steering, providing realistic simulations.

Multiple Agents: Supports simulations with multiple vehicles, including Controlled Autonomous Vehicles (CAVs) and Human Driven Vehicles (HDVs), enabling training in diverse scenarios.

Customizable: Offers extensive customization options, allowing adjustments to environment features like the number of lanes, vehicle densities, and presence of obsta-

cles.

Collision Detection and Rewards: Implements collision tracking and assigns rewards or penalties, aiding in the development of safe and efficient driving behaviors.

2.1.6 OpenAI Gym

Highway-env is implemented using the OpenAI Gym framework Dinneweth et al. [2022b] Brockman et al. [2016], a popular open-source library for developing and comparing Reinforcement Learning (RL) algorithms Towers et al. [2023]. It was created by OpenAI, but now it is renamed to "Gymnasium" and is being actively maintained by Farama Foundation. OpenAI Gym provides a wide variety of "gym" environments that can be used to train and test Reinforcement Learning (RL) agents. These environments can range from simple 2D environments to control problems like the inverted pendulum, to advanced environments like Atari video games to complex three-dimensional environments like simulated robotics Foundation [2023a]Foundation [2023b].

2.2 Related Work

2.2.1 CAVs in Mixed Traffic Scenarios

Connected and autonomous vehicles (CAVs) perceive their environment using a variety of sensors, including lidar, cameras, and radar Guanetti et al. [2018]. Considering the existing shortcomings in the perception process, He et al. [2022a] highlight the need to prepare road infrastructure for mixed traffic flow conditions, emphasizing the importance of addressing the imminent emergence of CAVs He et al. [2022a]. Without this, the envisioned future of CAVs to Improve traffic congestion and decrease the accidents caused by human error may remain unattainable.

Incorporating CAVs into mixed traffic conditions with human-driven vehicles (HDVs) can improve traffic flow stability and throughput and enhance safety Talebpour and Mahmassani [2016b]. Introducing CAVs into mixed traffic systems is associated with eco-driving, emphasizing their importance in significantly reducing energy consumption and pollutant emissions Wang et al. [2020]. Another critical benefit of CAVs in mixed traffic is the improvement in road safety. Research shows that CAVs significantly reduce accidents, providing compelling safety benefits even at low penetration rates Papadoulis et al. [2019b]. Moreover, dedicated lanes (DL) for CAVs on freeways have proven to enhance traffic efficiency and reduce traffic conflicts in mixed-traffic scenarios Kim et al. [2023]He et al. [2022a].

Despite the potential advantages of CAVs, they also present various security and privacy concerns Nanda et al. [2019]. The growing levels of automation and connectivity contribute to intensifying the security threat. This was demonstrated by Charlie Miller and Chris Valasek in 2015 by successfully hacking into a Jeep Cherokee via the Internet, exploiting a vulnerability in the vehicle's infotainment system Crede. Additionally, Song and Ding [2023] highlighted the safety risks associated with the transition of a CAV into an automated vehicle (AV) in mixed traffic due to communication failure, indicating a significant increase in safety risks Song and Ding [2023].

Overall, integrating CAVs into mixed traffic environments with HDVs holds the potential to improve traffic flow stability, throughput, and safety. However, careful planning and management are required to address the safety risks and facilitate a smooth integration of CAVs into existing traffic systems.

2.2.2 Communication Protocols in CAVs

A variety of viable wireless access technologies are available for vehicle-to-vehicle and vehicle-to-infrastructure communication purposes. Within the domain of CAVs, widespread wireless access technologies consist of communication via satellite, worldwide compatibility for microwave access (WiMAX), specialized short-range communications, cell phone networks, and WLAN.

Dedicated Short Range Communications (DSRC)

Dedicated short-range communications (DSRC) underpin the majority of vehicular communications Zhao et al. [2019b]. DSRC has been designed specifically to facilitate V2V and V2I communications with minimal latency and high reliability.

4G/5G Cellular Networks

4G cellular networks have the capability to deliver mobile ultra-broadband internet access Campos [2017]. Individuals are granted access to a multitude of networks without the need to switch between them manually. Certain technologies, such as microcell base stations and mobile communication systems with long-term evolution, are at the disposal of entities that aim for rapid transmission in particular regions.

WLAN and WiMAX

WLAN is a wireless communication technology that enables highly adaptable access points to connect to the broader internet. The coverage area of each access point is approximately

100 meters Deng et al. [2017]. It is simple to expand the range of a WLAN by incorporating one or more repeaters. The router's physical port limit does not constrain WLAN. As a result, dozens or even hundreds of devices may be supported.

On the other hand, WiMAX wireless broadband communication technology implements the IEEE 802.16 specification. For fixed stations, the maximum coverage range of WiMAX is 50 kilometers Malankar and Shah [2017].

Satellite Communications

Telecommunication signals may be transmitted and amplified via satellite communication Luo et al. [2019]. It is capable of establishing communication channels between geographically dispersed signal senders and receivers. The transmission data rate in satellite communication is restricted to a maximum of one thousand gigabits per second. The transmission data rate is limited to a maximum of 1000 Mbps. The range of its coverage extends from 100 km to 6000 km. Typically, satellite communication is linked to a 4G/5G cellular network in the context of CAVs.

2.2.3 MARL in Autonomous Vehicles

Autonomous vehicles (AVs) are being developed with the goal of reducing the occurrence of accidents by trying to eliminate human intervention. Recent advancements in the field of autonomous vehicle technologies have introduced increasing automation levels from level 1(essential assistance) to level 5(denoting full automation). In mixed traffic scenarios with both AVs and human drivers sharing the road, challenges arise due to the unpredictability of human behaviour, making it hard for AVs to adapt to mixed traffic scenarios. MARL has emerged as an important field of research for designing decision-making strategies in AVs that consider the unpredictability of human behaviour and adapt to mixed traffic scenarios Dinneweth et al. [2022c]. The application of MARL in the field of AVs has proven pivotal for developing adaptive, learning-based decision-making strategies, which is essential for AVs' co-existence in mixed and fully autonomous traffic scenarios Dinneweth et al. [2022c]Zhou et al. [2022b].

The implementation of MARL in the context of AVs is well investigated with its applications in different scenarios, including cooperative lane changing, traffic signal control, and highway on-ramp merging scenarios, demonstrating its adaptability in handling various complex driving scenarios in mixed traffic conditions Zhou et al. [2022b]Chu et al. [2020b]Chen et al. [2022]. Further, the study Lu et al. [2020] explores the implementation of MARL for hierarchical autonomous decision-making and motion planning of autonomous vehicles in complex dynamic traffic scenarios Lu et al. [2020].

Furthermore, the integration of MARL in autonomous driving scenarios has highlighted the potential of MARL in addressing challenges in enhancing communication and coordination in autonomous vehicles Chen et al. [2021]Xiao et al. [2023]Schmidt et al. [2022]. In the study Qu et al. [2020a], the authors have demonstrated the capability of MARL to mitigate traffic congestion in autonomous vehicle environments by adjusting the acceleration and speed of different vehicles Qu et al. [2020a].

Overall, the literature review highlights the growing importance of MARL in addressing the shortcomings of autonomous vehicle technologies. The continued research in the field of MARL for the improvement in autonomous vehicle technologies has the potential to revolutionise the future of autonomous vehicles.

2.2.4 MARL in Autonomous Driving

Autonomous driving in urban highway environments presents complex scenarios where multiple vehicles need to interact with each other frequently to execute safe and efficient actions Chen et al. [2019]. Multi-Agent Reinforcement Learning (MARL) frameworks hold the potential to train the control policies for these vehicles to navigate complicated scenarios like busy junctions, lane changing, roundabouts and on-ramp merges Lin et al. [2021]. In navigating such complex scenarios, the agents need to take both continuous (e.g. steering, acceleration/braking) and discrete actions (e.g. lane changing, turning) to avoid collisions and perform safe actions Crewe et al. [2023]. An agent interacts with other agents to receive information like position in the lane, orientation, and speed from other nearby vehicles in the road network. However, these observations might be affected by factors such as sensor noise and partial observability caused by occlusions, ie. Vehicles blocking the view of the agent leading to a scenario where the agent can not observe and communicate with its entire surrounding environment Chu et al. [2020a].

In MARL frameworks for autonomous driving, the reward design is a crucial component that considers multiple factors and significantly influences the agents' behaviour. Autonomous driving agents are expected to exhibit fast and efficient driving behaviour while avoiding collisions. So, a combination of both positive and negative rewards is typically used in designing the reward structure.

Negative rewards are usually assigned for undesirable actions like causing collisions, abrupt acceleration/braking (unnatural behaviour), and frequent lane changes Kim et al. [2021]. These negative rewards discourage the agent from choosing actions that display unnatural behaviour or cause accidents. On the other hand, positive rewards are assigned to actions that promote efficient, safe, and natural driving behaviours, like minimizing driving times while avoiding collisions Kim et al. [2021]. Assigning these positive rewards

encourages the agent to choose actions that promote safe and efficient driving practices, contributing to overall traffic flow and safety.

In this research, we will be considering a multi-agent scenario where all the agents work towards a common goal of performing safe and efficient highway on-ramp merging.

2.2.5 Challenges in the Application of MARL to Autonomous Driving

Applying Multi-Agent Reinforcement Learning frameworks to address the complexities of Autonomous driving in mixed traffic scenarios presents various challenges that are typically not encountered in single-agent scenarios. Most of the challenges arise due to the presence of multiple agents with conflicting goals, continuous optimization of the policies and the partial observability of the agent's environment.

Non-stationarity caused by learning agents

A vital challenge of the application of MARL in autonomous driving is the problem of non-stationarity caused due to continuous learning and adaptation of the agents. This will result in continually changing agents' policies, making it difficult for them to learn stable strategies. Each agent adapts to the other agents' policies, whose policies, in turn, adapt to the changes in other agents, causing cyclic and unstable learning dynamics. This issue is further complicated as different agents learn at different rates based on rewards and observations. For example, scenarios like changing traffic patterns, road conditions, or actions of other agents require constant adaptation to the evolving environment, making it difficult for the agents to learn effectively Dinneweth et al. [2022a]. The ability to handle the issue of non-stationarity is a crucial aspect of MARL frameworks and has been an area of research. Techniques such as considering the long-term influence of an agent's actions Kim et al. [2022], adapting to opponent agents' behaviours, influencing other agents' strategies Wang et al. [2021b] have been explored to resolve this issue. Dealing with non-stationarity is essential for stable and efficient learning of MARL agents Li et al. [2022].

Partial Observability

Partial Observability is another significant challenge for MARL frameworks where the agents lack complete information about the environment's state due to restricted communication between the agents Dinneweth et al. [2022a]Chu et al. [2020a]. In practical applications like autonomous driving, partial observability is caused by sensor faults and

occlusions caused by other vehicles blocking the view of the agent's environment Ding et al. [2022]. For example, in the context of autonomous driving, it is difficult for the agents to observe and predict the actions of all the other agents in the shared environment when their view is blocked by other vehicles Dinneweth et al. [2022a]. Various techniques like decentralized learning through communication Karten et al. [2023], Integrating knowledge compilation with reinforcement learning Ling et al. [2021], centralized training with decentralized execution Zhao et al. [2022] have been explored to solve the issue of partial observability. Further, approaches based on partially observable Markov decision processes (POMDPs) Wu et al. [2020] have also been proposed to handle decision-making in autonomous driving scenarios.

Curse of Dimensionality

Curse of Dimensionality Acito [2023] is another issue for MARL frameworks that refers to the exponential increase of the state-action space with the increase in the number of agents, leading to an exponential increase in the learning required, higher computational complexity, and resources required Wang et al. [2021a]. Due to this issue in scalability, MARL algorithms face difficulties in achieving complete exploration of the environment, making it an even bigger issue for autonomous driving Hao et al. [2022]Salem et al. [2023]. Various approaches like using Observation Embedding and Parameter Noise to enable scalable Deep MARL Zhang et al. [2019], Policy Distillation and Value Matching Wadhwania et al. [2019], use of Projection Exploration Tang et al. [2023] have been proposed to solve this problem of scalability in MARL frameworks.

Addressing these issues by refining the learning schemes is essential for developing MARL frameworks in complex mixed traffic scenarios.

2.2.6 MARL in Traffic Signal Control

Multi-Agent Reinforcement Learning has recently gained significant attention for its potential to address complex challenges in mixed traffic scenarios. The applications of MARL can be seen extensively in the areas of traffic control and management, addressing problems such as traffic signal control, congestion management and control systems.

Traditional traffic signal control methods often fall short in dynamic large-scale traffic signal scenarios. These multi-intersection traffic signal control shortcomings can be effectively addressed by leveraging MARL frameworks. The applicability of MARL in addressing traffic signal problems has been extensively researched.

MARL has been proven to effectively address the challenges of traffic demand, traffic

jams, and environmental pollution in multi-intersection scenarios in large-scale road networks. In the study Hu et al. [2023], the authors develop a decentralised MARL algorithm, MFDQL-DTC, that independently learns policies for each intersection to improve overall traffic efficiency. The MFDQL-DTC algorithm incorporates traditional traffic methods, intelligent control algorithms, and mean-field theory to reduce the complexity of joint action space and provide improved real-time traffic signalling in large-scale road networks. Additionally, MFDQL-DTC is efficient in handling convergence in large-scale road networks and outperforms the current state-of-the-art baseline models like MARL-DSTAN in terms of scalability and convergence Hu et al. [2023].

Similarly, Qu et al. [2020b] proposed a distributed control method for urban networks, MSNE-MARL, that integrates the notion of Mixed Strategy Nash-Equilibrium (MSNE) into the decision-making process of the MARL to prevent disturbance-based traffic congestion. The integration of MSNE and MARL enhanced the proposed method's ability to react rapidly and effectively to the disturbances in urban networks by accelerating the convergence process and reducing the learning time. The proposed MSNE-MARL method outperformed the baseline control strategies, FTC and II-MARL, in various traffic situations, demonstrating its effectiveness in managing traffic congestion Qu et al. [2020b].

Additionally, Chu et al. [2020b] proposed the method "Multi-agent Advantage Actor-critic (MA2C)" that extends the idea of independent Q-learning and independent A2C to address the challenges in adaptive traffic signal control in complex traffic networks. MA2C method addresses the scalability issues by distributing global control to local RL agents to make decisions based on local observations and limited communication. This approach outperforms both independent A2C and independent Q-learning algorithms in an extensive real-world traffic network of Monaco city. It proves the adaptability of the approach in large-scale traffic signal control scenarios Chu et al. [2020b].

Overall, MARL frameworks demonstrate a wide range of applications in addressing the shortcomings in traffic signal control, congestion prevention and traffic optimisation in urban networks and have the potential to revolutionise traffic and congestion management systems and improve traffic flow Qu et al. [2020b]Pan et al. [2020].

2.2.7 MARL in Cooperative Lane Changing

Integrating MARL to address the challenges associated with cooperative lane changing of Connected and Autonomous Vehicles (CAVs) has been pivotal for ensuring safety and enhancing traffic flow. Efficient lane-changing in CAVs helps overtake slow-moving vehicles, manage traffic flow and reduce traffic congestion.

Research shows that data-driven methods such as MARL have emerged as a promising

and scalable solution to address the complexities of decision-making tasks in highway lane changing in mixed traffic scenarios Zhou et al. [2022b]. Treating the lane-changing problem as a decentralized cooperative MARL problem and incorporating a multi-objective reward function that accounts for fuel efficiency, driving comfort, and safety enhanced the performance of the multi-agent advantage actor-critic network (MA2C) algorithm as proposed in the study Zhou et al. [2022b]. The MA2C algorithm outperforms other similar MARL algorithms, such as MADQN, MAACKTR, and MAPPO, in various traffic scenarios, displaying scalability, stability, and adaptive performance in response to different human driving behaviours in mixed traffic conditions Zhou et al. [2022b].

Most cooperative lane-changing algorithms are developed by considering not only the physical characteristics of the subject vehicle but also the leading and following vehicles on the target lane, highlighting the importance of considering the surrounding environment in developing safe and efficient lane-changing algorithms Shi et al. [2019]. MARL algorithms that take into account the surrounding environment involve an agent controlling the headway by providing merging advisory services at merging points for efficient outer-lane vehicle merging, whilst other agents focus on the lane-changing advisory services at advance lane-changing points to control the lane changes in AVs Zhu et al. [2021]. Furthermore, MARL has been used to develop lane-changing algorithms for CAVs in mixed-traffic environments, considering the motions of autonomous and human-driven vehicles(HDVs) before changing lines Zhou et al. [2022b].

The literature review highlights the growing importance of MARL frameworks in addressing the challenges associated with cooperative lane changing for CAVs in mixed traffic scenarios. The application of MARL to tackle the lane-changing problem is a promising approach to enhancing traffic flow and safety in CAV operations.

2.2.8 MARL in Highway On-Ramp Merging

On-ramp merging of connected and automated vehicles (CAVs) in mixed-traffic highway scenarios is crucial for traffic management and safety. Efficient and safe merging is crucial for minimising traffic congestion and avoiding the risk of accidents. Varying behaviour and decision-making of different drivers in mixed traffic scenarios can lead to unpredictable situations that challenge AVs in reacting to the dynamically changing environment. The need for coordination and communication between the vehicles further complicates the on-ramp merging process Chen et al. [2022]. Further, it is highlighted that the prediction of HDV behaviour and arrival times at on-ramps are crucial for effective coordination within CAVs Ma et al. [2023].

Integrating CAVs into mixed traffic during highway on-ramp merging has gained sig-

nificant attention in recent research, focusing on developing control strategies and optimisation frameworks that facilitate the efficient merging of CAVs at highway on-ramps Zhu et al. [2022]. These strategies ensure safety and minimise delays by optimising merging times, vehicle trajectories, and platoon coordination Mahbub et al. [2021]Ye et al. [2019]. Furthermore, the potential of decentralised control algorithms to coordinate CAVs in various traffic scenarios like highway on-ramp merging has been explored Zhao et al. [2019a].

Adjusting vehicle speed and regulating lane changes are two of the most challenging tasks that must be completed in on-ramp merging scenarios on urban highways Amezquita-Semprun et al. [2019]. Various control strategies for CAVs have been proposed to examine how system vehicles safely and efficiently navigate the convergence zone Xu et al. [2019]. Lu and Hedrick [2003] introduced the view of virtual vehicle platooning and transformed the ramp merging issue into a vehicle-following issue by mapping each ramp vehicle to the main road. The centralised controller regulates the velocity of every vehicle in the system to synchronise the moment the vehicle enters the convergence zone and prevent collisions. Cao et al. [2015] utilised a model predictive control framework to optimise the vehicle's trajectory and generate an appropriate distance by regulating the vehicle's speed to guarantee the safety of ramp vehicle merging.

The study Liu et al. [2021] presented a strategy for coordinating CAVs in multilane traffic on-ramp convergence. A model of uneven traffic flow was developed considering the need for uniformity in traffic flow across distinct lanes in the multilane scenario. Furthermore, a reinforcement learning model is developed based on this model to assist in lane selection to mitigate the congestion in the outside lane that arises from ramp passenger vehicle merging. The simulation results indicated that fuel economy and traffic efficiency increase constantly until the optimum allowable road capacity is reached, as vehicle flow and the dispersion of traffic flow between channels increase.

In the paper Schester and Ortiz [2019], the authors present an extended model that utilises continuous space of states and actions, integrating a MARL approach to train the controllers in an idealised environment. It leverages the recent developments in RL and employs artificial neural network (ANN) architectures for function approximation and policy modelling within the multi-agent Q-learning approach. Further, the research evaluates the performance of the trained controllers in preventing collisions through various simulations involving vehicles with diverse behaviours, highlighting the effectiveness of the proposed MARL approach in mixed-traffic AV scenarios.

Hu et al. [2019] proposes the decision-making with adaptive strategies (IDAS) method for resolving decision-making challenges associated with autonomous vehicle merging scenarios by incorporating driver type and road priority. By integrating driver type and road priority into self-driving vehicles, the authors aim to empower AVs to autonomously learn

from the actions of other drivers during interactions and utilise their cooperation to navigate different merging scenarios effectively. To address this within a MARL framework, the study introduced a double critic approach consisting of a centralised and decentralised action-value function. This method outperformed other approaches in terms of success rate and merging efficiency.

Zhou et al. [2022a] proposed a distributed multi-agent deep reinforcement learning approach for cooperative merging control in connected and automated vehicles (CAVs) called multi-agent Deep Deterministic Policy Gradient (MADDPG). This approach considers various factors such as energy consumption, rear-end safety, lateral safety, safe merging distances, and acceleration limits to optimise on-ramp merging scenarios. This method aims to enhance the efficiency and safety of on-ramp merging of CAVs. In order to tackle the issue of a dynamic environment that arises from decentralised learning of CAVs, Nakka et al. [2022] introduces a decentralised framework using MADDPG to coordinate CAVs during highway convergence. This framework enables the transmission and implementation of policies acquired by a limited subset of trained CAVs to unlimited CAVs. In addition, it employs a reward function that incentivises high-speed travel, promoting safer traffic flow and reducing rear-end and lateral collisions.

Sun et al. [2020] proposes a Cooperative Decision-Making for Mixed Traffic (CDMMT) mechanism specifically designed to facilitate efficient and smooth ramp merging of CAVs and reduce potential conflicts that may arise due to the non-cooperative behaviour of HDVs in mixed traffic. This study aims to improve traffic efficiency and safety in mixed traffic scenarios by leveraging discrete optimisation and bi-level dynamic programming. Additionally, the proposed CDMMT mechanism incorporates optimal control-based trajectory design for CAVs and implements cooperative and non-cooperative behaviours of HDVs in mixed traffic. The study also reviews the existing literature on cooperative merging models and trajectory design for CAVs and efficiently addresses the limitations of the current approaches. The CDMMT mechanism addresses the gaps in existing research by demonstrating smoother and more efficient ramp merging in mixed traffic environments through micro-simulations.

2.2.9 MARL Algorithms

In this research, we will use the multi-agent versions of the ACKTR, PPO, and DQN algorithms extended in the research by Chen et al. [2022], available at Chen [2023], to investigate the scalability of the above-mentioned algorithms to a multi-lane on-ramp merging scenario.

MAACKTR (Multi-Agent Actor-Critic using Kronecker-Factored Trust Region)

The Multi-Agent Actor-Critic using Kronecker-Factored Trust Region (MAACKTR) algorithm, an extension of the Actor-Critic using Kronecker-factored Trust Region (ACKTR) algorithm, is a significant development in the field of autonomous driving. ACKTR was developed at the University of Toronto and New York University by combining actor-critic methods, trust region optimization, and distributed Kronecker factorization OpenAI [2021]. ACKTR is an actor-critic method that learns the optimal policies by using Kronecker-factored approximation to optimize the natural gradient Wu et al. [2017].

The version of the MAACKTR algorithm used in this research is a Multi-Agent Reinforcement Learning (MARL) framework, which is an extension of the single-agent variant ACKTR. This was extended in the study Chen et al. [2022] to address the challenges of highway on-ramp merging of CAVs in mixed traffic scenarios Chen et al. [2022]. MAACKTR extends the ACKTR approach to a multi-agent setting by sharing the parameters and allowing the agents to learn collectively Chen et al. [2022].

MAPPO (Multi-Agent Proximal Policy Optimization)

The Multi-Agent Proximal Policy Optimization (MAPPO) framework is the extension of the single-agent Proximal Policy Optimization (PPO) framework to a multi-agent scenario Zabounidis et al. [2023]. MAPPO has been successfully used in various multi-agent settings to train the agents and achieves state-of-the-art performance in various cooperative multi-agent tasks Liang et al. [2023]Parada et al. [2022].

The multi-agent version of the PPO algorithm, MAPPO, used in this research is the extension of the single-agent variant PPO. This was extended in the study Chen et al. [2022] to address the challenges of highway on-ramp merging of CAVs in mixed traffic scenarios Chen et al. [2022]. By sharing observations and rewards, MAPPO leads to efficient navigation in mixed traffic scenarios.

MADQN (Multi-Agent Deep Q-Network)

The Multi-Agent Deep Q-Network (MADQN) algorithm is a significant advancement in the field of Multi-Agent Reinforcement Learning. It extends the single-agent Deep Q-Network (DQN) algorithm to multi-agent settings. MADQN agents update their Q-values by observing the states, exchanging knowledge, and performing actions Ibrahim et al. [2021]. The MADQN algorithm has been applied in various multi-agent scenarios and is proven to show efficient results.

The Multi-Agent DQN (MADQN) algorithm used in this research is the extension of the single-agent variant DQN. This was extended in the study Chen et al. [2022] to address the challenges of highway on-ramp merging of CAVs in mixed traffic scenarios Chen et al. [2022]. By encouraging cooperative behaviours, MADQN can enable multiple agents to learn and update their policies simultaneously.

2.3 Analysis

Analysis of the related work section highlights that most of the work done in the domain of highway on-ramp merging of CAVs is done on a single-lane on-ramp environment. This leaves a huge gap in multi-lane on-ramp scenarios, which is often overlooked in previous research.

So, I decided to explore this area by focusing on the study Chen et al. [2022], which provides a baseline for the comparison of the performance of the three different MARL frameworks (MADQN, MAPPO, and MAACKTR) developed for single-lane on-ramp environments, when extended to a multi-lane on-ramp scenario.

Chapter 3

Methodology

This chapter briefly discusses the research question and my approach to addressing it (Section 3.1), it explains the reason for choosing highway-env as the simulation environment (Section 3.2), it then discusses the architecture of highway-env (Section 3.3), then it delves into details of implementation of the modified simulation environment consisting of two merging lanes on the on-ramp and the challenges faced along the way (Section 3.4), and it ends with discussing the evaluation parameters (Section 3.5).

3.1 Introduction

To address the research question of whether existing Multi-Agent Reinforcement Learning (MARL) algorithms developed for single-lane on-ramp merging of Connected and Autonomous Vehicles (CAVs) in mixed traffic scenarios be scaled effectively to multi-lane merging scenarios, lane changing and lane merging in CAVs must be framed as a Multi-Agent Reinforcement Learning problem because, in any given scenario, we will have multiple CAVs controlled by MARL algorithms. Answering this research question would require testing the performance of the MARL algorithms developed for single-lane on-ramp merging in a multi-lane on-ramp merging scenario.

The existing simulation environments were only designed for and limited to a single-lane on-ramp. This necessitated the need to modify the existing simulation setup. So, the single-lane on-ramp simulation environment has been extended to include an additional lane on the on-ramp to simulate a multi-lane on-ramp merging environment. For the simulation environment, I decided to use highway-env (Section 2.1.5).

For the choice of the MARL algorithms developed for single-lane on-ramp scenarios, I chose MAPPO (Section 2.2.9), MADQN (Section 2.2.9), and MAACKTR (Section 2.2.9) developed in the study Chen et al. [2022]. These frameworks were then used to train

the agents in the extended multi-lane on-ramp environment to assess their scalability and performance.

After training the agents using the three MARL frameworks, I evaluated and compared the results across the single-lane and multi-lane on-ramp scenarios.

3.2 Choice of Simulation Environment

Reinforcement Learning (RL) and Multi-agent Reinforcement Learning (MARL) algorithms modelled for scenarios like lane changing and on-ramp merging of Connected Autonomous Vehicles (CAVs) in mixed traffic scenarios are mostly tested on traffic simulators as it is unsafe and expensive to perform experiments in real-world scenarios. For example, even running a single real-world experiment involving CAVs would take a lot of time as the agent must train against various scenarios. Until the agent is well trained, it will make a lot of random moves to explore its surrounding environment, which can lead to many accidents. These problems can be overcome by using open-source traffic simulators as they are very inexpensive to set up and are a much safer way to enable the agent to explore various scenarios and train based on the exploration.

There are various traffic simulators that are designed to replicate real-world driving scenarios. Some popular ones include Simulation Urban Mobility (SUMO) and highway-env.

SUMO (Simulation of Urban MObility) Lopez et al. [2018] is an open-source library designed to handle simulations of large road networks. SUMO is a very powerful tool for traffic simulations; however, it does have a few limitations. SUMO depends on other packages, such as the traffic control interface (TraCI) package and the Flow package, to run simulations. Another major downside of the SUMO library is that it is very computationally expensive, and even a single simulation takes hours to run. Considering the above drawbacks of SUMO, I decided to explore other simulation environments.

Highway-env Leurent [2018a] follows a minimalistic style and Pythonic implementation to simulate various traffic simulation scenarios. Some of the different environment offerings available within highway-env are highway, merge, parking, roundabout, etc. The original implementation of highway-env (2018) does not support multiple autonomous and connected vehicles in hybrid Multi-Agent Reinforcement Learning settings. However, it is possible to extend the support of this library to include multiple CAVs. Further, it is also relatively simple to customise this library to suit the requirements of our simulation environment.

Inspired by the simplistic implementation and various offerings of highway-env, I choose to use highway-env by Eduard Leurent as the simulation environment for this

research. Since this paper aims to explore the scalability of existing Multi-Agent Reinforcement Learning (MARL) approaches to multi-lane on-ramp merging scenarios, using the merge environment from highway-env was a viable starting point. To further simply the usability of this library, the study "Deep Multi-agent Reinforcement Learning for Highway On-Ramp Merging in Mixed Traffic" Chen et al. [2022] modified the original highway-env merge environment to include the support of multiple CAVs in a mixed traffic scenario. Inspired by this approach, I modified this environment to include an additional merge lane to the on-ramp.

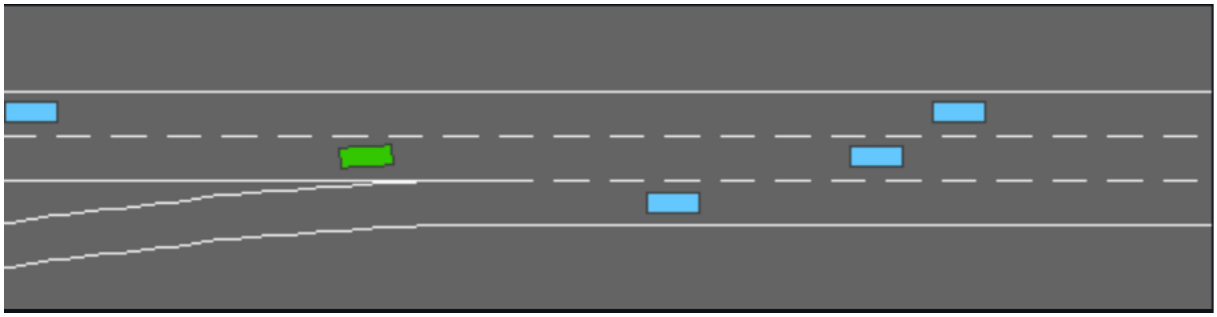

Figure 3.1: Original Merge environment from highway-env. Blue vehicles are the HDVs and the green vehicle is the autonomous vehicle.

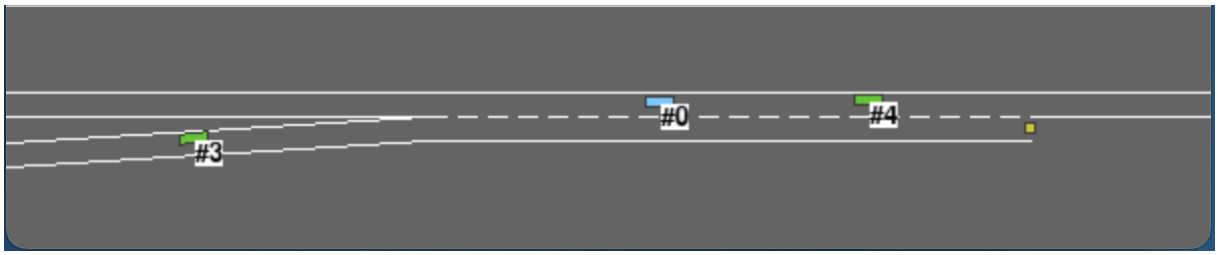

Figure 3.2: Merge environment modified by Dong Chen Chen et al. [2022]. Green vehicles are the HDVs and the blue vehicle is the autonomous vehicle.

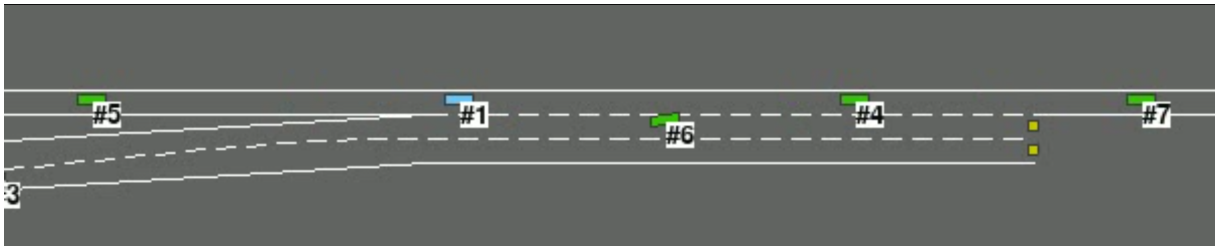

Figure 3.3: Modified merge environment with additional merge lane. Green vehicles are the HDVs and the blue vehicle is the autonomous vehicle.

Implementing these changes shown in "Figure 3.3" was not straightforward, and I was faced with various challenges along the way. These changes are discussed in detail below.

3.3 Highway-env Architecture

When it comes to the architecture of highway-env, for implementing or modifying any environment, the RoadNetwork and the Vehicle are the two of the most important class objects that need to be modified. Their modification is essential because any environment in highway-env essentially simulates various vehicles on different roads.

The RoadNetwork class is implemented using Lanes and Obstacles. To add an additional lane to the environment, we first have to initialise a lane and then add it to the road network. In RoadNetwork, a single highway lane is defined as a combination of smaller lanes. For example, as shown in Figure 3.4 , the lane "ad" is defined as a combination of 3 smaller lanes: "ab", "bc", and "cd". Here, "a" is the starting point of the lane, and "d" is the ending point of the lane. Each lane can be modelled into different types, such as a straight line or a sinelane (curved lane). Further, we can also define if the lane is continuous or stripped (a continuous lane does not allow for a lane change, whereas a striped lane does allow for a lane change).

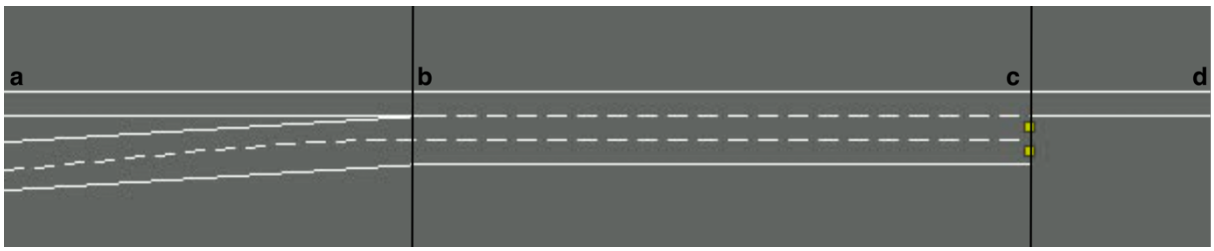

Figure 3.4: Lane "ad" that is split into 3 smaller roads "ab", "bc", and "cd".

To modify the vehicles simulated on the roads, the MDPVehicle and IDMVehicle classes that extend the ControlledVehicle class need to be modified. The MDPVehicle class defines the controls for the Connected Autonomous Vehicles (CAVs) in the simulated environment. The IDMVehicle class defines the behaviours of Human-driven vehicles (HDVs) in the environment. Only the vehicles defined by the MDPVehicle class (CAVs) will be trained using the Multi-Agent Reinforcement Learning (MARL) algorithms. The MARL algorithms do not train the vehicles defined by the IDMVehicles class (HDVs). However, the internal heuristics logic defined in the highway-env code allows the HDVs to drive without collision and demonstrate the natural behaviours of HDVs.

The environment class defines all the lanes and adds them to the road network. This class will also initialise and spawn the vehicles randomly on the road at different positions. The modified environment "multi_merge_env_v0" consists of one highway lane and two merge lanes populated with 1-6 CAVs and 1-5 HDVs, depending on the density of the simulation. Traffic density "1" spawns 1-3 CAVs and 1-3 HDVs, Traffic density "2" spawns 2-4 CAVs and 2-4 HDVs, and Traffic density "3" spawns 4-6 CAVs and 3-5 HDVs. Due to

time and hardware constraints, traffic density "2" has been chosen to run the simulations. It is an excellent middle ground with ample vehicles spawned in the environment to explore various scenarios. Further, for the purpose of easier representation (Figure 3.5), let us call the original merge lane on the on-ramp "merge lane 1" and the newly added merge lane on the on-ramp "merge lane 2".

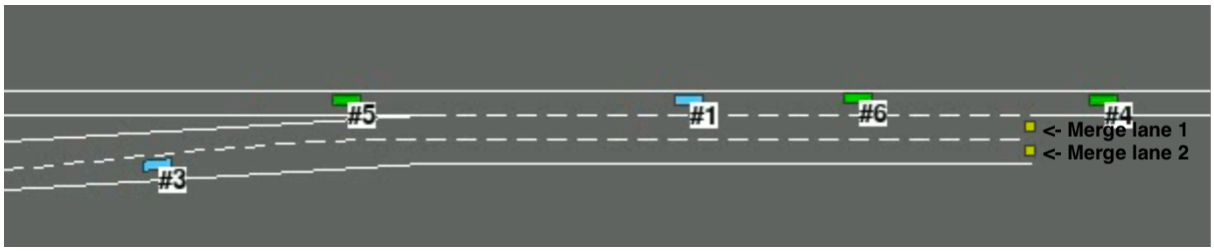

Figure 3.5: Modified highway-env merge environment. Blue vehicles are the CAVs and the green vehicles are the HDVs

3.4 Implementation

3.4.1 Adding Second Merge Lane

The original environment modified consisted of one highway lane and one on-ramp merging lane. To suit our requirements for this research, I modified and extended the existing environment to include an additional on-ramp lane that merges into the highway. As mentioned above, each lane is defined by multiple smaller lanes. So, to add the additional merge lane, I initialised and added three smaller lanes to the road network that are parallel to merge lane 1 (original on-ramp lane). Finally, I added an obstacle at the end of the newly added merge lane (merge lane 2) to indicate the end of the road.

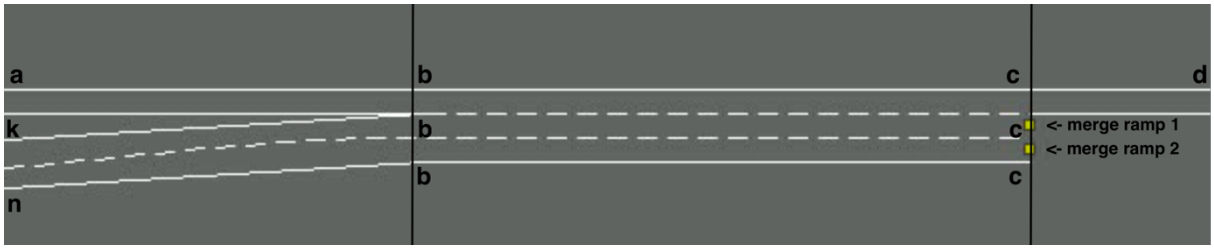

Figure 3.6: Road network of the modified environment

In the environment class, the _make_road method is used to design and incorporate lanes into the road network. The original environment consisted of only two lanes in the road network: a highway lane "ad" split into "ab", "bc", and "cd" and a merge lane "jc" (merge lane 1) split into "jk", "kb", and "bc". It is worth noting that the small lane

"jk" is a straight lane that comes before the point "k"; the small lane "kb" is a sinelane (curved lane); and "bc" is a straight lane. To this initial setup, I added another merge lane (merge lane 2), which is parallel to merge lane 1 and is defined by "mn", "nb", and "bc". Similar to lane "jk" in merge lane 1, "mn" is a straight lane before the point "n" in merge lane 2, which is parallel to "jk".

As we can observe from the "Figure 3.6", each lane in the environment is defined by multiple smaller lanes. The additional merge lane (merge lane 2) added to the environment defined by "mc" is divided into "mn", "nb", and "bc", where point "m" is the start of the road and point "c" is the end of the road. The road "mc" consists of multiple lanes, such as straight lanes and sinelanes (curved lanes). In the second merge lane, lanes "mn" and "bc" are defined as straight lanes that are parallel to lanes "jk" and "bc", and the lane "nb" is a sinelane that is parallel to "kb". All these smaller lanes are initialised separately and added to the road network. To initialise each of these lanes, we have to define a start_position, ending_position, line_type (assigns the lane as a continuous or a stripped lane), and forbidden (a Boolean value that allows or blocks vehicles from changing to that lane). Following the above steps, I have created an additional merge lane (merge lane 2) and added that to the road network for the vehicles to use. Next, using the Obstacle class, I added an obstacle at the end of the second merge lane to indicate the end of the road for the vehicles to follow. In the "Figure 3.6", yellow boxes placed at the end of the merge lanes are the obstacles that define the end of the road.

A critical aspect of lane implementation is the use of the argument "forbidden" while setting up the lane. When "forbidden" is set to True, vehicles are prohibited from changing their current lane and shifting into this lane. In the modified environment, I set the "forbidden" value of merge lane 1 to False, allowing vehicles to shift from merge lane 2 to merge lane 1. Conversely, the "forbidden" value is set to True for the second merge lane, preventing vehicles from moving from merge lane 1 to merge lane 2.

3.4.2 Spawning Vehicles on the Second Merge Lane

The code by Dong Chen Chen et al. [2022] Chen [2023] allows vehicles to be spawned on the highway lane and first merge lane. However, the addition of an additional merge lane (merge lane 2) required a change in the logic to spawn vehicles on the newly added second merge lane. In the existing code, $\frac{1}{2}$ of the vehicles were spawned on the highway lane and the other half on the merge lane. In the modified environment, I spawned $\frac{1}{2}$ of the vehicles on the highway lane; the other half of the vehicles were randomly spawned between the two merge lanes. The vehicles on the second merge lane will change lanes into the first merge lane before eventually merging into the highway lane.

_make_vehicle method is used to spawn the vehicles on the different lanes. As the existing code only accounted for vehicles being spawned on the highway and the first merge lane, the vehicle distribution logic had to be changed to spawn the vehicles onto the second merging lane. The existing code spawns the vehicles on the roads in four parts using a predefined list of spawn points containing the positions on the road where the vehicles must be spawned. First, it spawns the CAVs on the roads, and then it spawns the HDVs. In the existing code, $\frac{1}{2}$ of both CAVs and HDVs are spawned on the highway lane, and the other $\frac{1}{2}$ are spawned on the merge lane. To account for an additional merge lane, I decided to leave $\frac{1}{2}$ vehicles on the highway lane and only focus on modifying the code to split the other half of the vehicles between the two merge lanes. So, I modified this code to spawn vehicles on either of the two merging lanes randomly.

Spawning vehicles on the roads requires various arguments, such as the type of the vehicle (CAV or HDV), the name of the lane in the road network where the vehicle needs to be spawned, the random position on the road calculated using the spawn points, and the initial speed of the vehicle. The initial speeds of the vehicles are randomly generated and essential to simulating the real-world behaviours of various vehicles.

Even after modifying the "forbidden" argument of the first merge lane, I faced an issue while spawning vehicles on the second merge lane. The HDVs spawned on the second merge lane were not changing lanes into the first merge lane; they were going to the end of the second merge lane and stopping. This issue was only isolated to HDVs, as the CAVs were behaving normally. Upon troubleshooting, I realised that the problem is caused by the MOBIL (Minimising Overall Braking Induced by Lane change) function of the IDMVehicle class (a class that controls the behaviours of HDVs). In the MOBIL function, initially, the jerk (assess the change in acceleration) computed was always assigned to zero; this was the root cause of the issue. So, I added a statement to increment jerk by 0.11 if the acceleration of the HDV vehicle before and after the lane change (relative to the preceding vehicles) would be the same. This ensures that the jerk is not zero and fixes the issue. This change ensures that the HDVs are behaving normally and changing lanes.

The reward function calculates the rewards of each episode based on various factors like collisions, overall throughput, and speeds of the vehicles on the merging and the highway lanes. The individual rewards obtained by all the vehicles in an episode are cumulated to calculate the reward of each episode. The general idea is that vehicles that do not cause collisions and merge into the highway quickly and safely, maintaining a high speed, get higher rewards. This rewards function has also been modified to use the same logic but to include and consider vehicles on the second merge lane.

3.4.3 Forbidding Lane Changing of Vehicles from Left to Right Lanes

While modifying the road network to add the second merge lane, I have changed the "forbidden" argument of the first merge lane to False to allow vehicles from merge lane 2 to change to merge lane 1 before merging into the highway. However, this change caused an issue where vehicles from the highway lane were changing into the first merge lane to explore different actions. This would be a scenario that can lead to collisions and should not happen in real-world scenarios. So, I have modified the logic to enable lane changes only from the right lanes to the left lanes but not the other way around. I tried to address this problem in a few different ways, but this was the best possible solution.

Since the first merge lane's "forbidden" was set to False, the vehicles from the highway lane are changing to the first merge lane to explore different scenarios. I have tried to fix this problem using the RoadNetwork class, but there is no possible way to solve this using the RoadNetwork. So, I fixed this issue using the ControlledVehicle class. The environment consists of two types of vehicles: CAVs and HDVs. The logic for the HDVs is implemented using the IDMVehicle class, and the logic for the CAVs is handled by the MDPVehicle class. Both of these classes are extended from the ControlledVehicle class. So, I have changed the logic in this class to allow for lane changes only from the right to the left lanes. I have added a condition for action "LANE_RIGHT" and set the enable_lane_change to False. This change ensures that no vehicles change lanes to the right lane.

3.5 Evaluation Set-up

The modified highway-env merge environment, multi_merge_env_v0, implements a new environment setup (Figure 3.3) that introduces an extra lane to the on-ramp and strategically positions the vehicles on the road network. This setup simulates the highway on-ramp merging of CAVs in mixed traffic scenarios, particularly in the presence of a multi-lane on-ramp. Each simulation in multi-agent scenarios termed an episode, is a single sequence of states, actions, and rewards that the agents experience from the start of an environment until it reaches a terminal state. In our context, an episode starts with the spawning of the agents (CAVs) at the start of the road and ends when the agents (CAVs) either reach the end of the road, cause a collision or when the time limit expires.

In Reinforcement Learning scenarios, agents are trained for a specific number of episodes, allowing them to explore and learn from their environment. A cumulative reward, the average of all agents' rewards in the environment, is calculated for each episode.

In an ideal scenario where the agents learn from every training episode, the cumulative reward should increase as the training episodes progress.

In the study Chen et al. [2022], the authors evaluated the performance of various MARL algorithms over 20,000 episodes at 3 different traffic densities, adjusting the number of CAVs and HDVs varies in the environment (mentioned in section 3.3). Higher traffic densities, which have an increasing number of CAVs, would make it challenging to learn optimal strategies.

Due to the limitations of my current hardware, it takes approximately 33 hours to run 20,000 episodes for each algorithm. So, considering hardware and time constraints, I used the following settings to run the experiments.

Table 3.1: Parameters used for the Evaluation

Parameter	**Value**
Number of Training episodes	10,000
Number of Evaluation episodes	3
Evaluation Interval	20
Traffic density	2
Number of CAVs	2-4
Number of HDVs	2-4

Chapter 4

Evaluation

This chapter discusses the evaluation metrics (Section 4.1), the design of the experiments for evaluating the scalability of different MARL algorithms like MAACKTR, MAPPO and MADQN to the modified multi-lane highway on-ramp merging environment (Section 4.2), the experiment parameters used (Section 4.3), discussion on the results (Section 4.4), comparison of the various results (Section 4.5), and finally a discussion on the experiments (Section 4.6).

4.1 Evaluation Metrics

In this section, we will discuss the metrics used to assess the scalability of the MARL algorithms, such as MAACKTR, MAPPO, and MADQN, to multi-lane merging scenarios. The critical evaluation metric for evaluating the performance of the different MARL algorithms is the rewards obtained by the episodes. Each agent in a multi-agent scenario is rewarded based on its actions in the environment. A positive reward is assigned if an agent (CAV) follows actions that promote safe and efficient merging. Otherwise, a negative reward is assigned if the agent causes collisions or drives unnaturally. The highest reward is assigned if the agents follow the most optimal policy. An average of the rewards the agents earn in an episode is designated as the reward for that particular episode. Ideally, as the number of training episodes increases, the agents must learn to follow the optimal policy, increasing the rewards obtained.

For the baseline for our comparisons, I used the unmodified environment by Chen described in the study Chen et al. [2022] to generate the baseline results. I have not used the graphs mentioned in the study Chen et al. [2022] directly because they trained the model for 20,000 episodes. Still, I could only train the algorithms in modified environments for 10,000 episodes. The above-mentioned MARL algorithms have been trained for 10,000

using the same evaluation settings on the unmodified environment due to time constraints and to maintain consistency in the baseline results. I have used the results of Chen et al. [2022] as the baseline because the unmodified environment simulates the highway on-ramp merging of CAVs in mixed traffic scenarios in the presence of a single lane on-ramp. This is a good baseline as we are exploring the existing algorithms' scalability to a multi-lane on-ramp merging scenario.

For the evaluation, I have trained the MAPPO, MAACKTR, and MADQN algorithms for 10,000 episodes on the modified merge environment. Every 20 episodes, I ran 3 evaluation episodes and used the average rewards obtained from these three episodes as the evaluation metric. Ideally, the value of these rewards should increase as the training episodes increase, indicating that the agents are learning the optimal solution based on the algorithm.

4.2 Evaluation Design

Our primary objective is to devise more effective experiments for assessing the performance of the MARL algorithms: MAPPO, MAACKTR, and MADQN in a modified environment. This environment is crucial as it presents unique challenges and scenarios that are not encountered in the standard environment. The evaluation of these MARL algorithms' performance is conducted in two parts. The first part focuses on the scalability of these algorithms in multi-lane merging scenarios, comparing the average results against the baseline case. The second part involves a comprehensive performance evaluation of these algorithms against each other to determine the top performers.

One common challenge in evaluating RL/ MARL algorithms is their sensitivity to the random seeds used for environment initialization. Additional detailed information on this issue can be found in the research paper: Colas et al. [2018]. To address this, I have meticulously evaluated the performance of the MARL algorithms across a wide range of random test seeds. The average reward obtained over 3 evaluation episodes was plotted as the performance score. The random seed values used for environment initialization were: 0, 25, 50, 75, 100, 125, 150, 175, 200, 325, 350, 375, 400, 425, 450, 475, 500, 525, 550, 575. This comprehensive approach ensures the validity and reliability of the performance assessment.

This experiment design helps us better understand the scalability of the existing MARL algorithms by comparing the results of the modified environment to the original environment. It also allows us to understand which of the three algorithms tested is performing better by comparing the results of the algorithms among themselves.

Based on the original experiment's results for 10,000 training episodes, the general

expectation for the experiments is that MAPPO would be the best-performing algorithm, followed by MAACKTR and then MADQN.

4.3 Experiment Settings

All three Multi-Agent Reinforcement Learning algorithms are trained on the same environment parameters to maintain consistency in the comparisons.

Table 4.1: Environment parameters used for the experiment

Parameter	**Value**
Number of Training episodes	10,000
Number of Evaluation episodes	3
Evaluation Interval	20
Traffic density	2
Number of CAVs	2-4
Number of HDVs	2-4

Number of evaluation episodes defines the number of different random seeds used in the evaluation, the evaluation interval is the gap between evaluating the trained agents, and traffic density determines the total number of CAVs and HDVs spawned in each episode.

4.4 Results

The Y-axis represents the average evaluation reward received by the agents, which is the mean reward obtained from 3 consecutive evaluation episodes. The X-axis represents the evaluation episode intervals. Episodes are evaluated at regular intervals of every 20 episodes.

The dashed blue line indicates the average evaluation reward obtained by the agents. An overall upward trend in this line suggests that the agents are learning and improving their optimal policy. This line is the primary indicator of the agent's performance in the environment. The shaded area around the average rewards line is the standard deviation of rewards across multiple random seeds. Wider the shaded area suggests that the algorithm's performance is highly sensitive to the initial random seed. The shaded area represents the volatility in the agents' performance based on the random seeds.

In the reward graphs for Reinforcement Learning, the rewards often fluctuate and dip through the training episodes. This behaviour is typical as the Reinforcement Learning agents explore their environment in a trial-and-error process, and sometimes, in the process of exploring the environment, they take actions that result in lower rewards.

4.4.1 Unmodified environment results

MAACKTR

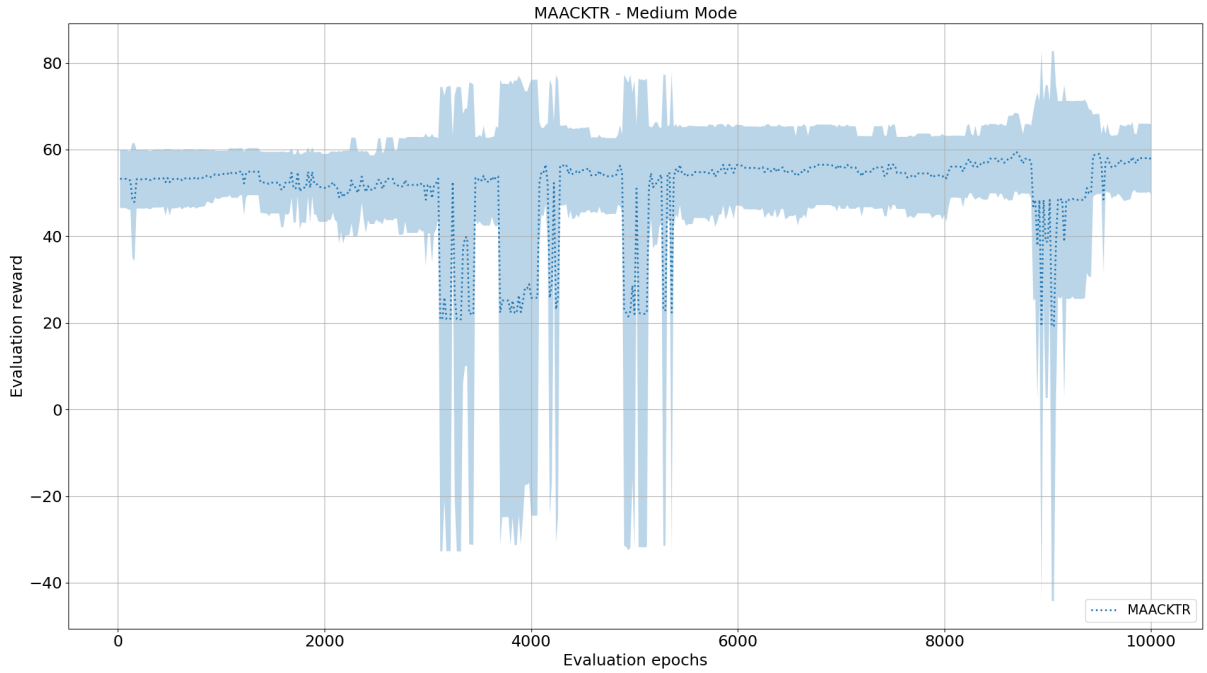

Figure 4.1: MAACKTR Rewards Graph on the unmodified environment

The following figure 4.1 represents the outcomes of training the agents over 10,000 episodes, utilizing the MAACKTR algorithm in medium mode, a crucial component of our training process.

From the figure 4.1, we can observe that the agents' rewards generally hoover around 40 to 60. This does not indicate high performance but suggests consistent learning from the agents. The fluctuations and occasional dips in the rewards indicate that the agents are exploring the environment. Another observation from the 4.1 is that the standard deviation of the rewards is relatively widespread. This suggests a high variability in the agents' performance across different random seed initializations. The analysis of the average rewards line reveals a modest overall positive slope, pointing to a gradual learning curve. This indicates that the agents are learning and improving their policies at a slow

learning rate. As the learning rate is slow, it would require a lot of training to achieve good results.

Overall, the MAACKTR algorithm exhibits high volatility and sensitivity to the initial seed values, which could pose challenges in environments where reproducibility is crucial. Despite this, in a traffic density two environment, the agents learning using the MAACKTR algorithm exhibit a positive learning rate, as evidenced in the figure. These results are used as the baseline for the performance of the MAACKTR algorithm.

MADQN

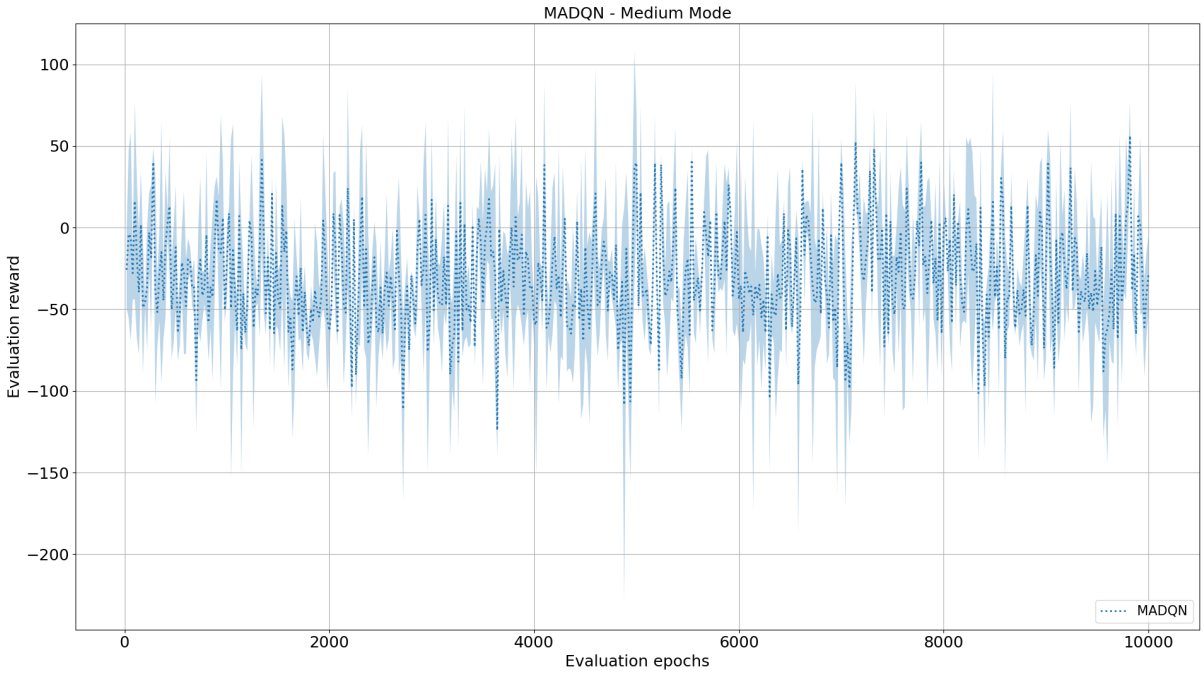

Figure 4.2: MADQN Rewards Graph on the unmodified environment

The following figure 4.2 represents the outcomes of training the agents over 10,000 episodes, utilizing the MADQN algorithm in medium mode, a crucial component of our training process.

From the figure 4.2, we can observe that the agents' rewards vary roughly between -100 and 50. The rewards dipping into negatives implies that the agents often take actions that result in penalties or losses, reflecting exploratory behaviour that involves testing various options. The sharp fluctuations observed for the evaluation rewards suggest that the agents encountered both high peaks and significant troughs in the performance, indicating an inconsistent model performance. Another observation from the figure 4.2 is that the MADQN algorithm's performance does not exhibit a clear upward trend, stating that there is no consistent improvement in learning. The standard deviation visualised by the

blue shaded area in the graph is consistently significant throughout the training episodes. This demonstrates that the MADQN algorithm is highly sensitive to initial random seed values.

Overall, the MADQN algorithm exhibits an unstable, erratic learning curve with high sensitivity to random seed values and no learning or policy improvement indication. This variability in the rewards obtained indicates that the MADQN algorithm is struggling to train the agents in a medium-density environment. These results are used as the baseline for the algorithm's performance.

MAPPO

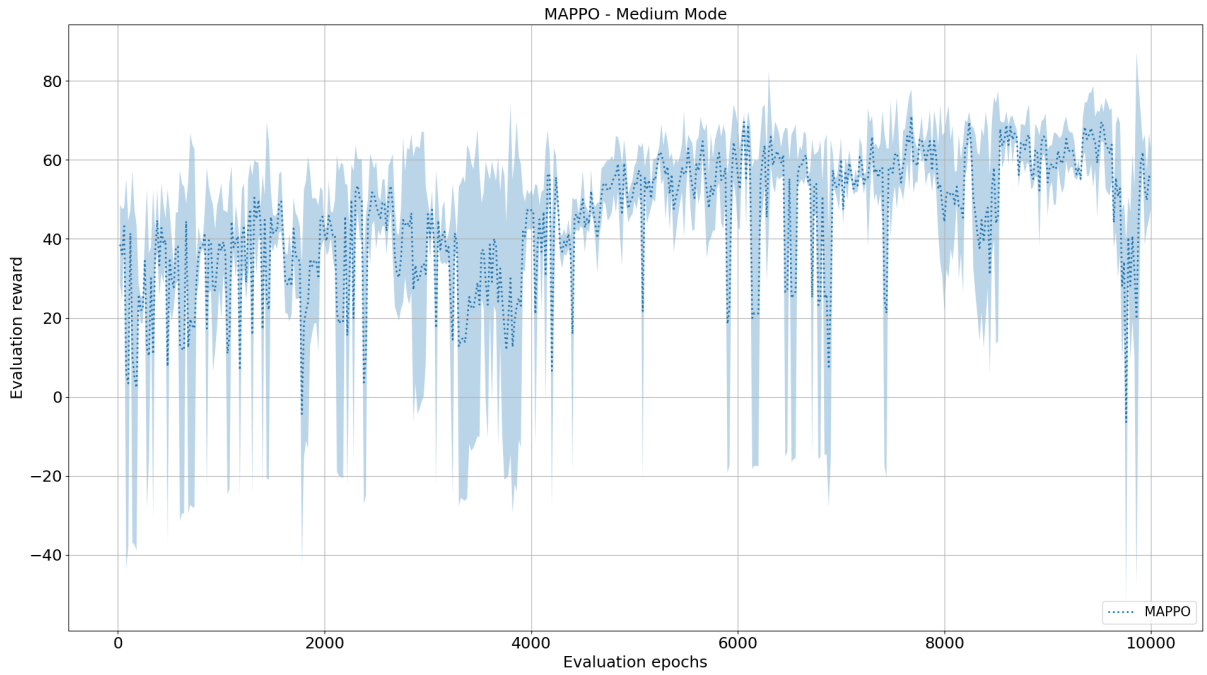

Figure 4.3: MAPPO Rewards Graph on the unmodified environment

The following figure 4.3 represents the outcomes of training the agents over 10,000 episodes, utilizing the MAPPO algorithm in medium mode, a crucial component of our training process.

From the 4.3, we can observe that the agents' rewards vary roughly between 0 and 70. In most cases, even while the algorithm explores the environment, the average rewards roughly vary in the range of 20 to 60, indicating that even exploratory actions do not result in significant losses. The average evaluation rewards oscillate notably but are less violent than the other two algorithms. This suggests that the agents are exploring various strategies but taking relatively better actions by learning from the previous episodes. Another

observation from the 4.3 is that although the MAPPO algorithm's performance fluctuates throughout the training episodes, it exhibits a clear upward learning trend, indicating that the agents are updating the optimal policy by learning from the previous episodes. Similar to the other algorithms, the standard deviation is considerable, suggesting that the MAPPO algorithm is sensitive to the initial random seed values chosen.

Overall, the MAPPO algorithm's learning curve shows a positive learning slope within a specific range, indicating that the agents efficiently learn based on the training episodes. MAPPO provides a good framework for the agents to learn and perform well in a medium-mode setting. These results are used as the baseline for the performance of the MAPPO algorithm.

4.4.2 Modified environment

MAACKTR

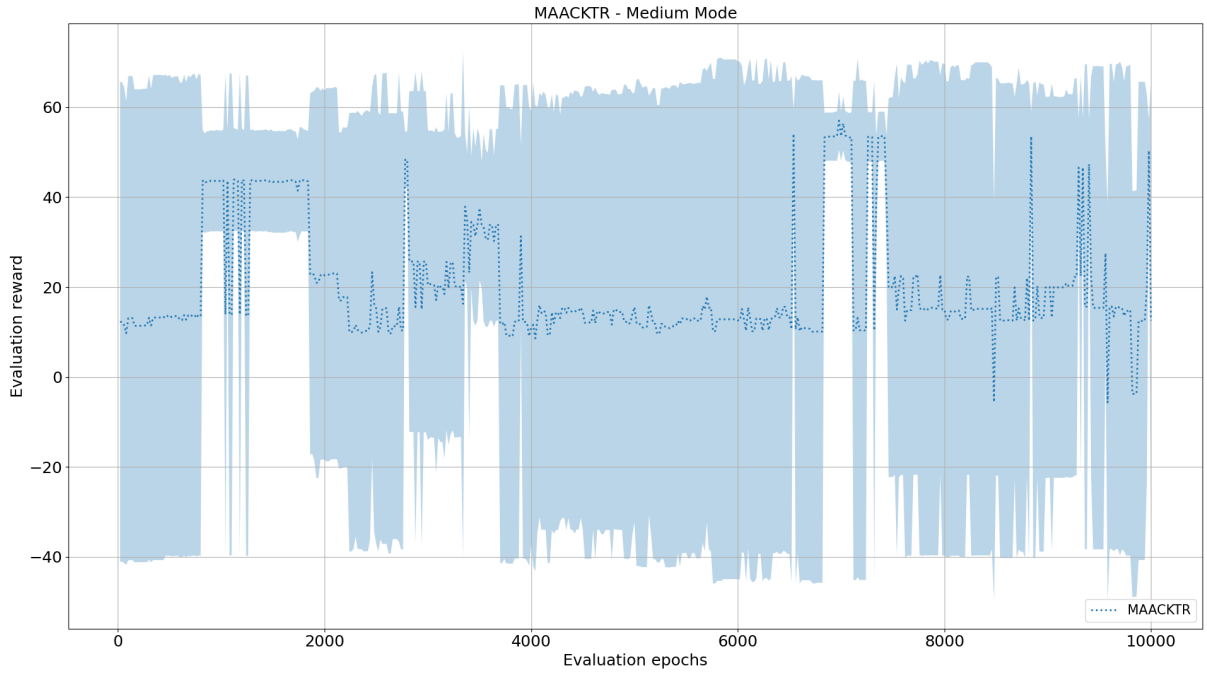

Figure 4.4: MAACKTR Rewards Graph on the modified environment

The following figure 4.4 represents the outcomes of training the agents over 10,000 episodes, utilizing the MAACKTR algorithm in medium mode, a crucial component of our training process.

The analysis of the average rewards from the 4.4 does not reveal any clear upward or a downward trend, indicating that the agents are neither learning nor declining based on the training episodes. The agents' rewards are generally varying between 10 and 20.

This shows that while the algorithm's performance is not great, it is consistent. The fluctuations in the evaluation rewards indicate that the rewards reach high peaks, but the agents are not learning from these episodes. Another observation from the 4.4 is that the standard deviation from the average rewards is quite significant. This indicates that the algorithm's performance heavily depends on the initial seed values.

Overall, the MAACKTR algorithm in the medium mode environment exhibits volatility and high sensitivity to the initial seed values. The agent's performance does not clearly indicate learning in this environment. The very high sensitivity to random seeds makes reproducibility challenging.

MADQN

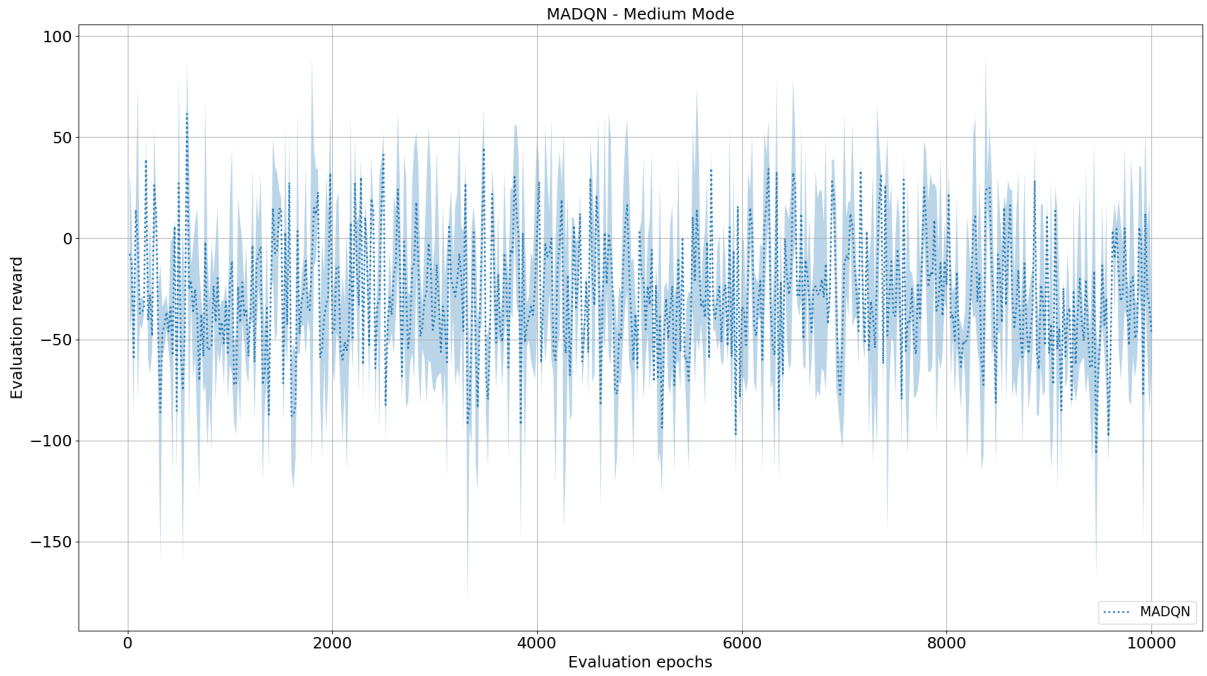

Figure 4.5: MADQN Rewards Graph on the modified environment

The following figure 4.5 represents the outcomes of training the agents over 10,000 episodes, utilizing the MADQN algorithm in medium mode, a crucial component of our training process.

We can observe that the MADQN's performance in the modified environment is very similar to that in the unmodified environment. From the 4.5, we can observe that the agents' rewards vary between -100 and 50, with the dip in the rewards implying that the agents often take exploratory actions that result in penalties or losses. The average rewards line going up and down randomly suggests that the agents encountered both high

peaks and significant troughs in the performance, indicating an unstable algorithm performance. Another observation from the 4.5 is that the MADQN algorithm's performance does not exhibit a clear upward or downward trend, stating that there is no consistent improvement in learning. This highlights that the model is in a cycle of trial-and-error exploration and needs to be learned. High variance in the rewards obtained is observed in the graph throughout the training episodes. This demonstrates that the MADQN algorithm is highly sensitive to initial random seed values.

Overall, the MADQN algorithm demonstrates an agent experiencing both high and low performance in quick successions. It exhibits an unstable, erratic learning curve with high sensitivity to random seed values and no indication of learning or policy improvement. This variability in the rewards obtained indicates that the MADQN algorithm struggles to train the agents in a medium-density environment.

MAPPO

Figure 4.6: MAPPO Rewards Graph on the modified environment

The following figure 4.6 represents the outcomes of training the agents over 10,000 episodes, utilizing the MAACKTR algorithm in medium mode, a crucial component of our training process.

From the figure 4.6, we can observe that the rewards are negative at the start of the training, and by the end of the 10,000 training episodes, the rewards are close to 75. This indicates a clear upward learning trend, indicating that the agents update

the optimal policy by learning from the previous episodes. In most cases, the average rewards are between 25 and 50, indicating that even exploratory actions do not result in significant losses. The occasional dips into negative rewards mean the agents explore new scenarios that cause negative rewards. However, this only happens a few episodes before the rewards reappear, indicating that the agents learn based on the training. The average evaluation rewards oscillate notably but are less violent than the other two algorithms. This suggests that the agents are exploring various strategies but taking relatively better actions by learning from the previous episodes. While some standard deviation is present, it is not as extreme, indicating that MAPPO is less sensitive to random seed values than other algorithms.

Overall, the MAPPO algorithm's learning curve shows a positive learning slope within a, indicating that the agents are efficiently learning based on the training episodes. It shows stability in its learning with comparatively fewer fluctuations. MAPPO provides a good framework for the agents to learn and perform well in medium-mode settings.

4.5 Comparisons

4.5.1 Scalability of the Algorithms

MADQN

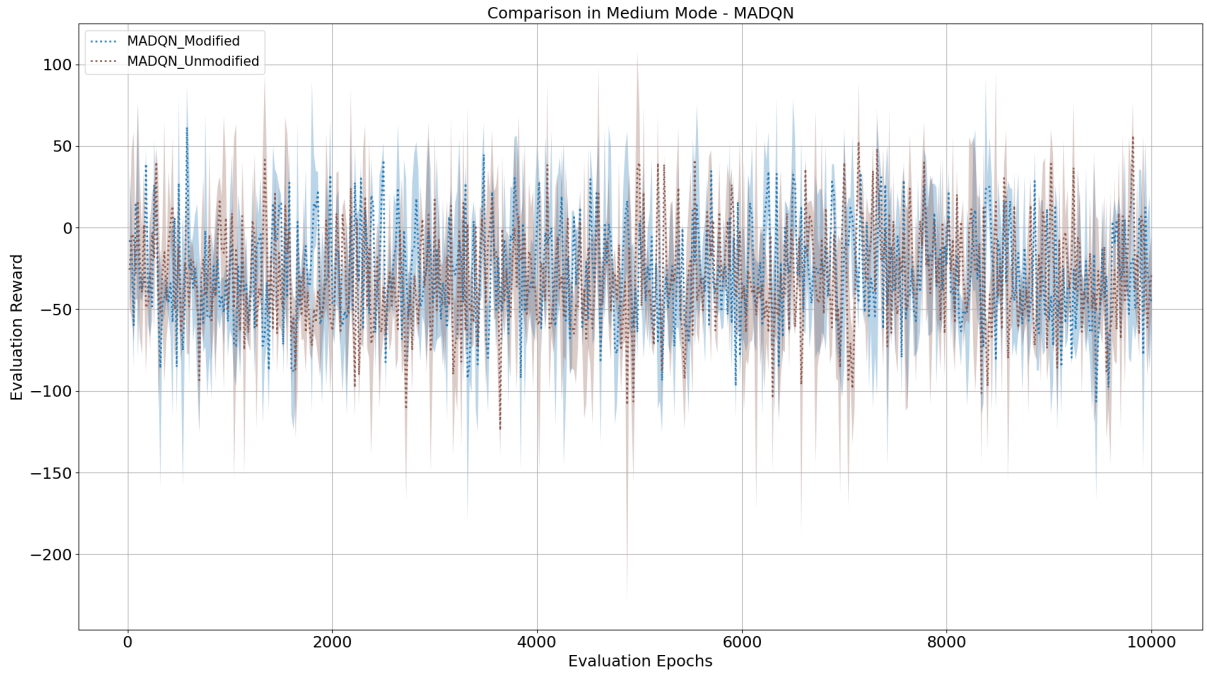

Figure 4.7: MADQN Rewards Graph comparing performance in the modified and the unmodified environment

The results in figure 4.7 are the results of training the agents for 10,000 episodes using the MADQN algorithm in medium mode on both the modified and unmodified environments.

From the figure 4.7 and the results from sections (section 4.4.1 and section 4.4.2), MADQN shows an upward trend in neither case, indicating that the agents are learning based on the training episodes. In both cases, the performance of the MADQN in terms of variability, stability, and consistency is identical. The MADQN algorithm in both cases follows a similar pattern. This indicates no degradation in performance when scaled to the new environment.

In summary, from the figure 4.7, the MADQN algorithm's performance in the modified environment is on par with its performance in the unmodified environment. This suggests that the MADQN algorithm is scalable to the new environment as there is no degradation in the algorithm's performance. However, no clear learning trend is visible in the average rewards; instead, they fluctuate heavily. This indicates that the agents are not learning

from the training episodes. So, although the MADQN algorithm is scalable to the new environment, there are better algorithms in this scenario of highway on-ramp merging of CAVs as the agents are not learning.

MAACKTR

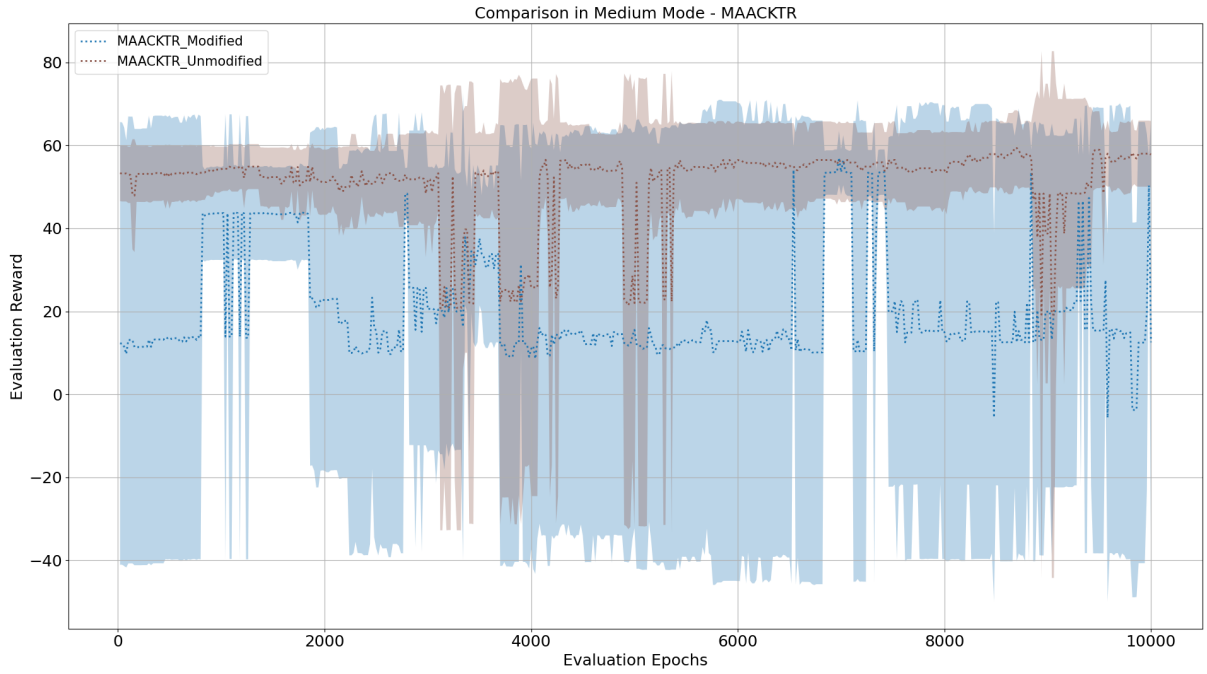

Figure 4.8: MAACKTR Rewards Graph comparing performance in the modified and the unmodified environment

The results in figure 4.8 are the results of training the agents for 10,000 episodes using the MAACKTR algorithm in medium mode on both the modified and unmodified environments.

From the results in sections(section 4.4.1 and section 4.4.2) and the figure 4.8, we can observe that the performance of the MAACKTR algorithm in the unmodified environment is much better than the performance of the MAACKTR algorithm in the modified environment. MAACKTR algorithm in the unmodified environment reaches higher reward values when compared to the modified environment. In the unmodified environment, the agents show a slow but positive learning trend, indicating that the agents are slowly learning based on the training episodes. However, in the modified environment, no such trend is shown. The standard deviation of the rewards is much higher in the modified environment compared to the unmodified environment, indicating that the MAACKTR algorithm is more sensitive to random seeds in the modified environment. Although both

scenarios display fluctuations, the fluctuations in the unmodified environment are much more tightly packed and indicate an upward trend in learning.

Considering these factors, in a medium traffic density setting, the MAACKTR algorithm performs much better in the unmodified environment than in the modified environment. Therefore, we can say that the MAACKTR algorithm is not scalable to the modified environment, as there is a considerable degradation in the algorithm's performance.

MAPPO

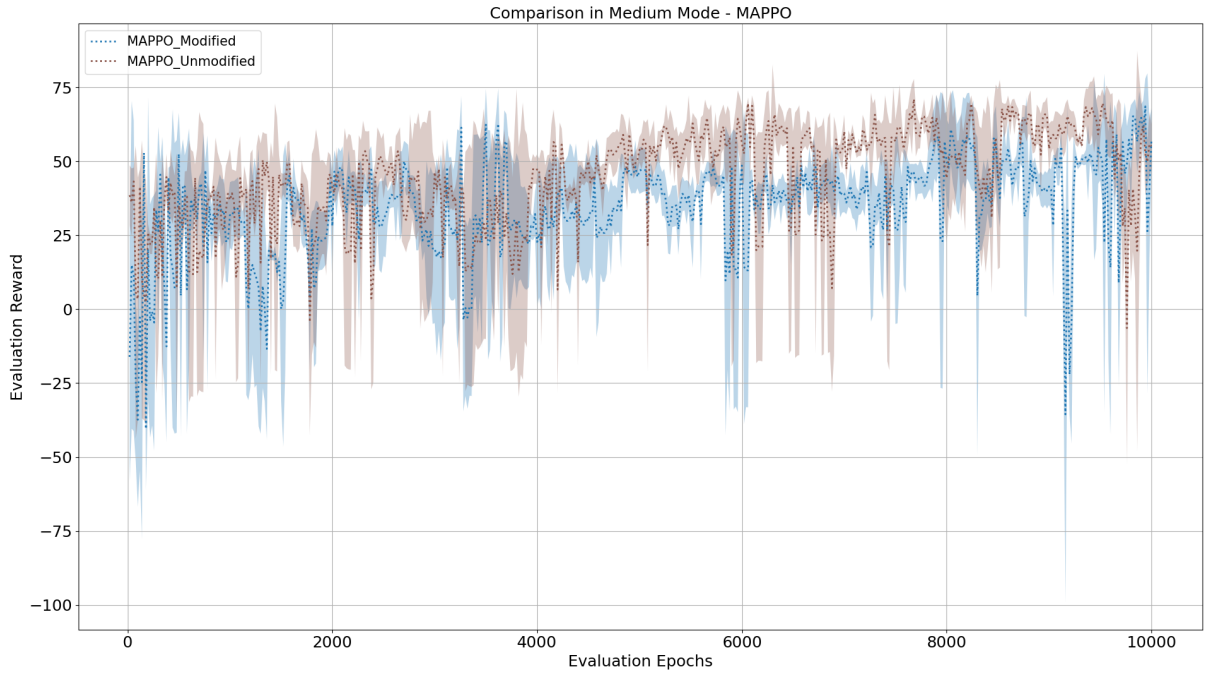

Figure 4.9: MAPPO Rewards Graph comparing performance in the modified and the unmodified environment

Figure 4.9 Are the results of training the agents for 10,000 episodes using the MAPPO algorithm in medium mode on both the modified and the unmodified environments.

Based on the figure 4.9 and the results from sections (section 4.4.1 and section 4.4.2), the MAPPO algorithm in both modified and unmodified environments shows an apparent positive learning curve where the agents learn from the previous training episodes. In both scenarios, the fluctuations in the rewards seem similar, and the standard deviation in the rewards shows a similar degree of variability. Looking at the average rewards line, MAPPO performs well in both environments, with a positive learning trend. MAPPO in the modified environment slightly outperforms the average rewards obtained and a somewhat tighter cluster of reward ranges. Still, overall, the performance of MAPPO in both environments is similar.

Overall, in a medium traffic density setting, the performance of the MAPPO algorithm is similar, if not better, to that of the modified environment compared to the unmodified environment. This indicates that the MAPPO algorithm is scalable to the modified environment, as there is no degradation in the algorithm's performance in the modified environment. The MAPPO algorithm shows good performance and a positive learning slope in both environments, making it a good choice for training the agents in both environments.

4.5.2 Which Algorithm Performs Better

Unmodified environment

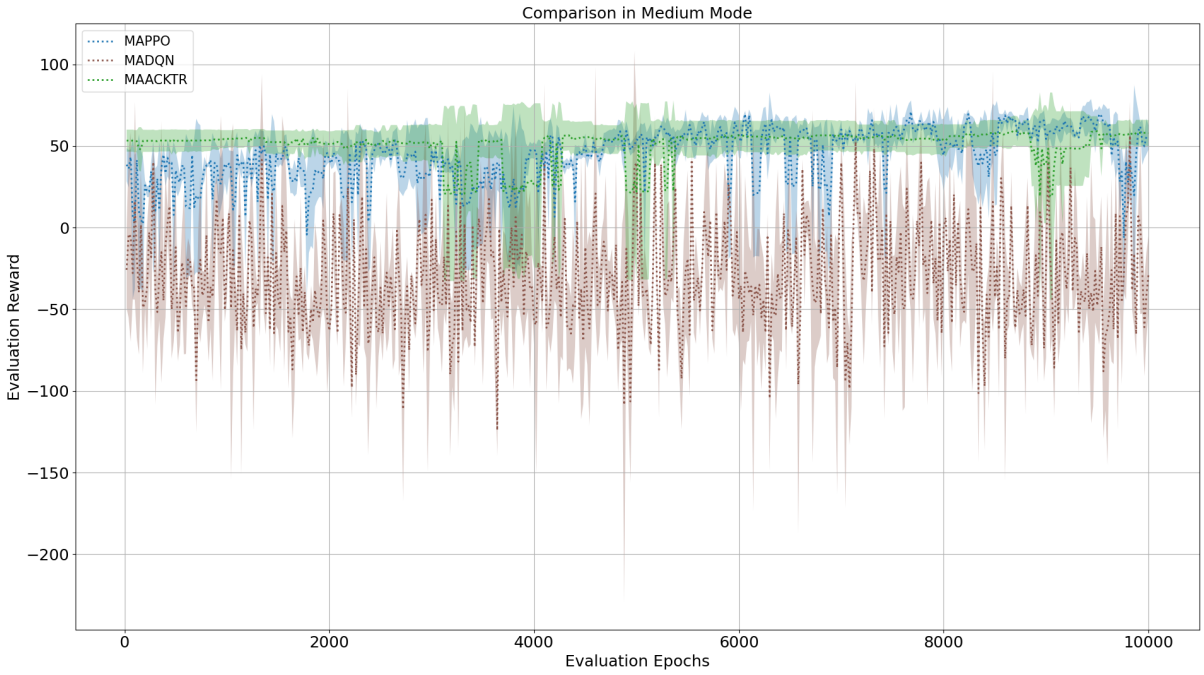

Figure 4.10: Comparison of the performance of different algorithms in the unmodified environment

Figure 4.10 are the results of training the agents for 10,000 episodes using different algorithms-MAPPO, MADQN, and MAACKTR-in medium mode on the unmodified environment.

From the figure 4.10, we can clearly observe that the performance of the MADQN algorithm is considerably worse and unstable than that of the other two algorithms. MAPPO and MAACKTR algorithms show similar performance, but the learning rate in the MAACKTR algorithm is slower than that of MAPPO. This is because the slope of the average rewards line for MAACKTR is less than the slope of the average rewards line

for MAPPO. All three algorithms show considerable fluctuations, but MADQN shows the highest fluctuations, making it highly inconsistent. The standard deviation is similar among all the three algorithms.

In summary, both MAACKTR and MAPPO perform well in the unmodified environment, with both algorithms showing a positive learning trend. However, MAPPO shows a slightly higher learning rate for the agents when compared with MAACKTR. MADQN performs significantly worse and needs a better algorithm to train the agents in the unmodified environment. So, we can conclude that the MAPPO and MAACKTR algorithms are good choices for training the agents in the unmodified environment.

Modified environment

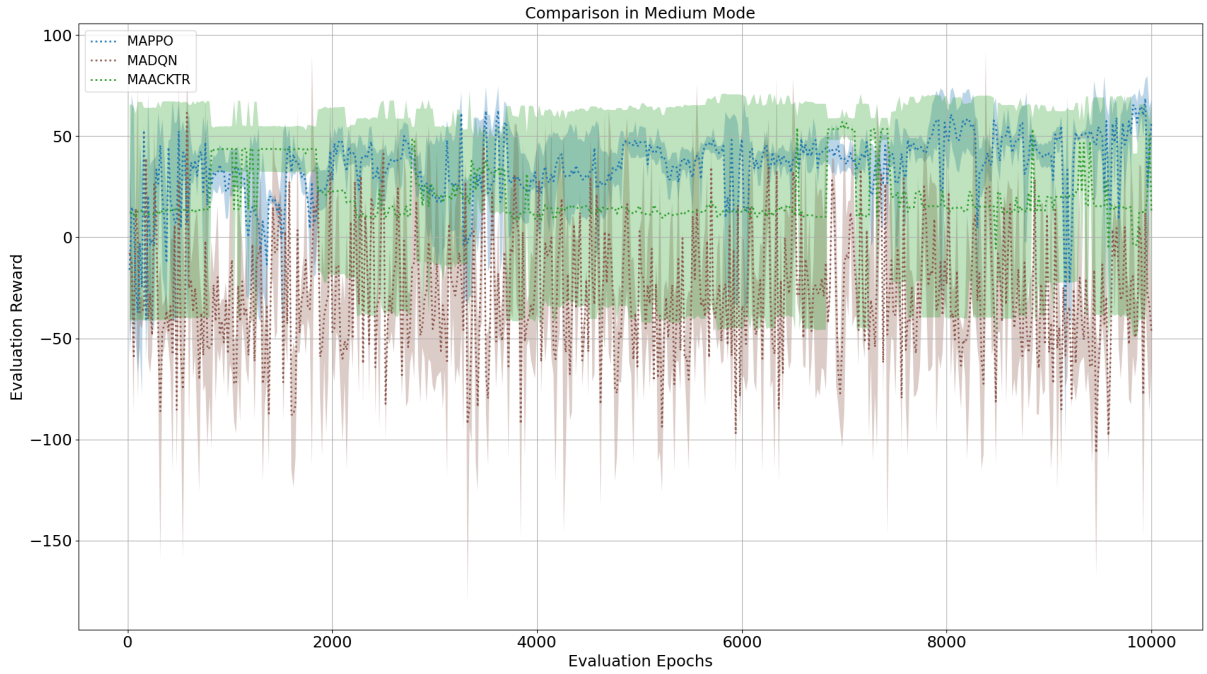

Figure 4.11: Comparison of the performance of different algorithms in the modified environment

Figure 4.11 are the results of training the agents for 10,000 episodes using different algorithms-MAPPO, MADQN, and MAACKTR-in medium mode on the modified environment.

From the figure 4.11, we can see that the MAPPO algorithm outperforms both MAACKTR and MADQN algorithms in the modified environment. MAPPO consistently achieves higher rewards and shows a positive learning trend compared to the other two algorithms. Also, the standard deviation of the average rewards is much lower for the MAPPO algorithm than for the other two algorithms. MAACKTR algorithm shows little to no learning

in the agents based on the training episodes, as there is no visible positive slope in the average rewards line. In addition, it shows a very high standard deviation of the average rewards, indicating that it is highly sensitive to random seed values. All three algorithms show notable fluctuations, but MADQN shows the highest fluctuations, making it highly inconsistent.

Overall, MAPPO is the best-performing algorithm in the modified environment as it consistently achieves higher rewards and shows a clear positive learning trend, indicating that the agents are learning from the previous episodes. MAACKTR shows no precise learning curve, meaning that the agents are not learning from the earlier episodes, and the average rewards achieved are considerably lower compared to MAPPO. MADQN's inconsistent performance makes it not a good algorithm for training the agents in the modified environment. So, the MAPPO algorithm is the best algorithm out of the three to train agents in the modified environment.

4.6 Summary

Comparing the performance of three different MARL algorithms-MAPPO, MADQN, and MAACKTR-in a medium traffic density setting in both the modified and the unmodified environment revealed meaningful insights on the performance and the scalability of these algorithms.

In the unmodified environment, MAPPO consistently achieves higher average rewards and exhibits an apparent positive learning curve. While MAACKTR also shows a positive learning trend, the algorithm's learning rate is slower than that of MAPPO. This suggests that MAACKTR will need to train the agents longer than MAPPO to learn optimal policies. In contrast to these algorithms, MADQN struggles to maintain consistent performance in an unmodified environment.

When scaled to the modified environment, MAPPO maintains its performance in achieving higher rewards and showing a positive learning trend. Its performance is similar, not improved, compared to the unmodified environment. This proves that MAPPO is scalable to the modified environment. However, the performance of the MAACKTR algorithm is degraded when scaled to the modified environment. This shows that the MAACKTR algorithm does not scale well to the modified environment. Similar to the unmodified environment, the MADQN algorithm struggles to maintain a consistent performance when scaled to the modified environment. It maintains a comparable performance to the unmodified environment. Although the MADQN algorithm performs similarly in the modified environment, it could be more consistent.

In conclusion, the MAPPO algorithm is the most scalable and efficient algorithm out of

the three training agents for highway on-ramp merging of CAVs in mixed traffic conditions. This can be evidenced by its performance and learning rate across both the modified and the unmodified environments. Although the MAACKTR algorithm performed well in the unmodified environment, its performance degraded and did not scale well to the modified environment. Finally, MADQN displayed the most inconsistent performance across both the modified and the unmodified environments, making it the least suitable out of the three for highway on-ramp merging of CAVs in mixed traffic conditions.

Chapter 5

Conclusions & Future Work

This chapter summarises the work presented in the dissertation, discusses the conclusions that can be drawn from the experiments (Section 5.1), and the possible future directions of this research (Section 5.2).

5.1 Summary

This dissertation aimed to verify the scalability of the existing Multi-Agent Reinforcement Learning (MARL) frameworks to a multi-lane highway on-ramp merging scenario involving Connected Autonomous Vehicles (CAVs) in mixed traffic scenarios. Most existing research in highway on-ramp merging of CAVs tests the algorithms on a single-lane on-ramp and often oversees the possibility of a multi-lane on-ramp. This leaves the performance of these algorithms largely unknown in such multi-lane on-ramp merging scenarios.

To address this crucial gap in the research, in this paper, I have extended (section **??**) the existing merge environment from "highway-env" to add an extra lane on the on-ramp. Further, in this new environment, I tested the scalability of three different MARL frameworks-MAPPO, MADQN, and MAACKTR Chen et al. [2022]. These three algorithms, proposed in the study Chen et al. [2022] and available at Chen [2023], were used to train the CAVs in both the modified multi-lane on-ramp environment and the unmodified single-lane on-ramp environment to assess their scalability and performance.

The experiments' findings indicate that the MAPPO algorithm is the most scalable and effective algorithm for managing the complexities of multi-lane merging scenarios. MAPPO obtains higher average rewards and demonstrates a positive learning curve in both the modified and unmodified environments. This indicates that the CAVs are learning from previous episodes, making it a good choice for training CAVs in highway on-ramp merging scenarios.

The MADQN algorithm proved to be the most unpredictable in terms of performance, with the average rewards fluctuating erratically in both environments. While the algorithm's performance has potential for scalability in the modified environment, its inconsistent learning trend in both environments renders it unsuitable for training CAVs in mixed traffic and highway on-ramp merging scenarios.

MAACKTR algorithm showed reasonable performance in an unmodified environment with agents slowing learning from the previous episodes. However, it did not scale well to the modified environment. The algorithm's performance declined drastically, and the average rewards obtained showed that the agents had not learned from the previous episodes. This indicates that MAACKTR is not scalable to the modified environment. Even though it was a good algorithm to train agents in the unmodified environment, its performance suggests that it needs to train the agents well in the modified environment.

In conclusion, this dissertation highlights the limitations in scaling the existing Multi-Agent Reinforcement Learning (MARL) frameworks to more complex, multi-lane on-ramp merging scenarios. The experiments concluded that not all existing MARL frameworks scale seamlessly to multi-lane on-ramp merging scenarios. Out of the three MARL frameworks tested-MAPPO, MADQN and MAACKTR-only MAPPO successfully scale to the modified multi-lane on-ramp merging environment. This work lays a foundation towards exploring the limitations and potential of Multi-Argent Reinforcement Learning in the context of highway on-ramp merging of CAVs in mixed traffic scenarios.

5.2 Future Work

This dissertation addresses the gap in exploring the scalability of existing MARL frameworks to a multi-lane on-ramp merging scenario. This research only discussed the scalability of three different algorithms: MAPPO, MADQN, and MAACKTR. One of the obvious directions for future work is to explore the scalability of various other existing MARL frameworks.

Moreover, there is a crucial need to enhance the existing frameworks to improve scalability and real-world performance. Another direction would be to enhance the existing frameworks to make them more scalable to various real-world scenarios. Focus can be put on strengthening the algorithms to make Connected Autonomous Vehicles (CAVs) better co-exist with Human Driven Vehicles (HDVs) while considering scalability, efficiency, and security. One prominent way would be exploring the application of Distributional Reinforcement Learning as it offers enormous potential for significant improvements and is relatively unexplored in this area.

Future research in these areas will be essential for the successful and seamless integra-

tion of Connected Autonomous Vehicles into mixed-traffic environments.

Bibliography

Part 2. navigate from rl to marl – marllib v1.0.0 documentation. https://marllib.readthedocs.io/en/latest/intro_marl/marl.html#. Accessed: Jan. 09, 2024.

What is reinforcement learning? https://aws.amazon.com/what-is/reinforcement-learning/, 2023. Accessed: 2024-04-15.

F. Acito. Dimensionality reduction. In *Predictive Analytics with KNIME*. Springer, Cham, 2023. doi: 10.1007/978-3-031-45630-5_5.

Khattab M. Ali Alheeti and Klaus McDonald-Maier. An enhanced aodv protocol for external communication in self-driving vehicles. In *2017 Seventh International Conference on Emerging Security Technologies (EST)*, pages 179–184, 2017. doi: 10.1109/EST.2017.8090420.

Karla Amezquita-Semprun, Yuvraj C. Pradeep, Pin-Chih Chen, Wei Chen, and Zhengyong Zhao. Experimental evaluation of the stimuli-induced equilibrium point concept for automatic ramp merging systems. *IEEE Transactions on Intelligent Transportation Systems*, 21(2):815–827, 2019. doi: 10.1109/TITS.2019.2899229.

Maarten Kroesen Andreia Martinho, Nils Herber and Caspar Chorus. Ethical issues in focus by the autonomous vehicles industry. *Transport Reviews*, 41(5):556–577, 2021. doi: doi:10.1080/01441647.2020.1862355. URL https://doi.org/10.1080/01441647.2020.1862355.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.

L. Busoniu, R. Babuska, and B. De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2):156–172, Mar 2008. doi: 10.1109/TSMCC.2007.913919.

Wenqi Cai, Arash B. Kordabad, Hossein N. Esfahani, Anastasios M. Lekkas, and Sébastien Gros. Mpc-based reinforcement learning for a simplified freight mission of autonomous surface vehicles. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2990–2995, 2021. doi: 10.1109/CDC45484.2021.9683750.

Ricardo S. Campos. Evolution of positioning techniques in cellular networks, from 2g to 4g. *Wireless Communications and Mobile Computing*, 2017, 2017. doi: 10.1155/2017/3082762.

Wei Cao, Masaki Mukai, Taketoshi Kawabe, Hiroshi Nishira, and Noboru Fujiki. Cooperative vehicle path generation during merging using model predictive control with real-time optimization. *Control Engineering Practice*, 34:98–105, 2015. doi: 10.1016/j.conengprac.2014.09.002.

Centre for Connected and Autonomous Vehicles. Uk government funding to boost self-driving transport technologies. GOV.UK, November 2023. URL `https://www.gov.uk/government/news/uk-government-funding-to-boost-self-driving-transport-technologies`. Accessed: 2024-04-22.

Dong Chen. Marl_cavs. `https://github.com/DongChen06/MARL_CAVs`, 2023.

Dong Chen, Mohammad Hajidavalloo, Zhaojian Li, Kaian Chen, Yongqiang Wang, Longsheng Jiang, and Yue Wang. Deep multi-agent reinforcement learning for highway on-ramp merging in mixed traffic, 2022.

Jianyu Chen, Bodi Yuan, and Masayoshi Tomizuka. Model-free deep reinforcement learning for urban autonomous driving. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 2765–2771, 2019. doi: 10.1109/ITSC.2019.8917306.

Sikai Chen, Jiqian Dong, Paul (Young Joun) Ha, Yujie Li, and Samuel Labi. Graph neural network and reinforcement learning for multi-agent cooperative control of connected autonomous vehicles. *Computer-Aided Civil and Infrastructure Engineering*, 36(7):838–857, 2021. ISSN 1467-8667. doi: 10.1111/mice.12702. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/mice.12702`. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/mice.12702.

Tianshu Chu, Jie Wang, Lara Codecà, and Zhaojian Li. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 21(3):1086–1095, 2020a. doi: 10.1109/TITS.2019.2901791.

Tianshu Chu, Jie Wang, Lara Codecà, and Zhaojian Li. Multi-Agent Deep Reinforcement Learning for Large-Scale Traffic Signal Control. *IEEE Transactions on Intelligent Transportation Systems*, 21(3):1086–1095, March 2020b. ISSN 1558-0016. doi: 10.1109/TITS.2019.2901791. URL https://ieeexplore.ieee.org/document/8667868. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

CNN. Uber self-driving car death: Operator found guilty. CNN, July 2023. URL https://edition.cnn.com/2023/07/29/business/uber-self-driving-car-death-guilty/index.html. Accessed: 2024-04-22.

Cédric Colas, Olivier Sigaud, and Pierre-Yves Oudeyer. How many random seeds? statistical power analysis in deep reinforcement learning experiments, 2018.

J. Connelly, W. S. Hong, R. B. Mahoney Jr., and D. A. Sparrow. Current challenges in autonomous vehicle development. In Grant R. Gerhart, Charles M. Shoemaker, and Douglas W. Gage, editors, *Unmanned Systems Technology VIII*, volume 6230, page 62300D. International Society for Optics and Photonics, SPIE, 2006. doi: 10.1117/12.666574. URL https://doi.org/10.1117/12.666574.

Karsten Crede. Connected cars – a potential target for hackers?! URL https://next.ergo.com/en/New-Mobility/2023/Karsten-Crede-connected-cars-hackers-software-cybersecurity.

Jacob Crewe, Aditya Humnabadkar, Yonghuai Liu, Amr Ahmed, and Ardhendu Behera. Slav-sim: A framework for self-learning autonomous vehicle simulation. *Sensors*, 23 (20), 2023. ISSN 1424-8220. doi: 10.3390/s23208649. URL https://www.mdpi.com/1424-8220/23/20/8649.

Jian Deng, Jian Li, Lei Zhao, and Liang Guo. A dual-band inverted-f mimo antenna with enhanced isolation for wlan applications. *IEEE Antennas and Wireless Propagation Letters*, 16:2270–2273, 2017. doi: 10.1109/LAWP.2017.2738024.

Lifu Ding, Gangfeng Yan, and Jianing Liu. Multiagent reinforcement learning for strictly constrained tasks based on reward recorder. *International Journal of Intelligent Systems*, 37(11):8387–8411, 2022. doi: https://doi.org/10.1002/int.22945. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/int.22945.

J. Dinneweth, A. Boubezoul, R. Mandiau, and S. Espié. Multi-agent reinforcement learning for autonomous vehicles: A survey. *Autonomous Intelligent Systems*, 2(1):27, Nov 2022a. doi: 10.1007/s43684-022-00045-z.

Joren Dinneweth, Abderrafiaa Boubezoul, René Mandiau, et al. Multi-agent reinforcement learning for autonomous vehicles: a survey. *Autonomous Intelligent Systems*, 2:27, 2022b. doi: 10.1007/s43684-022-00045-z. URL https://doi.org/10.1007/s43684-022-00045-z.

Joris Dinneweth, Abderrahmane Boubezoul, René Mandiau, and Stéphane Espié. Multi-agent reinforcement learning for autonomous vehicles: a survey. *Autonomous Intelligent Systems*, 2(1):27, November 2022c. ISSN 2730-616X. doi: 10.1007/s43684-022-00045-z. URL https://doi.org/10.1007/s43684-022-00045-z.

John Doe and Jane Smith. A theoretical exploration of deep learning, 2018.

European Commission. Cooperative, connected and automated mobility (ccam). European Commission - Mobility and Transport, April 2024. URL https://transport.ec.europa.eu/transport-themes/intelligent-transport-systems/cooperative-connected-and-automated-mobility-ccam_en. Accessed: 2024-04-22.

Hugging face. The bellman equation: simplify our value estimation. https://huggingface.co/learn/deep-rl-course/unit2/bellman-equation, N.d.

Daniel J. Fagnant and Kara Kockelman. Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations. *Transportation Research Part A: Policy and Practice*, 77:167–181, 2015. ISSN 0965-8564. doi: https://doi.org/10.1016/j.tra.2015.04.003. URL https://www.sciencedirect.com/science/article/pii/S0965856415000804.

Farama Foundation. Gym library, 2023a. URL https://www.gymlibrary.dev.

Farama Foundation. Gymnasium, 2023b. URL https://gymnasium.farama.org.

T. Fuchida, K. T. Aung, and A. Sakuragi. A study of q-learning considering negative rewards. *Artificial Life and Robotics*, 15:351–354, 2010. doi: 10.1007/s10015-010-0822-7.

General Motors. Path to autonomous driving, 2024. URL https://www.gm.com/commitments/path-to-autonomous. Accessed: 2024-04-22.

Roger Grosse, Chris Maddison, Juhan Bae, and Silviu Pitis. Csc 311: Introduction to machine learning. https://www.cs.toronto.edu/~rgrosse/courses/csc311_f20/slides/lec11.pdf, 2020.

Jacopo Guanetti, Yeojun Kim, and Francesco Borrelli. Control of connected and automated vehicles: State of the art and future challenges. *Annual Reviews in Control*, 45:18–40, January 2018. ISSN 1367-5788. doi: 10.1016/j.arcontrol.2018.04.011. URL https://www.sciencedirect.com/science/article/pii/S1367578818300336.

Xiaotian Hao, Hangyu Mao, Weixun Wang, Yaodong Yang, Dong Li, Yan Zheng, Zhen Wang, and Jianye Hao. Breaking the curse of dimensionality in multiagent state space: A unified agent permutation framework, 2022.

Shanglu He, Fan Ding, Chaoru Lu, and Yong Qi. Impact of connected and autonomous vehicle dedicated lane on the freeway traffic efficiency. *European Transport Research Review*, 14(1):12, April 2022a. ISSN 1866-8887. doi: 10.1186/s12544-022-00535-4. URL https://doi.org/10.1186/s12544-022-00535-4.

Sheng He, Feng Ding, Chao Lu, et al. Impact of connected and autonomous vehicle dedicated lane on the freeway traffic efficiency. *European Transport Research Review*, 14:12, 2022b. doi: 10.1186/s12544-022-00535-4. URL https://doi.org/10.1186/s12544-022-00535-4.

Tianfeng Hu, Zhiqun Hu, Zhaoming Lu, and Xiangming Wen. Dynamic traffic signal control using mean field multi-agent reinforcement learning in large scale road-networks. *IET Intelligent Transport Systems*, 17(9):1715–1728, 2023. ISSN 1751-9578. doi: 10.1049/itr2.12364. URL https://onlinelibrary.wiley.com/doi/abs/10.1049/itr2.12364. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1049/itr2.12364.

Yeping Hu, Alireza Nakhaei, Masayoshi Tomizuka, and Kikuo Fujimura. Interaction-aware decision making with adaptive strategies under merging scenarios. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 151–158. IEEE, 2019. doi: 10.1109/IROS40897.2019.8967767.

Yujing Hu, Qing Da, Anxiang Zeng, Yang Yu, and Yinghui Xu. Reinforcement learning to rank in e-commerce search engine: Formalization, analysis, and application, 2018.

Abdikarim Mohamed Ibrahim, Kok-Lim Alvin Yau, Yung-Wey Chong, and Celimuge Wu. Applications of multi-agent deep reinforcement learning: Models and algorithms. *Applied Sciences*, 11(22):10870, 2021. doi: 10.3390/app112210870. URL https://doi.org/10.3390/app112210870.

Seth Karten, Mycal Tucker, Huao Li, Siva Kailas, Michael Lewis, and Katia Sycara. Interpretable learned emergent communication for human–agent teams. *IEEE Trans-*

actions on Cognitive and Developmental Systems, 15(4):1801–1811, 2023. doi: 10.1109/TCDS.2023.3236599.

Dhanoop Karunakaran. Relationship between state (v) and action(q) value function in reinforcement learning. https://medium.com/intro-to-artificial-intelligence/relationship-between-state-v-and-action-q-value-function-in-reinforcement-learn 2021.

Shinpei Kato, Eijiro Takeuchi, Yoshio Ishiguro, Yoshiki Ninomiya, Kazuya Takeda, and Tsuyoshi Hamada. An open approach to autonomous vehicles. *IEEE Micro*, 35(6): 60–68, 2015. doi: 10.1109/MM.2015.133.

Hamza Khan, Petri Luoto, Sumudu Samarakoon, Mehdi Bennis, and Matti Latva-Aho. Network slicing for vehicular communication. *Transactions on Emerging Telecommunications Technologies*, 32(1):e3652, 2021. doi: https://doi.org/10.1002/ett.3652. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.3652. e3652 ett.3652.

Dong-Ki Kim, Matthew Riemer, Miao Liu, Jakob N. Foerster, Michael Everett, Chuangchuang Sun, Gerald Tesauro, and Jonathan P. How. Influencing long-term behavior in multiagent reinforcement learning, 2022.

Jong-Hoon Kim, Jun-Ho Huh, Se-Hoon Jung, and Chun-Bo Sim. A study on an enhanced autonomous driving simulation model based on reinforcement learning using a collision prevention model. *Electronics*, 10(18), 2021. ISSN 2079-9292. doi: 10.3390/electronics10182271. URL https://www.mdpi.com/2079-9292/10/18/2271.

Jongho Kim, Donghyun Lim, Younghoon Seo, Jaehyun (Jason) So, and Hyungjoo Kim. Influence of dedicated lanes for connected and automated vehicles on highway traffic flow. *IET Intelligent Transport Systems*, 17(4):678–690, 2023. ISSN 1751-9578. doi: 10.1049/itr2.12295. URL https://onlinelibrary.wiley.com/doi/abs/10.1049/itr2.12295. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1049/itr2.12295.

Henrietta Lengyel, Tamás Tettamanti, and Zsolt Szalay. Conflicts of automated driving with conventional traffic infrastructure. *IEEE Access*, 8:163280–163297, 2020. doi: 10.1109/ACCESS.2020.3020653.

Edouard Leurent. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018a.

Edouard Leurent. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018b.

Edouard Leurent. An environment for autonomous driving decision-making. https://github.com/Farama-Foundation/HighwayEnv, 2018c.

L. Li, V. Bulitko, and R. Greiner. Batch reinforcement learning with state importance. In J.-F. Boulicaut, F. Esposito, F. Giannotti, and D. Pedreschi, editors, *Machine Learning: ECML 2004*, volume 3201 of *Lecture Notes in Computer Science*, Berlin, Heidelberg, 2004. Springer. doi: 10.1007/978-3-540-30115-8_53.

Shengxiang Li, Ou Li, Guangyi Liu, Siyuan Ding, and Yijie Bai. Trajectory based prioritized double experience buffer for sample-efficient policy optimization. *IEEE Access*, 9:101424–101432, 2021. doi: 10.1109/ACCESS.2021.3097357.

Wenhao Li, Xiangfeng Wang, Bo Jin, Junjie Sheng, and Hongyuan Zha. Dealing with non-stationarity in marl via trust-region decomposition, 2022.

Yanchang Liang, Xiaowei Zhao, and Li Sun. A multiagent reinforcement learning approach for wind farm frequency control. *IEEE Transactions on Industrial Informatics*, 19 (2):1725–1734, 2023. doi: 10.1109/TII.2022.3182328.

Yiheng Lin, Guannan Qu, Longbo Huang, and Adam Wierman. Multi-agent reinforcement learning in stochastic networked systems, 2021.

Jiajing Ling, Kushagra Chandak, and Akshat Kumar. Integrating knowledge compilation with reinforcement learning for routes. *Proceedings of the International Conference on Automated Planning and Scheduling*, 31(1):542–550, May 2021. doi: 10.1609/icaps.v31i1.16002. URL https://ojs.aaai.org/index.php/ICAPS/article/view/16002.

Jun Liu, Wei Zhao, and Cheng Xu. An efficient on-ramp merging strategy for connected and automated vehicles in multi-lane traffic. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):5056–5067, 2021. doi: 10.1109/TITS.2021.3070946.

Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2018. URL https://sumo.dlr.de/docs/index.html.

Xiao-Yun Lu and J. Karl Hedrick. Longitudinal control algorithm for automated vehicle merging. *International Journal of Control*, 76(2):193–202, 2003. doi: 10.1080/0020717031000060541.

Yang Lu, Xin Xu, Xinglong Zhang, Lilin Qian, and Xing Zhou. Hierarchical Reinforcement Learning for Autonomous Decision Making and Motion Planning of Intelligent Vehicles. *IEEE Access*, 8:209776–209789, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3034225. URL `https://ieeexplore.ieee.org/document/9241055`. Conference Name: IEEE Access.

Thorsten Luettel, Michael Himmelsbach, and Hans-Joachim Wuensche. Autonomous ground vehicles—concepts and a path to the future. *Proceedings of the IEEE*, 100 (Special Centennial Issue):1831–1839, 2012. doi: 10.1109/JPROC.2012.2189803.

Zhen Luo, Zhongliang Pei, and Bihua Zou. Directional polarization modulation for secure dual-polarized satellite communication. In *2019 International Conference on Communications, Information System and Computer Engineering (CISCE)*, pages 270–275. IEEE, 2019. doi: 10.1109/CISCE.2019.00062.

Lijing Ma, Shiru Qu, Lijun Song, and Bo Liu. Exploring the effect of connected autonomous vehicles in mixed traffic flow. In *Third International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI 2022)*, volume 12509, pages 201–206. SPIE, January 2023. doi: 10.1117/12.2656036. URL `https://www.spiedigitallibrary.org/conference-proceedings-of-spie/12509/125090W/Exploring-the-effect-of-connected-autonomous-vehicles-in-mixed-traffic/10.1117/12.2656036.full`.

A. M. Ishtiaque Mahbub, Behdad Chalaki, and Andreas A. Malikopoulos. A Constrained Optimal Control Framework for Vehicle Platoons with Delayed Communication, November 2021. URL `http://arxiv.org/abs/2111.08080`. arXiv:2111.08080 [math].

Neelesh R. Malankar and Raj Shah. Qos analysis over wimax network with varying modulation schemes and efficiency modes. *International Journal of Computer Applications*, 162(8):9–16, 2017. doi: 10.5120/ijca2017914286.

M.H. Martens and A.P. van den Beukel. The road to automated driving: Dual mode and human factors considerations. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 2262–2267, 2013. doi: 10.1109/ITSC.2013.6728564.

Marilo Martin-Gasulla, Peter Sukennik, and Jochen Lohmiller. Investigation of the impact on throughput of connected autonomous vehicles with headway based on the leading vehicle type. *Transportation Research Record*, 2673(5):617–626, 2019. doi: 10.1177/0361198119839989. URL https://doi.org/10.1177/0361198119839989.

Maike M. Mayer, Axel Buchner, and Raoul Bell. Humans, machines, and double standards? the moral evaluation of the actions of autonomous vehicles, anthropomorphized autonomous vehicles, and human drivers in road-accident dilemmas. *Frontiers in Psychology*, 13, 2023. ISSN 1664-1078. doi: 10.3389/fpsyg.2022.1052729. URL https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.1052729.

Tim Miller. Policy gradients. Available: https://gibberblot.github.io/rl-notes/single-agent/policy-gradients.html#policy-improvement-using-gradient-ascent, 2023.

A.J.M. Muzahid, S.F. Kamarulzaman, M.A. Rahman, et al. Multiple vehicle cooperation and collision avoidance in automated vehicles: survey and an ai-enabled conceptual framework. *Sci Rep*, 13:603, 2023. doi: 10.1038/s41598-022-27026-9.

Srikanth K. S. Nakka, Behdad Chalaki, and Andreas A. Malikopoulos. A multi-agent deep reinforcement learning coordination framework for connected and automated vehicles at merging roadways. In *2022 American Control Conference (ACC)*, pages 3297–3302. IEEE, 2022. doi: 10.23919/ACC53348.2022.9867412.

Ashish Nanda, Deepak Puthal, Joel J. P. C. Rodrigues, and Sergei A. Kozlov. Internet of Autonomous Vehicles Communications Security: Overview, Issues, and Directions. *IEEE Wireless Communications*, 26(4):60–65, August 2019. ISSN 1558-0687. doi: 10.1109/MWC.2019.1800503. URL https://ieeexplore.ieee.org/document/8809661. Conference Name: IEEE Wireless Communications.

Thanh Tang Nguyen, Sunil Gupta, and Svetha Venkatesh. Distributional reinforcement learning via moment matching, 2020.

Alexandros Nikitas, Ioannis Kougias, Elena Alyavina, and Eric Njoya Tchouamou. How can autonomous and connected vehicles, electromobility, brt, hyperloop, shared use mobility and mobility-as-a-service shape transport futures for the context of smart cities? *Urban Science*, 1(4), 2017. ISSN 2413-8851. doi: 10.3390/urbansci1040036. URL https://www.mdpi.com/2413-8851/1/4/36.

OpenAI. Part 1: Key concepts in rl. https://spinningup.openai.com/en/latest/spinningup/rl_intro.html, 2018. Accessed: 2024-04-15.

OpenAI. Openai baselines: Acktr & a2c, 2021. URL https://openai.com/research/openai-baselines-acktr-a2c. Accessed: [Insert date of access here].

Zhaotian Pan, Zhaowei Qu, Yongheng Chen, Haitao Li, and Xin Wang. A Distributed Assignment Method for Dynamic Traffic Assignment Using Heterogeneous-Adviser Based Multi-Agent Reinforcement Learning. *IEEE Access*, 8:154237–154255, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3018267. URL https://ieeexplore.ieee.org/document/9172059. Conference Name: IEEE Access.

Alkis Papadoulis, Mohammed Quddus, and Marianna Imprialou. Evaluating the safety impact of connected and autonomous vehicles on motorways. *Accident Analysis Prevention*, 124:12–22, 2019a. ISSN 0001-4575. doi: https://doi.org/10.1016/j.aap.2018.12.019. URL https://www.sciencedirect.com/science/article/pii/S0001457518306018.

Alkis Papadoulis, Mohammed Quddus, and Marianna Imprialou. Evaluating the safety impact of connected and autonomous vehicles on motorways. *Accident Analysis & Prevention*, 124:12–22, March 2019b. ISSN 0001-4575. doi: 10.1016/j.aap.2018.12.019. URL https://www.sciencedirect.com/science/article/pii/S0001457518306018.

Leandro Parada, Eduardo Candela, Luis Marques, and Panagiotis Angeloudis. Safe and efficient manoeuvring for emergency vehicles in autonomous traffic using multi-agent proximal policy optimisation, 2022.

Scott Drew Pendleton, Hans Andersen, Xinxin Du, Xiaotong Shen, Malika Meghjani, You Hong Eng, Daniela Rus, and Marcelo H. Ang. Perception, planning, control, and coordination for autonomous vehicles. *Machines*, 5(1), 2017. ISSN 2075-1702. doi: 10.3390/machines5010006. URL https://www.mdpi.com/2075-1702/5/1/6.

Xiaobo Qu, Yang Yu, Mofan Zhou, Chin-Teng Lin, and Xiangyu Wang. Jointly dampening traffic oscillations and improving energy consumption with electric, connected and automated vehicles: A reinforcement learning based approach. *Applied Energy*, 257: 114030, January 2020a. ISSN 0306-2619. doi: 10.1016/j.apenergy.2019.114030. URL https://www.sciencedirect.com/science/article/pii/S0306261919317179.

Zhaowei Qu, Zhaotian Pan, Yongheng Chen, Xin Wang, and Haitao Li. A Distributed Control Method for Urban Networks Using Multi-Agent Reinforcement

Learning Based on Regional Mixed Strategy Nash-Equilibrium. *IEEE Access*, 8: 19750–19766, 2020b. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.2968937. URL https://ieeexplore.ieee.org/document/8967108/.

Ashish Rauniyar, Desta Haileselassie Hagos, Manish Shrestha, and Claudio Agostino Ardagna. A crowd-based intelligence approach for measurable security, privacy, and dependability in internet of automated vehicles with vehicular fog. *Mobile Information Systems*, 2018:7905960, 2018. ISSN 1574-017X. doi: 10.1155/2018/7905960. URL https://doi.org/10.1155/2018/7905960.

Wutthigrai Boonsuk Rendong Bai and Peter P. Liu. Autonomous driving and related technologies. In *2019 ASEE Annual Conference & Exposition*, number 10.18260/1-2–32137, Tampa, Florida, June 2019. ASEE Conferences. https://peer.asee.org/32137.

William B. Ribbens. Chapter 12 - autonomous vehicles. In William B. Ribbens, editor, *Understanding Automotive Electronics (Eighth Edition)*, pages 573–593. Butterworth-Heinemann, eighth edition edition, 2017. ISBN 978-0-12-810434-7. doi: https://doi.org/10.1016/B978-0-12-810434-7.00012-0. URL https://www.sciencedirect.com/science/article/pii/B9780128104347000120.

Francisca Rosique, Pedro J. Navarro, Leanne Miller, and Eduardo Salas. Autonomous vehicle dataset with real multi-driver scenes and biometric data. *Sensors*, 23(4), 2023. ISSN 1424-8220. doi: 10.3390/s23042009. URL https://www.mdpi.com/1424-8220/23/4/2009.

Hammam Salem, M.D. Muzakkir Quamar, and Adeb Magad et al. Data-driven integrated sensing and communication: Recent advances, challenges, and future prospects. TechRxiv, July 2023.

Leon Schester and Luis E. Ortiz. Longitudinal position control for highway on-ramp merging: A multi-agent approach to automated driving. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3461–3468. IEEE, 2019. doi: 10.1109/ITSC.2019.8917375.

Lukas M. Schmidt, Johanna Brosig, Axel Plinge, Bjoern M. Eskofier, and Christopher Mutschler. An Introduction to Multi-Agent Reinforcement Learning and Review of its Application to Autonomous Mobility, March 2022. URL https://arxiv.org/abs/2203.07676v2.

David Shepardson. Tesla driver in fatal 'autopilot' crash got numerous warnings: U.s. government. Reuters, June 2017. URL https://www.reuters.com/article/idUSKBN19A2XC. Accessed: 2024-04-22.

Yunpeng Shi, Qing He, and Zhitong Huang. Capacity Analysis and Cooperative Lane Changing for Connected and Automated Vehicles: Entropy-Based Assessment Method. *Transportation Research Record*, 2673(8):485–498, August 2019. ISSN 0361-1981. doi: 10.1177/0361198119843474. URL https://doi.org/10.1177/0361198119843474. Publisher: SAGE Publications Inc.

Ziyu Song and Haitao Ding. Modeling car-following behavior in heterogeneous traffic mixing human-driven, automated and connected vehicles: considering multitype vehicle interactions. *Nonlinear Dynamics*, 111(12):11115–11134, June 2023. ISSN 1573-269X. doi: 10.1007/s11071-023-08377-y. URL https://doi.org/10.1007/s11071-023-08377-y.

Sergiu C. Stanciu, David W. Eby, Lisa J. Molnar, Renée M. St. Louis, Nicole Zanier, and Lidia P. Kostyniuk. Pedestrians/bicyclists and autonomous vehicles: How will they communicate? *Transportation Research Record*, 2672(22):58–66, 2018. doi: 10.1177/0361198118777091. URL https://doi.org/10.1177/0361198118777091.

Statista. Projected size of the global autonomous vehicle market by vehicle type. https://www.statista.com/statistics/428692/projected-size-of-global-autonomous-vehicle-market-by-vehicle-type/, 2023. [Online; accessed 20-April-2024].

Statista. Impact of vehicle automation on collision rates. https://www.statista.com/statistics/1238242/impact-of-vehicle-automation-on-collision-rates/, 2024. [Online; accessed 20-April-2024].

Haojie Sun, Shuo Feng, Xiangbin Yan, and Henry X. Liu. Corner case generation and analysis for safety assessment of autonomous vehicles. *Transportation Research Record*, 2675(11):587–600, 2021. doi: 10.1177/03611981211018697. URL https://doi.org/10.1177/03611981211018697.

Zhiyuan Sun, Tao Huang, and Peng Zhang. Cooperative decision-making for mixed traffic: A ramp merging example. *Transportation Research Part C: Emerging Technologies*, 120:102764, 2020. doi: 10.1016/j.trc.2020.102764.

S. Susilawati, W. J. Wong, and Z. J. Pang. Safety effectiveness of autonomous vehicles and connected autonomous vehicles in reducing pedestrian crashes. *Transportation Research Record*, 2677(2):1605–1618, 2023. doi: 10.1177/03611981221108984.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. Second edition edition, 2020. URL `http://incompleteideas.net/book/RLbook2020.pdf`. Accessed: 2024-04-15.

Árpád Takács, Dániel András Drexler, Péter Galambos, Imre J. Rudas, and Tamás Haidegger. Assessment and standardization of autonomous vehicles. In *2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES)*, pages 000185–000192, 2018. doi: 10.1109/INES.2018.8523899.

Alireza Talebpour and Hani S. Mahmassani. Influence of connected and autonomous vehicles on traffic flow stability and throughput. *Transportation Research Part C: Emerging Technologies*, 71:143–163, 2016a. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2016.07.007. URL `https://www.sciencedirect.com/science/article/pii/S0968090X16301140`.

Alireza Talebpour and Hani S. Mahmassani. Influence of connected and autonomous vehicles on traffic flow stability and throughput. *Transportation Research Part C: Emerging Technologies*, 71:143–163, October 2016b. ISSN 0968-090X. doi: 10.1016/j.trc.2016.07.007. URL `https://www.sciencedirect.com/science/article/pii/S0968090X16301140`.

Alireza Talebpour, Hani S. Mahmassani, and Samer H. Hamdar. Modeling lane-changing behavior in a connected environment: A game theory approach. *Transportation Research Part C: Emerging Technologies*, 59:216–232, 2015. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2015.07.007. URL `https://www.sciencedirect.com/science/article/pii/S0968090X15002478`. Special Issue on International Symposium on Transportation and Traffic Theory.

Hainan Tang, Juntao Liu, and Zhenjie Wang et al. Projection exploration for multi-agent reinforcement learning. PREPRINT (Version 1) available at Research Square, Apr 2023. URL `https://doi.org/10.21203/rs.3.rs-2759603/v1`. Accessed: [Insert today's date or the date you accessed the information].

Zuoyin Tang and Jianhua He. Noma enhanced 5g distributed vehicle to vehicle communication for connected autonomous vehicles. In *Proceedings of the ACM MobiArch 2020 The 15th Workshop on Mobility in the Evolving Internet Architec-*

ture, MobiArch'20, page 42–47, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450380812. doi: 10.1145/3411043.3412505. URL https://doi.org/10.1145/3411043.3412505.

Voot Tangkaratt, Abbas Abdolmaleki, and Masashi Sugiyama. Guide actor-critic for continuous control, 2018.

Tesla, Inc. Support for tesla autopilot, 2024. URL https://www.tesla.com/en_ie/support/autopilot. Accessed: 2024-04-22.

Mark Towers et al. Gymnasium. https://github.com/Farama-Foundation/Gymnasium, 2023.

Saber Fallah Mehrdad Dianati Alan Stevens David Oxtoby Umberto Montanaro, Shilp Dixit and Alexandros Mouzakitis. Towards connected autonomous driving: review of use-cases. *Vehicle System Dynamics*, 57(6):779–814, 2019. doi: 10.1080/00423114.2018.1492142. URL https://doi.org/10.1080/00423114.2018.1492142.

Martijn van Otterlo and Marco Wiering. Reinforcement learning and markov decision processes. In Marco Wiering and Martijn van Otterlo, editors, *Reinforcement Learning*, volume 12 of *Adaptation, Learning, and Optimization*. Springer, Berlin, Heidelberg, 2012. doi: 10.1007/978-3-642-27645-3_1.

Samir Wadhwania, Dong-Ki Kim, Shayegan Omidshafiei, and Jonathan P. How. Policy distillation and value matching in multiagent reinforcement learning. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8193–8200, 2019. doi: 10.1109/IROS40897.2019.8967849.

Mingyu Wang. An improved research of in-vehicle Internet of Things based on PID algorithm. In Xiangjie Kong and Francisco Falcone, editors, *3rd International Conference on Internet of Things and Smart City (IoTSC 2023*, volume 12708, page 127080Y. International Society for Optics and Photonics, SPIE, 2023. doi: 10.1117/12.2683896. URL https://doi.org/10.1117/12.2683896.

Sung-Jung Wang, S. K. Jason Chang, and Saber Fallah. Autonomous bus fleet control using multiagent reinforcement learning. *Journal of Advanced Transportation*, 2021:6654254, 2021a. doi: 10.1155/2021/6654254. URL https://doi.org/10.1155/2021/6654254.

Woodrow Z. Wang, Andy Shih, Annie Xie, and Dorsa Sadigh. Influencing towards stable multi-agent interactions, 2021b.

Xinshui Wang, Ke Meng, Xu Wang, Zhibin Liu, and Yuefeng Ma. Dynamic user resource allocation for downlink multicarrier noma with an actor–critic method. *Energies*, 16: 2984, 03 2023. doi: 10.3390/en16072984.

Ziran Wang, Guoyuan Wu, and Matthew J. Barth. Cooperative Eco-Driving at Signalized Intersections in a Partially Connected and Automated Vehicle Environment. *IEEE Transactions on Intelligent Transportation Systems*, 21(5):2029–2038, May 2020. ISSN 1558-0016. doi: 10.1109/TITS.2019.2911607. URL `https://ieeexplore.ieee.org/document/8704319`. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

Waymo LLC. Waymo - autonomous vehicle technology, 2024. URL `https://waymo.com`. Accessed: 2024-04-22.

Winder.AI. Predicting rewards with the state-value function. `https://rl-book.com/learn/mdp/state_value_function/`, N.d. Accessed: 2024-04-15.

M. K. Wong, T. Connie, M. K. O. Goh, et al. A visual approach towards forward collision warning for autonomous vehicles on malaysian public roads. *F1000Research*, 10:928, 2022. doi: 10.12688/f1000research.72897.2. [version 2; peer review: 2 approved].

Yuhuai Wu, Elman Mansimov, Shun Liao, Roger Grosse, and Jimmy Ba. Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation, 2017.

Zhenyu Wu, Kai Qiu, and Hongbo Gao. Driving policies of v2x autonomous vehicles based on reinforcement learning methods. *IET Intelligent Transport Systems*, 14 (5):331–337, 2020. doi: https://doi.org/10.1049/iet-its.2019.0457. URL `https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-its.2019.0457`.

Baidi Xiao, Rongpeng Li, Fei Wang, Chenghui Peng, Jianjun Wu, Zhifeng Zhao, and Honggang Zhang. Stochastic Graph Neural Network-based Value Decomposition for MARL in Internet of Vehicles, March 2023. URL `http://arxiv.org/abs/2303.13213`. arXiv:2303.13213 [cs].

R. Xie, Z. Meng, Y. Zhou, Y. Ma, and Z. Wu. Heuristic q-learning based on experience replay for three-dimensional path planning of the unmanned aerial vehicle. *Science Progress*, 103(1), 2020. doi: 10.1177/0036850419879024.

Tao Xu, Cheng Wen, Lei Zhao, Ming Liu, and Xiaoxiang Zhang. The hybrid model for lane-changing detection at freeway off-ramps using naturalistic driving trajectories. *IEEE Access*, 7:103716–103726, 2019. doi: 10.1109/ACCESS.2019.2931726.

Takaya Yamazato. V2x communications with an image sensor. *Journal of Communications and Information Networks*, 2:65–74, 2017. doi: 10.1007/s41650-017-0044-4. URL https://doi.org/10.1007/s41650-017-0044-4.

Ronghan Yao, Xiaojing Du, Wenyan Qi, and Li Sun. Evolutionary dynamics of mandatory lane changing for bus exiting. *Journal of Advanced Transportation*, 2021:Article ID 2958647, 2021. doi: 10.1155/2021/2958647. URL https://doi.org/10.1155/2021/2958647.

Fei Ye, Jianlin Guo, Kyeong Jin Kim, Philip V. Orlik, Heejin Ahn, Stefano Di Cairano, and Matthew J. Barth. Bi-level Optimal Edge Computing Model for On-ramp Merging in Connected Vehicle Environment. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 2005–2011, June 2019. doi: 10.1109/IVS.2019.8814096. URL https://ieeexplore.ieee.org/document/8814096. ISSN: 2642-7214.

Renos Zabounidis, Joseph Campbell, Simon Stepputtis, Dana Hughes, and Katia Sycara. Concept learning for interpretable multi-agent reinforcement learning, 2023.

Betina Carol Zanchin, Rodrigo Adamshuk, Max Mauro Santos, and Kathya Silvia Collazos. On the instrumentation and classification of autonomous cars. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2631–2636, 2017. doi: 10.1109/SMC.2017.8123022.

Hui Zhang, Yanyong Guo, Ninghao Hou, Jianhua Zhang, Xuyi Li, and Yan Huang. Evaluating the safety impact of connected and autonomous vehicles with lane management on freeway crash hotspots using the surrogate safety assessment model. *Journal of Advanced Transportation*, 2021:5565343, 2021. doi: 10.1155/2021/5565343. URL https://doi.org/10.1155/2021/5565343.

Jian Zhang, Yaozong Pan, Haitao Yang, and Yuqiang Fang. Scalable deep multi-agent reinforcement learning via observation embedding and parameter noise. *IEEE Access*, 7:54615–54622, 2019. doi: 10.1109/ACCESS.2019.2913235.

Jian Zhao, Xunhan Hu, Mingyu Yang, Wengang Zhou, Jiangcheng Zhu, and Houqiang Li. Ctds: Centralized teacher with decentralized student for multi-agent reinforcement learning, 2022.

Liuhui Zhao, Andreas A. Malikopoulos, and Jackeline Rios-Torres. On the Traffic Impacts of Optimally Controlled Connected and Automated Vehicles. In *2019 IEEE Conference on Control Technology and Applications (CCTA)*, pages 882–887, August

2019a. doi: 10.1109/CCTA.2019.8920630. URL https://ieeexplore.ieee.org/document/8920630.

Xiangmo Zhao, Shubin Jing, Fei Hui, Rongjie Liu, and Asad J. Khattak. Dsrc-based rear-end collision warning system—an error-component safety distance model and field test. *Transportation Research Part C: Emerging Technologies*, 107:92–104, 2019b. doi: 10.1016/j.trc.2019.08.011.

Shun Zhou, Weihua Zhuang, Guodong Yin, Hongchao Liu, and Chunxiang Qiu. Cooperative on-ramp merging control of connected and automated vehicles: Distributed multi-agent deep reinforcement learning approach. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 402–408. IEEE, 2022a. doi: 10.1109/ITSC55140.2022.9920596.

Wei Zhou, Dong Chen, Jun Yan, Zhaojian Li, Huilin Yin, and Wanchen Ge. Multi-agent reinforcement learning for cooperative lane changing of connected and autonomous vehicles in mixed traffic. *Autonomous Intelligent Systems*, 2(1):5, March 2022b. ISSN 2730-616X. doi: 10.1007/s43684-022-00023-5. URL https://doi.org/10.1007/s43684-022-00023-5.

C. Zhu, M. Dastani, and S. Wang. A survey of multi-agent deep reinforcement learning with communication. *Autonomous Agents and Multi-Agent Systems*, 38(1):4, Jan 2024. doi: 10.1007/s10458-023-09633-6.

Jie Zhu, Said Easa, and Kun Gao. Merging control strategies of connected and autonomous vehicles at freeway on-ramps: A comprehensive review. *Journal of Intelligent and Connected Vehicles*, 5(2):99–111, 2022. ISSN 2399-9802. doi: 10.1108/JICV-02-2022-0005. URL https://ieeexplore.ieee.org/document/10004548. Conference Name: Journal of Intelligent and Connected Vehicles.

Tong Zhu, Xiaohu Li, Wei Fan, Changshuai Wang, Haoxue Liu, and Runqing Zhao. Trajectory Optimization of CAVs in Freeway Work Zone considering Car-Following Behaviors Using Online Multiagent Reinforcement Learning. *Journal of Advanced Transportation*, 2021:e9805560, November 2021. ISSN 0197-6729. doi: 10.1155/2021/9805560. URL https://www.hindawi.com/journals/jat/2021/9805560/. Publisher: Hindawi.