Machine Learning Assisted Ultra Reliable and Low Latency Vehicular Optical Camera Communications

Amirul Islam

A thesis submitted for the degree of

Doctor of Philosophy

School of Computer Science and Electronic Engineering
University of Essex

April 2022

To my son **Aaryan Islam**, who is the best gift in my life.

Abstract

Optical camera communication (OCC) has emerged as a key enabling technology for the seamless operation of future autonomous vehicles. By leveraging the supreme performance of OCC, the stringent requirements of ultra-reliable and low-latency communication (uRLLC) can be met in vehicular OCC. In this thesis, a rate maximization approach is presented to vehicular OCC that aims to optimize vehicle speed, channel code rate, and modulation order while adhering to uRLLC requirements. The reliability is modelled by satisfying a target bit error rate (BER) and latency as transmission latency. To improve transmission rate and reliability, low-density parity-check codes and adaptive modulation are adopted in this thesis. First, the rate maximization problem is formulated as an optimization problem aimed at determining vehicle speed, channel code rates, and modulation order given reliability and latency constraints. Even for a small set of modulation orders, this problem is mixed integer programming, which is NP-hard. To overcome the complexity of the NP-hard problem, the proposed optimization problem is modelled as a Markov decision process and then solved it distributively using multi-agent deep reinforcement learning (DRL). Then, the optimization problem is solved using the actor-critic DRL framework with Wolpertinger architecture. A deep deterministic policy gradient algorithm is employed to operate over continuous action spaces. The proposed model and optimization formulation are justified through numerous simulations by comparing capacity, BER, and latency. From the findings, it is clear that the multi-agent DRL framework in vehicular OCC leads to improved performance in terms of maximizing the communica-

tion rate while respecting uRLLC. This work constitutes a significant step towards addressing the challenges in vehicular OCC to respect uRLLC.

Keywords: Deep reinforcement learning, optical camera communications, autonomous vehicular communications, uRLLC, LDPC code.

Acknowledgements

My deepest gratitude must foremost go to my supervisors, Prof. Leila Musavian and Prof. Nikolaos Thomos, for their invaluable guidance throughout the development of this thesis. It was a genuine pleasure and honour for me to work with them. Working with them was a genuine pleasure and honour for me. Throughout my entire PhD period, I was always inspired about how to flesh out and develop my research projects thanks to their wise guidance, immense knowledge, and incredible patience. I am very grateful that my experimental demonstration skills are well-developed because of the valuable experience provided by them.

I'd also like to thank Dr Manoj Thakur for his help and advice, as well as for agreeing to serve on my PhD progress monitoring committee. I would also like to express my gratitude to Prof. Tasos Dagiuklas and Dr Jianhua He for agreeing to serve on my dissertation examination committee. I sincerely thank all of my friends and colleagues in the School of CSEE for the stimulating and enjoyable discussions.

Last but most important of all, Last but most important of all, my heartfelt gratitude is for my beloved family, especially my parents, Mostafa Sardar and Jahanara Khatun, my brother, Jamirul Islam as well as my wife, Jesmin Nahar, who experienced all of the ups and downs of this journey but always provided me with unconditional love and support.

Publications

Journal Papers

J1 **A. Islam**, N. Thomos, and L. Musavian,"Deep Reinforcement Learning Based Ultra Reliable and Low Latency Vehicular OCC," submitted in *IEEE Transactions on Communications*. (Under Review)

J2 **A. Islam**, N. Thomos, and L. Musavian, "Multi-agent deep reinforcement learning for spectral efficiency optimization in vehicular optical camera communications," submitted in *IEEE Transactions on Mobile Computing*. (Under Review)

Conference Papers

C1 **A. Islam**, N. Thomos, and L. Musavian, "Achieving uRLLC with Machine Learning Based Vehicular OCC," Accepted for publication to *2022 IEEE GLOBECOM*, Rio de Janeiro, Brazil, Dec. 2022.

C2 **A. Islam**, L. Musavian, and N. Thomos, "Multi-agent deep reinforcement learning in vehicular OCC," in *Proc. 2022 IEEE 95th Vehicular Technology Conference*, Helsinki, Finland, Jun. 2022, pp. 1-6.

C3 **A. Islam**, L. Musavian, and N. Thomos, "Performance analysis of vehicular optical camera communications: Roadmap to uRLLC," in *Proc. IEEE Global Communications Conference (Globecom'19)*, Waikoloa, HI, USA, Dec. 2019, pp. 1-6.

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

Adam Adaptive Moment Estimation

AoI Angle of Incidences

AV Autonomous Vehicle

BER Bit Error Rate

bps bit per second

BPSK Binary Phase-Shift Keying

CDF Cumulative Distribution Function

DC Direct Current

DDPG Deep Deterministic Policy Gradient

DNN Deep Neural Network

DQN Deep Q-Network

DRL Deep Reinforcement Learning

EMI Electromagnetic Interference

E2E End-to-End

FoV Field of View

fps frame per second

GF Galois Field

GPS Global Positioning System

GUI Graphical User Interface

ITS Intelligent Transportation System

KNN K-nearest Neighbour

LDPC Low-Density Parity-Check

LED Light-Emitting Diode

LiDAR Light Detection and Ranging

LiFi Light Fidelity

LoS Line-of-Sight

LTE Long Term Evolution

MARL Multi-Agent Reinforcement Learning

Mbps Megabits per second

MDP Markov Decision Process

M-QAM M-ary Quadrature Amplitude Modulation

MSA Min-Sum algorithm

NN Neural Network

NP Non-deterministic Polynomial-time

NR New Radio

OCC Optical Camera Communication

PD Photodiode

QC Quasi-Cyclic

QoS Quality of Service

ReLU Rectified Linear Unit

RF Radio Frequency

RL Reinforcement Learning

RMSPro Root Mean Square Propagation

RSU Roadside Unit (RSU)

RV Receiver Vehicle

SARL Single Agent Reinforcement Learning

SLO Single Link Optimization

SNR Signal-to-Noise Ratio

SPA Sum-Product Algorithm

SUMO Simulation of Urban Mobility

TDMA Time Division Multiple Access

TraCI Traffic Control Interface

TV Transmitter Vehicle

uRLLC ultra-Reliable and Low-Latency Communication

VANET Vehicular Ad-Hoc Network

VLC Visible Light Communication

V2I Vehicle-to-Infrastructure

V2V Vehicle-to-Vehicle

WAP Wireless Access Point

3D Three Dimensional

5G Fifth Generation

List of Symbols

H	Channel gain
θ	AoI w.r.t. the receiver axix
t	Time-frame index
A_{eff}	Effective signal collection area of the image sensor
$\Re$	Transmitter radiant intensity
ϕ	Angle of irradiance w.r.t. the emitter
θ_l	FoV of the camera lens
d	Inter-vehicular distance
δ	Distance between the left and right LED array units
f	Lens's focal length
a	Image pixel size
m	Order of Lambertian radiation pattern
$\Phi_{1/2}$	LED semi-angle at half luminance
A	Image sensor physical area
T_{s}	Transmission efficiency of the optical filter
g	Lens's gain
n	Internal refractive index of the lens
P_{r}	Received optical power
P	Optical transmitting power
γ	Received SNR
ρ	Receiver's responsitivity
σ	Total noise power
q	Electron charge

W_{fps}	Camera-frame rate
W_{s}	Spatial-bandwidth
M	Constellation size
$\mathcal{M}$	Set of available modulation schemes
N_{LEDs}	Number of LEDs at each row
N_{row}	Captured number of row pixel lines
ϱ	Size of the LED
w	Resolution of image
L	Packet size
τ	End-to-end latency
B	Number of V2V links
b	Index of the backward V2V link
v	Relative speed of the vehicle
λ, ν	Lagrangian multiplier
$\mathcal{S}$	Set of all possible states
$\mathcal{A}$	Set of all possible actions
r	Reward function
ϵ	Exploration rate
ζ	Discount factor
α	Learning rate
ϱ	5G NR LDPC Code rates
$\mathcal{X}$	Set of 5G NR LDPC Code rates
β	Soft target updates rate

CHAPTER 1

Introduction

1.1 Motivations

Driven by vehicular networks, the automotive industry is undergoing key technological transformations through Autonomous Vehicles (AVs). In the modern world, the number of vehicles and vehicle-assisting infrastructures is increasing rapidly, making the transportation system more vulnerable than ever, resulting in more traffic congestion, road causalities, and overall less road safety. These rapid growths in the number of vehicles will open a significantly challenging but profitable market for the future Intelligent Transportation Systems (ITSs) [1]. To cope with the current ever-growing and complex vehicular networks, the practice of sharing information and cooperative driving on the road is substantially increasing. Moreover, there is the consumption of data along the way, where the users spend time on vehicles and want to consume more content. Besides, the vast amounts of generated data (each vehicle can generate up to 750 Megabits per second (Mbps)) need to be communicated in vehicular networks. However, The deployment of AVs can help in reducing traffic congestion, increasing road safety, minimizing fuel consumption, and enhancing the overall driving ex-

perience [2], [3]. Though several Vehicle-to-Vehicle (V2V) applications, such as lane changing alert, and automotive braking systems, have already been deployed, mission-critical services, e.g., collision avoidance, autonomous driving, and other safety-related issues, are still creating severe challenges. Therefore, providing efficient V2V communications is necessary for enabling future ITS [4].

The concept of establishing communication among devices is promising, and inter-vehicular communication has been attracting massive attention from academia and industry. AV communication will play an essential role in the next-generation networks and is considered as one of the most promising enablers for intelligent transportation systems [2], [5]. Typically, AV safety applications are believed to be time-critical, as they rely on acquiring real-time status updates from individual vehicles. To effectively operate AVs, reliable communication between vehicles and infrastructure is required. The performance of the growing transportation systems depends on the availability of V2V communication links at ultra-low latency and errors. As a result, data should be delivered within a short time, providing a high probability of success.

Every year, the data sharing within the vehicular networks are continuously increasing, thus incurring enormous network overhead [3, 6]. As a result, the current congested and saturated Radio Frequency (RF) spectrum cannot accommodate the increasing demand for data traffic although RF-based communication systems (e.g., cellular, Wi-Fi, and sensor networks) are essential parts of existing wireless communication systems. Recently, Optical Camera Communication (OCC) has emerged as a potential technology for ITS [7], [8] and as an alternative to RF due to the fact it offers license-free unlimited spectrum, longer lifespans, lower implementation cost, lower power consumption, and enhanced security [7]. OCC systems belong to the family of Visible Light Communication (VLC) systems. In typical OCC systems, Light-Emitting Diodes (LEDs) are usually used as transmitters and cameras are employed as receivers. VLC systems using Photo-

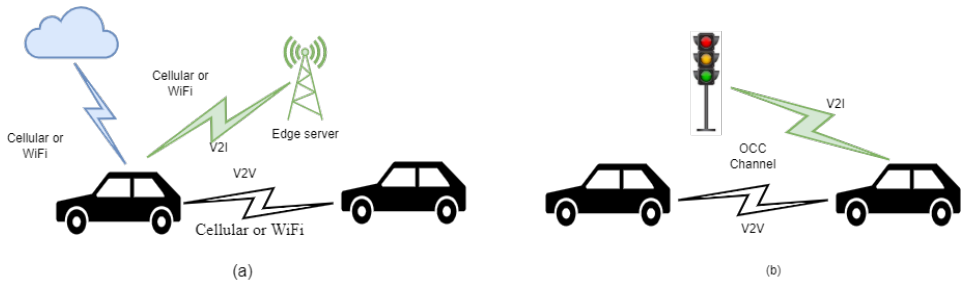

Figure 1.1: Overview of (a) RF and (b) OCC-based communication system.

diode (PD) as the receiver are called Light Fidelity (LiFi), which is termed as PD-based system in this thesis. In traditional VLC, the receiver often consists of a non-imaging device, i.e., PD, and its performance is limited by the trade-off between transmission range and signal reception. Different from PD-based systems, OCC can spatially separate and process different sources independently on its image plane, which allows the receiver to discard noise sources, e.g., Sun, streetlights, and other light sources, and focuses mainly on the pixels to which the LEDs strikes [8]. Thus, it reduces interference by a great margin. Furthermore, it is easier to integrate OCC with the existing vehicular communication systems at a minimum additional cost and without any significant infrastructure changes because the LED lights are already existing in traffic lights, infrastructures, and vehicles. OCC can face challenges due to its Line-of-Sight (LoS) properties, i.e., communication links can be obstructed by objects or bad weather conditions, for example, buildings, walls, rain, cloud, or fog. Studying the effect of weather conditions is beyond the scope of this thesis though it is an interesting topic.

Fig. 1.1 illustrates the key difference between RF and OCC-based system. From the Fig. 1.1(a), it is seen that RF-based systems employ cellular or WiFi, whereas OCC system uses direct LoS channel (Fig. 1.1(b)). In RF-based communication systems, the centre base station or edge server is a compulsory element. Moreover, they mostly rely on centralized resource management, where fast and efficient distributed algorithms are needed to

Table 1.1: Comparison between OCC, PD, and RF

Parameter	**VLC**		**RF**
	OCC	**PD**	
Carrier bandwidth	Unlimited (400 - 700) nm	Unlimited (400 - 700) nm	300 GHz (saturated and regulated)
Electromagnetic Interference	No	No	Yes
Transmitter	LED	LED or Laser Diode (LD)	Antenna
Receiver	Camera	PD	Antenna
Power consumption	Relatively low	Higher than OCC	Medium
Interference	Negligible	Low	Very high
Communication distance	200 m	10 m	$>$ 100 km using Microwave
Noise	No	Sun and ambient light	Electrical, electronic appliances
Security	High	High	Low
Data rate	54 Mbps	10 Gbps using LED, 100 Gbps using LD	6 Gbps (IEEE 802.11ad at 60GHz)
Main purpose	Illumination, communication	Illumination, communication	Communication, positioning
Limitation	Low data rate	Short distance, no mobility guaranty	Interference

manage tasks in dense vehicular networks. The information is processed centrally in the base station, which takes time to process and send back to the vehicles. As a result, it induces extra latency. Please note that in this dissertation, only V2V communication is considered though Vehicle-to-Infrastructure (V2I) is an interesting direction to work on in future. Table 1.1 summarizes the key differences of OCC, PD and RF communication systems, which shows that OCC suffers from almost negligible interference and consumes less power than RF. Further, OCC supports almost 20 times longer distance than the PD-based systems. Although having a low data rate, OCC can be a better alternative to the congested and saturated RF system due to its negligible noise and interference characteristics. Besides, OCC offers LoS communication, which guarantees security. The system will also be fully decentralized communication where each vehicle will process the surrounding information individually or collectively.

Another challenge of vehicular networks arises from the fact that, they are time-varying and highly dynamic, while the data should be delivered reliably within stringent time constraints for ensuring safety. This makes it challenging to respect ultra-Reliable and Low-Latency Communication (uRLLC). Various technologies have been proposed in recent years to ensure reliability and low latency in ITS using traditional optimization schemes, such as [9, 10], which reflect on delay minimization and reliability guarantee. Specifically, in [9], the vehicular network transmission power is minimized by grouping vehicles into clusters and modelling reliability as queuing delay violation probability. In [10], a joint resource allocation and power control algorithm is proposed to maximize the communication rate considering latency and reliability constraints. Vehicular communication systems become even more complex when they involve controlling various decision-making parameters, e.g., channel code rate, speed, distance and modulation scheme. Using traditional distributed methods, it is difficult to solve these decision-making problems because of the inherent complexity and the time needed to solve them.

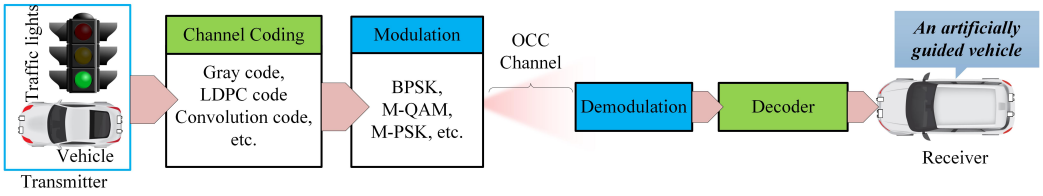

Figure 1.2: Basic OCC communication system.

Fortunately, Reinforcement Learning (RL) methods can serve as an effective solution to overcome the complexity of the above systems [11] due to the fact that it is possible to apply them distributively. The RL is modelled as a Markov Decision Process (MDP), where each vehicle acts as an agent, and everything beyond the particular vehicle is regarded as the environment. Every vehicular agent interacts with the environment to have a better understanding of it to decide its policy. The agents explore the environment and improve the policies based on their observations of the environmental state. Despite MDP providing an efficient way to express the vehicular problem, traditionally used methods to solve them, like value-iteration, require the knowledge of the state-action transition probability matrix that is difficult to be obtained in dynamic problems such as the one examined in this thesis. These limitations are overcome through Q-Learning [11]. However, Q-Learning has slow convergence and cannot solve large-scale problems. To address this limitation of the Q-Learning algorithm, Deep Reinforcement Learning (DRL) is utilized [12]. DRL has also emerged as a possible candidate to solve autonomous vehicular problems [13, 14]. DRL uses a Deep Q-Network (DQN) which combines Q-learning with Deep Neural Networks (DNNs) to approximate the state-action value function by adjusting the weights of the neural networks.

Channel coding scheme strongly affects the capacity and reliability of the system. In some cases, channel coding can be an optional choice, where only grey codes are used. Fig. 1.2 illustrates a basic OCC system consisting of transmitter, channel coding, modulation, demodulation, decoding and the receiver to show different parts of OCC system. The details

of this framework will be discussed in Chapter 4. Choice of channel coding and modulation depend on the user requirement. However, meeting uRLLC constraints necessitate the use of strong channel coding. There are different channel codes to ensure uRLLC, e.g., LDPC codes and convolution codes. Low-Density Parity-Check (LDPC) codes are a promising candidate for uRLLC, which has been adopted in the Fifth Generation (5G) New Radio (NR) services [15]. LDPC codes can help achieving a higher transmission rate, low latency and high reliability. In light of the fact that the proposed vehicular OCC system requires ultra-reliability and low-latency, 5G NR LDPC codes are employed in this thesis, which have already been implemented for adaptive modulation schemes [16].

1.2 Challenges of uRLLC in Vehicular OCC

Having discussed the motivations of vehicular OCC to ensure uRLLC in this thesis, the following research challenges are required to overcome:

> **Research Challenge 1**: How to meet the stringent transmission latency in vehicular OCC?
>
> **Research Challenge 2**: How to respect the ultra-reliability requirement while meeting low-latency?
>
> **Research Challenge 3**: How to satisfy uRLLC in multi-links vehicular networks?
>
> **Research Challenge 4**: How to meet uRLLC in AV communication while mitigating the mobility?

These research challenges are examined and resolved throughout this dissertation. A brief explanation of solving the above challenges is presented in the next section as contributions.

1.3 Contributions

In this thesis, we aim at maximizing the communication rate by optimizing code rates and modulation schemes to respect uRLLC constraints in vehicular OCC. To this end, we first justify whether OCC is suitable for employing uRLLC in Chapter 3. In this chapter, we demonstrate the OCC system model, which will be utilized for the rest of the thesis. To the best of our knowledge, this is the first time where OCC will be used to examine whether OCC is suitable for employing uRLLC that formulates the communication link performance with an adaptive modulation scheme in automotive vehicles. We analyze the performance of vehicular OCC in terms of Bit Error Rate (BER), spectral efficiency, and transmissions latency at different inter-vehicular distances and Angle of Incidences (AoI). Further, we investigate the use of adaptive modulation to improve spectral efficiency. Please note that Chapter 3 is also a research contribution where we test the validity of the OCC system model to employ uRLLC in vehicular communication. This has been published at the 2019 IEEE Global Communications Conference (GLOBECOM) [C3]. The major contributions of this dissertation are covered in Chapter 4, Chapter 5, and Chapter 6, which are presented briefly in next subsections.

1.3.1 Multi-Agent Deep Reinforcement Learning for Spectral Efficiency Optimization in Vehicular OCC

The first research question is examined in Chapter 4. In this chapter, a spectral efficiency maximization scheme in vehicular OCC is proposed that satisfies BER and latency constraints. In doing so, the optimal modulation order and speed of the vehicles using DRL are determined. A decentralized, independent and Multi-Agent Reinforcement Learning (MARL) scheme is considered to solve this problem. To the best of my knowledge, this is the first time where DRL is applied in vehicular OCC for resource allocation.

The formulated optimization problem aims at maximizing the spectral efficiency subject to BER, latency and a small set of modulation schemes constraints. The optimization function is a Non-deterministic Polynomial-time (NP) hard problem leading to a difficult search for the optimal solution. Hence, the optimization problem is formulated as an MDP problem to reduce the complexity of the NP-hard problem, which enables us to find an optimal solution. The reward function is designed considering the objective function and satisfying users' requirements. The constrained problem is relaxed into an unconstrained one using the Lagrangian relaxation method by relaxing the BER and latency constraints, which essentially simplifies the solution of the complex problem. We then solve the spectral efficiency maximization problem using deep Q-Learning to deal with large state-action spaces. We evaluate the performance of the proposed DRL-based optimization scheme and compare it with various variants of our scheme as well as RF-based communication schemes. The results demonstrate that a DRL-based optimization scheme can effectively learn to maximize spectral efficiency while meeting the constraints. The Cumulative Distribution Function (CDF) of latency and BER are evaluated, which confirm that the proposed system can satisfy ultra-low latency communication and BER constraints while the rest of the schemes fail. Further, the results show that the proposed vehicular system achieves better sum spectral efficiency and lower average latency compared to all the schemes under comparison. This work has been published in a conference paper at the IEEE 95th Vehicular Technology Conference [C1] and submitted for publication in IEEE Transactions on Mobile Computing, which is under revision currently.

1.3.2 Deep Reinforcement Learning based Ultra Reliable and Low Latency Vehicular OCC

In continuation to the finding of Chapter 4, where spectral efficiency is maximized, the answer to the second question is examined in Chapter 5.

In this chapter, 5G NR LDPC codes are presented in vehicular OCC, which offers variable rate, low latency and high reliability. To the best of my knowledge, this is the first where code rates are optimized using actor-critic based DRL scheme in vehicular OCC to ensure uRLLC. This method aims at maximizing the achievable rate while respecting the uRLLC constraints. In doing so, the communication rate is maximized subject to selecting the optimal modulation schemes, deciding appropriate code rates and adjusting the relative speed of the vehicle to the optimal value while respecting uRLLC requirements and dealing with the massive continuous state-action spaces. Similar to the problem in Chapter 3, the presented problem is an NP-hard problem, and it also contains non-linear operations in latency and BER formulations. Hence, the problem is modelled as a DRL framework. However, DQN cannot be straightforwardly applied to continuous state-action spaces [17], which is the case for the proposed vehicular system. The issues with continuous state-action space can be alleviated by adopting the actor-critic DRL frameworks [17]. The Wolpertinger architecture [18] along with the actor-critic network achieves convergence faster than the vanilla actor-critic method over a large actions space by considering the nearest neighbour's actions. Hence, an actor-critic DRL framework is employed by adopting the Wolpertinger policy for the vehicular OCC system. A Deep Deterministic Policy Gradient (DDPG) [17] is used to train the model, which updates both the critic and actor networks. A multi-layer neural network is employed as a function approximator for the actor and critic functions. The performance of the proposed DRL framework is evaluated in terms of achievable capacity, BER, and transmission latency. Then, the performance is compared with several variants of our scheme and an RF communication-based scheme. The average goodput of our proposed scheme shows a considerably higher value compared to other schemes under comparison. The proposed scheme can guarantee uRLLC while maximizing the goodput, whereas other methods fail most of the time. The results show that the proposed actor-critic based DRL scheme can achieve prom-

ising results and maximize the transmission rate while satisfying the uRLLC requirements and outperforming the comparison schemes. This work has been submitted for publication in the IEEE Transactions on Communication, and it is under review at this moment.

1.3.3 Multi-agent Deep Reinforcement Learning for uRLLC in Vehicular OCC

This dissertation is completed by presenting the final contribution in Chapter 6, where the third research question is solved. In this chapter, the single link problem of Chapter 5 is extended to a multi-link vehicular scenario considering multiple lanes. If the multi-link problem using the proposed approach in Chapter 5 is solved, the solution will be sub-optimal. This happens because the decision for all the links is taken by observing the state of a particular link and thereby optimizing the policies for them. This could be sub-optimal for all other links most of the time because the state of the other links is unknown to the agent. In this chapter, a multi-agent DRL vehicular OCC system is proposed while considering all the possible communication links. To this aim, the communication rate is maximized subject to selecting an optimal code rate, deciding the optimal modulation scheme, and choosing the optimal relative speed of the vehicle while respecting the uRLLC constraints. The 5G NR LDPC code is used similarly to Chapter 5, which helps to achieve a higher transmission rate, ultra-reliability and low latency that requires for this case. The decisive difference between this Chapter and Chapter 5 is as follows. In this chapter, the parameters for multiple vehicular links, i.e., channel code rate, modulation scheme, and relative speed, are optimized, whereas it is done for a single link in Chapter 5. The major challenge in this work is to satisfy uRLLC conditions for all the links and optimize the decision parameters. An actor-critic based DRL framework is employed by adopting the Wolpertinger architecture for the multi-agent system in large continuous state-action spaces. The Wolpertinger architecture avoids the complexity of ex-

ploring the large action space over all the decision intervals. The model is trained using DDPG, which updates the actor-critic network parameters. The performance of the proposed multi-agent vehicular DRL scheme is evaluated for achievable rate, latency and BER. Then, the performance with different variants of the proposed scheme, a single link optimization scheme, and a scheme without coding is presented. The results illustrate that the proposed scheme achieves a better rate and average latency than other schemes under comparison. The results further demonstrate that our proposed DRL based vehicular OCC always satisfies the uRLLC constraints, whereas the other methods under comparison fail to meet most of the time. Finally, it can be concluded that uRLLC is achieved in a multi-agent DRL based vehicular OCC system. This work is planned to submit for publication in the IEEE Transactions on Vehicular Technology journal.

1.4 Organization of the Thesis

The remainder of this thesis is organized as follows. Chapter 2 gives the relevant works of the optical vehicular communication, vehicular uRLLC, DRL and channel codes. In Chapter 3, the performance analysis of vehicular OCC is presented starting from the system model toward the performance parameters modelling. Afterwards, Chapter 4 illustrates the DRL based multi-agent vehicular OCC system to maximize the sum spectral efficiency, where coding mechanism is not considered. Then, a code rate optimization scheme is used to ensure uRLLC while maximizing the goodput of the transmission link for a single vehicular link in Chapter 5. The contribution of this thesis is finalized in Chapter 6 by maximizing the communication rate for multi-link vehicular OCC systems while satisfying uRLLC by optimizing the code rate, selecting an optimal modulation scheme, and optimal relative speed. Finally, Chapter 7 outlines the concluding remarks of this dissertation and provides a discussion of future research directions.

CHAPTER 2

Related Works

2.1 Introduction

In this chapter, an overview of the literature related to this thesis is presented. This chapter is organized as follows: Section 2.2 discusses different existing technologies used in vehicular networks before presenting works related to OCC in Section 2.3. Then various approaches in vehicular networks are provided to ensure uRLLC in Section 2.4 while outlining their advantages and disadvantages. Afterwards, reinforcement learning in vehicular networks is discussed in Section 2.5. Finally, in Section 2.6, an overview of related works in channel coding is given.

2.2 Traditional Vehicular Networks

In the modern world, the number of vehicles and Roadside Unit (RSU)s (RSUs) are increasing rapidly, making the transportation system more vulnerable than ever. This results in more traffic congestion, road causalities, accidents, and less road safety. The RSU is a fixed infrastructure component that can connect with other similar components and supports vehicu-

lar communications. To cope with the current complex traffic system, a unique network is required to accumulate vehicular-system information and ensure an effective transportation system, such as Vehicular Ad-Hoc Networks (VANETs) [19], thus providing proficient communication on the road with the help of pre-established infrastructure. VANETs connect all vehicles and infrastructure within their coverage area through a wireless router or Wireless Access Point (WAP). The connection between the vehicle and the network can be lost when a vehicle moves away from the signal range of the network. As a consequence, a new free WAP is generated in the existing VANET for other vehicles outside of the network. Improving traffic safety and enhancing traffic efficiency by reducing time, cost and pollution are two major reasons behind the demand for VANETs.

Though creating greater opportunity in the transportation system at lower operational cost [20], VANETs suffer drawbacks such as lack of pure ad-hoc network architecture [21], incompatibility with personal devices [22], unreliable Internet service [23], lower service accuracy, unavailability of cloud computing [24], and cooperative operational dependency of the network. Concurrently, there are a limited number of access points for the particular networks. Several countries, e.g., the United States and Japan, have tried to implement the basic VANET architecture but not the whole system due to the lack of commercialization. This leads to demand for more reliable and market-oriented architecture for modern transportation systems [25].

There are also some vehicular localization systems, such as Global Positioning System (GPS), Light Detection and Ranging (LiDAR) for positioning or ranging applications [26]. However, GPS is not a reliable positioning technique in the vehicular environment. The LiDAR system requires more complex and heavy systems, and its deployment is costly. Also, the LiDAR system does not include any communication mechanism with the surrounding vehicles or infrastructures, and it is only used for remote sensing or building Three Dimensional (3D) image points. But, the information about

the surrounding vehicles or infrastructures is the most significant factor for next-generation intelligent autonomous vehicles and intelligent transport systems as with the case of our system.

2.3 Optical Camera Communication

In recent years, OCC have attracted attention in various camera-based applications and services, such as multimedia, security tracking, localization, broadcasting [27], and ITS [8], [28]. Since LEDs and cameras are already installed in vehicles and traffic lights, there has been rapid advancement in AV communications. OCC is a promising technology with the functionality of LoS service and LED illumination, and it has considerable superiority over existing communication technologies in the wireless domain, e.g., radio waves or single-element PDs-based communication [29], [30]). In the general OCC system, LED arrays act as transmitters that are embedded in the vehicle or on traffic lights, and the camera performs as a receiver. Cameras can build image pixels projected from various light sources within their Field of View (FoV), which helps to achieve LoS and directed communications.

Recently, various studies of the capabilities, potentials and advantages of the OCC system have already been conveyed [8], [31]. The existing works mainly targeting to increase the data rate, but they do not consider the uRLLC aspects that we study here [7, 32–34]. Based on variation in LED light intensity, a flag image was generated via communication pixels with a 10-Mbps data rate [7]. To increase the data rate, in [32], the authors proposed an optical orthogonal frequency-division multiplexing where they achieve a transmission data rate of 54 Mbps based on the IEEE802.11p standardization. In other research, the data rate was improved to 15 Mbps with 16.6-ms real-time LED detection [33]. The transmission performance was further improved to 54 Mbps with a BER $< 10^{-5}$. Recently, in [35], the authors tried to improve the BER performance by driving a close form

expression for BER in V2I applications. But they did not consider the ultra-reliability and low-latency aspects. However, the above mentioned schemes tried to enhance performance by improving the data rate, but none of them have considered the resource allocation or uRLLC performance analysis.

2.4 Ensuring uRLLC in Vehicular Networks

Ensuring low-latency and ultra-reliable communication for future wireless networks is of capital importance. To date, no work has been done on combining latency and reliability into a theoretical framework. Besides that, no wireless communication systems have been proposed for systems with latency constraints on the order of milliseconds and with system reliability requirements. We would like to emphasize that we are aiming for uRLLC in vehicular OCC, so we design the constraints to meet the reliability and latency requirements. The requirements for uRLLC vary depending on the use case; for example, ultra-reliability in terms of packet error rate can range from 10^{-5} to 10^{-9} [36] and low-latency can range from 1-10 ms [37]. In the case of vehicular communication, the required reliability is 1-10^{-5} and the latency is 3-10ms for a packet size of 300 bytes [37]. Please note that the latency requirement reduces to 1 ms for packet sizes of 32 bytes [37]. So, for large packet sizes (5 kbits in our case), maximum of 10 ms latency will be ideal.

Furthermore, the requirement to meet both latency and reliability requirements simultaneously makes the vehicular communication a very challenging problem. Hence, to cope with these issues, resource management, e.g., communication rate, latency, BER, plays a vital role in this system in order to achieve both efficiency and reliability in vehicular networks. The existing radio resource management schemes in V2V communication mainly focus on maximizing data rate. In the freeway scenario, a location-dependent uplink resource management scheme for V2V communication is proposed to maximize the sum rate of V2V links [38]. In an urban scen-

ario, to satisfy the latency and reliability requirements of V2V services, a resource block allocation and power control algorithm are proposed, taking into account the intra-cell interference [10].

For enabling uRLLC in ITSs, several methods are examined in literature, such as delay minimization [9], reliability guarantee [10], vehicle clustering [39], and excess queue length evaluation [40]. Specifically, in [9], the vehicular network transmission power is minimized by grouping vehicles into clusters and modelling reliability as queuing delay violation probability. In [10], a joint resource allocation and power control algorithm is proposed to maximize the V2V rate considering latency and reliability constraints. In [39], the authors study the impact of transmission time interval on the performance of low-latency vehicular communications. Recently, several principles for supporting uRLLC from the perspective of traditional assumptions and models applied in communication theory are discussed in [40]. In [41], the authors survey various software-defined latency control schemes in V2I networks. The work in [42] proposes different radio resource management methods for achieving low-latency vehicular communications. Recently, packet duplication was proposed to achieve high reliability in [43], [44], high availability (in an interference-free scenario) using multi-connectivity was studied in [45].

Moreover, edge computing is also considered as an attractive solution to minimize latency. This is done by processing the requested tasks locally at the edge servers, without relying on the remote servers, e.g., base stations and cloud servers [46, 47]. Reliable V2V communication and mobile edge computing with uRLLC guarantees were studied in [48]. From an ultra-reliable communication perspective, a maximum average rate was derived in [49] guaranteeing a minimum signal-to-interference ratio coverage. In [50], an edge computing framework is developed to reduce computational latency for vehicular services. Finally, a recent (high-level) uRLLC survey can be found in [40] highlighting the building principles of uRLLC.

The above schemes employ RF systems that face interference issues.

These can be solved using OCC, which is also used in this dissertation [32].

2.5 Reinforcement Learning in Vehicular Networks

Recently, deep learning has made great stride in speech recognition [51], image recognition [52], and wireless communications [53]. With deep learning techniques, reinforcement learning has shown impressive improvement in many applications, such as playing videos games [54] and playing Go games [55]. It has also been applied in resource allocation in various areas. In [56], a deep reinforcement learning framework has been developed for scheduling to satisfy different resource requirements. In [57], a DRL-based approach has been proposed for resource allocation in the cloud radio access network to save power and meet the user demands. In [14], the resource allocation problem in vehicular clouds is solved by reinforcement learning, where the resources can be dynamically provisioned to maximize long term rewards for the network and avoid myopic decision making. A deep reinforcement learning-based approach has been proposed in [58] to deal with the highly complex joint resource optimization problem in virtualized vehicular networks.

To deal with the time-varying nature of the optimization problems in vehicular applications, DRL has already been applied for solving resource allocation problems [13, 14, 59]. In [13], a deep reinforcement learning framework has been developed for spectral sharing in an RF-based centralized system, where each V2V link acts as an agent. In this paper, the agents collectively interact with the communication environment, receive a common reward, and learn to improve spectrum and power allocation through MARL. In [14], the authors address the resource provisioning problem in vehicular clouds to dynamically meet resource demands and stringent Quality of Service (QoS) requirements with minimal overhead. The authors in [59] study a transmission delay minimization problem in

software-defined vehicular networks, where the problem is formulated as partially observable MDP and solved with an online distributed learning algorithm. But these methods only optimize the spectral efficiency in RF-based systems without considering uRLLC constraints. Even though DNNs has improved the scalability of RL, training a centralized RL agent is still infeasible for large scale V2V environments. First, we need to collect all vehicle parameters in the network and feed them to the agent as the global state. This centralized state processing itself will cause high latency and failure rate in practice, and the topological information of the traffic network will be lost. Further, the joint action space of the agent grows exponentially in the number of signalized intersections. Therefore, it is efficient and natural to use MARL based problem representation, where each agent decides its policy by considering the local observation, which requires limited communication from other agents or servers. A simpler and more common alternative is independent Q-learning [60], in which each local agent learns its policy independently by modelling other agents as parts of the environment dynamics.

Since the vehicular environment is time-varying and decision making parameters, e.g., speed, distance, are continuous, general DQN cannot be applied to this system without sacrificing the performance [17]. As these discretize the state-action spaces, we may lose some state or actions, which is important for decision making. Recently, actor-critic based deep reinforcement learning frameworks are proposed to solve the continuous problem in various applications [61–63]. In [61], the authors address a power allocation problem to maximize the sum rate and ensure fairness in free-space communication. In [62], the authors propose a deep reinforcement learning-based user scheduling, phase shift control, beamforming optimization algorithm to maximize the aggregate throughput and achieve the proportional fairness while improving the trade-off between throughput and fairness in an intelligent reflecting surface. A platooning control of vehicular communication is proposed in [63]. To the best of my knowledge,

actor-critic based DRL frameworks have not been applied in vehicular OCC up to the present time.

2.6 Channel Coding

Various coding schemes are available, but most of them do not function consistently well for a vast variety of code rates and block lengths. Convolutional, turbo, polar, and LDPC codes are the four general considered coding systems [64].

Turbo codes have a variety of uses, including 3G/4G mobile communication, deep-space communications, and universal mobile telecommunications system, since they offer a lower error probability and low complexity. Because of the interleaving and iterative process, turbo codes have high decoding complexity, which induces extra latency [65]. Convolutional codes are widely used codes because of their benefits of low-complexity encoding, easy rate-adaptation ability, and hardware-friendly decoding algorithms [66]. In contrast to other coding schemes, their BER does not improve with the increase of message length, which makes them inappropriate for long-range communication. They, however, perform comparably for small message lengths using maximum-likelihood decoding [65]. Moreover, the convolutional code has a loss in rate performance [67]. Polar code being a very efficient channel code has a vast range of applications in wireless communications [68]. However, for a short block length, the redundancy to join with codes becomes higher, thus reducing the overall efficiency [69]. LDPC code is a very useful coding approach for error correction and BER reduction in communication channels. Since LDPC codes offer performance close to Shannon's, the chance of information loss is lower.

To satisfy the uRLLC requirements in vehicular OCC, effective channel coding is required in addition to the interference mitigation and DRL framework. Recently, 5G NR LDPC codes are used to provide reliability

and low latency while improving transmission rate [15]. To the best of my knowledge, LDPC codes have not been applied in vehicular OCC yet, along with the DRL framework. LDPC codes have also been tested to improve the reliability of communications using adaptive modulation schemes both in wireless [70] and optical communication [16]. These techniques use traditional optimization methods to solve the underlying optimization problem, which is inefficient in a time-varying vehicular environment because of the entailed computational complexity.

Channel coding scheme strongly affects the capacity and reliability of the system. Due to the fact that the proposed vehicular OCC system requires ultra-reliability and low-latency, 5G NR LDPC codes are embraced in this thesis, which have already been applied for adaptive modulation schemes [16]. 5G NR system uses Quasi-Cyclic (QC)-LDPC as the data channel coding scheme because of the advantages of efficient implementation and offering improved performance [71]. The QC-LDPC coded modulation can also resolve the weaknesses of having poor reliability and latency performance for arbitrary order of modulation formats [16, 72] while guaranteeing a low error rate for all code rates. A notable feature of the 5G NR LDPC codes is the flexibility to support a wide range of information block lengths ranging from 40 to 8448 bits and various code rates ranging from 1/5 to 8/9 [15, 73]. 5G NR codes use a feedback channel to adapt protection, which makes them more reliable. Therefore, we use 5G NR QC-LDPC channel coding over the Galois Field (GF)(Q) for Q-ary QAM transmissions in our vehicular OCC systems.

CHAPTER 3

Performance Analysis of Optical Camera Communications

3.1 Introduction

In this chapter, we study communication link performance with adaptive modulation scheme to examine whether OCC is suitable for employing uRLLC in automotive vehicles. To the best of our knowledge, this is the first OCC based vehicular systems that focus on uRLLC aspects. To this aim, we introduce a novel low latency V2V communications framework that ensures reliability using OCC. The proposed system is fully decentralized and each vehicle process the communicated information either individually or collectively. We provide a mathematical framework to model the OCC channel in order to find out the probability of errors, achievable spectral efficiency, and transmission latency as a function of inter-vehicular distances and AoIs while considering the adaptive modulation. To model the latency, we only consider transmission latency, as a small amount of data is processed in our system that is related to the action or safety information, and hence, the computational latency is negligible. To improve the efficiency of OCC-based communication, we use an adaptive modula-

tion scheme. By increasing the modulation order, higher spectral efficiency and lower latency can be achieved. In our evaluation, we consider satisfying the target BER as an indication of reliability in our system. If the reliability requirement for a certain modulation scheme is not met, the system can reduce the AoI at the receiver. Finally, we analyze the performance of the proposed system in terms of BER, spectral efficiency, transmission latency for various inter-vehicular distances and AoIs of LED lights at the receiver. We investigate how to achieve uRLLC by introducing a mechanism of varying the AoI at the receiver vehicle when the transmitter changes the modulation scheme depending on the size of the transmitting data.

The rest of this chapter is organized as follows: Section 3.2 outlines the vehicular OCC overview. Next, we present vehicular OCC system modelling in Section 3.3 before presenting the performance analysis of the vehicular OCC system in Section 3.4. Finally, in Section 3.5, we summarize the contribution of this chapter.

3.2 Overview of Vehicular OCC

In this section, we first discuss the advantages of OCC over other existing communication systems. Then, we illustrate the general architecture of the vehicular OCC. Finally, a overview of several existing studies on vehicular OCC is presented.

In recent years, OCC has gained new momentum as a promising complementary technology over existing communication systems, e.g., RF or PD-based communication, [29, 30]. The advantages of the license-free unlimited spectrum, longer lifespans, lower implementation cost, negligible interference, less power consumption, and enhanced security have prompted the OCC technology as a viable candidate for future wireless communication applications [7]. Also, OCC does not harm the human body or eyes and is not affected by electromagnetic interference. It is, in fact, very easy to integrate OCC to the existing vehicular systems without making any

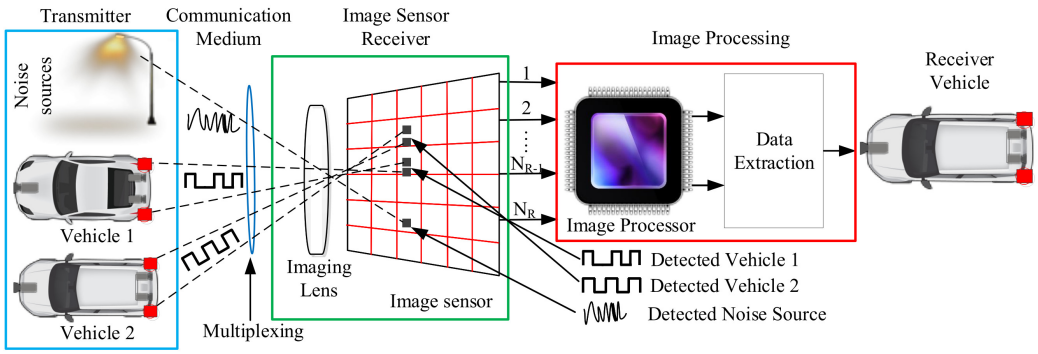

Figure 3.1: An illustration of vehicular optical camera communication operation.

significant changes. This is because LEDs already exists in vehicles, traffic lights, or road infrastructures. Besides achieving a low data rate, OCC can be a better alternative to the congested and saturated RF systems due to its negligible noise and interference and higher security [29]. However, OCC can face challenges due to its LoS requirements for communication. The speed of LED switching frequency is kept high enough so that it is not perceivable by the human eye. As a result, LED lights maintain their main purpose of illumination or indication. The camera can receive a signal which lies within its FoV. The radiated signal passes through an optical filter and a lens to ensure maximum light within the FoV of the receiver.

In general vehicular OCC architecture, LED arrays located on the rear side of a vehicle or other light sources act as transmitters, and cameras act as receivers (see Fig. 3.1). As shown in the figure, Vehicle 1 and Vehicle 2, communicate information through LED lights, which are called hereafter transmitters. Other light sources, e.g., Sunlight, ambient lights, traffic lights, and digital signages, are considered as noise sources. Meanwhile, the camera at the receiver vehicle captures the video frames within its FoV, which then passes it through an imaging lens. The captured images are fed into the image processor, which identifies the LEDs pattern from the captured images. After processing the image, the signal is passed to the data extraction unit to recover the communicated information. In vehicu-

lar OCC, no complex signal processing algorithm is required to filter out the light sources that do not convey information. The noise and data sources can easily be distinguished and captured on the image plane of the image sensor because the cameras only focus on the pixels in which the LED lights strike [33]. In this manner, interference-free and secure communications can be achieved using an image sensor.

Besides OCC, VANETs created immense opportunities in the ITS at lower operational cost [19, 20]. But, VANETs have shortcomings, such as lower accuracy, unreliable internet service, and lack of pure network architecture [25]. Alternatively, AV communication uses wireless access in vehicular environments, i.e., IEEE 802.11p standard [34]. However, OCC has several advantages over IEEE 802.11p, including unlicensed frequency spectrum access, BSs independency, and simultaneous lighting and communication. Moreover, there are several existing experimental methods to improve the performance of vehicular OCC systems. Specifically, in [7], the authors have achieved 10 Mbps data rate based on the LEDs intensity variation by generating a flag image from the communication image pixels in which the high-intensity light sources appear. In [33], the authors have proposed an image sensor based VLC system, which achieved a 20 Mbps/pixel data rate without LED detection and 15 Mbps/pixel data rate with 16.6 ms real-time LED detection. In [74], the transmission rate was improved to 45 Mbps without bit errors and to 55 Mbps with BER $< 10^{-5}$.

The above mentioned OCC schemes investigated the data rate enhancement through experimental study, and none of them examined the uRLLC aspects of vehicular OCC and optimization of system parameters, e.g., rate. To the best of our knowledge, this is the first time in vehicular OCC where uRLLC is examined.

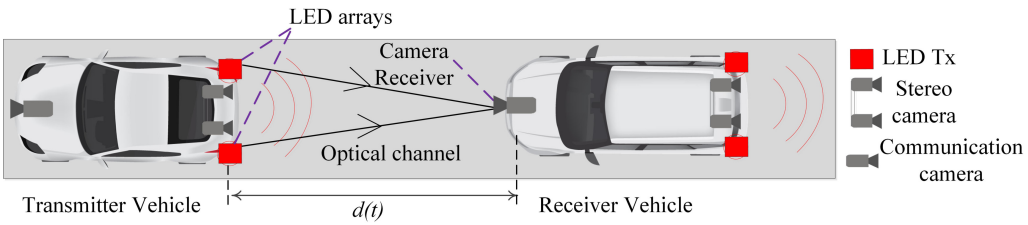

Figure 3.2: Proposed system model of vehicular optical camera communication.

3.3 System Model

In this section, we present the considered system model and parameters of vehicular OCC. Then, we specify the performance defining metrics of OCC in terms of the BER, the achievable rate, and the observed transmission latency.

3.3.1 System Modelling

Fig. 3.2 outlines the proposed vehicular OCC system model, where vehicles communicate with each other. In this scenario, the vehicle conveying information is denoted as “Transmitter Vehicle (TV)". Whereas the vehicle which follows TV and receives the transmitted information is defined as “Receiver Vehicle (RV)". In our system, the LED lights located at the back side of TV is the transmitter, and a high-speed camera (also known as image sensor and has a frame rate of 1000 frame per second (fps)) located at the front side of RV, is the receiver. If cameras of low frame rate were used, e.g., 30 fps, the data rate per pixel would be limited to 15 bits per second (bps) or less to satisfy the Nyquist frequency requirement [8], which is low for the considered applications. Therefore, high-speed cameras should be utilized in the receiver systems to achieve higher data rates or receive high-speed optical signals. The communicated information between the vehicles is mainly vehicle’s internal information, e.g., speed, next action, position,

and/or other safety and action-related information from the transmitter. We denote the distance between transmitter and receiver by $d(t)$. In our system, each vehicle has two camera sets, one in the front and another in the back. The front camera, i.e., high-speed camera, has dual functionality. Firstly, it measures forward distance, $d(t)$, between the TV and RV using the distance between the LEDs and pixel information on the image sensor, which we will discuss later in the next sub-section. Secondly, the camera acts as the receiver, which decodes transmitted data from the LED transmitters. The back camera, i.e., vision camera, measures the backward distance, $d(t)$, between the vehicles using a stereo-vision camera as the one discussed in [75].

M-ary Quadrature Amplitude Modulation (M-QAM) is used to modulate the signal in VLC [76] as it is a multi-level, high-order, and spectrally-efficient modulation technique that is relatively easy to implement and offers very low BER, high-speed, and flicker-free communication [77]. For employing M-QAM, at the transmitter, the data bit-streams are first mapped into symbols by splitting amplitude and phase into in-phase and quadrature parts, respectively. The symbol is modulated to a square wave signal with an amplitude, period and a shifting phase, i.e., period $\times$ phase$/2\pi$. A preamble is inserted with the data bits for synchronization and modulation scheme estimation at the receiver. The resulting signal is then transmitted through the optical channel by modulating intensity of the LEDs. At the receiver, the camera captures the modulated light waveform within its exposure time. During this time, the image sensor captures the intensity of the light coming in as different LED states, e.g., on, off, mid. The camera integrates the signal during its exposure time, which is recorded as the pixels of the image. We can extract the original signal information from this detected intensity in these pixels using an efficient M-QAM demodulation scheme [78]. At the receiver, frame synchronization, modulation scheme estimation, and post-equalization are carried out to eliminate the effect of the channel with the help of the preamble. In [78], a simple mathemat-

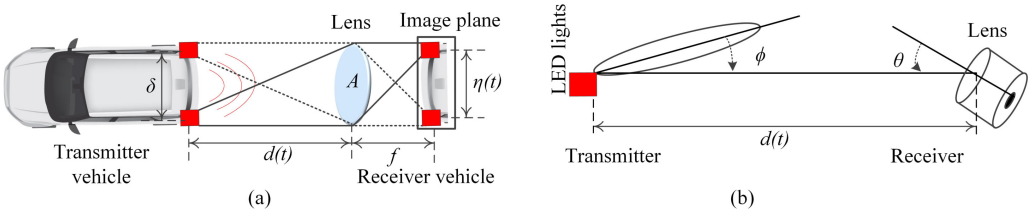

Figure 3.3: (a) LoS channel model of OCC and (b) Inter-vehicular distance measurement [7].

ical formulation for encoding and decoding of the amplitude and phase of transmitted symbols is proposed, where the modulated symbol is sampled in three consecutive frames by the image sensor. From the captured frames, the LED states, e.g., on, off, mid, are identified, and a lookup table is developed. Then, the phase position is retrieved using the lookup table, and the reconstructed phase is converted to radian so that it can be mapped to M-QAM. Finally, the original signal is perfectly recovered from the detected amplitude and phase.

3.3.2 Optical Channel Model

We assume an uninterrupted LoS link between the transmitter and camera receiver, ensuring the vehicles are free from obstruction to communicate with each other continuously. Depending on the channel conditions, OCC has either a flat-fading or diffuse channel. Generally, OCC channel has two types of light propagation components: (i) LoS component resulting from direct light propagation from the transmitter to the receiver and (ii) diffuse components resulting from the reflected lights from other vehicles or reflective surfaces. Usually, the diffuse propagation has much lower energy than the LoS component, and therefore, the diffuse light component is neglected in this thesis. Accordingly, considering the LOS channel, the Direct Current (DC) channel gain, $H(\theta, t)$, between the transmitter and the receiver is given by [79]

$$H(\theta,t)=\begin{cases}\frac{A_{\text{eff}}(\theta)}{d^2(t)}\,\Re(\phi), & 0\leq\theta\leq\theta_l\\ 0, & \theta>\theta_l\end{cases} \tag{3.1}$$

where $A_{\text{eff}}(\theta)$ is the effective signal collection area of the image sensor, θ is the AoI with respect to the receiver axis, ϕ is the angle of irradiance with respect to the emitter, $\Re(\phi)$ is the transmitter radiant intensity, θ_l denotes the FoV of the image sensor lens, and finally, t is the time-frame index. The $d(t)$ is expressed as [7]:

$$d(t)=\frac{f}{a}\cdot\frac{\delta}{\eta(t)}, \tag{3.2}$$

where δ is the distance between the left and right LED array units, f is the lens focal length, $\eta(t)$ is the distance in terms of number of pixels between the left and right LED array units on the captured image, and a is the image pixel size. The inter-relation between the distance measurement parameters is illustrated in Fig. 3.3(a). The backward distance can be estimated with a stereo vision camera using a similar method to the one in [75], which is presented in Appendix A.

Regarding the above parameters: δ is sent from TV to RV through LEDs, and f and a are known values for any system, such as 15 mm and 7.5 μm, respectively, as we considered in this work [7]. The value of $\eta(t)$ can be obtained via simple image processing techniques or by calculating the pixel values using data pointer. In this way, using both the received data and the captured image, the RV can estimate the inter-vehicular distance, $d(t)$ and the channel gain. Please note that positioning accuracy is not emphasized in the literature because the focus is to ensure communication quality. The positioning error is estimated to be 10 cm [7].

We assume that the LED follows a Lambertian radiation pattern and has wider directivity. Therefore, the light emission from the LED transmitters can be modelled using a generalized Lambertian radiant intensity [79, 80]

and following the link geometry, as shown in Fig. 3.3(b):

$$\Re(\phi) = \frac{(m+1)}{2\pi} \cos^m(\phi), \tag{3.3}$$

where m is the order of Lambertian radiation pattern, which is derived from the LED semi-angle at half luminance, $\Phi_{1/2}$, as

$$m = \frac{-\ln(2)}{\ln(\cos(\Phi_{1/2}))}. \tag{3.4}$$

Also, $A_{\text{eff}}(\theta)$ can be expressed as [79]

$$A_{\text{eff}}(\theta) = \begin{cases} A\, T_s(\theta)\, g \cos(\theta), & 0 \leq \theta \leq \theta_l \\ 0, & \theta > \theta_l \end{cases} \tag{3.5}$$

where A is the area of the entrance pupil of the camera lens, $T_s(\theta)$ is the signal transmittance of the optical filter, and g is the gain of the lens. An ideal lens has a gain: $g = n^2/\sin^2(\theta_l)$, where n is the internal refractive index of the lens.

Based on the above definitions and considering (3.3) and (3.5), finally, $H(\theta, t)$ can be formulated as

$$H(\theta, t) = \begin{cases} \frac{(m+1)A}{2\pi d^2(t)} \cos^m(\phi)\, T_s(\theta)\, g \cos(\theta), & 0 \leq \theta \leq \theta_l \\ 0. & \theta > \theta_l \end{cases} \tag{3.6}$$

From (3.6), we observe that if A and g are fixed for an image sensor, the channel power gain $H(\theta, t)$ can be increased by either (a) decreasing the distance, $d(t)$ and/or (b) increasing the signal collection area, i.e., by decreasing the AoI of the camera lens. Lower AoI of the camera lens means the strength of light beam will be stronger on the image sensor, which in turn, will increase the channel power gain. Alternatively, higher AoI reduces the $H(\theta, t)$ as the LED light beam will spread out at the wide angle of the camera lens. So, maintaining narrower AoI at the receiver will provide improved performance because of having higher gain.

Finally, the received optical power $P_{\mathrm{r}}(\theta, t)$ can be derived using the optical transmitted power P from LED lights

$$P_{\mathrm{r}}(\theta, t) = P \cdot H(\theta, t). \tag{3.7}$$

I would like to note that in this work, I neglect the signal detection overhead of recognizing the desired light sources under mobile scenarios. This is motivated by [30], where the authors have proposed a statistical vehicle motion model in an image plane and showed that the vehicle motion along the vertical and horizontal axes of the image plane is limited to within one pixel in most cases, which is very small compared to entire image pixels on the captured image. Moreover, the DC gain, and as a result, the Signal-to-Noise Ratio (SNR) at a pixel remains constant as long as the projected image of the transmitter LED occupies several pixels. Further, a simple design of an LED detection and tracking system is proposed using the result and the vehicle motion model of [30], which limits the tracking area of the VLC transmitter and reduces the computational cost. Thus, the vehicle motion and the pixel illumination model is used as a guideline for our system to overlook the overhead of recognizing the desired light sources for the mobile environment.

3.3.3 Parameters Modelling

In order to analyze the system performance, we first formulate the SNR of the optical link.[1] We consider SNR as a measure of communication link quality of the signal transmission. Therefore, according to [81], the received SNR, $\gamma(\theta, d)$ of visible light link can be expressed by

$$\gamma(\theta, d) = \frac{(\rho P_{\mathrm{r}}(\theta, d))^2}{\sigma(d)} \tag{3.8}$$

[1]From (3.6), we see that the channel gain depends on θ and t, where t represents the changes in inter-vehicular distance, i.e., d over time. So, the only changing variable is the distance, and from now we can drop the variable t by only keeping d. Therefore, for formulating the SNR, we use d by leaving t.

where $\sigma(d)$ represents the total noise power, which can be computed as in [79]

$$\sigma(d) = q\rho P_n A(d) W_{\text{fps}}, \tag{3.9}$$

where q is the electron charge, P_n is the power in background light per unit area, and W_{fps} is the sampling rate of the camera in fps.

Regarding the calculation of $A(d)$: Points at different distances are imaged as a little circle on the image plane. Hence, a LED occupies a circle having a diameter [82], $l' = \frac{fl}{d}$, where l is the diameter of a LED and f is the focal length. To conservatively account for the quantization effects, measurements are commonly made at the nodes of a square grid of points. This means that, the LED will occupy a square area of size l'^2. When the LED moves away from the camera, the projected diameter l' will eventually become smaller than the size of a photodiode. We refer to the distance where the LED generates an image that falls onto exactly one pixel as the critical distance $d_c = fl/s$, where s is the edge-length of a pixel.

So, (3.9) can be rewritten as

$$\sigma(d) = \begin{cases} q\rho P_n W \frac{f^2 l^2}{d^2}; & \text{if } \ d < d_c \ , \\ q\rho P_n W s^2; & \text{if } \ d \geq d_c \ . \end{cases} \tag{3.10}$$

Finally, from (3.7) and (3.10), (3.8) can be summarized as,

$$\gamma(d) = \begin{cases} \frac{\rho k^2 P^2}{q P_n W f^2 l^2 d^2}; & \text{if } \ d < d_c \ , \\ \frac{\rho k^2 P^2}{q P_n W s^2 d^4}; & \text{if } \ d \geq d_c \ . \end{cases} \tag{3.11}$$

where $k = \frac{(m+1)A}{2\pi} \ \cos^m(\phi) \ T_{\text{s}}(\theta) \ g \ \cos(\theta)$.

Motivated by the trade-off among modulation order, achieved BER, and improved spectral efficiency, we consider adaptive modulation that permits us to adapt the modulation order by satisfying the target BER requirement of the system. The adaptive scheme can deal with the time-varying nature of the channel while maintaining the desired link quality and maximizing the rate for the given channel conditions [83]. Furthermore, the adaptive

modulation scheme transmits at high-speed under favourable channel conditions, and the rate decreases when the channel conditions worsen. It is worth noting that different users might have a different rate since they do not have precisely the same SNR and, consequently, they provide varying BERs. However, in practice, a particular discrete modulation set is used to examine how the performance varies after limiting the system to a small modulation set. For the considered system, we study Binary Phase-Shift Keying (BPSK) and uncoded M-QAM with the square constellation as an example because they offer higher spectral efficiency, low BER and easy implementation. Still, our scheme is general and other modulation schemes can also be employed. The BER of the optical wireless channel at the receiver using the BPSK and M-QAM scheme is evaluated similarly to [84] and [85]as:

$$\text{BER}(\theta, d) = \begin{cases} Q\left(\sqrt{2\gamma(\theta, d)}\right), & \text{for BPSK} \\ \frac{4}{\log_2(M(\theta,d))} \cdot Q\left(\sqrt{\frac{3\ \gamma(\theta,d)\ \log_2(M(\theta,d))}{M(\theta,d)-1}}\right), & \text{for M-QAM} \end{cases} \tag{3.12}$$

where M is the constellation size and $Q(x) = \frac{1}{2}\ \text{erfc}\left(\frac{x}{\sqrt{2}}\right)$ stands for the Q-function. So, the spectral efficiency of the BPSK and M-QAM modulation schemes can be expressed as $\text{SE}_{\text{BPSK}} = 1$ and $\text{SE}_{\text{M-QAM}} = \log_2(M(\theta, d))$, respectively.

It is worth noting that the adjustment of modulation depends on the road scenarios. At normal conditions, when there is nothing to communicate, RV maintains the wider AoI to understand the whole scenario of the road. If the TV wants to transmit any critical information, it chooses a higher modulation based on the size of the transmitted data. On the receiver side, if the RV notices any sudden change in the TV transmitted signal and fails to decode it using the current modulation scheme, the RV switches to another modulation from the chosen limited modulation set. If there are consecutive failures, the receiver employs the closest possible modulation. In the meantime, RV decreases the AoI of the camera lens to focus on the LED transmitter and decodes the transmitted signal within the

shortest possible time. The adaptive modulation in our system is adjusted as follows. Suppose there is any change in the modulation scheme during communication. In that case, the transmitter informs the receiver regarding the employed modulation by appending a small overhead, e.g., some extra bits, in each transmitted packet. This overhead can be neglected because, in practice, as a small set of modulation scheme is used, e.g., 6, in our system. We require only 3 bits to be appended in the transmitted data for the receiver. Hence, the overhead will be minimal compared to the transmitted packet size, i.e., 5 kbits, in our system.

We should note that the transmission rate (measured in bit per second (bps)) of a camera based-communication system depends on the received SNR as shown in [32] and is given by

$$C(\theta, d) = \frac{W_{\text{fps}}}{3} W_{\text{s}}(d) \cdot \log_2(M(\theta, d)), \tag{3.13}$$

where $W_{\text{s}}(d)$ is the spatial-bandwidth, which denotes the number of information carrying pixels per image frame. The term $W_{\text{fps}}/3$ refers to the fact that the camera must sample the modulated signal three times of the sampling frames to decode the original M-QAM signal [78]. In other words, for reconstructing the amplitude and phase perfectly, a modulated symbol is sampled in three consecutive frames. Finally, $W_{\text{s}}(d)$ is defined as

$$W_{\text{s}}(d) = N_{\text{LEDs}} \cdot N_{\text{row}}(d), \tag{3.14}$$

where N_{LEDs} is the number of LEDs at each row of the transmitter and $N_{\text{row}}(d)$ represents the captured number of row pixel lines in each frame. Considering a rolling shutter camera, the actual number of samples, $N_{\text{row}}(d)$ acquired from the captured image at d can be expressed as

$$N_{\text{row}}(d) = w \, \frac{\varrho}{2 \tan\left(\frac{\theta_l}{2}\right) d}, \tag{3.15}$$

where w is the image resolution and ϱ is the normalized length (diameter) of the LEDs along the width. Taking into account (3.14) and (3.15), $C(\theta, d)$

is re-written as:

$$\begin{aligned} C(\theta, d) =& \frac{W_{\text{fps}}\ N_{\text{LEDs}}\ w\varrho}{6\ \tan\left(\frac{\theta_l}{2}\right) d} \cdot \log_2(M(\theta, d)) \\ =& \frac{l_0}{d} \cdot \log_2(M(\theta, d)), \end{aligned} \tag{3.16}$$

where $l_0 = \frac{W_{\text{fps}}\ N_{\text{LEDs}}\ w\varrho}{6\ \tan\left(\frac{\theta_l}{2}\right)}$. Considering the communications between the vehicles, the overall end-to-end latency, $\tau(\theta, d)$ can be found as [46]

$$\tau(\theta, d) = \frac{L}{C(\theta, d)}, \tag{3.17}$$

where L is the packet size in bits. Recall that in our system, we consider that the end-to-end latency is dominated by transmission latency, and therefore, we neglect the computational latency. This is because we process a small amount of data, i.e., the decision information from TVs to the RVs, and hence, the computational time will be short.

Since, the goal of the system is to avoid critical conditions, i.e., avoid collisions between vehicles, a minimum distance has to be maintained between the vehicles so that the collision between the vehicles can be avoided. However, by increasing the distance between the vehicles, the quality of the communication deteriorates. Specifically, increasing distance beyond a threshold would lead uRLLC conditions to be violated. So, in order to maintain uRLLC, we can vary the modulation order at the transmitter depending on the size of the transmitting data and the AoIs at the RV to satisfy the target BER. In our system, we analyze the performance of our proposed OCC system by varying both the inter-vehicular distances and AoIs. However, as it is challenging to change AoI sharply in a realistic scenario, which would introduce additional delays in changing the AoI inside the vehicle mechanically, in the next section, we consider distance as the only free variable and fix AoI in a value that guarantees system requirements.

Table 3.1: Simulation Parameters

Parameter, Notation	Value
Angle of irradiance w.r.t. the emitter, ϕ	70^o [79]
Semi-angle at half luminance of the LED, $\Phi_{1/2}$	60^o [79]
Inter-vehicular distance, d	$(0 - 150)$ m
AoI w.r.t. the receiver axis, θ	0^o to 90^o
FoV of the camera lens, θ_l	90^o [79]
Image sensor physical area, A	10 cm^2 [79]
Transmission efficiency of the optical filter, T_s	1 [79]
Refractive index of concentrator/lens, n	1.5 [79]
Constellation size, M	4, 8, 16, 32, 64
Camera-frame rate, W_{fps}	1000 fps [7]
Optical transmitting power, P	1.2 Watts
Number of LEDs in the transmitter, N_{LEDs}	300 (30 $\times$ 10) [7]
Electron charge, q	1.6×10^{-19} C
Focal length of the camera lens, f	15 mm [7]
Image pixel size, a	7.5 μm [7]
Distance between left and right LED array, δ	50 cm [7]
Size of the LED, ϱ	15.5 $\times$ 5.5 cm^2 [7]
Resolution of image, w	512 $\times$ 512 pixels [7]

3.4 Simulation Results and Performance Analysis

In this section, simulations are conducted to investigate the performance of the proposed system model in vehicular OCC. We start by evaluating different performance metrics of the proposed system model to get a better understanding of the interplay among the various parameters of our system. We consider the vehicular OCC system as described in Section 3.3.1 and 3.3.2. We demonstrate the performance for distances up to 150 m, which

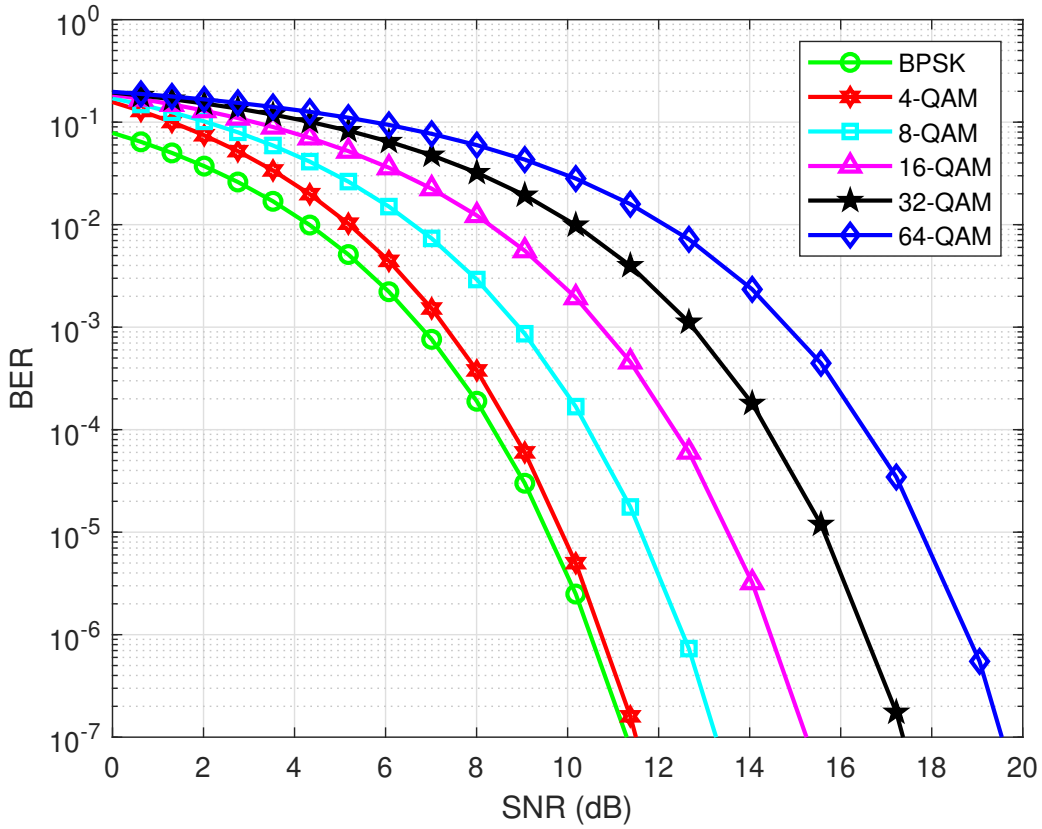

Figure 3.4: BER versus SNR (dB) for various modulation schemes considering AoI = 60^0 and fixed transmit power at 1.2 W, when d is varying.

we believe is sufficient to maintain communication and avoid collisions. We chose an AoI range of 0-90 to accommodate the FoV of the camera lens. We employ the OCC modelling parameter described in [7,79] as presented in Table 3.1. We propose an adaptive modulation scheme using BPSK and M-QAM with five different constellations, $M = \{4, 8, 16, 32, 64\}$ as example, still, other modulation schemes can also be employed. Here, we consider a transmitter size of 300 (30×10) LEDs with 0.5 cm spacing between each LED and a 1000 fps camera for the receiver, where the resolution of the received image is 512×512. Target BER is set to 10^{-4} and 10^{-5} for performance comparison to be compliant with uRLLC requirements. The rest of the simulation-related parameters are summarized in Table 3.1.

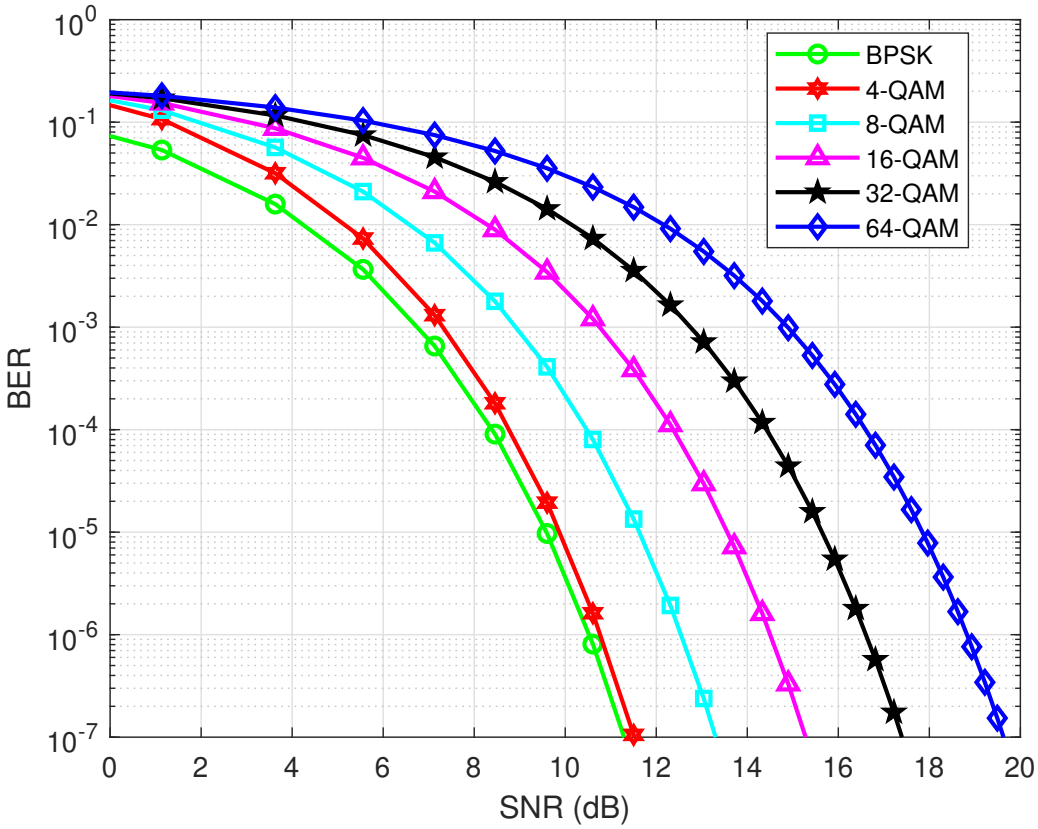

Figure 3.5: BER versus SNR (dB) for various modulation schemes considering d = 50 m and fixed transmit power at 1.2 W, when AoI is varying.

3.4.1 Performance of BER Modelling

In this sub-section and the following sub-section, we analyze the performance of the proposed system model for BER, spectral efficiency, and latency at different inter-vehicular distances and AoIs. We start by comparing the BER versus SNR (dB) for the chosen modulation set with a fixed transmit power of 1.2 Watt. The results for different inter-vehicular distances and AoIs are illustrated in Fig. 3.4 and Fig. 3.5, respectively. The plots demonstrate that we achieve better BER performance at higher-order modulation, but this comes at the cost of higher SNR level. In our evaluation, we do not vary the distance and AoI at the same time. While varying distance, we change it from 0 m to 150 m by keeping the AoI at 60^{o}, similarly we vary

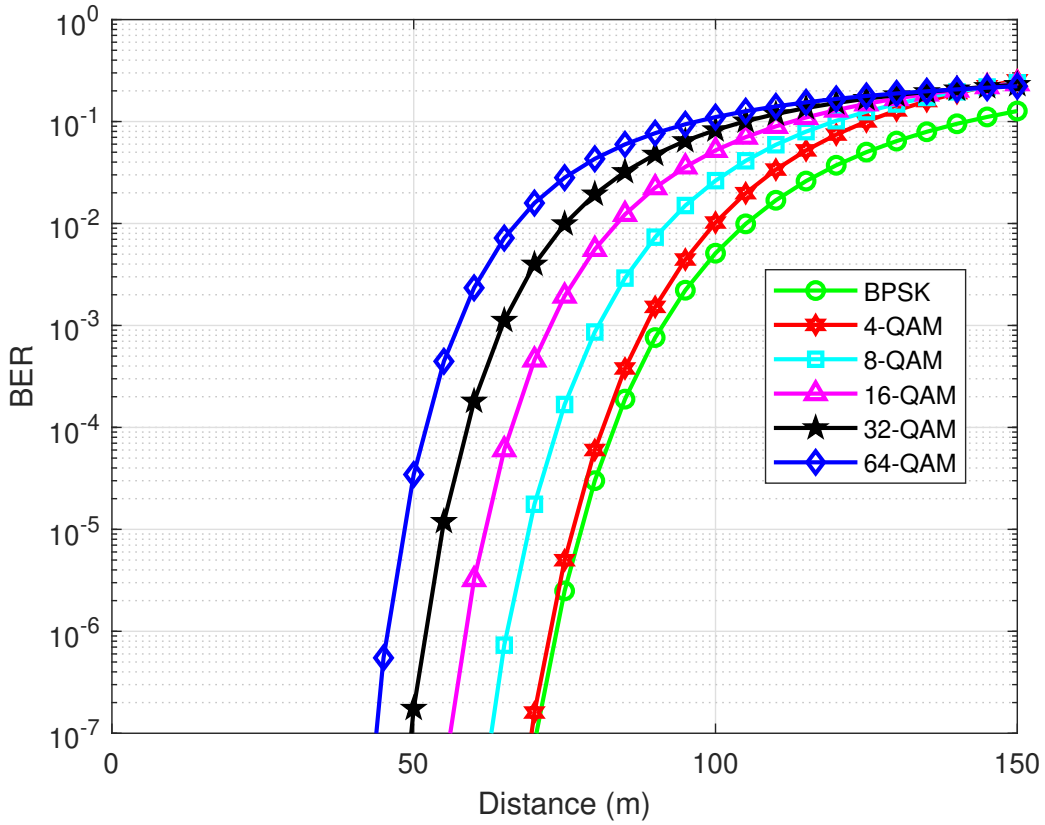

Figure 3.6: BER versus Distance (m) for various modulation schemes considering AoI = 60^0 and fixed transmit power at 1.2 W.

AoIs between 0^o to 90^o by keeping the distance to 50 m. In this manner, we justify that the same BER performance can be achieved at various distances and AoIs while using different modulation schemes.

In Fig. 3.6 and Fig. 3.7, we evaluate the achieved BER performance for the different modulation schemes at varying distances and AoIs, respectively. Fig. 3.6 shows that BPSK satisfies target BER (10^{-4}) up to 82 m, and for 64-QAM, it is satisfied at 52 m. Similarly, in Fig. 3.7, target BER (10^{-4}) is satisfied at 80^o and 62^o for BPSK and 64-QAM, respectively. The plots confirm that at a shorter distance and narrower AoI, the modulation order will be higher due to higher SNR at the receiver. This is due at the narrower AoI, the strength of the light beam on the image sensor is strong,

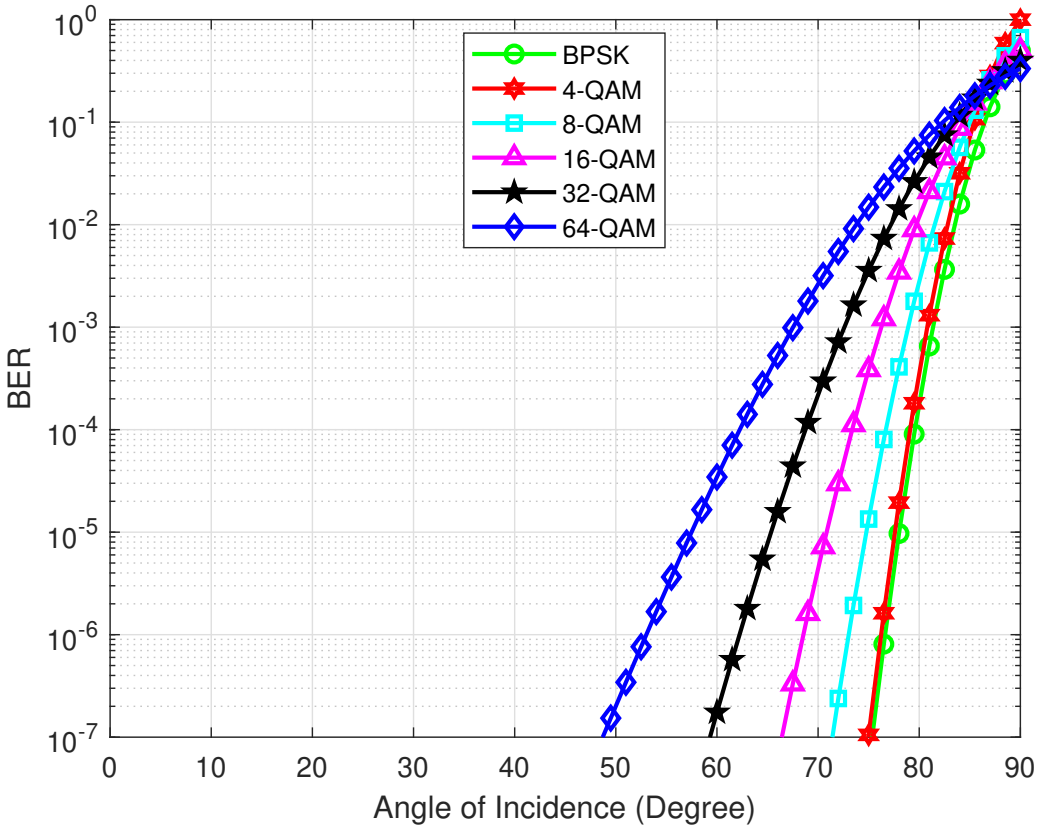

Figure 3.7: BER versus AoI (Degree) for M-QAM scheme considering d = 50 m and fixed transmit power at 1.2 W.

which increases channel gain. Alternatively, at the shorter distance, the SNR gets higher. Thus, the target BER can be achieved while maintaining the trade-off between the modulation order and distances or AoIs.

3.4.2 Spectral Efficiency and Latency Performance

The achieved spectral efficiency and observed latency improvements of the proposed system are presented in Fig. 3.8 for various distance values. In this evaluation, we consider, 10^{-4} and 10^{-5}, as the target BER for performance comparison. We determine the distance that satisfies the target BER, and then adopt the highest modulation scheme from the available schemes using Fig. 3.6. Then, we calculate spectral efficiency at that correspond-

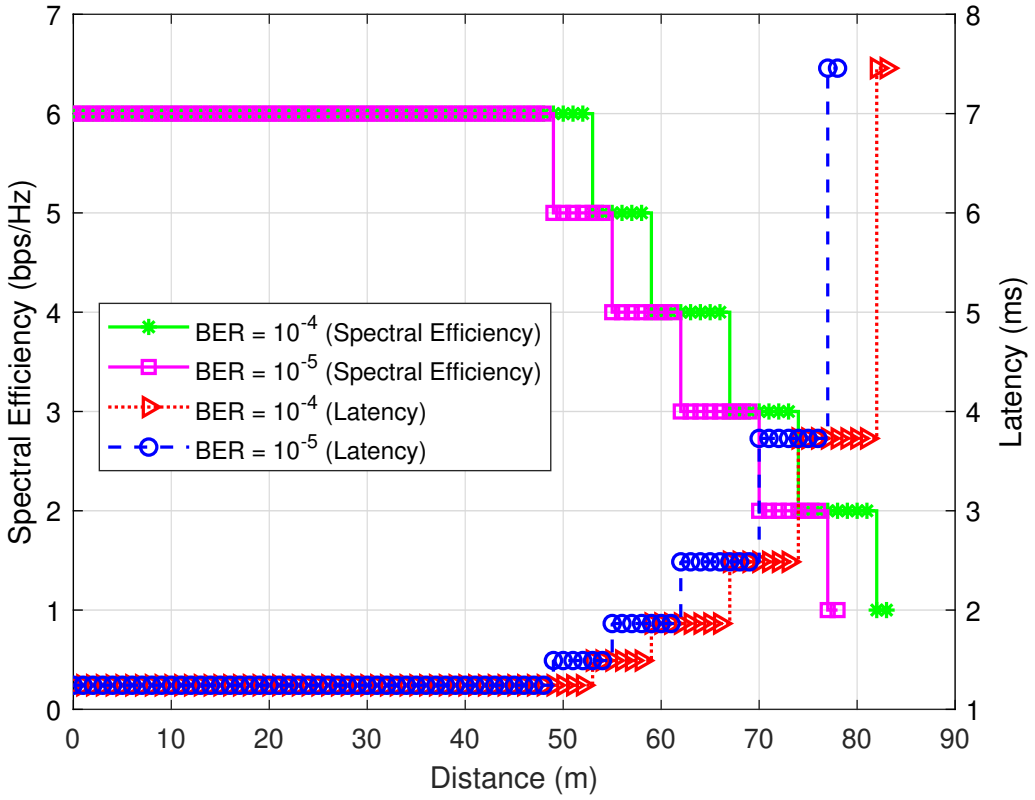

Figure 3.8: Spectral efficiency and latency versus distance at target BER of 10^{-4} and 10^{-5}.

ing modulation scheme and distance. We achieve spectral efficiency of 6 bps/Hz (Fig. 3.8) for distance until 48 m (for BER = 10^{-5}) and 52 m (for BER = 10^{-4}) using 64-QAM. Likewise, we notice a spectral efficiency of 2 bps/Hz from 74 m to 81 m and 69 m to 76 m at the target BER of 10^{-4} and 10^{-5}, respectively. Please note that the above evaluation is ideal since the modulation level is perfectly adapted, and the target BER is known to both the transmitter and receiver in advance. As a result, the transmitter and receiver can choose the modulation order and target BER from the predefined sets.

For latency evaluation, we first calculate the achievable rate using (3.16), considering w as 512×512 pixels. Then, we compute the transmission latency for a packet size of 5 kbits using (3.17). Here, we consider trans-

mission latency to be equal to the end-to-end latency because we process small amount of data in our system. The results are presented in Fig. 3.8 for distances from 0 m to 90 m and two different target BERs, i.e., 10^{-4} and 10^{-5}. This evaluation shows that our system can achieve the latency of 1ms at 52 m and 48 m at target BER of 10^{-4} and 10^{-5}, respectively. From Fig. 3.8, it can be seen that we gain 1ms latency and 6 bps/Hz spectral efficiency using 64-QAM at constant power. Therefore, we note that both latency and BER requirements are satisfied at 60^0. As a result, we consider AoI as 60^0 for our optimization problem formulation to deal with the complexity of changing AoI in practice and its induced latency.

From Fig. 3.8, we can conclude that the use of adaptive modulation offers higher spectral efficiency and lower latency. Whereas a single modulation scheme offers fixed-rate and latency having limitations in distance coverage and BER requirements. For example, 64-QAM can satisfy a target BER of 10^{-4} and a latency of 1.2 ms up to 52 m. Thus, beyond this distance, we need another modulation scheme for satisfying the target BER, i.e., 32-QAM, 16-QAM, and so on. Similarly, BPSK meets the target BER of 10^{-4} up to 83 m but offers a lower rate, i.e., 1 bps/Hz, and higher latency of 7.5 ms. Thus, it can be said that adaptive modulation provides better performance while satisfying the trade-off between BER and latency requirements.

3.5 Summary

In this chapter, the performance of adaptive modulation has been analyzed for automotive vehicular uRLLC considering OCC. The latency is modelled based on the capacity of the vehicular OCC while considering the transmission latency only. Further, the BER performance is studied for various sets of the AoI and inter-vehicular distance. In our system, the spectral efficiency of vehicular OCC is adjusted adaptively using adaptive modulation which ensures reliability by maintaining the BER to a pre-determined target value. We carried out simulations to get an understanding of how

to adjust the employed modulation scheme as well as AoIs so that it meets the BER requirements. Interestingly, the proposed model provides about 7 ms latency while satisfying the reliability requirement of 10^{-4} or 10^{-5} when the AoI is varied between 0^o to 90^o.

CHAPTER 4

Multi-Agent Deep Reinforcement Learning for Spectral Efficiency Optimization in Vehicular OCC

4.1 Introduction

Following the performance analysis in Chapter 3, we justified that OCC can satisfy BER and low latency requirements in AVs. Therefore, we utilize OCC in multi vehicular networks to maximize the communication rate while meeting the BER and latency requirements. One of the major challenges in vehicular networks arises from the fact that they are time-varying and highly dynamic, while the vast amounts of generated data (each vehicle can generate up to 750 Mbps, i.e., sensors data, videos or images data from cameras) should be delivered reliably within stringent time constraints for ensuring safety. Various technologies have been proposed in recent years to ensure reliability and low latency in ITS using traditional optimization schemes, such as [9, 10], reflect on delay minimization and reliability guarantee. However, ensuring reliability and low latency have been challenging due to the complexity of the system. The complexity arises when it

involves decision-making in controlling different parameters, e.g., speed, distance and modulation schemes. Using traditional distributed methods, it is difficult to solve these decision-making problems because of the inherent complexity and the time needed for solving them. Fortunately, RL methods can serve as an effective alternative solution to overcome the complexity of such systems [11] due to the fact that it is possible to be applied distributively.

In this chapter, we propose an OCC-based vehicular communication system to maximize the communication rate by satisfying the latency requirements while respecting BER. The studied problem can be modelled as a MDP. Despite MDP providing an efficient way to express our framework, traditionally used methods to solve them, like value-iteration, require the knowledge of the state-action transition probability matrix that is difficult to be obtained in dynamic problems such as the one we examine in this chapter. These limitations are overcome through Q-Learning [11]. However, Q-Learning has slow convergence and cannot solve large-scale problems. To address this limitation of the Q-Learning algorithm, we use the DRL [12]. DRL approximates the state-action value function by adjusting the weights of a neural network.

Even though DRL has improved the scalability of RL, training a centralized RL agent is still infeasible for large-scale V2V environments as the one considered in this chapter. This is due to the fact that we need to collect all the observation states from the vehicular network and communicate them to an agent (e.g., base station), which optimizes the policies of all the vehicles centrally. After determining the policies, the central agent should communicate them to the vehicles. This centralized decision-making is problematic as it causes higher latencies due to communicating data back and forth, it worsens congestion in the network, and it may lead to finding inefficient policies in particular when the information is lost or delayed. To avoid the above problems, we formulate the problem as a MARL, where each agent considers only local observations and does not require global

communication. In particular, we adopt independent Q-Learning [60], in which each local agent learns its policy independently by modelling other agents as part of the environment. It has been shown that independent Q-Learning can lead to well-performing solutions though there are no theoretical guarantees [86].

To the best of our knowledge, optimizing the performance of vehicular OCC employing DRL has not been investigated in the literature. In this chapter, we propose a spectral efficiency maximization scheme in vehicular OCC that satisfies BER and latency constraints. In doing so, we determine the optimal modulation order and speed of the vehicles using DRL. We consider a decentralized, independent and MARL scheme in solving this problem.

The major contributions of this chapter are summarized as follows:

- We propose a multi-vehicular spectral efficiency maximization scheme based on independent deep reinforcement learning in vehicular OCC;
- We formulate the maximization problem subject to BER, latency and a small set of modulation schemes constraints. As the optimization function is a NP hard problem leading to a difficult search for the optimal solution, we model the problem as an MDP, where the reward function is designed to satisfy users' requirements;
- We relax the constrained problem into an unconstrained one using the Lagrangian relaxation method, which essentially simplifies the solution of the complex problem. We then solve the spectral efficiency maximization problem using deep Q-Learning;
- We evaluate the performance of the proposed DRL-based optimization scheme through simulations. The results demonstrate that DRL-based optimization algorithm can effectively learn to maximize the spectral efficiency while meeting the constraints. Further, the results show that our scheme outperforms significantly methods based on other communication technologies.

Chapter 3 is organized as follows. We present the vehicular OCC system model and the proposed problem formulation in Section 4.2. Section 4.3 outlines the RL based MDP formulation and solution to the multi-agent problem using deep Q-Learning. The simulation set up and training procedure of our proposed DRL algorithm is explained in Section 4.4. Section 4.5 provides the simulation results using the proposed DRL-based optimization scheme. Finally, concluding remarks are drawn in Section 4.6.

4.2 System Model and Problem Formulation

In this section, we first present the considered vehicular OCC system model. Then, we specify the performance defining metrics of OCC in terms of the BER, the achievable rate, and the observed transmission latency. Finally, we formulate a sum spectral efficiency maximization problem setting BER and latency constraints to a predefined target value while adjusting the constellation size.

4.2.1 System Model

We consider a vehicular OCC system as shown in Fig. 4.1, where each vehicle is an individual agent. Let B be the number of V2V links at the back of each vehicle, where $\mathcal{B} = \{1, 2, \cdots B\}$ is the set of V2V links. We express the distance with the backward vehicles as d^b where $b \in \mathcal{B}$ represents the index of the backward V2V link.

Our system employs an adaptive modulation scheme that consists of M-QAM and Time Division Multiple Access (TDMA). The transmitter contains arrays of LEDs, which transmit at a different rate for different users under the adaptive modulation scheme. To support transmission at different modulation orders for different backward vehicles (link), we introduce TDMA in our system, similar to [87], where specific time slots are assigned to each vehicular link at the back. In this way, different time slots are allocated to each V2V link for either transmission or reception. However, since

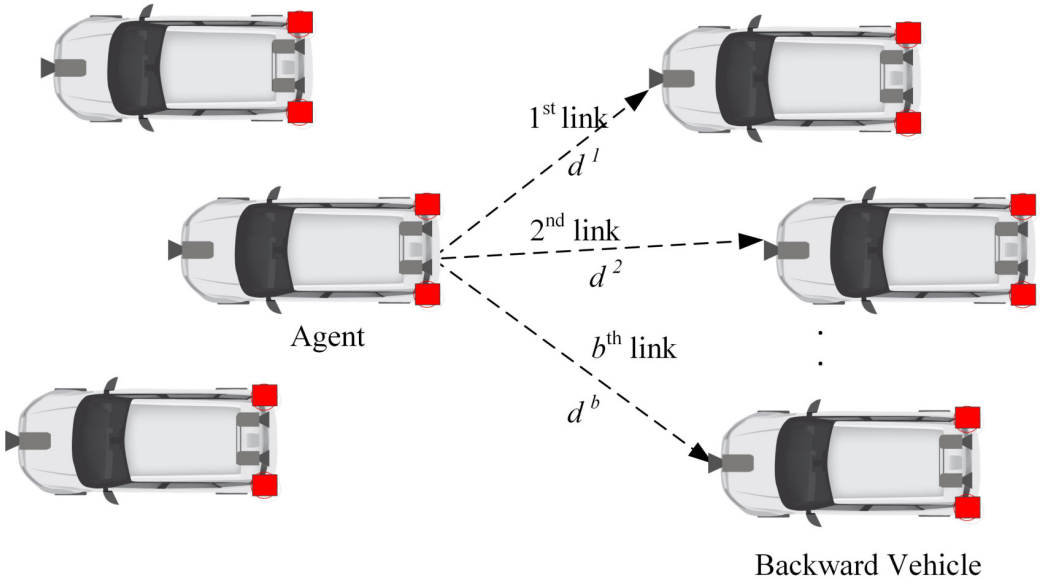

Figure 4.1: Proposed system model for vehicular optical camera communication.

each link of the vehicle transmits information only at specific times, the sum spectral efficiency is divided by the number of available vehicles, B, at the back.

4.2.2 Optical Channel Model

We considering the LoS channel between transmitter and receiver, according to [79,88] and (3.11), the received SNR, γ_t^b, of the link b at time t for a single LED-camera communication can be expressed as[1]

$$\gamma^b = \begin{cases} \frac{\rho k^2 P^2}{q P_n W f^2 l^2 (d^b)^2}; & \text{if } d^b < d_c \ , \\ \frac{\rho k^2 P^2}{q P_n W s^2 (d^b)^4}; & \text{if } d^b \geq d_c \ . \end{cases} \tag{4.1}$$

Motivated by the trade-off among modulation order, achieved BER, and improved spectral efficiency, we consider adaptive modulation that permits

[1]For notational simplicity, we drop t from the notation in the remainder of the chapter unless it is necessary; hence, we will adopt γ^b instead of γ_t^b and so on. Also, it is clear from the context that distance is our working variable.

us to adapt the modulation order by satisfying the target BER requirement of the system. In this considered system, we study uncoded M-QAM with the square constellation as an example. Still, our scheme is general and other modulation schemes can also be employed. The BER of the optical wireless channel at the receiver using the M-QAM scheme is evaluated similarly to [89] as:

$$\text{BER}^b = \frac{2\left(\sqrt{M^b}-1\right)}{\sqrt{M^b}\log_2(M^b)}\ \text{erfc}\left(\sqrt{\frac{3\ \gamma^b\ \log_2(M^b)}{2\left(M^b-1\right)}}\right), \tag{4.2}$$

where M^b is the available constellation points for each V2V link b, e.g., $M = 4, 8, 16, \cdots$ and erfc$(\cdot)$ is the complementary error function.

For a given M^b, the spectral efficiency of the M-QAM scheme can be expressed as:

$$\text{SE}^b = \log_2(M^b). \tag{4.3}$$

The channel capacity of a camera-based communication system depends on the employed modulation scheme as has been shown in [32] where the transmission rate of link b is expressed as

$$C^b = \frac{(W_{\text{fps}}/3)N_{\text{LEDs}}w\varrho}{2\ \tan\left(\frac{\theta_l}{2}\right)\cdot d^b}\cdot\log_2(M^b), \tag{4.4}$$

where N_{LEDs} is the number of LEDs at each row of the transmitter, w is the image width (in case the rolling axis is along the width of the image sensor), and ϱ is the size of LED lights in cm^2. Please note that the distance d^b in (4.4) is affected by relative speed of the vehicle, which affects the position of the vehicle on the road. Let us assume a slotted time. The inter-vehicular distance at current time t is adjusted using $d_t = d_{t-1} + v_t \cdot \Delta t$, where d_{t-1} is the distance of the previous time instance, v_t represents the velocity of vehicle, and Δt is the time elapsed between time instant t and $t-1$.

The transmission latency of packet size, L, can be expressed similarly to [88] as:

$$\tau^b = \frac{L}{C^b}. \tag{4.5}$$

4.2.3 Proposed Problem Formulation

Considering the proposed framework and ultra-low latency and BER requirements, we formulate an optimization scheme that aims at maximizing the sum spectral efficiency of the vehicular OCC system by selecting the optimal modulation order from an available set and adjusting the relative speed of the vehicle to the optimal value. The BER and latency are constrained so that they meet the values imposed by the system. Mathematically, our constrained maximization problem is, hence, formulated as:

$$\max_{\mathcal{M},v} \quad \frac{1}{B}\sum_{b=1}^{B}\log_2\left(M^b\right), \tag{4.6}$$

$$\text{s.t.} \quad \text{BER}^b \leq \text{BER}_{\text{tgt}},\ \forall b; \tag{4.7}$$

$$\tau^b \leq \tau_{\max},\ \forall b; \tag{4.8}$$

$$M^b \in \mathcal{M},\ \forall b; \tag{4.9}$$

where $\mathcal{M}$ is the set of QAM modulation orders, BER_{tgt} is the maximum target BER, and $\tau_{\max}$ is the maximum affordable latency. Constraints (4.7) and (4.8) indicate that the BER and latency thresholds are satisfied. The modulation scheme is chosen from a small set of available M-QAM schemes, as shown in (4.9).

4.3 DRL-based Problem Formulation and Proposed Solution

The studied problem in (4.6) is mixed-integer programming (MIP) with nonlinear constraints for BER (4.7) and delay (4.8). This makes our problem NP-hard [90]. It is known that MIP problems have high computational complexity [91] and although it may be possible to solve them using dynamic programming or exhaustive search techniques, these methods cannot be used in dynamic systems as the one we investigate in this paper since they are extremely time-consuming or computationally demanding.

As in our problem, we simultaneously control the speed and modulation for multiple links, the decision space is large. Due to the entailed computational and time complexities in solving the proposed problem, we first express the problem as an MDP problem in the next subsection. This gives us the opportunity to use other tools, such as deep RL, to solve the problem with less complexity. Note that vehicular communication must satisfy the maximum latency and BER requirements to ensure that the information is received reliably within the shortest time. We adopt an independent learning framework, where each vehicle independently decides its action, but they all affect the environment. It has been shown that this leads to well-performing solutions without requiring explicit communication [91]. Preceding to presenting our solution, we first model the optimization problem in (4.6) as an MDP in the next subsection.

4.3.1 Modelling of MDP

We model the proposed multi-agent RL problem as an MDP, where each vehicle acts as an agent, and everything beyond the particular vehicle is regarded as the environment. Every vehicular agent interacts with the environment to have a better understanding of it to decide its own policy. The agents explore the environment and improve the spectral efficiency maximization policies based on their observations of the environmental state. The optimization problem (4.6) is modelled as a MDP with a tuple ($\mathcal{S}$, $\mathcal{A}$, p, r, ζ) [11], where $\mathcal{S}$ is the set of all possible states; $\mathcal{A}$ denotes the set of all possible actions; $p(s_{t+1}, r_t|s_t, a_t)$ denotes the transition probability which describes the probability that an agent selects an action $a_t \in \mathcal{A}$ and transits to a new state $s_{t+1} \in \mathcal{S}$ from the current state $s_t \in \mathcal{S}$; while r represents the reward. The parameter $\zeta \in [0, 1]$ is the discount factor, which gradually discounts the effect of an action to future rewards. A discount factor $\zeta = 0$ provides a short-sighted goal that maximizes the immediate reward. When ζ is close to 1, the agent focuses more on the future reward, and the scheme becomes far-sighted. In practice, a far-sighted approach is desirable as it

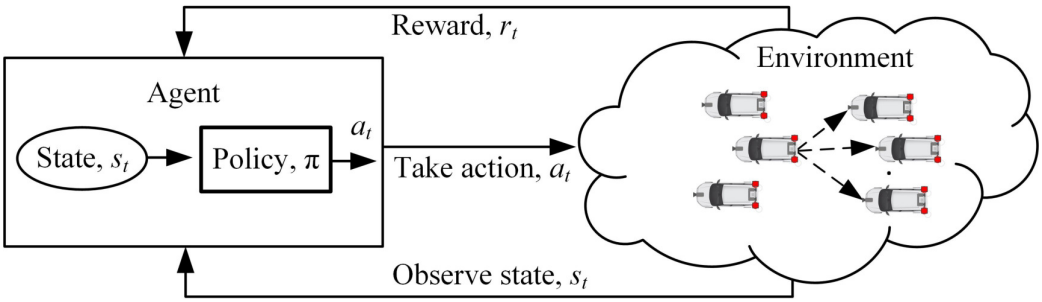

Figure 4.2: An illustration of basic reinforcement learning framework for V2V communications.

achieves better returns by focusing on future discounted rewards. It is also notable that an algorithm with lower discount factors converges faster, especially during early learning. But of course, a small valued discount factor can lead to highly sub-optimal policies that are too myopic.

We present a general RL framework in Fig. 4.2 consisting of agents and environment. From this figure, we see that at each time t, an agent observes a state, $s_t \in \mathcal{S}$ and accordingly takes an action, $a_t \in \mathcal{A}$ based on the policy, π and receives a reward, r_t, from the environment. Next, we express the state space $\mathcal{S}$, the action space $\mathcal{A}$, and the reward function, r of the considered RL framework.

State Definition

In our system, the observed state from the environment to each agent couples two components: the backward distance vector, $\mathbf{d}_t^b = (d_t^1, \cdots, d_t^B)$ and transmitting modulation order for the backward vehicles, M_t^b from the set $\mathcal{M} = \{4, 8, 16, 32, 64\}$. In summary, the state is expressed as $s_t = \left\{\mathbf{d}_t^b, M_t^b\right\}$.

Action Definition

At each time t, the agent takes an action a_t, a decision regarding the relative speed of the agent (i.e., vehicle), v_t and selecting modulation order from

the set $\mathcal{M}$, based on the current state, s_t by following a policy π. Overall, the action space is summarized as $a_t = \{v_t, M_t^b\}$.

Reward Definition

At each time slot t, when agent takes an action a_t in state s_t, it will immediately receive a reward r_t. Note that, an effective reward framework is imperative for the learning algorithm to achieve the desired goal, which is achieved through exploration. Therefore, the reward function that guides the overall learning should be consistent with the objective.[2] First, we express the reward related to distance as follows:

$$r_t^{\mathrm{d},i} = \begin{cases} -1 \times (d_{\text{stop}} - d_t^b), & d_t^b < d_{\text{stop}} \ , \\ \frac{1}{d_t^b - d_{\text{stop}}}, & d_t^b > d_{\text{stop}} \ , \end{cases} \tag{4.10}$$

where i is the index of the agent. Recall that, d_t^b represents the backward distance of the vehicle, but in designing our reward, we only consider the vehicle behind residing at the same lane on the road. The priority is to avoid collision with the vehicle on the same lane. This is the decisive vehicle since it has the possibility of coming closer to the agent vehicle in the following time step or near future. d_{stop} is the stopping distance, which is equal to the sum of covered distance by the vehicle to travel after the brakes are activated, and the covered distance to travel due to driver's reaction time after observing a situation [92]. In our system, each vehicle will carry out the same process individually. As a result, for notational simplicity, we drop i hereafter. Since our objective is to maximize the sum spectral efficiency, we design our reward function as a weighted sum of a reward related to the backward distance and the sum spectral efficiency (4.6). As the goal of RL is to maximize the reward, it will conclusively maximize the spectral efficiency while maintaining a safe distance. Hence,

[2]From hereon, we will use backward distance and distance interchangeably though it indicates the same meaning.

considering the objective function (4.6), the overall reward, R_t, can be expressed as

$$R_t = \omega_d \, r_t^{\mathrm{d}} + \omega_r \, \frac{1}{B} \sum_{b=1}^{B} \log_2 \left(M_t^b \right), \tag{4.11}$$

where ω_d and ω_r are positive weights that balance distance and sum spectral efficiency rewards. The weights are adjusted based on the system requirements. It sets the priority depending on its distance and modulation scheme changes.

4.3.2 RL-based Problem Formulation

After interaction with the environment in each time slot t, the agent receives a reward $r_t = r(s_t, \pi(s_t))$ by taking an action a_t and following a policy π at the current state s_t. The goal of RL is to find the optimal policy that maximizes the expected return from the state s_t, whereas the return, G_t, is defined as the cumulative discounted reward, as follows:

$$G_t = \sum_{j=0}^{\infty} \zeta^j R_{t+j+1}, \qquad 0 \leq \zeta \leq 1. \tag{4.12}$$

The objective is as follows: An agent (i.e., a vehicle) selects the speed and modulation order while respecting the BER and latency constraints to find a policy that maximizes the expected cumulative discounted rewards. Finally, the constrained reward maximization problem is expressed in the RL framework as

$$\max \quad \mathbb{E}\left[G_t \left(s_t, a_t \right) \right], \ \forall t \tag{4.13}$$

$$\text{s.t.} \quad \mathrm{BER}_t^b \leq \mathrm{BER}_{\mathrm{tgt}}, \ \forall t; \tag{4.14}$$

$$\tau_t^b \leq \tau_{\max}, \ \forall t; \tag{4.15}$$

4.3.3 The Lagrangian Approach

According to [93], constrained MDP problems can be solved by recasting them as unconstrained ones via the Lagrange relaxation method. Hence,

we reformulate the constrained optimization problem in (4.13) - (4.15) by introducing Lagrange multipliers associated with the BER and latency constraints, $c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t)$, as:

$$c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) = R_t\left(s_t, a_t\right) - \sum_{b=1}^{B} \lambda^b \cdot (\text{BER}_t^b - \text{BER}_{\text{tgt}}) - \sum_{b=1}^{B} \nu^b \cdot (\tau_t^b - \tau_{\max}), \tag{4.16}$$

where $\boldsymbol{\lambda} = (\lambda^1, \lambda^2 \cdots, \lambda^b)$ and $\boldsymbol{\nu} = (\nu^1, \nu^2, \cdots, \nu^b)$ are vectors representing the Lagrange multipliers corresponding to the constraints in (4.14) and (4.15), respectively. The optimal value of the constrained MDP problem can be computed as [94]:

$$\begin{aligned} L_\delta^{\pi^*,\boldsymbol{\lambda}^*,\boldsymbol{\nu}^*}(s) =& \min_{\pi\in\phi} \max_{\boldsymbol{\lambda},\boldsymbol{\nu}\geq 0} V^{\pi,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) - \sum_{b=1}^{B} \lambda^b\delta_1 - \sum_{b=1}^{B} \nu^b\delta_2 \\ =& \max_{\boldsymbol{\lambda},\boldsymbol{\nu}\geq 0} \min_{\pi\in\phi} V^{\pi,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) - \sum_{b=1}^{B} \lambda^b\delta_1 - \sum_{b=1}^{B} \nu^b\delta_2, \end{aligned} \tag{4.17}$$

where $\delta = \{\delta_1, \delta_2\}$, with $\delta_1 = \text{BER}_{\text{tgt}}$ and $\delta_2 = \tau_{\max}$. ϕ denotes the set of all possible stationary policies,

$$V^{\pi,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \zeta c^{\boldsymbol{\lambda},\boldsymbol{\nu}}\left(s_t, \pi\left(s_t\right)\right) \mid s_0 = s\right]. \tag{4.18}$$

A policy π^* is optimal for the constrained MDP, if and only if

$$L_\delta^{\pi^*,\boldsymbol{\lambda}^*,\boldsymbol{\nu}^*}(s) = \max_{\boldsymbol{\lambda},\boldsymbol{\nu}\geq 0} V^{\pi^*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) - \sum_{b=1}^{B} \lambda^b\delta_1 - \sum_{b=1}^{B} \nu^b\delta_2. \tag{4.19}$$

For a fixed $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$, the rightmost maximization of (4.17) is equivalent to solving the following dynamic programming equation:

$$V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t) = \min_{a_t\in\mathcal{A}} \left\{ c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) + \zeta \sum_{s_{t+1}\in\mathcal{S}} p(s_{t+1} \mid s_t, a_t) V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}) \right\}, \forall s \in \mathcal{S}, \tag{4.20}$$

where $V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}} : \mathcal{S} \mapsto \mathbb{R}$ is the optimal state-value function and s_{t+1} is the state at time slot $t+1$.

We also define optimal action-value function $Q^{*,\boldsymbol{\lambda},\nu} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ which represents the Q-value of action a_t in a given state s_t.

$$Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) = \ c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) + \zeta \sum_{s_{t+1} \in \mathcal{S}} p(s_{t+1} \mid s_t, a_t) V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}), \tag{4.21}$$

where $V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}) = \max_{a_t \in \mathcal{A}} Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}, a_t), \forall s \in S$. In words, $Q^{*,\boldsymbol{\lambda}\boldsymbol{\nu}}(s_t, a_t)$, is the infinite discounted cost achieved after taking action a_t in state s_t and therefore, following the optimal policy $\pi^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}$, which is given by

$$\pi^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t) = \arg\max_{a_t \in \mathcal{A}} \ Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t), \forall s \in S. \tag{4.22}$$

[3]In practice, the optimal policy, π^*, cannot be determined using value-iteration method [11] as it requires transition probabilities to be known beforehand. For the considered problem, continuous computation of the transition probability matrix is necessary, which is computationally demanding. To solve this problem, we adopt a model-free RL approach known as Q-Learning, which learns Q^* and π^* online, without requiring the model of the environment and computing the transition probability matrix. Q-Learning uses the $Q_t(s_t, a_t)$ values instead of the value function in (4.20). $Q_t(s_t, a_t)$ represents how good it is to take action a_t when starting from state s_t, and thereafter follow the policy π. To determine the optimal policy π^*, the Q-Learning algorithm employs the following recursive formula to update the $Q_t(s_t, a_t)$ values:

$$Q_{t+1}(s_t, a_t) = (1 - \alpha_t) Q_t(s_t, a_t) + \alpha_t \left[c_t(s_t, a_t) \ + \zeta \max_{a_{t+1} \in \mathcal{A}} Q_t(s_{t+1}, a_{t+1}) \right], \tag{4.23}$$

where $\alpha_t \in [0, 1]$ is a time-varying learning rate and a_{t+1} is the greedy action in state s_{t+1} at time slot $t+1$. The learning rate refers to the rate at which newly updated information overrides old one.

[3]For notational simplicity, we drop the Lagrangian multipliers from the notation in the remainder of the chapter unless it is necessary, for example, we will write $c(s_t, a_t)$, $Q^*(s_t)$, instead of $c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t)$, $Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t)$, respectively.

Q-Learning can select actions using policies such as the ϵ-greedy, where $\epsilon \in [0, 1]$ [95]. It has been shown in [94] that the Q-Learning algorithm will eventually converge to the optimal Q, $Q^*(s_t, a_t)$ with probability 1 when all the state-action pairs are visited often, and the learning rate α_t respects the following conditions:

$$\alpha_t \in [0, 1], \quad \sum_{t=0}^{\infty} \alpha_t = \infty, \quad \sum_{t=0}^{\infty} (\alpha_t)^2 < \infty. \tag{4.24}$$

We discuss how to learn the optimal $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ in Section 4.3.4.

4.3.4 Deep Q-Learning

Q-Learning is a well-known method [11] that is used to solve problems expressed as MDP. The convergence speed of this algorithm depends on the state-action space size. Q-Learning converges faster for small state-action spaces since the agent can quickly explore the state-action pairs and determine the optimal policy. For larger state-action spaces, the convergence is slow which makes the determination of the optimal actions not feasible within the stringent time constraints imposed by the dynamic nature of the environment in problems like the one we study here. Although some linear function approximation approaches exist for solving large-scale RL problems, their capabilities are limited to medium-scale problems. In high-dimensional and complex systems, conventional RL methods cannot learn the informative features of the environment quickly, despite employing effective approximation functions. This is due to the fact that most of the state-action pairs are rarely visited, and thus the corresponding Q-values are not updated regularly, leading to a much longer time to converge. More importantly, distance and speed are continuous values that lead to a large state-action space; hence, the tabular Q-learning algorithm cannot be used because it works with discrete values. Discretization may be applied, but this affects the quality of the solution.

However, this problem can be resolved by employing deep learning-based function approximators, in which DNNs are trained to learn the op-

timal policy. In a DQN, a DNN function approximator with weights θ is employed as Q-network, and then Q-Learning is combined with deep learning. Once weights θ are determined, Q-values, $Q(s, a)$, will be the outputs of the DNN. DNN addresses sophisticated mappings between the channel information and the desired output through excessive training data, which are then used to determine the Q-values.

Target Network

In order to stabilize the learning of DQN, we follow the target network approach. The DQN consists of two separate networks known as the main network that approximates the Q-function and the target network that gives the target for updating the main network. In the training phase, while the main network parameters $\boldsymbol{\beta}$ are adjusted after every action, target network parameters $\boldsymbol{\beta}_-$ are updated after a certain period of time. The target network is not updated after each iteration because it adjusts the main network updates to control the value estimations. If both networks are updated simultaneously, the change in the main network would be exaggerated due to the feedback loop from the target network, which results in an unstable network. To ensure the stability in learning, the neural network aims to minimize the loss function, $L(\boldsymbol{\beta})$, which is expressed as

$$L(\boldsymbol{\beta}) = \mathbb{E}\left[y_t - Q\left(s_t, a_t; \boldsymbol{\beta}\right)\right]^2, \tag{4.25}$$

where $y_t = c(s_t, a_t) + \zeta \max_{a_{t+1} \in \mathcal{A}} Q\left(s_{t+1}, a_{t+1}; \boldsymbol{\beta}_-\right)$ is the target for each iteration. Note that, $\boldsymbol{\beta}_-$ are held fixed when optimizing the loss function $L(\boldsymbol{\beta})$.

Optimal Lagrange Multipliers

The optimal value of the Lagrange multipliers λ^b, ν^b in (4.16) depend on the BER constraint, BER_{tgt} and latency constraint, $\tau_{\max}$, respectively and can be learned online using a stochastic sub-gradient method as presented

in [96]

$$\lambda_{t+1}^{b} = \Lambda\left[\lambda_{t}^{b} + \varpi_{t}(\mathrm{BER}_{t}^{b} - \mathrm{BER}_{\mathrm{tgt}})\right], \tag{4.26}$$

$$\nu_{t+1}^{b} = \Lambda\left[\nu_{t}^{b} + \varpi_{t}(\tau_{t}^{b} - \tau_{\max})\right], \tag{4.27}$$

where we apply the projection operator Λ in order to project λ^b and ν^b onto $[0, \lambda_{\max}]$ and $[0, \nu_{\max}]$. To ensure the boundedness of $\lambda_{\max}$ and $\nu_{\max}$, we consider $\lambda_{\max}, \nu_{\max} > 0$ to be large enough. ϖ_t corresponds to a time-varying learning rate, which obeys the same conditions as α_t in (4.24). The following additional conditions must be jointly satisfied by α_t and ϖ_t to guarantee the convergence of (4.26) and (4.27) to $\boldsymbol{\lambda}^*$ and $\boldsymbol{\nu}^*$, respectively:

$$\sum_{t=0}^{\infty}(\alpha_t + \varpi_t) < \infty \quad \text{and} \quad \lim_{t\to\infty}\frac{\varpi_t}{\alpha_t} \to 0. \tag{4.28}$$

4.4 Experimental Set up

This section describes the implementation details of our proposed DRL-based vehicular OCC scheme. Specifically, we build upon the simulation environment upon microscopic traffic simulator Simulation of Urban Mobility (SUMO) [97] and DRL framework within SUMO.

4.4.1 SUMO Framework

Our simulation framework maintains the connection between SUMO and the DRL agent using Traffic Control Interface (TraCI). SUMO is an open-source, microscopic, multi-model traffic and extensible simulator and has been widely used in research projects with worldwide community support. It allows the users to simulate specific traffic scenarios performed in given road maps. In our experiments, SUMO is used as the traffic simulator because: (i) it performs an optimized traffic distribution method based on vehicle types or driver behaviors to maximize the capacity of the urban transportation network; (ii) it provides flexibility and scalability to create

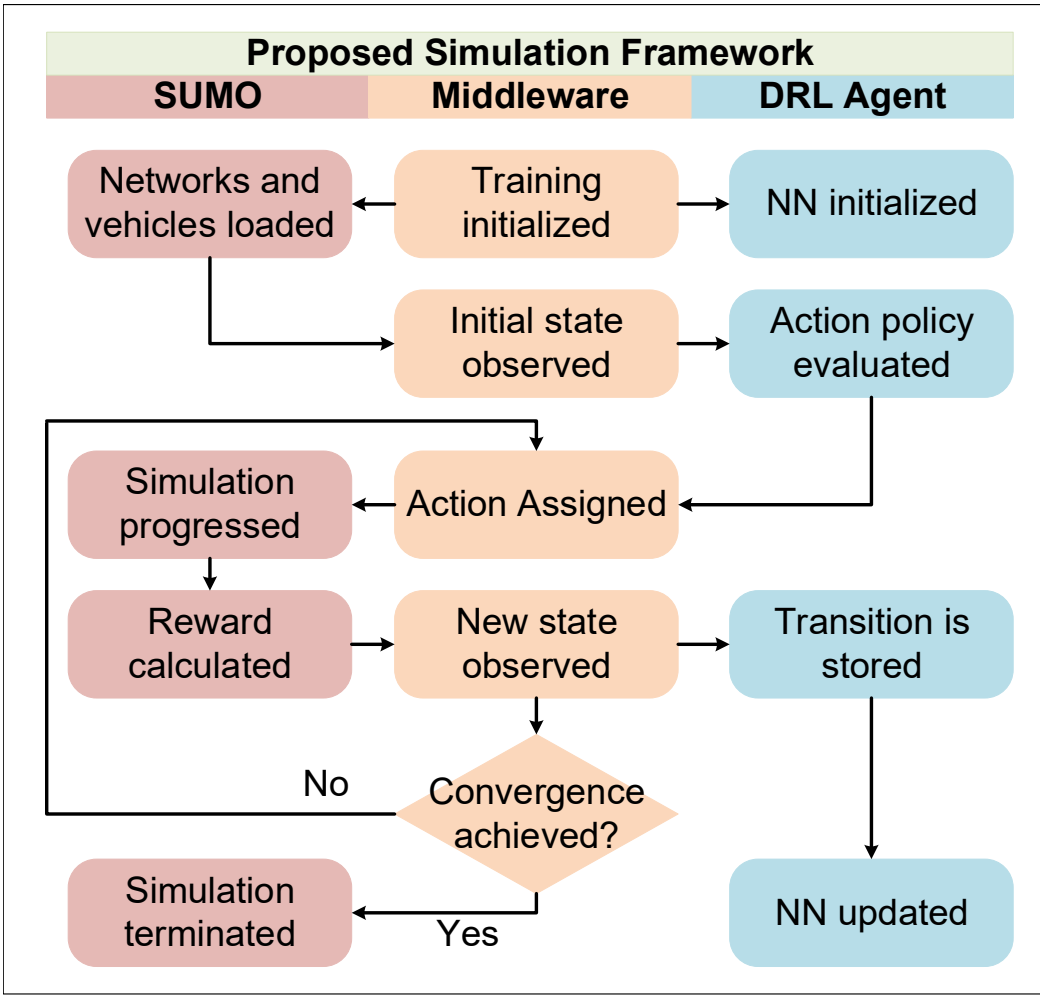

Figure 4.3: Proposed simulation framework combining SUMO simulator, middleware and DRL agent for the vehicular communication.

the scenario maps; and (iii) it supports TraCI, a Python-based API to communicate the traffic simulation with the controls from the smart agents.

In order to simulate the proposed vehicular framework in a more practical scenario, we convert the proposed environment into a corresponding SUMO map. Each vehicle is considered as an agent and accordingly modelled to test the proposed DQN method in the integrated environment. The vehicles enter randomly in the SUMO environment and then move or leave the network following SUMO mobility models.

Figure 4.4: Illustration of proposed scenario in SUMO GUI interface.

As shown in Fig. 4.3, the proposed simulation framework consists of three parts: firstly, SUMO, which is the simulator environment for creating traffic scenarios; secondly, the middleware that connects the SUMO environment with the DRL agents; and finally, the DRL agents, which maintain and update the network policies and execute actions for the simulation. After the training is initialized, the SUMO simulator is loaded, progressed, and reset with required information such as the transportation network and vehicles via TraCI. During the simulation, TraCI interacts with the SUMO environment and extracts the data from SUMO to produce the observation for state space and aggregate rewards. Moreover, TraCI retrieves different features from the network, e.g., the number of vehicles on each road, the speed of the vehicle, and the current position of the agent. Based on the current observations, the DRL agent evaluates the current traffic environment and assigns an action based on the policy of the neural network. Accordingly, the agent updates the state and moves to the next step in the SUMO environment and this process continues until all the simulation steps finish. The reward is then computed and transferred to the DRL agent for optimization at the end of each simulation run. The objective is to train the policy network that ensures higher communication quality in the form of spectral efficiency, delay, and BER.

We modified the SUMO environment according to the requirement of our proposed multi-agent vehicular system. For example, we have a window size of the simulation of 180 m. To introduce randomness. we put an aggressive vehicle in the SUMO model, which moves freely in the environment. The simulation parameters of the SUMO framework are presented in Table 4.1. We illustrate a screenshot of the simulated vehicular model represented on SUMO Graphical User Interface (GUI) interface in Fig. 4.4.

Table 4.1: SUMO modelling parameters

Parameter	Value
Initial velocity of vehicle	5 miles per hour
Window size of the simulation	180 m
Maximum number of vehicle per window	20
Number of lane	3
Step length	1 m
Lateral movement of vehicle	0.64 m per timestep

As shown in this figure, we have three lanes, where vehicles move at different velocity and each vehicular agent has potentially multiple vehicles in front and back. The agent extracts the various parameters of the vehicle and surrounding environment related to our modelling and exports them to the DRL agent using TraCI. Please recall that the agent must satisfy the constraints of the system to generate a higher reward and minimize the loss.

4.4.2 DQN Settings

Network Architecture

This subsection provides the details of the employed DNN architecture as well as the training parameters we employed. The DQN consists of three fully connected layers, including an input layer, a hidden layer, and an output layer. Recall that distance and modulation order define the state space; hence, the input layer consists of $M^b + d^b$ nodes. The output layer consist of $M^b + v$ nodes, as we have $M^b + v$ actions. The hidden layer has 250 neurons. We use Rectified Linear Unit (ReLU) as the activation function [95], defined as $f(x) = \max(0, x)$. We adopt Root Mean Square Propagation (RMSPro) optimizer [98] as the training algorithm to minimize the loss function and update DQN network parameters, which is one of the most used optimizers

Algorithm 1 DQN Training Algorithm

Initialization: Initialize SUMO environment, DQN parameters, replay memory according to system requirements.
Output: Action-value function, loss (4.25).
for each episode **do**
  Update vehicle speed and modulation order
  for each link, b **do**
    Observe state s_t
    Choose action a_t according to the ϵ-greedy policy
    Execute action a_t, observe reward r_t and next state s_{t+1}
    Store transitions (s_t, a_t, r_{t+1}, s_{t+1}) in the replay memory
  end for
  Agent takes actions and receive reward r_t using (4.11).
  Update Lagrange multipliers $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ using sub-gradient method as in (4.26) and (4.27), respectively.
end for
Sample a mini-batch from the replay memory.
Optimize error between Q-network and target Q, defined in (4.25), using RMSProp optimizer gradient descent.

in neural networks. We set the initial learning rate α to 0.001, which will be sufficient to balance the convergence time. It is known that a large learning rate leads to fast convergence behaviour, but at the same time, may incur a poor convergent point with unsatisfactory performance, e.g., local minima, saddle point. On the contrary, intensive training computations are required for a small α as it results in slow convergence. Therefore, an appropriate α should carefully be chosen. In our case, the RMSPro optimizer is used to vary the learning rate over time. In our simulations, to implement deep reinforcement learning, we use TensorFlow [99], which allows us to debug better and track the training process. We implement ϵ-greedy policy to balance between exploration and exploitation while avoiding overfitting.

Table 4.2: List of DRL hyper-parameters and their values

Parameter, Notation	Value
Mini-batch size	32
Replay memory size	100000
Number of hidden layer (Neurons)	$1(250)$
Exploration rate, ϵ	0.05
Discount factor, ζ	0.98
Activation function	ReLU
Optimizer	RMSProp
Learning rate (used by RMSProp)	0.001
Gradient momentum (used by RMSProp)	0.95

According to ϵ-greedy policy, the action with maximum $Q_t(s_t, a_t)$ value is chosen with probability $1 - \epsilon$ while a random action is selected with probability ϵ.

Training Procedure

The training procedure of our proposed DQN algorithm is summarized in **Algorithm** 1. The input of the algorithm is the current observations (distance and modulation scheme), and the output is the chosen actions (speed and modulation scheme) by the vehicle. The agents map the actions with the corresponding action-value functions, i.e., Q-value. We train the DQN algorithm for multiple episodes and, at each training step, all the agents execute the ϵ-greedy policy to explore the state-action space. Following the environment transition due to channel variation and actions taken by all agents, each agent observes and stores the transition tuple, (s_t, a_t, r_{t+1}, s_{t+1}), in the replay memory. At each episode, a uniformly sampled mini-batch of experiences are taken from the memory for updating $\boldsymbol{\beta}$ parameters of (4.25) using stochastic gradient descent methods and the loss is estimated using (4.25).

For the simulations, we train the DQN for 10000 episodes. The exploration rate, ϵ is set to 0.05. The target Q-network parameters are updated every 400 learning steps, where each episode contains 100 steps. We choose a discount factor, $\zeta = 0.98$. For our simulation run, we use a track size of 180 m, and we measure the density of vehicles as the number of vehicles per 180 m. The total replay memory size for storing the transactions is 100000, and the mini-batch for training is 32. The training and testing parameters of the DRL are presented in Table 4.2.

Normalization

The goal of normalization is to bring the different sub-rewards corresponding to delay and spectral efficiency in (4.16) to be on a similar scale. This normalization improves the performance and provides training stability of the Neural Network (NN) model. Specifically, we normalize the reward function (4.11), BER and latency constraints of (4.16) to keep the scale between 0 and 1. Please note that, we perform quantization on the continuous values of distance and speed of the vehicle to convert them into discrete values. For example, we quantize the values of distance into step length of 1 m and the speed into 0.5 ms^{-1} step.

4.5 Performance Evaluation

In this section, we evaluate the performance of the proposed multi-agent RL based spectral efficiency maximization scheme for vehicular OCC. The simulation parameters for OCC system model are listed in Table 4.3.

4.5.1 Overview of Comparison Schemes

We investigate the performance of the proposed multi-agent DRL based vehicular scheme, termed hereafter as the proposed scheme against different methods for comparison. We present a brief summary of all the schemes

Table 4.3: Vehicular OCC modelling parameters

Parameter, Notation	Value
Angle of irradiance w.r.t. the emitter, ϕ	70^o
AoI w.r.t. the receiver axis, θ	60^o
FOV of the camera lens, θ_l	90^o
Image sensor physical area, A	10 cm^2
Transmission efficiency of optical filter, T_s	1
Concentrator/lens gain, g	3
Optical transmitting power, P	1.2 Watts
Constellation size, M	4, 8, 16, 32, 64
Camera-frame rate, W_{fps}	1000 fps
Number of LEDs at each row, N_{LEDs}	30
Packet size, L	5 kbits
Size of the LED, ϱ	15.5 $\times$ 5.5 cm^2
Resolution of image, w	512 $\times$ 512 pixels

under comparison below:

- **Proposed scheme**: By the proposed scheme, we refer to our multi-agent DRL-based vehicular OCC system, where each agent performs independent learning considering all other vehicles as environment. In this case, we employ the settings as we discuss in Sections 4.4.2 and 4.4.2. We set the discount factor to 0.98.

- **Greedy**: The greedy method is when we assume $\zeta = 0$ in (4.25). This method is a variant of our scheme, where we set the discount factor to $\zeta = 0$ in (4.25), while we keep all other parameters of the systems as reported in Table 4.2. In this scenario, the agent chooses the action which maximizes only the immediate reward.

- **Far-sighted**: This method is a variant of our scheme, where we set the discount factor to $\zeta = 1$ in (4.25), while we keep all other parameters

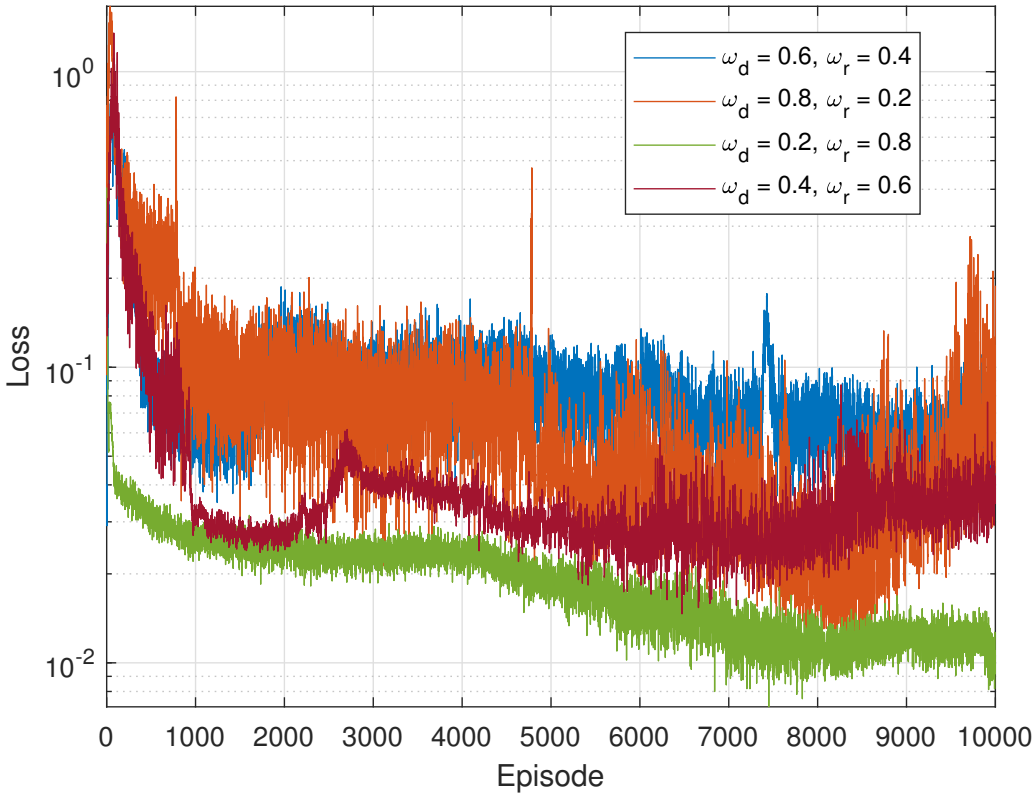

Figure 4.5: Convergence of loss function for $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

of the systems as reported in Table 4.2. This scheme takes future rewards into account more strongly and ignores immediate rewards.

- **Random**: This is a scheme, where the actions are chosen randomly for all the vehicles at each time slot. In this case, the system parameters are not optimized and the agent chooses speed and modulation schemes randomly.

- **RF-based MARL [13]**: This is a multi-agent RL based resource allocation scheme presented in [13]. This method is based on RF technology. For this scheme, we adapt the hyper-parameters according to our proposed scheme while keeping the environment unchanged. This scheme considers centralized learning and distributed imple-

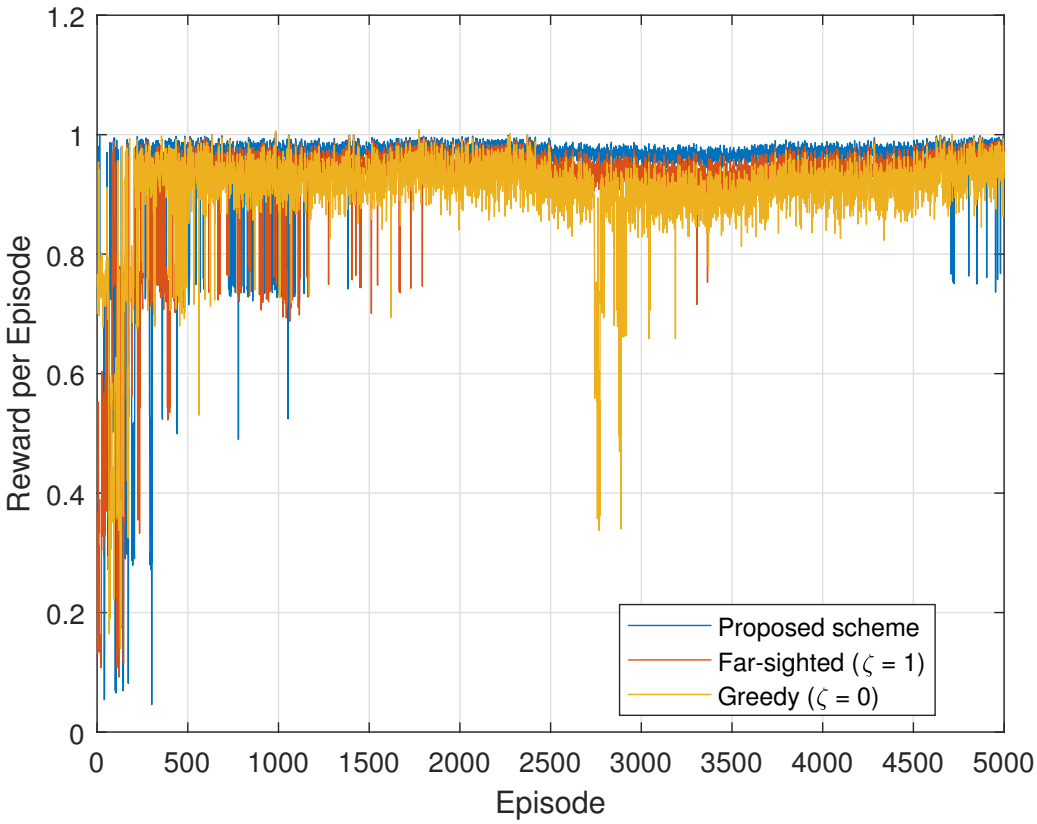

Figure 4.6: Reward per training episode for three different approaches when $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

mentation. The system performance-related reward is available to each individual agent through a centralized base station in the cellular network. Then the agent adjusts its action towards the optimal policy by updating its DQN and utilises its local observation and trained DQN to select the best action. Finally, the agent communicates the updated DQN towards the base station.

- **RF-based SARL [13]**: This is a single agent RL based scheme proposed in [13], specified as Single Agent Reinforcement Learning (SARL), where at each time only an agent, i.e., V2V link, updates its action based on the locally observed information, whereas other agents' action remains unchanged. A single DQN policy is shared over the

vehicular network towards all the vehicles.

4.5.2 Simulation Results

The convergence trend of the training algorithm confirms the suitability of the proposed scheme. To this end, we investigate the convergence of the proposed algorithm. First, we perform an ablation study to determine the weight values corresponding to distance and rate rewards in (4.11). In doing so, we examine our algorithm for different weight settings of distance and rate rewards, but for simplicity of representation, we only demonstrate four settings, including $\omega_d = 0.2$ and $\omega_r = 0.8$, $\omega_d = 0.4$ and $\omega_r = 0.6$, $\omega_d = 0.6$ and $\omega_r = 0.4$, $\omega_d = 0.8$ and $\omega_r = 0.2$, as shown in Fig. 4.5. We observe that we achieve lower loss when we allocate higher weight value to the spectral efficiency component. By observing Fig. 4.5, we can see that our scheme converges at around 8000 episodes for $\omega_d = 0.2$ and $\omega_r = 0.8$. On the contrary, other weight sets require longer times for convergence and show frequent variations in the loss and offer higher loss than $\omega_d = 0.2$ and $\omega_r = 0.8$ set. So, we adopt this weights setting for the rest of our performance evaluation.

We then present the rewards per training episode to analyze the convergence behaviour of the multi-agent vehicular OCC system at three different discount factors, i.e., the proposed scheme ($\zeta = 0.98$), greedy ($\zeta = 0$) and far-sighted ($\zeta = 1$). The results are shown in Fig. 4.6. Please note that for the ease of visualization, we present the reward until 5000 episodes as they follow the same trend after that. From this figure, we observe that until 1500 episodes, the greedy and far-sighted approaches achieve better performance than the proposed scheme. This happens because the agent requires time at the start to fit in an optimal solution through a perfect exploration and exploitation policy. However, the cumulative reward for the proposed scheme improves as the training advances and reaches to lower loss. Instead, the rewards for greedy and far-sighted schemes fluctuate throughout the training episodes. We can conclude that the proposed

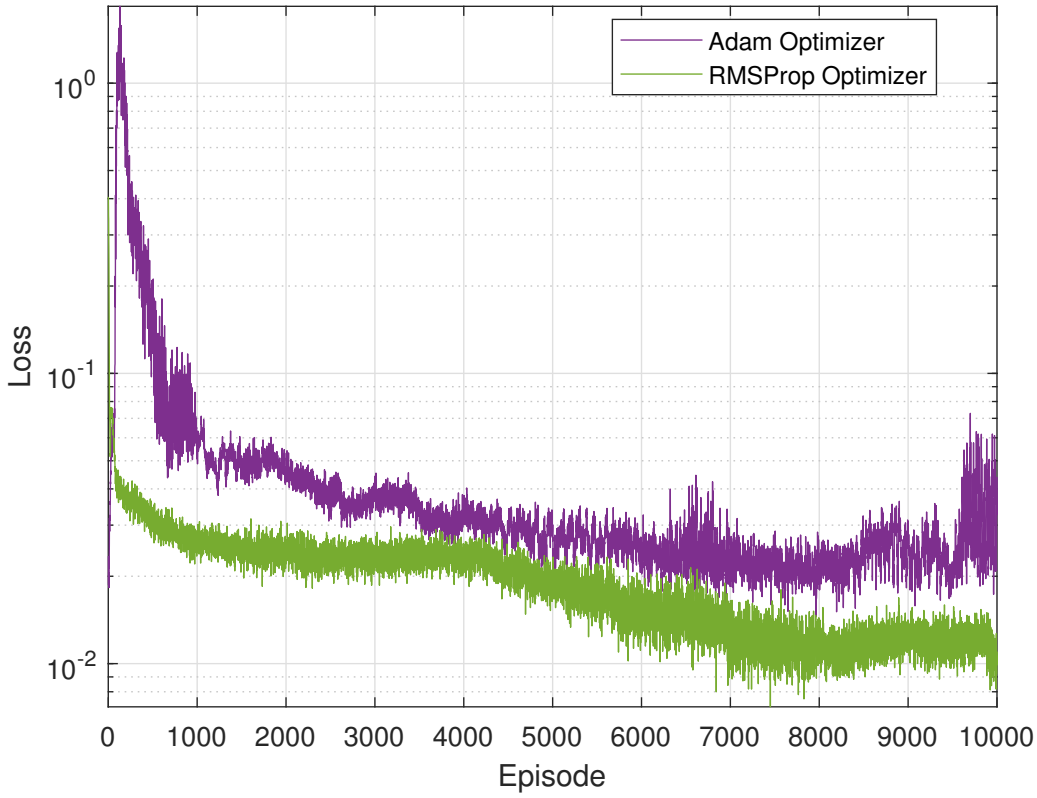

Figure 4.7: Performance comparison between RMSProp and Adam gradient optimizer versus training episode.

scheme achieves higher rewards than other variants of our scheme.

To minimize the loss in the DQN, there are different gradient descent optimizers, which vary the learning rate adaptively. Here, we investigate the loss performance of two mostly applied optimizers, namely, RMSPro and Adaptive Moment Estimation (Adam) optimizers for 10000 episodes. The results are illustrated in Fig. 4.7. From this figure, we can see that the RMSPro optimizer achieves lower loss throughout the training period than the Adam optimizer. More specifically, while Adam optimizer does not converge within 10000 episodes, the RMSPro converges at around 7000 episodes. Therefore, we adopt an RMSPro optimizer in our framework.

To justify the superiority of the proposed multi-agent DRL-based vehicular OCC scheme, we compare its performance with MARL and SARL method

presented in [13] and a random scheme. We utilize the same DQN parameter to optimize the problem in [13]. For example, we implement a single hidden layer with 250 neurons instead of three hidden layers, a fixed discount factor, and 10000 training episodes. We also formulate the spectral efficiency and latency according to our formulation. Though the system proposed in [13] have not considered latency, we estimated it to study how the latency requirements are satisfied. As the MARL and SARL methods require base stations to communicate with each other, it involves uplink and downlink latency in addition to processing latency. Whereas our system has only transmission latency as it is a decentralized scheme, RF-based MARL and SARL in [13] require centralized communication, which incurs additional latency.

Fig. 4.8 shows the maximized sum spectral efficiency performance with respect to the density of vehicles for all schemes under comparison. From this figure, we observe that the sum spectral efficiency increases with an increase in density of vehicles for all the methods using our proposed framework, namely, greedy, far-sighted, proposed scheme as well as the random scheme. On the contrary, the performance drops with increasing density of vehicles for RF-based MARL [13] and SARL systems [13]. For our OCC system, an increase in vehicle density means that the distance between vehicles is smaller, and hence, the communication quality improves, which boosts the spectral efficiency. Whereas for RF-based MARL and SARL, an increased vehicles' density causes higher interference, and thus, it reduces the spectral efficiency. The results show that the proposed algorithm obtains approximately 2.4 times better rates in comparison to the MARL, 2.9 times for the SARL, and about 1.6 times for the random scheme when the density of vehicles is 16. Whereas, it is lower by 0.73 times for MARL and 0.82 times for SARL when vehicle density is 6. From this comparison, we can conclude that our OCC system performs better in urban scenarios or highways with dense traffic where the density of vehicles is always higher.

We present the comparison results of average latency versus the density

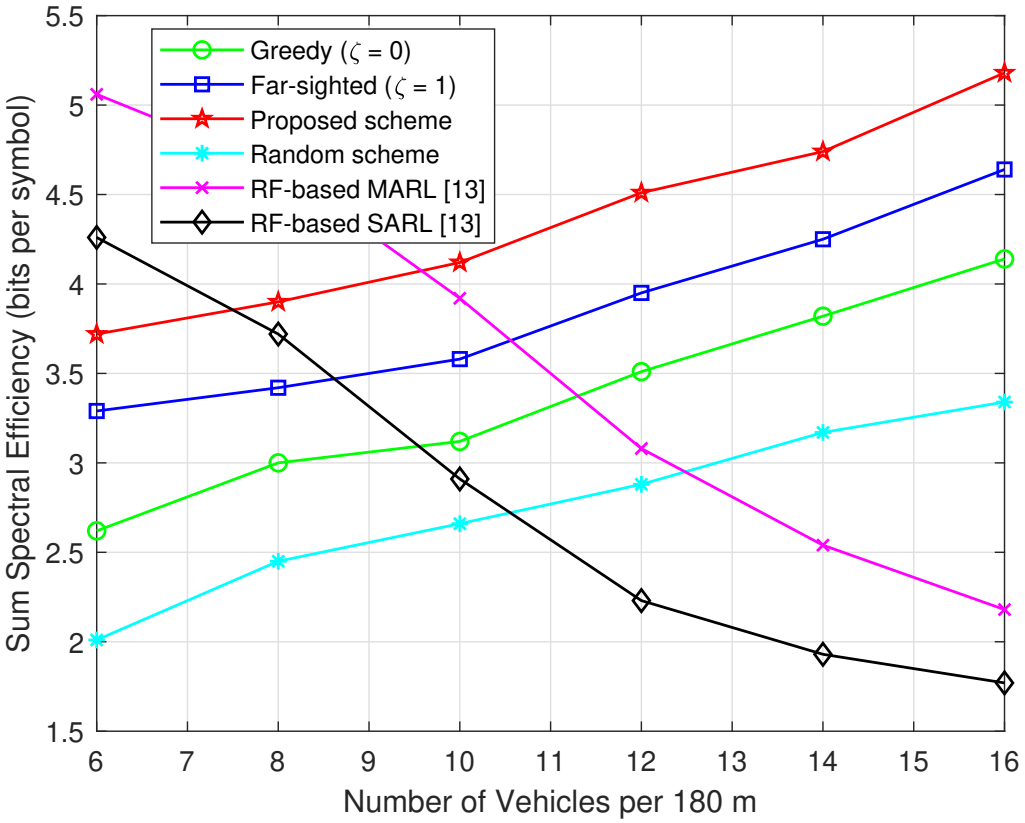

Figure 4.8: Comparison of sum spectral efficiency with different approaches when $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

of vehicles in Fig. 4.9, which shows that the latency reduces for different variants of our scheme using our proposed algorithm as the number of vehicles increases. Whereas for SARL and MARL schemes, it follows the opposite trend. This is because when the density of vehicles increases, the latency increases, and therefore, the delay performance falls. More importantly, the interference becomes stronger with an increase in vehicles density. Also, there is latency involved in the RF-based centralized system as it needs to communicate with the base station and receive feedback. Therefore, the latency increases with the increase of vehicles density. For our proposed scheme, there is no interference, which improves the spectral efficiency and hence, the latency with the increase of vehicles density. Our

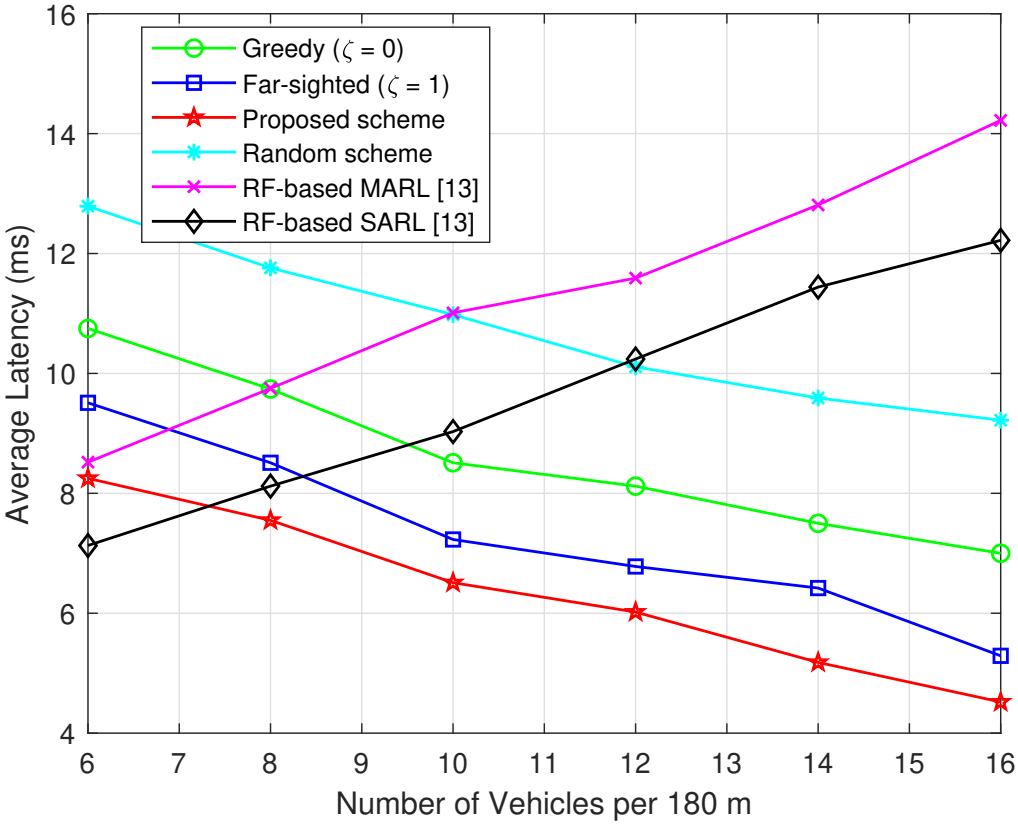

Figure 4.9: Comparison of average latency versus density of vehicle with different schemes when $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

scheme achieves the lowest average latency of 4.5 ms and the maximum of 8.2 ms when the density of vehicles is 16 and 6, respectively. Whereas for MARL, SARL and the random scheme, the average latency is 8.5 ms and 14.2 ms, 7.1 and 12.2, 12.9 ms and 9.2 ms, respectively, when the number of vehicles is 6 and 16. From this comparison, it is seen that our proposed algorithm achieves lower latency compared to other schemes.

To explore whether the proposed scheme can maximize the spectral efficiency and at the same time respect the latency and BER constraints, we present the CDF of BER and latency for the schemes under comparison. First, we compare the CDF of the observed latency considering the maximum latency of all available links at each time slot for 10000 epis-

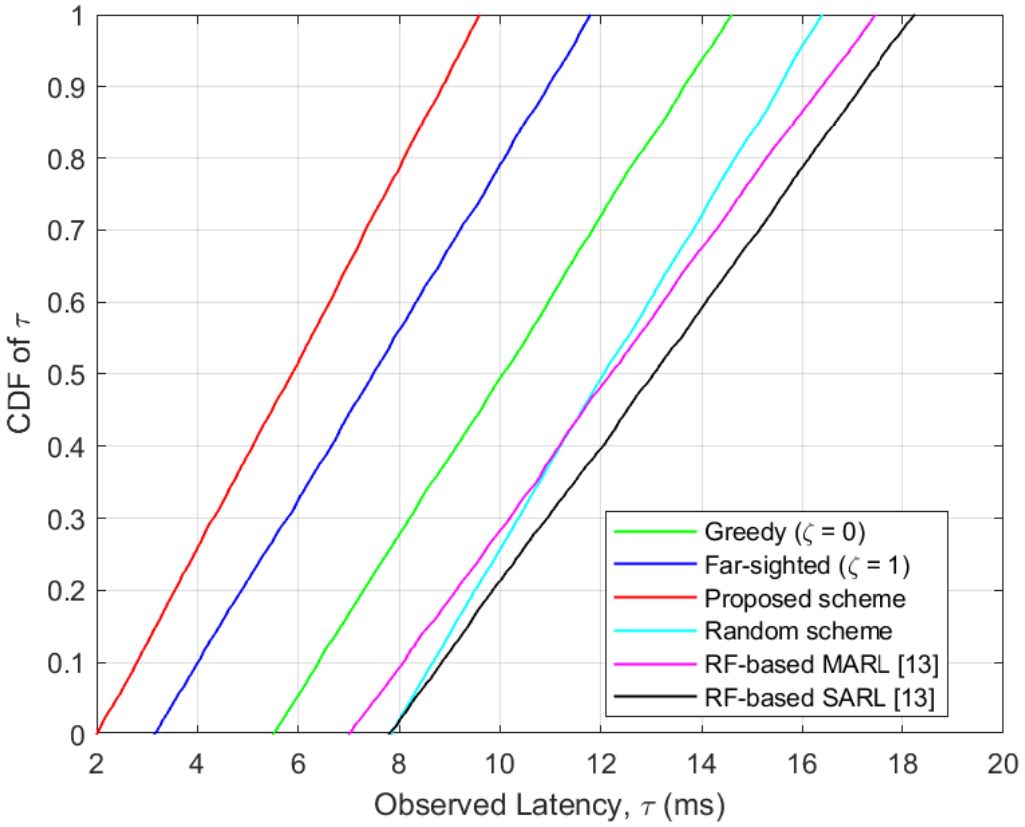

Figure 4.10: CDF of observed latency while considering the maximum latency of all the available link behind the agent for $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

odes in Fig. 4.10. From the figure, we observe that the proposed scheme can always satisfy the latency requirements of 10 ms whereas, the greedy, far-sighted and random methods, satisfy the constraint only 50%, 78%, and 27% of the time, respectively. At the same time, the RF-based MARL and SARL schemes meet the latency requirement for 29%, and 20% of the time, respectively. Therefore, we can conclude that our proposed multi-agent DRL-based vehicular OCC system can maximize the rate by satisfying latency constraints, whereas other schemes fail to respect the requirement most of the time.

Finally, Fig. 4.11 illustrates the comparison of CDF of the observed BER

for different schemes under comparison when the schemes have been optimized for 10000 episodes. In doing so, we examine only the maximum observed BER of all available links at each time slot, which will respect the minimum BER. From this figure, we note that our algorithm always satisfies the BER constraints of 10^{-4}. We can also see that the other algorithms violate the BER constraints most of the time. Specifically, far-sighted schemes satisfy BER requirements a maximum of 40%, whereas greedy and random schemes meet 27% and 8% of the time, respectively. Similarly to what we observed in Fig. 4.10, the proposed method also respects the BER requirement when other schemes satisfy it only for some time.

From the presented results, we can summarize that using the proposed adaptive modulation scheme and besides following a decentralized approach, we achieve better performance than the fixed modulation and centralized RF system.

4.6 Summary

In this chapter, we present a DRL-based spectral efficiency optimization scheme for a multiple vehicular OCC scenario while respecting BER and latency requirements. Firstly, we model the OCC channel and several performance parameters. Then, we formulate a sum spectral efficiency maximization problem considering a small set of modulation orders, as well as the BER and latency constraints. To reduce the complexity of the NP-hard problem, we formulate the optimization problem as an MDP problem, which enables us to find an optimal solution. We design the reward function considering the objective function. We then convert the constrained problem into an unconstrained problem through the Lagrangian relaxation method by relaxing the BER and latency constraints. To solve the problem, we employ deep Q-Learning to deal with large state-action spaces. We verify the performance of our proposed scheme through extensive simulations and compare it with various variants of our scheme as well as

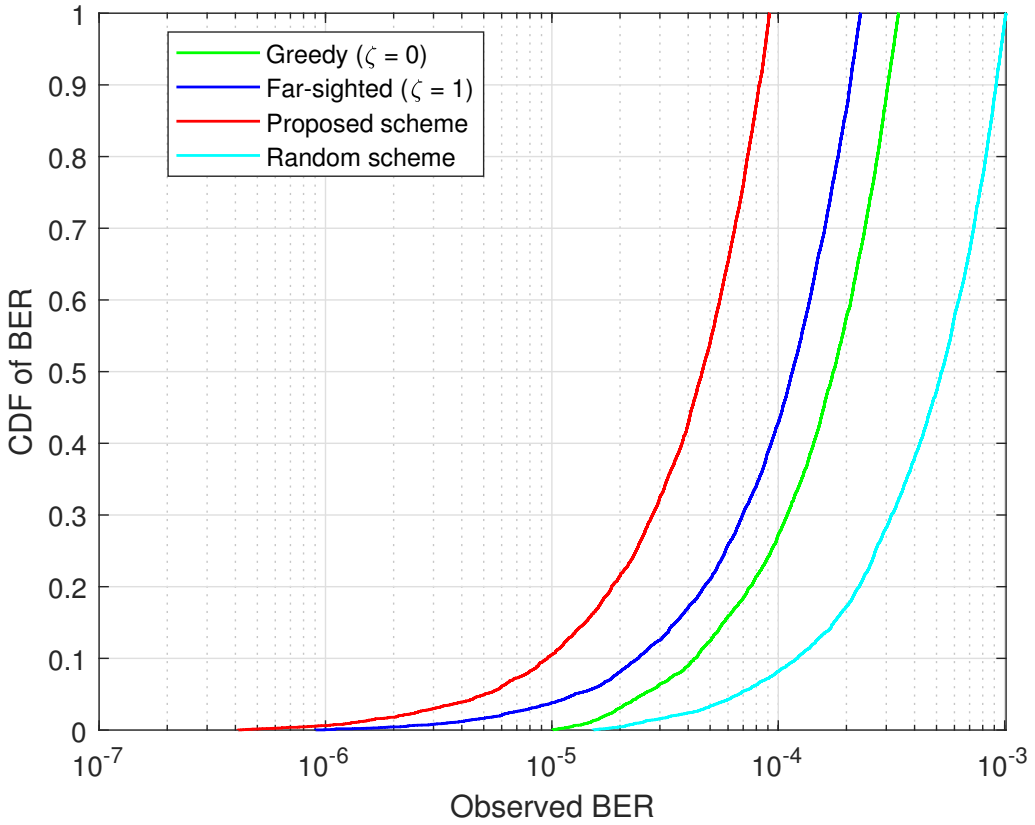

Figure 4.11: CDF of BER while considering the maximum BER of all the available link behind the agent for $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

schemes based on RF communications. Our system achieves better sum spectral efficiency and lower average latency compared to all the schemes under comparison. By observing the CDF of latency and BER, we can conclude that our system can satisfy ultra-low latency communication and BER constraints, while the rest of the schemes fail.

CHAPTER 5

Deep Reinforcement Learning based Ultra Reliable and Low Latency Vehicular OCC

5.1 Introduction

From the findings of Chapter 4, we see that we can maximize the communication rate while meeting low latency and BER requirements. We maintain low latency but could not ensure ultra-reliability. Providing efficient V2V communications is necessary, while the performance of the growing transportation systems depends on the availability of V2V communication links at extreme low latency and ultra-reliability [4]. The requirement to respect both latency and reliability requirements simultaneously makes vehicular communication a very challenging problem.

Since vehicular environments are time-varying and dynamic, it is challenging to respect uRLLC constraints. Further, vehicular communication systems become even more complex when they involve controlling various decision-making parameters, e.g., code rates, speed, distances, and modulation schemes. It is hard to solve these problems using traditional methods

because of their inherent complexity and the time required to solve them. DRL has emerged as a possible candidate to solve autonomous vehicular problems [13, 14]. DQN cannot be straightforwardly applied to continuous state-action spaces [17], which is the case for our proposed vehicular system. One of the approaches to solve continuous problems is to discretize the state-action spaces. However, this introduces suboptimality, as we may not find the optimal action because of discretization. This happens as inexperienced discretization needlessly discards information, which can be critical for solving the underlying problems. These issues can be alleviated by adopting the actor-critic DRL frameworks [17], where the DRL agent incorporates two parts, namely, the actor network and the critic network. The actor network controls the agent's behaviour by selecting actions, whereas the critic network refines the actor's choices to accomplish the optimal policy approximation. The Wolpertinger architecture [18] along with the actor-critic network converges faster than the vanilla actor-critic method over a large actions space by considering the nearest neighbour's actions of a proto-actor action selected by the actor network.

However, meeting uRLLC constraints necessitate the use of channel coding. LDPC codes are a promising candidate for uRLLC, which has been adopted in the 5G NR services [15]. As LDPC codes can help achieve a higher transmission rate, low latency and high reliability, we use them in our system. In this chapter, we propose an actor-critic DRL approach in vehicular OCC that aims at maximizing the achievable rate while respecting the uRLLC constraints. We apply the actor-critic DRL framework by adopting the Wolpertinger policy for our vehicular OCC system. In doing this, we optimize the achievable rate by selecting the optimal code rate, modulation scheme and speed of the vehicle. We use DDPG [17] to train the model. The main contributions of this chapter are summarized below:

- To the best of our knowledge, this is the first to use 5G NR LDPC codes in vehicular OCC to ensure uRLLC.
- We present a DRL based capacity maximization scheme subject to

selecting adaptive modulation schemes, deciding appropriate code rates and adjusting the speed of the vehicles while respecting uRLLC requirements and dealing with the massive continuous state-action spaces.

- We adopt the Wolpertinger architecture along with the actor-critic network to avoid exploring large action spaces over all the decision intervals.

- We evaluate the performance of the proposed DRL framework in terms of achievable capacity, BER, and transmission latency. The results show that the proposed actor-critic based DRL scheme achieves promising results and maximizes the transmission rate while satisfying the uRLLC constraints and outperforms the comparison schemes.

The remainder of this chapter is organized as follows. Section 5.2 outlines the OCC channel model and mathematical representation of the performance parameters of the proposed V2V system, while the formulation of the maximization problem and RL is presented in Section 5.3. Section 5.4 introduces the actor-critic deep reinforcement learning framework with Wolpertinger architecture. The simulation setup for the proposed system's performance evaluation is given in Section 5.5 followed by Section 5.6 where we provide the simulation results with respect to different performance parameters and comparison with various schemes under consideration. Finally, we summarize the contribution of this chapter in Section 5.7.

5.2 System Modelling

We start this section by introducing the considered vehicular OCC system parameters. We, then, discuss the employed LDPC channel codes and adaptive modulation schemes. Finally, we present the performance defining

parameters of the proposed vehicular OCC systems involving the achievable channel capacity and the observed transmission latency.

5.2.1 System Overview

We consider the same system model of Fig. 3.1, where each vehicle is an individual agent. Considering the advantages of adaptive modulation to improve the transmission rates and maintain the quality of service, we employ M-QAM. However, different modulation schemes can still be applied to our system. M-QAM has already been used in optical communications [89], which offers very low BER, high-speed, and flicker-free communication [77]. To further improve the transmission rate and guarantee low BER, i.e., ultra-reliability and low latency, we utilize the 5G NR LDPC code with the M-QAM scheme. We illustrate the overall block diagram of the OCC system employing the transmitter, OCC channel, and receiver in Fig. 5.1. The transmitter consists of an LDPC encoder, an M-QAM modulator, and a LED transmitter, whereas the receiver consists of an image sensor receiver, an M-QAM demodulator, and an LDPC decoder. We will describe the employed LDPC codes in Section 5.2.2. At the transmitter, the data bitstreams are first encoded using LDPC codes before mapping the channel encoded codewords into M-QAM symbols. Then, the coded data are transmitted over an OCC channel through LEDs. At the receiver, the camera captures the modulated light intensity as three different LED states, i.e., on, off, and mid. The originally transmitted information is then extracted from the detected intensity using M-QAM demodulation [78].

5.2.2 Channel Coding

Channel coding strongly affects the achieved throughput and reliability of a communication system. In light of the fact that our vehicular OCC system requires ultra-reliability and low-latency, 5G NR LDPC codes we have used, which have already been applied in optical communications [16]. 5G NR

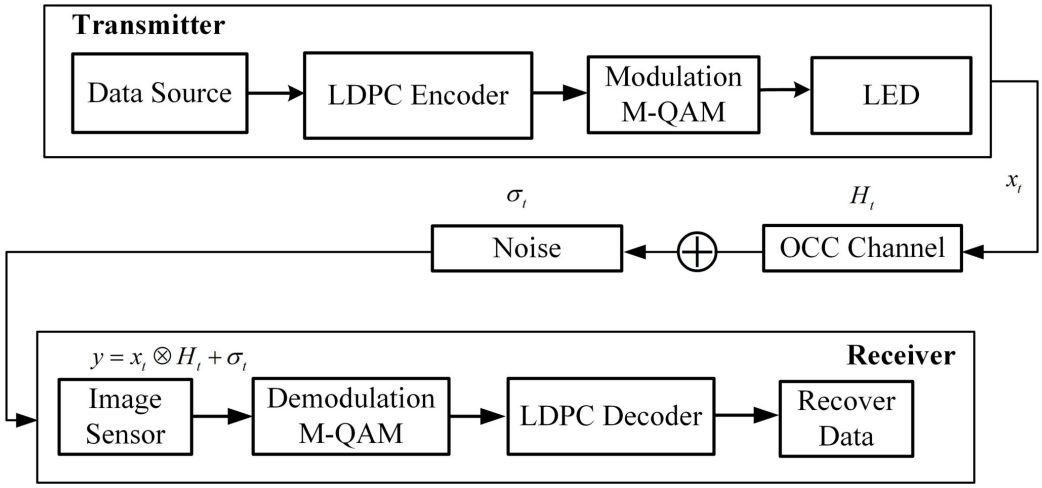

Figure 5.1: Block diagram of LDPC coded M-QAM for vehicular OCC.

system uses QC-LDPC as the data channel coding scheme because of the advantages of efficient implementation and offering improved performance [71]. The QC-LDPC coded-modulation can also resolve the weaknesses of having low reliability and high latency performance for arbitrary order of modulation formats [16, 72] while guaranteeing a low error rate for all code rates. A notable feature of the 5G NR LDPC codes is the flexibility to support a wide range of information block lengths ranging from 40 to 8448 bits and various code rates, $\varkappa$, ranging from 1/5 to 8/9 [15, 73]. 5G NR codes use a feedback channel to adapt protection, which makes them reliable and efficient. Therefore, we use 5G NR QC-LDPC channel coding over the GF(Q) for Q-ary QAM transmissions in our vehicular OCC systems.

For a GF size of $Q = 2^M$, the transmitter encodes the original data using Q-ary LDPC codes. Then, the encoded bits are sequentially mapped to symbol constellations with M-QAM modulation schemes. On the receiver side, the modulated symbols, i.e., codeblock, are accumulated for demodulating and decoding the originally transmitted information. Among the LDPC decoding algorithms, the Sum-Product Algorithm (SPA) is the most efficient in terms of BER performance [100]. As SPA has a higher computation cost, we have not employed it in our system, but instead, we use the Min-

Sum algorithm (MSA) [101]. MSA reduces LDPC decoding complexity by decreasing the number of multiplication operations on the SPA with only minor performance loss [102]. After LDPC decoding, the receiver uses a standard M-QAM demodulator to demodulate the incoming message in order to recover the original information message. In the Appendix B, LDPC encoder and decoder details are provided.

5.2.3 Optical Channel Model

We can model our OCC system as equivalent baseband model [79], and, thus, the received signal Y_t for a transmitted symbol X_t is given by

$$Y_t = X_t \otimes \rho H_t + \sigma_t, \tag{5.1}$$

where the $\otimes$ symbol denotes convolution, H_t is the channel DC gain, t is the time-frame index, and ρ is the receiver's responsitivity. The channel input X_t represents instantaneous optical power, which is non-negative, $X_t \geq 0$, and the transmitted optical power is given by

$$P = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} X_t dt \ . \tag{5.2}$$

5.2.4 Capacity and Latency Modelling

The channel capacity of a camera-based communication system for a code rate, $\varkappa$, with the employed modulation scheme is expressed as in (3.16) and [32]

$$\begin{aligned} C(d,\varkappa) =& \varkappa \, \frac{W_{\text{fps}} \, N_{\text{LEDs}} \, w\varrho}{6 \, \tan\left(\frac{\theta_l}{2}\right) d} \cdot \log_2(M(d)) \\ =& \varkappa \, \frac{l_0}{d} \cdot \log_2(M(d)), \end{aligned} \tag{5.3}$$

where $l_0 = \frac{W_{\text{fps}} \, N_{\text{LEDs}} \, w\varrho}{6 \, \tan\left(\frac{\theta_l}{2}\right)}$.

The transmission latency, $\tau(d,\varkappa)$, for a packet size, L, can be expressed as [46]

$$\tau(d,\varkappa) = \frac{L}{C(d,\varkappa)}, \tag{5.4}$$

Recall that in our system, we consider that the end-to-end latency is dominated by transmission latency, and therefore, we neglect the computational latency.

5.3 Problem Statement and MDP Formulation

5.3.1 Constrained Problem Formulation

Considering the proposed vehicular environment and ultra-reliable and low-latency communication requirements, we formulate an optimization problem that aims at maximizing the goodput of the considered vehicular OCC system by selecting the optimal modulation order and LDPC code rate from the available sets in 5G NR and adjusting the relative speed of the vehicle to the optimal value. The BER and latency are constrained to meet uRLLC conditions. Hence, our constrained maximization problem is formulated as:

$$\max_{\boldsymbol{\mathcal{M}},\, \boldsymbol{\mathcal{X}},\, \boldsymbol{v}} \quad C(d, \varkappa) = \varkappa \, \frac{l_0}{d} \cdot \log_2(M(d)) \tag{5.5}$$

$$\text{s.t.} \quad \text{BER}(d, \varkappa) \leq \text{BER}_{\text{max}}, \tag{5.6}$$

$$\tau(d, \varkappa) \leq \tau_{\text{max}}, \tag{5.7}$$

$$M(d) \in \mathcal{M}, \tag{5.8}$$

$$\varkappa \in \mathcal{X}, \tag{5.9}$$

where $\mathcal{M}$ is the set of QAM modulation orders, $\mathcal{X}$ is the set of LDPC codes, v is the relative speed of the vehicle, BER_{max} is the maximum target BER, and τ_{max} is the maximum allowable latency. To ensure uRLLC, the reliability is satisfied by maintaining the target BER as in (5.6), and the latency requirement is respected as in (5.7). The modulation scheme is chosen from a small set of available M-QAM modulation schemes, as shown in (5.8). The code rates are adjusted using the set of available 5G NR codes [73], as defined in the IEEE standard as presented in (5.9). We adapt the distance

by $d_t = d_{t-1} + v_t \cdot \Delta t$, where d_{t-1} is the distance at the prior state, and Δt is the time difference between two states.

By observing the optimization problem in (5.5), we notice that we have a NP-hard combinatorial problem [90], where finding the optimal solution is hard. We also have non-linear operations in (5.5) - (5.7). Solving this problem using a traditional optimization technique is time-consuming, where each vehicle should choose the speed, code rate, and modulation scheme individually. We can overcome these limitations using RL. In RL, the vehicles (agents) interact with the unknown environment to decide the optimal policy, i.e., selecting optimal code rate, speed, and modulation order, while adapting to the environmental changes. Before driving to the solution in the following section, we first formulate the optimization problem of (5.5) as a Markov Decision Process (MDP) in the next subsection.

5.3.2 MDP Modelling

The proposed optimization problem in (5.5) is formulated as an MDP, where each vehicle acts as an agent, and everything beyond the particular vehicle is regarded as the environment. The agent explores and interacts with the environment to have a better understanding of it and decides the capacity maximization policies based on their observations of the environmental state. Next, we present the state space $\mathcal{S}$, the action space $\mathcal{A}$, and the reward function, r of the considered RL framework.

State Definition

At each time t, the agent observes the state s_t from the environment. In our system, the state consists of three parameters: the backward distance, d_t, the transmitting modulation scheme, M_t, from the set $\mathcal{M} = \{4, 8, 16, 32, 64\}$, and the code rate, $\varkappa_t$, from the set $\mathcal{X} = \{\text{5G NR codes}\}$ [73]. We summarize the state at time t as $s_t = \{d_t, M_t, \varkappa_t\}$.

Action Definition

At the current state s_t, the agent chooses an action a_t from the action set $\mathcal{A}$ following a policy π. For our considered system, the action space is the combination of selecting a modulation scheme from the set $\mathcal{M}$, code rate from the set of $\mathcal{X}$, and adjusting the relative speed, v_t. In summary, the action space is expressed as $a_t = \{\triangle M_t, \triangle \varkappa_t, \triangle v_t\}$, where $\triangle$ represents the change of values of the respective parameters, e.g., $\triangle M_t$ refers to the change in modulation scheme.

Reward Function

Following the action taken at the current state, the agent receives a reward. Note that, an effective design of the reward is imperative for the learning algorithm to obtain the desired goal, which is achieved by experience and a multitude of attempts. Therefore, the reward function that controls the learning should be relevant to the objective. In our framework, the reward function is the weighted sum of the rewards corresponding to inter-vehicular distance, goodput (5.5), BER constraint (5.6), and latency constraint (5.7). Firstly, we model the reward for the distance changes, r_t^{d}, as follows:

$$r_t^{\mathrm{d}} = \begin{cases} -1 \times (d_{\text{stop}} - d_t), & d_t < d_{\text{stop}} \\ \frac{1}{d_t - d_{\text{stop}}}, & d_t > d_{\text{stop}} \end{cases} \tag{5.10}$$

where d_{stop} is the stopping distance, which is equal to the sum of covered distance by the vehicle to travel after the brakes are activated, i.e., braking distance, and the covered distance to travel due to driver's reaction time, i.e., reaction distance, after observing a situation [103]. We, then, model the reward for the reliability, i.e., BER, r_t^r, as:

$$r_t^r = \mathbb{1}_{\mathrm{b}}(\text{BER}_{\text{max}} \geq \text{BER}_t), \tag{5.11}$$

where $\mathbb{1}_{\mathrm{b}}$ stands for the indicator function for the BER. The indicator function returns 1 if the condition for BER requirement is satisfied or 0 otherwise. Similarly, the latency is constrained so that it meets the low latency

requirement. Accordingly, the reward for latency, r_t^{τ}, is modelled as follows:

$$r_t^{\tau} = \mathbb{1}_{\tau}(\tau_{\text{max}} \geq \tau_t), \tag{5.12}$$

where, $\mathbb{1}_{\tau}$ is the indicator function for latency that returns 1 for true condition and 0 otherwise.

Finally, from the above modelling, the overall weighted sum of rewards, r_t, is expressed as

$$r_t = \omega_d \, r_t^{\text{d}} + \omega_b r_t^{r} + \omega_{\tau} r_t^{\tau} + \omega_c \, C(d, \varkappa), \tag{5.13}$$

where, ω_d, ω_b, ω_{τ}, and ω_c are positive weights to balance between the distance, BER, latency, and communication rate rewards. The weights can be adjusted based on the system requirements. For instance, a higher value for ω_c gives higher priority to selecting actions that maximize the goodput at every step.

The return from a state in the MDP is the discounted sum of future rewards received by the agent, $G_t = \sum_{j=0}^{\infty} \zeta^j r_{t+j+1}$. The goal of the RL is to maximize the expected return over all episodes, i.e., $\max \ \mathbb{E}[G_t(s_t, a_t)]$, is the expected return starting from a given state, s_t, and taking an action, a_t, following a policy, π_t, thereafter. The Q-learning based action-value function is commonly used in RL algorithms. It can be expressed in a recursive relationship using the Bellman equation:

$$Q^{\pi}\left(s_t, a_t\right)) = r(s_t, a_t) + \zeta \mathbb{E}_{a_{t+1} \sim \pi}\left[Q^{\pi}\left(s_{t+1}, a_{t+1}\right)\right], \tag{5.14}$$

where $\mathbb{E}_{a_{t+1} \sim \pi}$ stands for expectation of future accumulated reward $Q^{\pi}\left(s_{t+1}, a_{t+1}\right)$, while taking an action following a policy π at time $t+1$. Here, the agent considers both current and next state to calculate the Q value for each action $\mathcal{A}$.

5.4 Proposed Solution

The rate of convergence of the Q-learning algorithm depends on the size of the state-action space. When the state-action space is small, the RL agent

can explore all the state-action pairs rapidly and find the optimal policy. However, if the state-action space is large, the Q-learning convergence rate slows down since many state-action pairs may not be explored by the RL agent and the storage size of the Q-table is extremely large. In particular, when the state-action space is infinitely large, the Q-learning algorithm requires significant time to converge and significant storage space for the Q-table. The problem quickly becomes intractable when the cost of evaluating the Q function increases since the execution complexity grows linearly with the increase in state-action spaces.

Moreover, if the environment is time-varying, similarly to our case, we need to deal with the continuous state-action spaces. Discretizing the state, action spaces is a way of dealing with the issue of the continuous problem, but there is a trade-off between the discretization and the size of the state-action space. Thus, we have to sacrifice the performance because we may require to generalize the state-action space while discretizing them. Unfortunately, Q-learning cannot be straightforwardly applied to continuous action spaces. This is because, in continuous spaces, we require optimization at every timestep to find the greedy policy in Q-learning. Then the optimization becomes too slow to be practical when we have large, unconstrained function approximators and nontrivial action spaces. This motivates the approach described in this cap. In particular, we use an actor-critic framework based on the DDPG algorithm [104], where we utilize a new policy architecture termed as Wolpertinger architecture [18]. This architecture avoids the heavy computational cost of evaluating Q-function on every action taken.

5.4.1 Wolpertinger Architecture

As we have already mentioned that the proposed policy architecture follows the Wolpertinger architecture [18], this policy builds upon the actor-critic [11] framework. The Wolpertinger architecture consists of three main components: actor network, K-nearest Neighbour (KNN), and critic net-

work, which works in three steps. First, the actor network takes states as its input and provides a single proto-actor $\hat{a}$ at its output. Then, KNN receives the proto-actor as its input and calculates the L_2 distance between every valid action and the proto-actor in order to expand the proto-actor to action space, $\mathcal{A}_K$, with K elements and each element being a possible action $a \in \mathcal{A}$. Finally, the critic network takes $\mathcal{A}_K$ as its input and refines the actor network based on the Q value. We train the policy using the DDPG algorithm [17], which is applied to update both critic and actor networks. We use multi-layer neural networks as function approximators for the actor and critic functions.

We provide a more detailed description of the key components of the Algorithm 2.

The actor network

The actor network maps the state s from the state space $\mathcal{S}$ to the action space and chooses a proto-actor $\hat{a} \in \mathcal{A}$ from the valid actions. The network is expressed as a function and characterized by θ^μ. Finally, the proto-actor is defined as follows:

$$\begin{aligned} &\mu(s \mid \theta^\mu) : \mathcal{S} \rightarrow \mathcal{A} \\ &\mu(s \mid \theta^\mu) = \hat{a}. \end{aligned} \tag{5.15}$$

K-nearest neighbours (KNN)

The generation of the proto-actor can help reduce the potentially high computational complexity due to the large size of the action space. However, reducing the high-dimensional action space to only a single actor will lead to poor decision making. To resolve this, the KNN mapping, g_K, is applied to expand the actor's choice of action to a subset of valid actions from $\mathcal{A}$. The set of actions returned by g_K is denoted by $\mathcal{A}_K$:

$$\mathcal{A}_K = g_K(\hat{a}_t), \tag{5.16}$$

where

$$g_K = \arg\min_{a \in \mathcal{A}}^{K} \mid a - \hat{a} \mid^2 . \tag{5.17}$$

We determine the K nearest neighbours of the proto-actor using (5.17). Here, $\mid a - \hat{a} \mid^2$ is the distance of the features between the chosen action a and the proto-actor $\hat{a}$. When the actor network selects the proto-actor, the agent will traverse the action space to find the K nearest feature distances, and then the action set will be determined accordingly.

The critic network

To avoid selecting an action that leads to a low Q-value frequently, a critic network is introduced to refine the actor. The deterministic target policy is characterized as below:

$$Q\left(s_t, a_t \mid \theta^Q\right)\big) = \mathbb{E}\left[r(s_t, a_t) + \zeta Q\left(s_{t+1}, a_{t+1} \mid \theta^Q\right)\right], \tag{5.18}$$

where θ^Q is the parameters of the critic network. The critic network evaluates all actions in the expanded action space, and the action that provides the maximum Q-value is chosen as

$$\pi^*(s_t) = \arg\max_{a_t \in \mathcal{A}} \; Q^*(s_t, a_t), \forall s \in S. \tag{5.19}$$

The critic calculates the Q value while considering the current state s_t and the next state s_{t+1} as its input. The critic network evaluates all actions in $\mathcal{A}_K$, and chooses the action that provides the maximum Q-value, as follows:

$$a_t = \arg\max_{a_t \in \mathcal{A}_K} Q(s_t, a_t \mid \theta^Q). \tag{5.20}$$

Update: At each timestep, the actor and critic networks are updated by sampling uniformly a minibatch from the replay buffer. Because DDPG is an off-policy algorithm, the reply buffer can be large, allowing the algorithm to benefit from learning across a set of uncorrelated transitions. Therefore,

the actor policy is updated using DDPG with a minibatch size $N_{\mathcal{B}}$, which is given as

$$\nabla_{\theta^{\mu}} J \approx \frac{1}{N_{\mathcal{B}}} \sum_{t} \nabla_{a} Q\left(s, a \mid \mu^{Q}\right) |_{s=s_t, a=\mu(s_t)} \nabla_{\theta_{\mu}} \mu\left(s \mid \theta^{\mu}\right) \mid s_t, \tag{5.21}$$

and the critic is updated by minimizing the loss:

$$L = \frac{1}{N_{\mathcal{B}}} \sum_{t} \left(y_t - Q\left(s_t, a_t \mid \theta^{Q}\right)\right)^2, \tag{5.22}$$

where

$$y_t = r_t + \zeta Q'\left(s_{t+1}, \mu'(s_{t+1} \mid \theta^{\mu'}) \mid \theta^{Q'}\right). \tag{5.23}$$

Implementing (5.22) directly with neural networks becomes unstable in many environments. Since the updated $Q(s, a \mid \theta^{\mu})$ network is used to calculate the target value (5.23), the Q update is prone to divergence. Instead of directly copying the weights, we present a similar target network used in [54] as the solution but is adjusted for actor-critic while using "soft" target updates. In doing so, we calculate the target values by creating a copy of the actor and critic networks, $Q'(s, a \mid \theta^{\mu'})$ and $\mu'(s \mid \theta^{\mu'})$, respectively. The weights of these target networks are then updated by having them slowly track the learned networks as

$$\theta^{Q'} \leftarrow \beta\theta^{Q} + (1-\beta)\theta^{Q'}, \tag{5.24}$$

$$\theta^{\mu'} \leftarrow \beta\theta^{\mu} + (1-\beta)\theta^{\mu'}, \tag{5.25}$$

where $\beta \ll 1$ is the soft target update rate. This means that the target values are constrained to change slowly while improving the stability of learning.

In contrast to the general Q-learning, where the balance between exploration and exploitation is controlled using a ϵ-greedy method [11], a major challenge in continuous action spaces learning is exploration. Fortunately, the DDPG algorithm can separately deal with the exploration problem from the learning algorithm. Hence, we define an exploration policy

Algorithm 2 Actor-Critic Algorithm

Randomly initialize critic network $Q\left(s, a \mid \theta^{Q}\right)$ and $\mu(s \mid \theta^{\mu})$ with weights θ^{Q} and θ^{μ}.

Initialize target network Q' and μ' with weights $\theta^{Q'} \leftarrow \theta^{Q}$, $\theta^{\mu'} \leftarrow \theta^{\mu}$

Initialize SUMO environment and replay memory according to system requirements.

for episode **do**

  for each timestep t **do**

    Receive observation state s_t

    Actor: Receive proto-action from actor network $\hat{a}_t = \mu(s_t \mid \theta^{\mu})$.

    KNN: Retrieve k approximately closest actions $\mathcal{A}_K = g_K(\hat{a}_t)$

    Critic: Select action $a_t = \arg\max_{a_t \in \mathcal{A}_K} Q(s_t, a_t \mid \theta^{Q})$ according to the current policy

    Execute action a_t, and compute reward r_t and observe new state s_{t+1}

    Store transition (s_t, a_t, r_t, s_{t+1}) in replay memory.

    Sample a random mini batch of $N_{\mathcal{B}}$ transitions (s_t, a_t, r_t, s_{t+1}) from replay memory

    Set target $y_t = r_t + \zeta Q'\left(s_{t+1}, \mu'(s_{t+1} \mid \theta^{\mu'}) \mid \theta^{Q'}\right)$

    Update critic by minimizing the loss using (5.22)

    Update the actor policy using the sampled policy gradient using (5.21)

    Update the target networks with $\beta \ll 1$ using (5.24) and (5.25)

    Update the state

    Update features space $\mathcal{F}$

    Update rate

  end for

end for

μ' by adding sampled noise from a noise process n_t to the actor policy

$$\mu'(s_t) = \mu(s_t \mid \theta_t^{\mu}) + n_t, \tag{5.26}$$

where n_t is chosen to suit the environment. We consider temporally cor-

related noise to explore well in the environment using the similar process introduced in [105].

5.5 Experimental Setup

In this section, we present the simulation setup for the proposed actor-critic based DRL scheme in vehicular OCC system. In particular, we start by presenting the microscopic traffic simulation of SUMO [97]. We, then, present the considered parameters for the proposed actor-critic scheme and training workflow.

5.5.1 SUMO Framework

In order to implement our vehicular environment, we have chosen SUMO which already includes a set of different driver models and where is relatively easy to include additional models. Thus, we transform the proposed vehicular environment into a corresponding SUMO map, where each vehicle is an agent. The vehicles enter randomly in the SUMO environment and then move or leave the map following the SUMO mobility model set by the system. The interaction between the SUMO framework and the DRL agent is managed by a middleware, which is termed as Traffic Control Interface (TraCI). The agent can retrieve various features of the vehicle from the SUMO network, such as the inter-vehicular distance, the speed of the vehicle, the current position of the agent, and so on.

5.5.2 OCC System Design

To present the efficiency of the proposed OCC-based communication scheme, we consider the communication of 10^{11} bits and a packet size of 5 kbits. Please note that our BER requirement is 10^{-7}. For our simulation, we consider the 5G NR LDPC codes set from the IEEE standard [106]. The required stimulation parameters are shown in Table 4.3. We train the sys-

tem model with transmission of zero codewords, i.e., all the bits of the codeword are zero, which are sufficient for the training as the channel is symmetric. On the transmitter side, the zero codewords are encoded by the LDPC encoder and, after M-QAM modulation, are transmitted through the LoS OCC channel. On the receiver side, the data is first demodulated by the M-QAM demodulator and then decoded by the LDPC decoder. The error is computed by comparing the received codeword with the zero codeword.

5.5.3 Actor-critic DRL Framework

Training Parameters Settings

In this subsection, we introduce the actor-critic-based DRL network settings and the considered training parameters. The individual actor and critic network has three fully connected layers, including an input layer, a hidden layer, and an output layer. The input layer has $(d + \mathcal{M} + |\mathcal{X}|)$ nodes since the state space combines the distance, modulation scheme, and code rate, where $d = 150, \mathcal{M} = 5$. We consider distance up to 150 m, which is sufficient to maintain communication quality and avoid collisions and $\mathcal{M} = 5$ because we use only 5 modulation schemes as the set. Whereas the output layer has $(\triangle M + \triangle \varkappa + \triangle v)$ nodes, as in our proposed system, the action includes the change in modulation scheme, code rate, and velocity where $\triangle M = 5$, and $\triangle v = 60)$. The hidden layer has 250 neurons. We adopt a typical ReLU as the actor and critic networks' activation functions [95]. For learning the neural network parameters, we set the initial learning rate α to 10^{-4} for both the actor and critic networks. Whereas, for the soft target updates we set $\beta = 0.001$, which is sufficient to balance between the optimality and computational cost. The final layer weights and biases of both the actor and critic are initialized from a uniform distribution to ensure the initial outputs for the policy and value estimates were near zero. We use TensorFlow [99] in our simulations to implement deep reinforcement learning algorithms. We use RMSPro optimizer [98] as the training

Table 5.1: List of DRL hyper-parameters and their values

Parameter, Notation	**Value**
Mini-batch size, $N_{\mathcal{B}}$	64
Replay memory size	10^{11}
Number of hidden layer (Neurons)	$1(250)$
Discount factor, ζ	0.98
Exploration rate, ϵ	0.05
Activation function	ReLU
Optimizer	RMSProp
Learning rate (used by RMSProp), α	10^{-4}
Soft target updates rate, β	0.001
Gradient momentum (used by RMSProp)	0.95

algorithm, which minimizes the loss function and updates DQN network parameters.

In our implementation, we train the actor-critic based DRL scheme for 10000 episodes, which we find sufficient to have better performance convergence. For the exploration noise process, we use temporally correlated noise to effectively explore the environments. We use the Ornstein-Uhlenbeck process models [105] with mean value equal to 0.15 and variance equal to 0.2, which results in temporally correlated values centered around 0. We set the discount factor ζ to 0.98 for our proposed scheme. For exploration, we consider $\epsilon = 0.05$ for the ϵ-greedy algorithm. We train the network with minibatch sizes of 64 while having a replay buffer size of 10^{11} to store the transitions in the memory. We also perform normalization to bring the different sub-rewards corresponding to distance, BER, latency, and transmission rate in (5.13) to a similar scale. This normalization improves the performance and provides training stability for the NN model. Specifically, we normalize the reward function of distance (5.10) and rate of (5.5) to keep the scale of (5.13) between 0 and 1. The training

parameters are listed in Table 5.1.

Training Procedure

The training workflow of our proposed actor-critic based DRL algorithm is introduced in Algorithm 2. At each training step, t, the agent observes the current state s_t (distance, modulation scheme, and code rate) from the environment. Then, the proto-actor obtained by the actor network based on the current policy is passed to the KNN algorithm, and the expanded action set (change in modulation scheme, code rate, and velocity) will be evaluated by the critic network. After the chosen action is executed in the environment, the transition (s_t, a_t, r_t, s_{t+1}) will be stored in the replay buffer at the end of this epoch. Next, a minibatch with size, $N_{\mathcal{B}}$, will be randomly sampled from the memory and replayed to update the actor and critic networks. Then critic network is updated by minimizing the loss (5.22), and the actor policy is updated using the sampled policy gradient (5.21). Finally, the target network is updated by slowly varying the weights of (5.24) and (5.25). To evaluate the K-nearest neighbour actions, we consider K ratio as 0.1 of the action space $\mathcal{A}$. Please note that, throughout our simulation, we refer to the episode or timestep as the decision interval of our scheme.

In this section, simulations are conducted to investigate the performance of the proposed system model and rate optimization schemes in vehicular OCC. We start by evaluating different performance metrics of the proposed system model to get a better understanding of the interplay among the various parameters of our system.

5.5.4 Comparison Schemes

We investigate the performance of the proposed DRL-based actor-critic scheme, termed hereafter as the proposed scheme against different methods for to get insights on the system performance. We present a brief summary of all the schemes under comparison below:

- **Proposed scheme**: By the proposed scheme, we refer to our DRL-based vehicular OCC system, where each vehicles is an agent considering other vehicles as environment. In this case, we employ the settings as we discuss in Sections 5.5 and 5.5.3. We set the discount factor ζ to 0.98.

- **Greedy**: This is one of the variants of our scheme, where we set the discount factor to $\zeta = 0$ in (5.22), while we keep all other parameters of the system as reported in Table 5.1. In this scenario, the agent chooses the action which maximizes only the immediate reward.

- **Farsighted**: This method is a variant of our scheme, where we set the discount factor to $\zeta = 1$ in (5.22), while we keep all other parameters of the system as reported in Table 5.1. This scheme focuses on the future rewards and ignores immediate rewards.

- **RF-based Scheme [13]**: This is a RF technology based resource allocation scheme presented in [13]. For this scheme, we adapt the hyper-parameters according to our proposed scheme while keeping the environment unchanged. This scheme considers centralized learning, which involves communication between server and the agent. This system incurs extra delay due to having feedback loop.

5.6 Performance Evaluation

5.6.1 Simulation Results

We start by exploring the training convergence of the proposed actor-critic based DRL scheme by performing an ablation study for addressing the trade-off between the different weight settings of the total rewards in (5.13), namely distance ω_d, BER ω_b, latency ω_τ, and rate ω_r rewards. For ease of visual representation, we only demonstrate five particular settings, including (i) $\omega_d = 0.1$, $\omega_b = 0.1$, $\omega_\tau = 0.1$, $\omega_c = 0.7$; (ii) $\omega_d = 0.1$, $\omega_b = 0.4$,

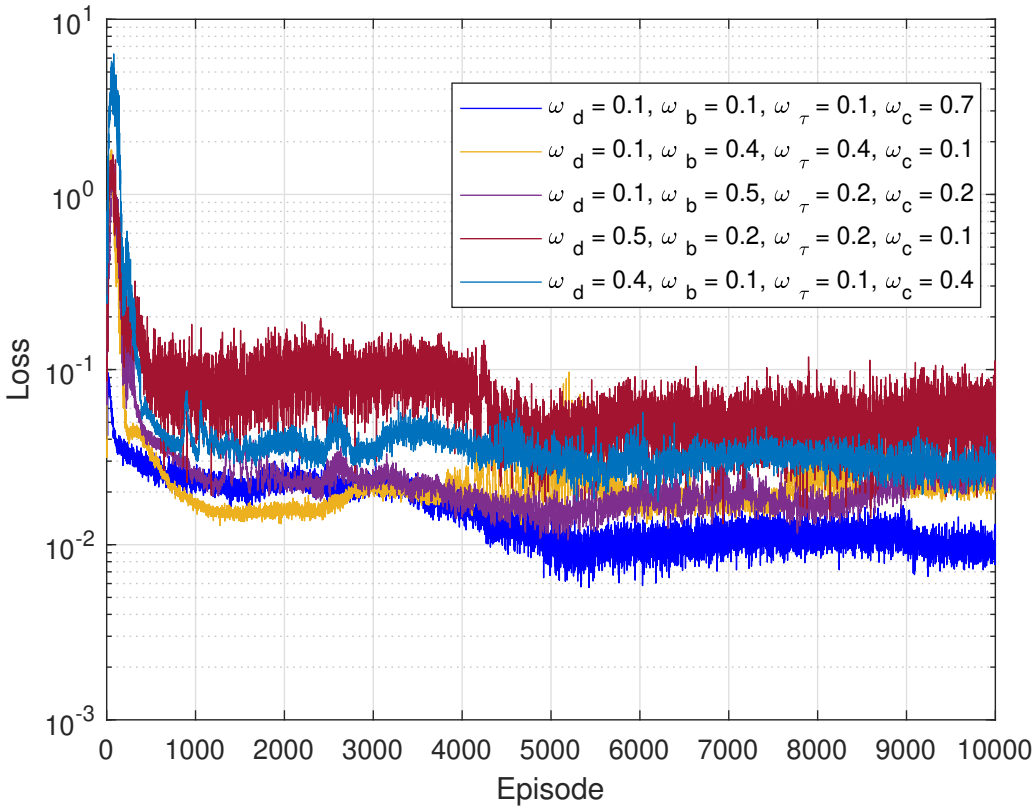

Figure 5.2: Convergence of loss function for different weight settings of sub-reward function with learning rate $= 10^{-4}$.

$\omega_\tau = 0.4$, $\omega_c = 0.1$; (iii) $\omega_d = 0.1$, $\omega_b = 0.5$, $\omega_\tau = 0.2$, $\omega_c = 0.2$; (iV) $\omega_d = 0.5$, $\omega_b = 0.2$, $\omega_\tau = 0.2$, $\omega_c = 0.1$; and (v) $\omega_d = 0.4$, $\omega_b = 0.1$, $\omega_\tau = 0.1$, $\omega_c = 0.4$, as shown in Fig. 5.2. From the figure, we observe that setting (i) converges after 5000 decision episodes and presents better loss performance when we assign higher weight related to goodput. Other settings have more elevated losses compared to setting (i). Though setting (ii) demonstrates better performance until 3000 episodes, setting (i) overcomes (ii) after 3000 episodes as the DRL agent takes some time to provide a balance between the action and the achieved rewards. Therefore, we adopt this weights setting (i) for the rest of our performance evaluation.

To verify the improvement of rewards over decision interval, we illustrate the cumulative rewards for the different variants of our proposed

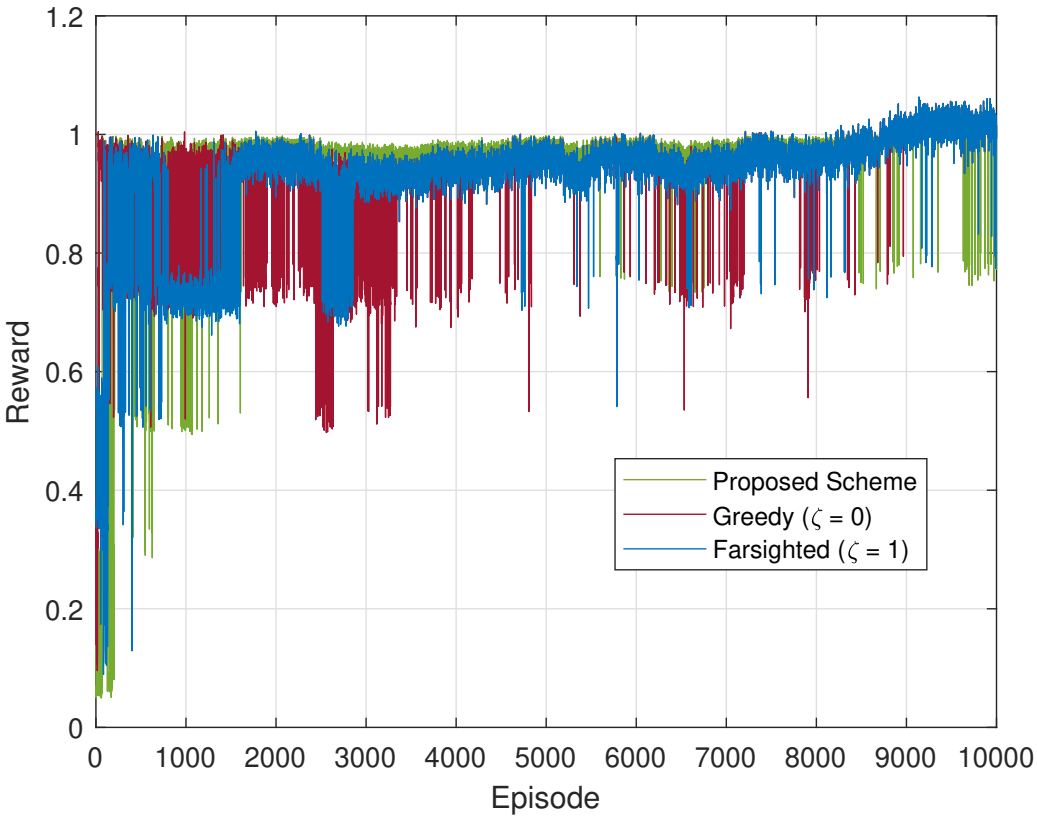

Figure 5.3: Reward per training episode for the proposed scheme and its variants with learning rate $= 10^{-4}$.

scheme, i.e., the proposed scheme ($\zeta = 0.98$), greedy ($\zeta = 0$) and farsighted ($\zeta = 1$) for 10000 episodes in Fig. 5.3. From the figure, we see that the proposed scheme demonstrates higher rewards over all the decision intervals, whereas the greedy and farsighted schemes display fluctuating rewards in most cases. We also observe that the reward traverses over 1 after 9000 episodes. This happens because we present the cumulative rewards over the episodes, which has the effect of the discount factor towards the future rewards for both of the schemes. In contrast, the reward never exceeds 1 for the greedy scheme because the discount factor has no impact on the future rewards. Therefore, we can conclude that the proposed scheme achieves higher rewards than the other variants of our scheme.

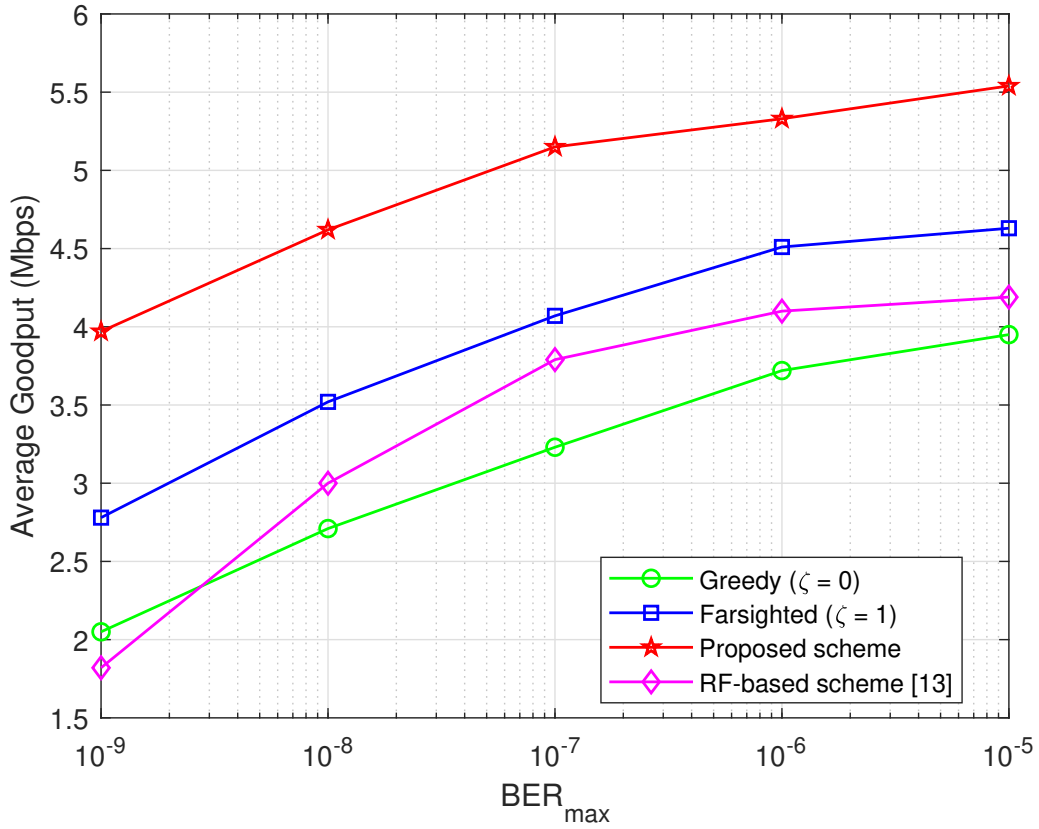

Figure 5.4: Comparison of average rate by varying the BER_{max} requirement for all schemes under comparison.

After demonstrating the training implementation, we now examine different communication performance metrics for our actor-critic based vehicular OCC system. Consequently, we evaluate the schemes under comparison with respect to goodput, latency and reliability. Please note that for the RF-based scheme, we require communication between the server and agent back and forth through feedback link, which involves an extra delay that is usually 1 ms to 12 ms. For our simulation, we consider 2 ms as the additional delay for the feedback, which is a favourable setting for the RF scheme though other settings can also be used as user necessities.

First, we investigate the effect of BER on the average goodput (Mbps) in Fig. 5.4 to measure the goodput performance for various BER_{max}. From

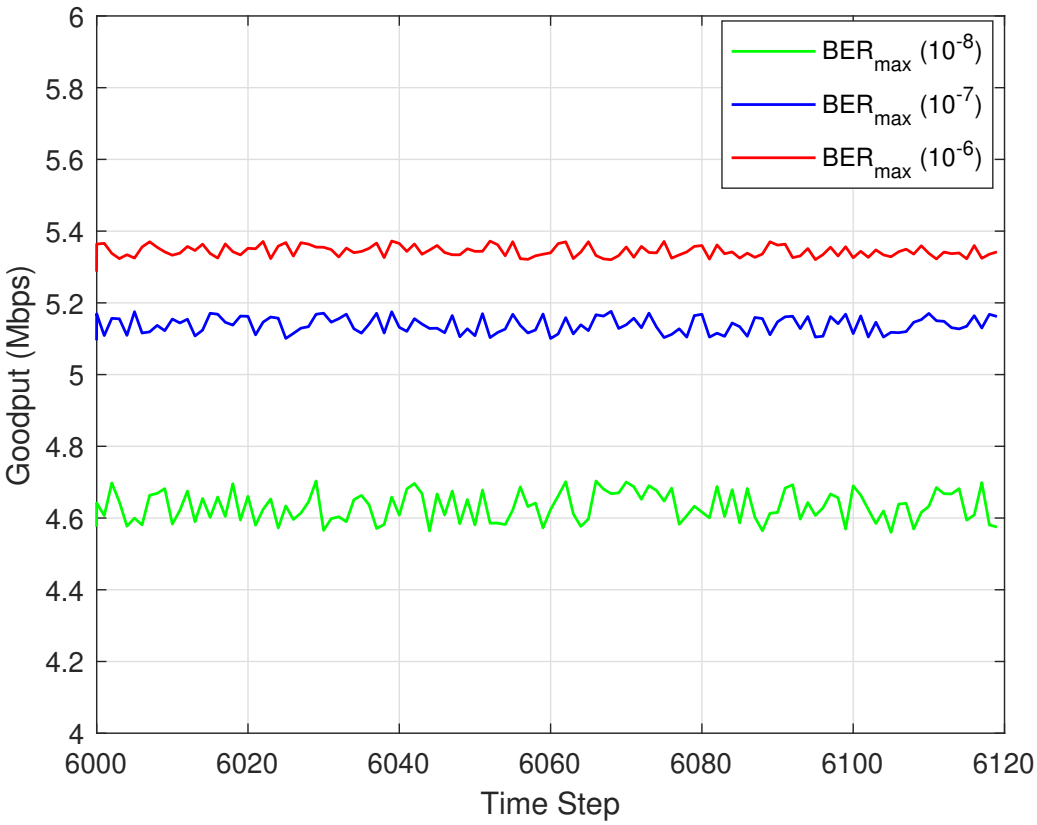

Figure 5.5: Comparison of achievable goodput by our scheme over timestep considering different BER_{max} requirement.

the figure, we see that the average goodput increases as we reduce the BER_{max} requirements from 10^{-9} to 10^{-5}. This happens because of using less strong LDPC codes. We also observe that the proposed scheme outperforms all the other schemes under comparison. Initially, the RF-based method achieves the lowest rate of all the schemes but performs better than the greedy method beyond $\text{BER}_{\text{max}} = 10^{-8}$. Hence, it is seen that when the BER requirements are more tight, other schemes fail to meet the constraint (5.6), and therefore performance degrades. As a result, the average goodput is lower than that of the proposed scheme all the time. For example, for the proposed scheme when $\text{BER}_{\text{max}} = 10^{-9}$, the average goodput is 4 Mbps, whereas, for greedy, farsighted and RF-based schemes, it is 2 Mbps, 2.75

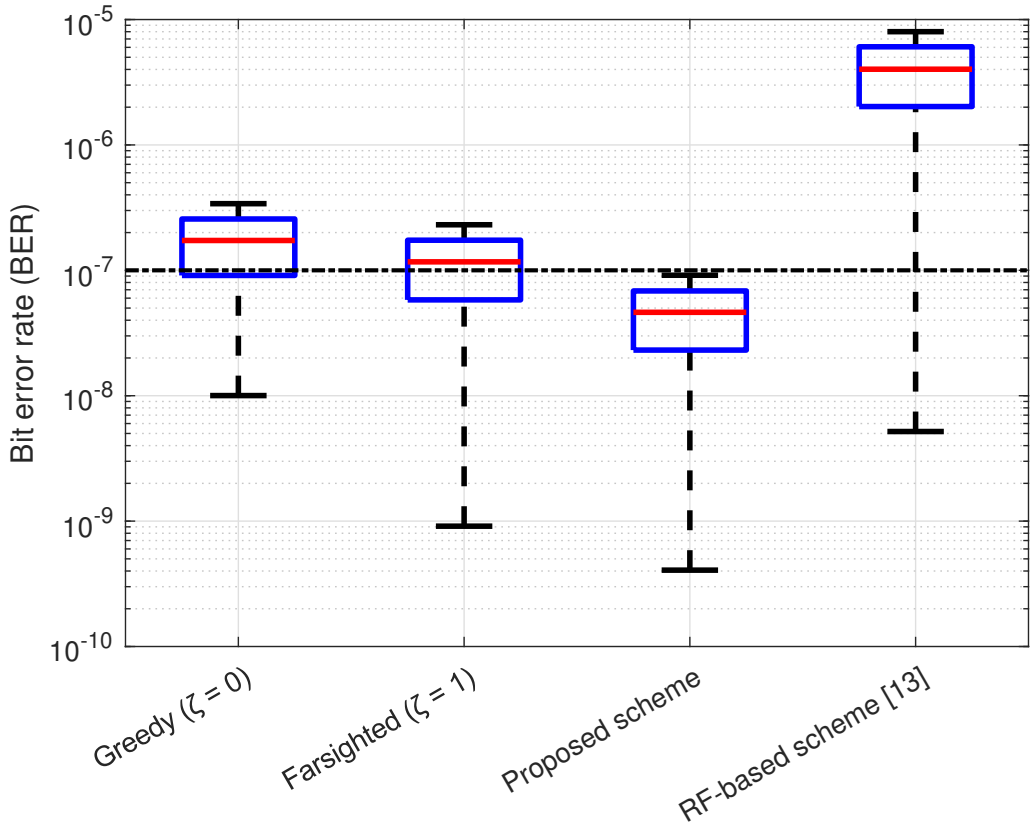

Figure 5.6: Box plot to justify how the reliability requirement is satisfied considering our maximum allowable BER 10^{-7}.

Mbps, and 1.8 Mbps, respectively. Meanwhile, the average goodput increases to 5.5 Mbps, 3.95 Mbps, 4.6 Mbps, and 4.2 Mbps for the proposed, greedy, farsighted and RF-based scheme, respectively, when $\text{BER}_{\text{max}} = 10^{-5}$. We will examine the effect of BER further in detail in the later part of this section.

In an effort to present the robustness of selecting the code rate, we evaluate the goodput (Mbps) for three different BER_{max} requirements in Fig. 5.5. For this simulation, we illustrate the goodput across 120 runs, where a sample is taken from the whole run at 6000 - 6120 timesteps. From the figure, we observe that the goodput varies near the average values for all BER_{max}. We also notice that we achieve higher goodput, and there is less

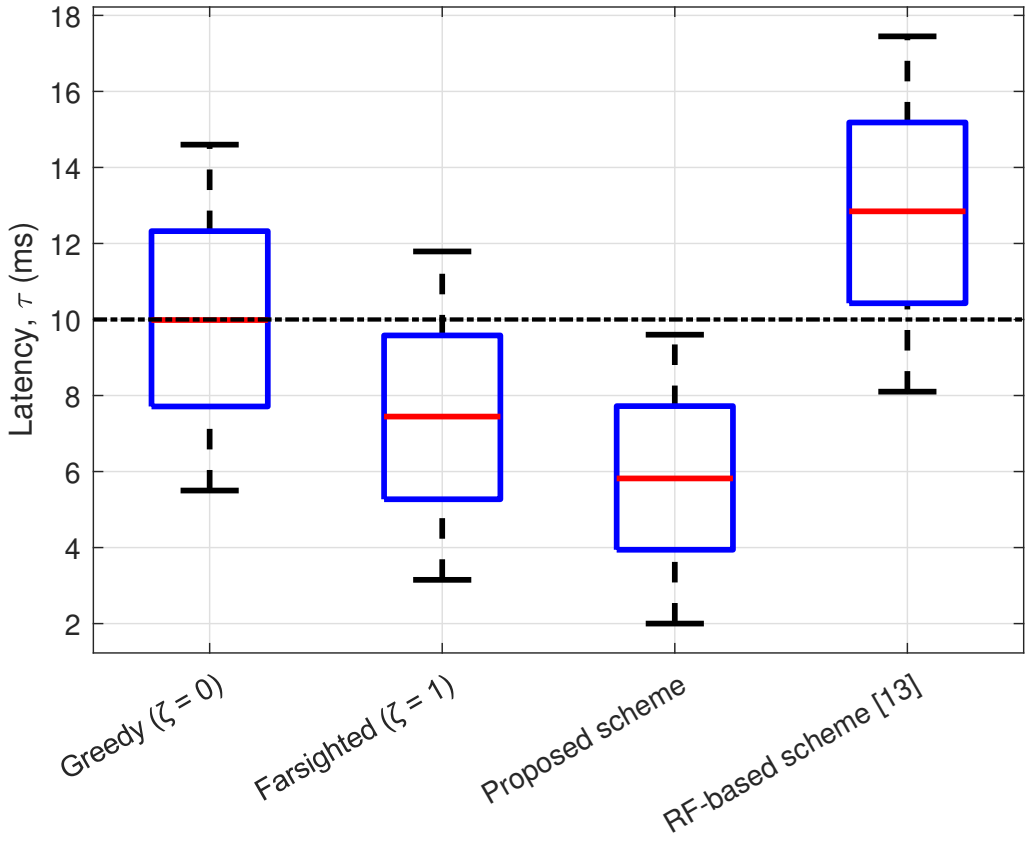

Figure 5.7: Box plot to verify how the latency requirement is satisfied considering our latency requirement 10 ms

fluctuation between decision intervals when the BER requirement is lower, e.g., 5.36 Mbps for 10^{-6}. Since at lower BER_{max}, the probability of violating (5.11) is becoming smaller, the goodput is higher, and variation between one decision interval to another is less and vice versa.

To visualize how the proposed scheme respects the uRLLC requirements while maximizing the goodput, we analyze the BER and latency performance for the various schemes under comparison. Please note that in this cap, we consider meeting BER of 10^{-7} and latency of 10 ms as the ultra-reliability and low-latency requirements, respectively. We execute the simulation for 10000 decision episodes to investigate the BER and latency data. We then generate boxplots over all the available data and compare the res-

ults with all the schemes under comparison. First, we illustrate the boxplot of the BER to demonstrate whether all the schemes under comparison meet the reliability requirement in Fig. 5.6. We have also plotted a reference line to present our BER constraint of 10^{-7} (dashed black line). From the figure, we observe that our proposed algorithm always satisfies the reliability requirement, whereas other schemes cannot respect the constraint most of the time. Specifically, for greedy, farsighted and RF-based schemes, the maximum BER is 3.4×10^{-7}, 2.3×10^{-7}, and 8×10^{-6}, respectively.

Finally, we present the boxplot of observed latency for all the schemes under comparison in Fig. 5.7. Similar to the BER performance, we evaluate the boxplot to examine how the low latency requirement (10 ms) is satisfied by the different comparison schemes. Like in the previous comparison, we also show a reference line for the latency constraint (dashed black line) in Fig. 5.7. From the figure, we can note that our proposed scheme always respects the low latency requirements of 10 ms, while the other three schemes fail to meet the constraint most of the time in our simulation. In particular, for the greedy, farsighted and RF-based schemes, the maximum observed latency is 14.5 ms, 11.8 ms, and 17.5 ms, respectively. From this comparison, we can conclude that our proposed vehicular OCC system can maximize the goodput while guaranteeing uRLLC, while the RF-based schemes cannot meet the delay requirements.

5.7 Summary

In this chapter, we introduce an actor-critic DRL framework in vehicular OCC by selecting the optimal code rates and modulation schemes as well as changing the relative speed of the vehicles while respecting uRLLC requirements. First, we model the vehicular OCC system. To support variable rate and ultra-reliability, we use 5G NR LDPC code rate optimization for the M-QAM scheme. We solve the continuous optimization problem using an actor-critic algorithm with Wolpertinger architecture. We verify our pro-

posed scheme through numerous simulations and compare it with several variants of our scheme and an RF communication-based scheme. The average goodput of our proposed scheme shows a considerably higher value compared to other schemes under comparison. We neglect the effect of weather conditions in this chapter. Our proposed scheme can guarantee uRLLC while maximizing the goodput, whereas other methods fail most of the time. This happens because interference free OCC DRL-based systems achieve higher rates even at low BER requirements, and the code rate optimization scheme offers ultra-reliability.

CHAPTER 6

Multi-agent Deep Reinforcement Learning for uRLLC in Vehicular OCC

6.1 Introduction

In the previous two chapters, we showed that DRL helps to solve the large scale and continuous problems and the requirement of reliability is met through code rate optimization. Whereas OCC helps to respect low-latency constrain. In Chapter 4, we presented a DRL scheme by discretizing the state-action spaces, where we can meet the low latency but could not achieve ultra-reliability. Discretization reduces the overall system performance. If the discretization is too coarse probably it will lead to a sub-optimal solution, if it is too fine it will require enormous time to find a solution and there will be no optimality guarantee. In Chapter 5, we utilize code rate optimization to ensure uRLLC [15] and actor-critic based DRL scheme to solve the continuous problem, where we considered a single link. In this case, the solution can be sub-optimal when we have multiple links scenarios in a real urban road scenario. This happens because the information of other links is unknown to the vehicular agent. The performance of other links is optimized using the optimal policy from the observed single link performance. To

mitigate the above challenges, we present a multiple links vehicular OCC system in this chapter. In this scheme, we propose a rate maximization scheme to meet the uRLLC of multiple links while optimizing the parameters of all links, i.e., speed, code rate, and modulation scheme. To this aim, we use a multi-link actor-critic based DRL framework in the proposed OCC system for continuous and large state-action spaces. We use Wolpertinger architecture with actor-critic to limit the search for optimal action to the nearest neighbour's actions of a proto-actor action selected by the actor network. We evaluate the performance of the actor-critic based DRL scheme in vehicular OCC system and compare it with the performance of the different schemes under comparison, e.g., No Coding scheme, Single Link Optimization (SLO) scheme, greedy scheme and farsighted scheme. The results demonstrate that the proposed method shows superiority against its counterpart schemes. The results further make it obvious the benefit of using code rate optimization and actor-critic based DRL scheme in multi-agent vehicular OCC system to meet uRLLC.

The rest of the chapter is organized as follows. We outline the vehicular OCC system model in Section 6.2. Afterwards, in Section 6.3, we present the proposed optimization problem before presenting the simulation setup in Section 6.4. We then show the evaluation results of the proposed scheme in Section 6.5. Finally, we draw the conclusion remarks in Section 6.6.

6.2 System Model

In this section, we present the considered system model and parameters of vehicular OCC. Then, we specify the performance defining metrics of OCC in terms of the BER, the achievable rate, and the observed transmission latency.

For the employed M-QAM modulation scheme, the achievable capacity of the OCC system for the link b for the 5G NR LDPC codes with code rate,

$\varkappa$, is expressed by following (5.3) and [107]:

$$C^b(\varkappa) = \varkappa \frac{W_{\text{fps}} N_{\text{LEDs}} w \varrho}{6 \tan\left(\frac{\theta_l}{2}\right) \cdot d^b} \cdot \log_2(M^b). \tag{6.1}$$

We have already mentioned in Section 3.3.3 that the End-to-End (E2E) latency is contributed by the transmission latency because we process a small amount of data, i.e., the decision information from transmitter to receiver. Therefore, the transmission latency for packet size, L, is given by following [107]:

$$\tau^b(\varkappa) = \frac{L}{C^b(\varkappa)}. \tag{6.2}$$

6.3 Proposed Problem Formulation

6.3.1 Constrained Problem Formulation

The objective of this chapter is to present an optimization framework to maximize the communication rate of the proposed vehicular environment which meeting uRLLC requirements. To this aim, we formulate an optimization problem that aims at maximizing the sum rate of the vehicular OCC system by selecting the optimal modulation order and code rate and adjusting the relative speed of the vehicle to the optimal value. We set the BER and latency to a predefined value to respect the uRLLC conditions imposed by the system. Finally, we formulate the constrained maximization problem as:

$$\max_{\mathcal{M},\mathcal{X},v} \quad \frac{1}{B} \sum_{b=1}^{B} C^b(\varkappa), \tag{6.3}$$

$$\text{s.t.} \quad \text{BER}^b(\varkappa) \leq \text{BER}_{\text{tgt}}, \ \forall b; \tag{6.4}$$

$$\tau^b(\varkappa) \leq \tau_{\max}, \ \forall b; \tag{6.5}$$

$$M^b \in \mathcal{M}, \ \forall b; \tag{6.6}$$

$$\varkappa^b \in \mathcal{X}. \ \forall b; \tag{6.7}$$

Constraint (6.4) indicates that the reliability is satisfied by maintaining a target BER, and the latency requirement is respected by (6.5), for ensuring uRLLC. The modulation scheme is chosen from a small set of available modulation schemes, as shown in (6.6), whereas the 5G-NR codes are adjusted from the set as in (6.7).

Our formulated problem (6.3) is an NP-hard problem, which is hard to solve using distributed optimization methods [90]. Further, there are non-linear operations in (6.3) - (6.5), which makes it time-consuming and complex to find the optimal solution. This happens because in traditional optimization methods, the decision-making parameters, i.e., code rate, modulation scheme and speed, should be selected in a distributed way. To overcome these challenges, we introduce RL in our system, which interacts with the environment and finds the optimal policy by adapting to the environmental changes.

We start the next subsections by describing our problem spaces as MDP and then detail our policy architecture, demonstrating how we train it using DDPG methods in an actor-critic framework.

6.3.2 Modelling of MDP

Our proposed maximization problem can be modelled as an MDP, with a tuple ($\mathcal{S}$, $\mathcal{A}$, p, r, ζ) [11]. We outline the state space $\mathcal{S}$, the action space $\mathcal{A}$, and the reward function, r of the considered RL framework as follows:

State space

At time t, each agent interacts with the environment and observes the state, s_t. The state in our system has three components: the backward distance vector, $\mathbf{d}_t^b = (d_t^1, \cdots, d_t^B)$, the transmitted modulation scheme, $\mathbf{M}_t^b = (M_t^1, \cdots, M_t^B)$, from the set $\mathcal{M}$ = {4, 8, 16, 32, 64}, and the code rate vector, $\boldsymbol{\varkappa}_t^b = (\varkappa_t^1, \cdots, \varkappa_t^B)$, from the set $\mathcal{X}$. In summary, the state is outlined as $s_t = \left\{\mathbf{d}_t^b, \mathbf{M}_t^b, \boldsymbol{\varkappa}_t^b\right\}$

Action space

From the state s_t, the agent takes an action a_t from the set $\mathcal{A}$, consisting of adjusting the relative speed, v_t, selecting modulation scheme $\mathbf{M}_t^b \in \mathcal{M}$, and code rate $\boldsymbol{\varkappa}_t^b \in \mathcal{X}$. We summarize the action space as $a_t = \left\{v_t, \mathbf{M}_t^b, \boldsymbol{\varkappa}_t^b\right\}$.

Reward function

The agent receives a reward based on the action, a_t, taken from the state, s_t. In our framework, the reward function is the weighted sum of the rewards corresponding to inter-vehicular distance, BER constraint (6.4), latency constraint (6.5), and goodput (6.3). We first model the reward for the distance changes, r_t^{d}, as follows:

$$r_t^{\mathrm{d},i} = \begin{cases} -1 \times (d_{\text{stop}} - d_t^b), & d_t^b < d_{\text{stop}} \ , \\ \frac{1}{d_t^b - d_{\text{stop}}}, & d_t^b > d_{\text{stop}} \ , \end{cases} \tag{6.8}$$

where i is the index of the agent. For notational simplicity we drop the i hereafter. To satisfy the BER requirement, we model the reward for BER, r_t^b, as:

$$r_t^r(\boldsymbol{\varkappa}) = \mathbb{1}_{\mathrm{b}}(\text{BER}_t^b(\boldsymbol{\varkappa}) \leq \text{BER}_{\text{max}}), \tag{6.9}$$

Similarly, the latency is maintained so that it obeys the latency constraint. Accordingly, the reward for latency, r_t^τ, is modelled as follows:

$$r_t^\tau(\boldsymbol{\varkappa}) = \mathbb{1}_\tau(\tau_t^b(\boldsymbol{\varkappa}) \leq \tau_{\text{max}}), \tag{6.10}$$

Considering the above definition, we express the overall weighted sum of the rewards, R_t, is expressed as

$$R_t = \omega_d \, r_t^{\mathrm{d}} + \omega_b r_t^r(\boldsymbol{\varkappa}) + \omega_\tau r_t^\tau(\boldsymbol{\varkappa}) + \omega_c \, \frac{1}{B} \sum_{b=1}^{B} C^b(\boldsymbol{\varkappa}), \tag{6.11}$$

where, ω_d, ω_b, ω_τ, and ω_c are positive weights to balance between distance, BER, latency, and communication rate rewards.

After each interaction with the environment in time slot, t, the agent receives a reward r_t The goal of RL is to maximize the total future discounted reward: $G_t = \sum_{j=0}^{\infty} \zeta^j r_{t+j+1}$.

6.3.3 Proposed Solution

The Q-Learning convergence is slow when it involves large state-action space. To overcome the slow convergence, we use DRL in our system. Moreover, the vehicular networks are time-varying and dynamic, where the state-action spaces are continuous. Therefore, we propose actor-critic based DRL framework with Wolpertinger architecture to solve the continuous and large state-action problems. This helps us apply the closest action to proto-actor in the set directly to the environment or select the highest valued action from the set related to the cost function.

As we have already explained the Wolpertinger architecture at Section 5.4.1 in Chapter 5, we will discuss them briefly in here.

The actor network: The actor network maps the state $s \in \mathcal{S}$ to the action space $\mathcal{A}$ and chooses a proto-actor $\hat{a}$ from the valid actions set $\mathcal{A}$. The actor network is characterized by θ^{μ}, where we define the proto-actor as in (5.15).

K-nearest neighbours (KNN): We determine the K nearest neighbours of the proto-actor using (5.17). Through KNN mapping, the actor can expand its choice to the nearest neighbour of proto-actor action determined by the actor network.

The critic network: The critic network refines the actor's choices to accomplish the optimal policy approximation. The deterministic target policy for the critic network is expressed similar to as (5.18):

$$Q\left(s_t, a_t \mid \theta^{Q})\right) = \mathbb{E}\left[r(s_t, a_t) + \zeta Q\left(s_{t+1}, a_{t+1} \mid \theta^{Q}\right)\right], \tag{6.12}$$

The critic assesses all the actions from the expanded action space $\mathcal{A}_K$ and calculates the Q-value. Finally, the critic selects the action that offers the maximum Q-value using

$$a_t = \arg\max_{a_t \in \mathcal{A}_K} Q(s_t, a_t \mid \theta^{Q}). \tag{6.13}$$

Update: We update the actor policy using DDPG algorithm for a mini-

batch size $N_{\mathcal{B}}$ as

$$\nabla_{\theta^{\mu}} J \approx \frac{1}{N_{\mathcal{B}}} \sum_{t} \nabla_{a} Q\left(s, a \mid \mu^{Q}\right) |_{s=s_t, a=\mu(s_t)} \nabla_{\theta_{\mu}} \mu\left(s \mid \theta^{\mu}\right) \mid s_t, \tag{6.14}$$

The update of critic network is done through minimization of the loss:

$$L = \frac{1}{N_{\mathcal{B}}} \sum_{t} \left(y_t - Q\left(s_t, a_t \mid \theta^{Q}\right)\right)^2, \tag{6.15}$$

where

$$y_t = r_t + \zeta Q'\left(s_{t+1}, \mu'(s_{t+1} \mid \theta^{\mu'}) \mid \theta^{Q'}\right). \tag{6.16}$$

The weight parameters of actor network θ^{μ} and critic network θ^{Q} through "soft" target updates. This helps to improve the stability of learning. This done by slowing changing the weight parameters and are given by

$$\theta^{Q'} \leftarrow \beta\theta^{Q} + (1-\beta)\theta^{Q'}, \tag{6.17}$$

$$\theta^{\mu'} \leftarrow \beta\theta^{\mu} + (1-\beta)\theta^{\mu'}, \tag{6.18}$$

6.4 Simulation Setup

In this section, we start by providing a brief overview SUMO framework. We then provide the details of the considered training parameters in our simulation.

6.4.1 SUMO Framework

We implement our vehicular environment in SUMO framework, where the vehicles are modelled according to our proposed system modelling. TraCI maintains the interaction between the SUMO framework and the DRL agent from where the agent receive different information about the vehicular network, including distance, speed, position of the vehicle, and can feed the decision back to the SUMO again.

6.4.2 Training Parameters

For OCC system design, we consider the communication of 10^{11} bits and a packet size of 5 kbits, where we train the model with transmission of zero codewords. We consider the code rates of the 5G NR LDPC codes as defined in the IEEE standard [106]. In our simulation, we consider the following training parameters and setting for the actor-critic based DRL framework. Each of the actor and critic networks have four fully connected layers, including an input layer, two hidden layers, and an output layer. The two hidden layers have 500 and 250 neurons, respectively. The input layer has $(d + M + |\mathcal{X}|)$ nodes since the state space combines the distance, modulation scheme, and code rate, where $d = 150$ and $M = 5$. Whereas the output layer has $(\triangle M + \triangle \varkappa + \triangle v)$ nodes, as in our proposed system, the action includes the change in modulation scheme, code rate, and velocity. We utilize ReLU activation function [95]. We employ TensorFlow [99] as the training algorithm to minimize the loss, where we set the initial learning rate, α, to 10^{-4}. We set the soft target value to $\beta = 0.001$.

We run the training of the proposed the actor-critic based DRL scheme for 10000 episodes. We use temporally correlated noise from the Ornstein-Uhlenbeck process models [105] with mean 0.15 and variance 0.2 for the exploration. In our proposed scheme, we set the discount factor, ζ to 0.98. We consider minibatch sizes of 64 and a replay buffer size of $10^1 1$ to store the transition in the memory. We normalize the sub-rewards values corresponding to distance, BER, latency, and transmission rate in (6.11) to be on a similar scale between 0 and 1. The stimulation parameters are summarized in Table 4.2.

6.5 Performance Evaluation

In this section, we perform numerous simulations to understand the performance of our proposed multi-agent actor-critic based DRL scheme in vehicular OCC. Before presenting the simulation results, we first provide a

overview of the various schemes under comparison.

6.5.1 Comparison Scheme

In this subsection, we provide a brief summary of all the schemes under comparison in this chapter.

- **Proposed scheme**: We refer our actor-critic vehicular OCC system to the proposed scheme. In our scheme, we employ the settings as we have mentioned in Section 6.4.2. We set the discount factor to 0.98. Further, we perform code rate optimization in multi-agent RL system, where we observe all the links behind the agent vehicle and optimize the policy based on the observations.

- **No Coding**: In this scheme, we consider a multi-link system similar to our proposed scheme, without using channel coding. This scheme helps us understand the impact of channel coding to the system performance. Hence, we termed this scheme as 'No Coding' scheme. This is an extension of our previous scheme (Chapter 4), where we maximized the sum spectral efficiency without performing the code rate optimization. This scheme considered only latency constraint, the extension takes into account also the reliability constraint.

- **Single Link Optimization**: This is a variant of our proposed scheme, where we observe the state of a single link. Then, we apply the optimized code rate and modulation order to all other links. In this scheme, we consider the states of other links are unknown to the agent while keeping track of the observed single link only. Therefore, we called it as SLO scheme.

- **Greedy**: This method is a variant of our scheme, where we set the discount factor to $\zeta = 0$ in (6.15), while we keep all other parameters of the systems as reported in Table 4.2. In this scenario, the agent chooses the action which maximizes only the immediate reward.

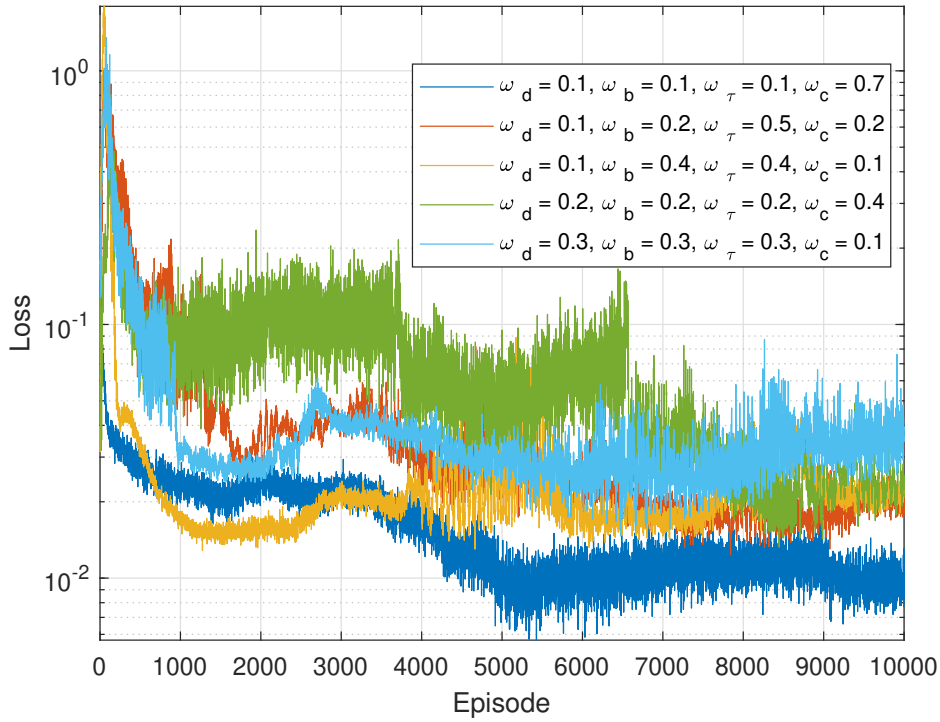

Figure 6.1: Convergence of loss function for different weight settings of sub-reward function with $\alpha = 10^{-4}$.

- **Far-sighted**: This method is another variant of our scheme where, we consider the discount factor to be $\zeta = 1$ in (6.15). In this scheme, we maintain all other parameters of the systems to be same as reported in Table 4.2. This scheme takes future rewards into account more strongly and ignores immediate rewards.

6.5.2 Simulation Results

We first start our evaluation by presenting an ablation study of different weight values of reward function (6.11) to select the setting that leads to faster convergence of the loss function. In particular, we present five different settings weight values of distance, ω_d; BER, ω_d; latency, ω_d and

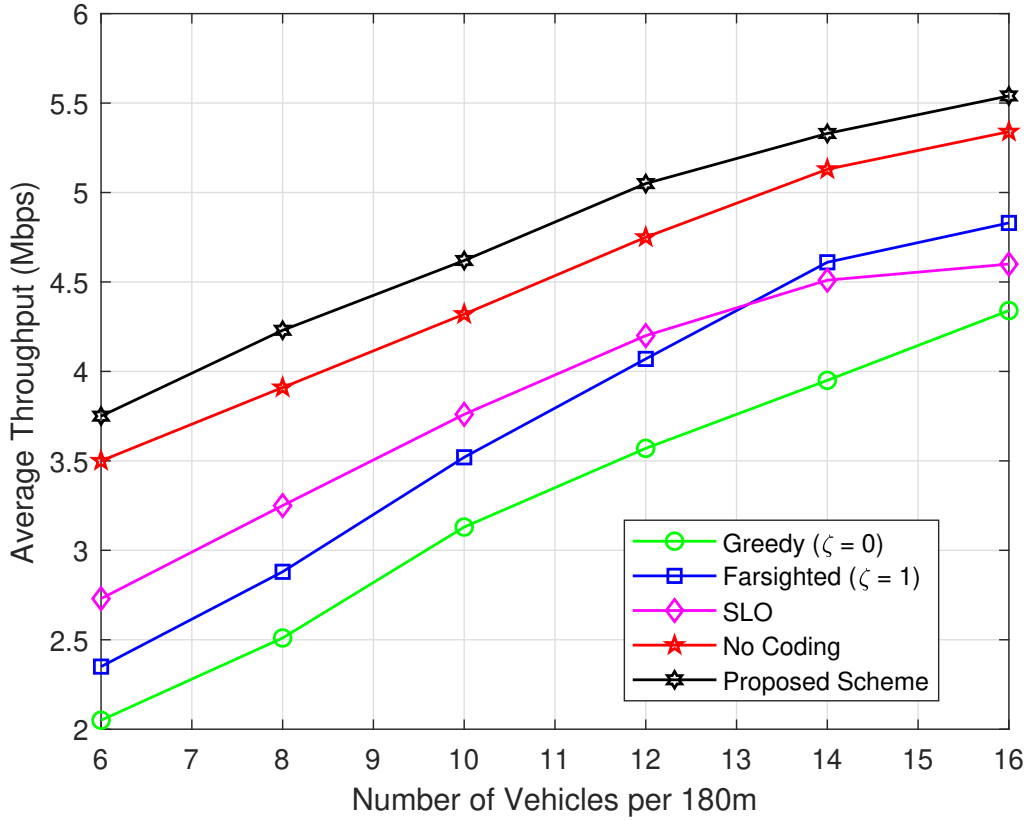

Figure 6.2: Comparison of sum goodput with different approaches for learning rate $\alpha = 10^{-4}$.

rate, ω_d for easy visualization in Fig. 6.1 though more settings could be illustrated. From the figure we see that the setting $\omega_d = 0.1$, $\omega_b = 0.1$, $\omega_\tau = 0.1$, $\omega_c = 0.7$ provides faster convergence and leads to lower loss. Therefore, we employ this setting for the rest of our evaluation.

We now study the impact of training over the different performance parameters for the schemes under comparison with respect to goodput, latency and reliability. We first evaluate the effect of the density of vehicles on the average goodput and average latency for different schemes under comparison. To this end, we vary the density of the vehicle in the range from 6 to 16. We first study the average goodput for various schemes under comparison in Fig. 6.2. We can note from figure that the proposed

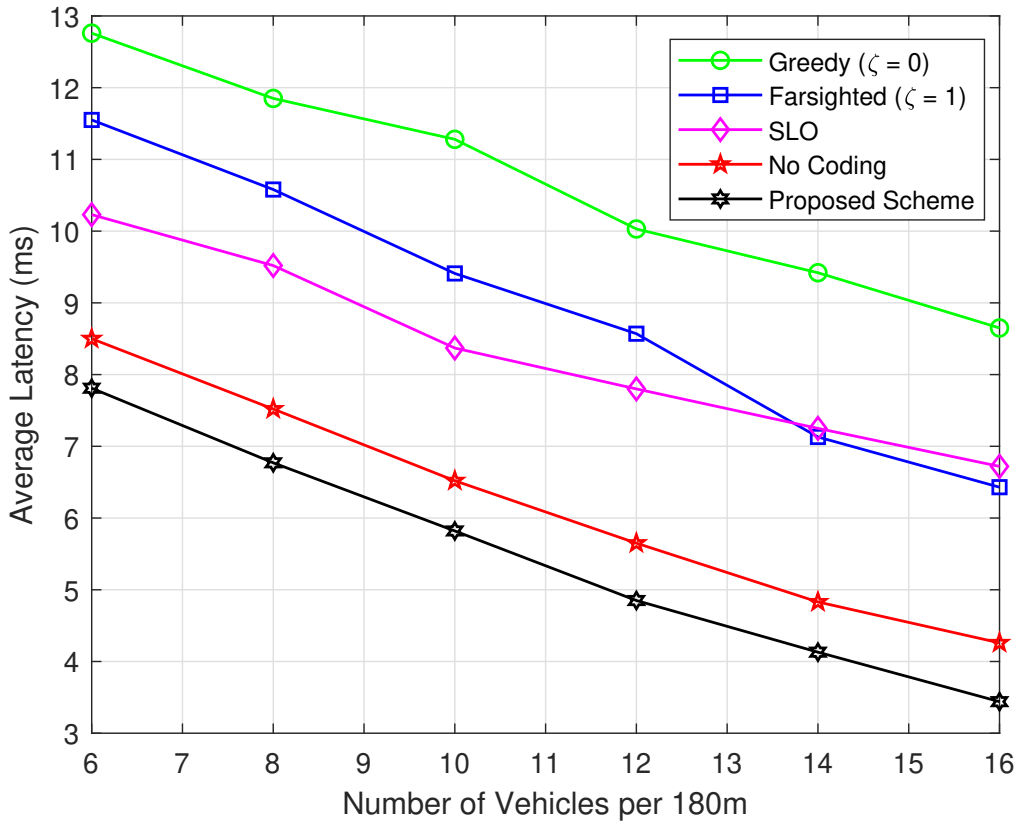

Figure 6.3: Comparison of average latency versus density of vehicle with different schemes with learning rate $\alpha = 10^{-4}$.

scheme outperforms the schemes under comparison significantly in terms of the average goodput in all the range of density of vehicles. Specifically, for low density of vehicles, i.e., 6, the performance gap between the proposed scheme and the No Coding, SLO, farsighted, and greedy schemes is approximately 0.3 Mbps,1 Mbps, 1.4 Mbps, and 1.7 Mbps, respectively. For the highest density of vehicles, i.e., 16, the gap increases to about 0.2 Mbps, 0.9 Mbps, 0.7 Mbps, and 1.2 Mbps, respectively. From the figure, it is evident that No Coding scheme offers the second-best performance, which shows the advantage of using code rate optimization in our proposed scheme. Whereas for the SLO scheme, the gap is considerably bigger than the No Coding scheme. This happens because, in SLO, we observe the

state of one link while optimizing the parameters of other links based on the policy of the observed link. So, all the constraints may not be satisfied, and therefore there is a big performance gap from the proposed scheme. For farsighted and greedy schemes, the gap grows further because of considering only the future rewards and immediate rewards, respectively.

We then illustrate the average latency for the various schemes under comparison at different density of vehicles in Fig. 6.3. From the figure, we observe that the proposed scheme outperforms other schemes. Specifically, for low density of vehicles, i.e., 6, the gap between the proposed scheme and No Coding, SLO, farsighted, and greedy schemes is about 0.7 ms, 2.4 ms, 3.7 ms, and 5 ms, respectively. Whereas for high density of vehicles, i.e., 16, the gap grows to 0.8 ms, 3.3 ms, 3 ms, and 5.2 ms, respectively. Similar to the average goodput in Fig. 6.2, the No Coding scheme shows the lowest latency gap and hence, we can see the effect of code rate optimization in our proposed scheme. Similarly, the SLO scheme offers more gaps in performance, and even it obtains higher latency when the density of vehicles passes 14. So, we can conclude that we achieve a higher rate and low latency if we use code rate optimization and multi-link policy maximization.

We then visualize how the proposed scheme meets the uRLLC requirements while maximizing the goodput. To this aim, we analyze the BER and latency performance for the various schemes under comparison. Please note that in this paper, we set the requirements ultra-reliability to meet BER of 10^{-7} and low-latency to satisfy 10 ms latency. We execute the simulation for 10000 decision episodes to investigate the BER and latency data. We then generate boxplots over all the available data and compare the results with all the schemes under comparison. First, we illustrate the boxplot of the BER to demonstrate whether all the schemes under comparison meet the reliability requirement in Fig. 6.4. We have also plotted a reference line to present our BER constraint of 10^{-7} (dashed black line). From the figure, we observe that our proposed algorithm always satisfies the reli-

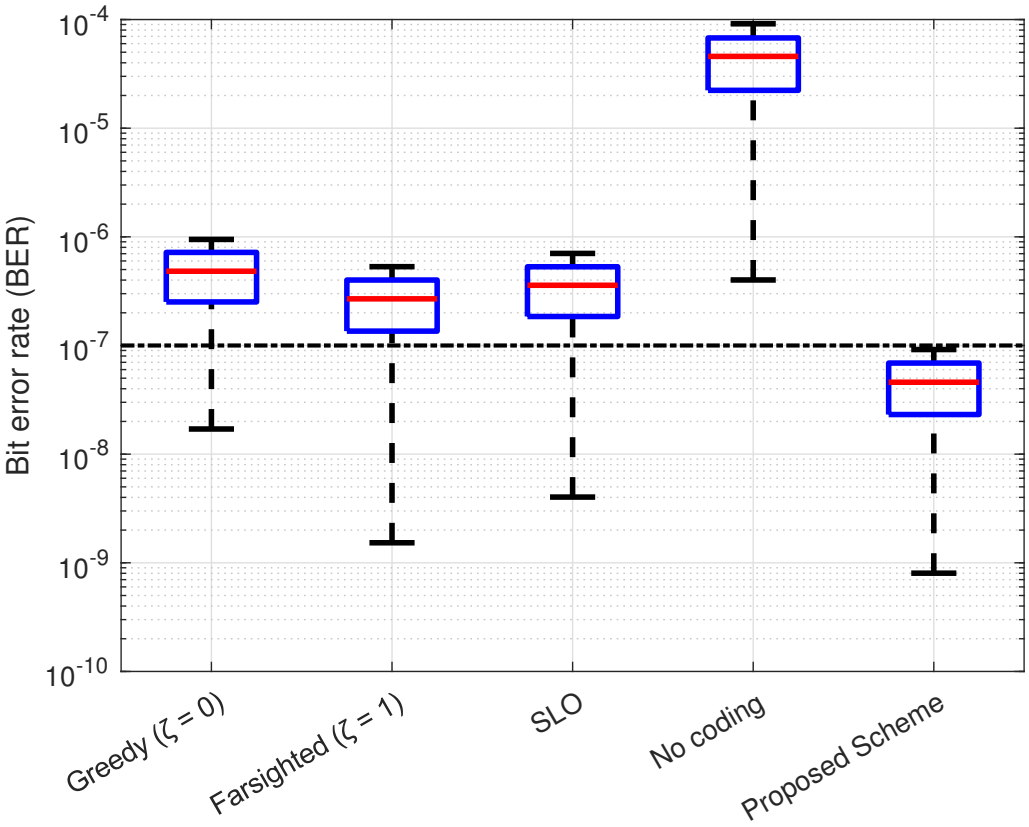

Figure 6.4: Box plot showing the maximum and minimum BER offered for all the schemes under comparison to justify how the reliability requirement is satisfied considering our maximum allowable BER 10^{-7}.

ability requirement, whereas other schemes cannot respect the constraint most of the time. In particular, no coding scheme can never satisfy BER. Therefore, it is evident that we cannot meet the reliability requirements without channel coding. The SLO scheme can respect the BER for 60% of the time because we optimize the performance of multiple links from the observation of single link parameters. In this scenario, there is the possibility of bad policies for other vehicles, which was considered good for the observed link.

Finally, Fig. 6.5 illustrates the boxplots of the observed latency to examine how the low latency requirement (10 ms) is met by the different

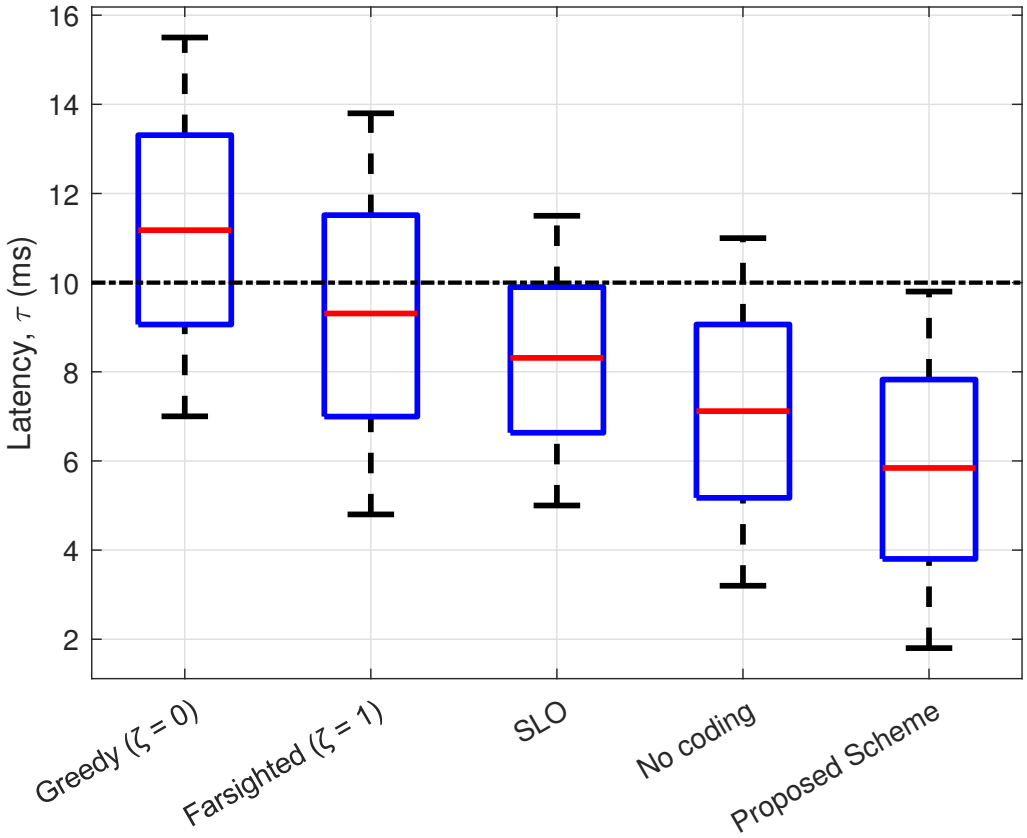

Figure 6.5: Box plot showing the maximum and minimum latency offered for different schemes under comparison to verify how the latency requirement is satisfied considering our latency requirement 10 ms.

comparison schemes when we optimize all the schemes for 10000 episodes. Similar to the BER performance, we also draw a reference line for the latency constraint (dashed black line) in Fig. 6.5. From the figure, we can note that our proposed scheme always respects the low latency requirements of 10 ms, while the other schemes fail to meet the constraint most of the time in our simulation. In particular, for the greedy, farsighted, SLO, and No coding schemes, the maximum observed latency is 15.5 ms, 13.8 ms, 11.5 ms, and 11 ms, respectively.

From all the above performance comparisons, we can conclude that our proposed vehicular OCC system can maximize the goodput while guar-

anteeing uRLLC, while the other schemes cannot meet the reliability and delay requirements. It is also obvious that we get better performance if we perform code rate optimization in the multi-agent vehicular OCC system.

6.6 Summary

In this chapter, we study a multi-link DRL based rate maximization scheme to ensure uRLLC in vehicular OCC. To this aim, we choose the optimal code rate, modulation scheme and the speed of vehicles for multiple vehicular links. We apply actor-critic based DRL frameworks with Wolpertinger architecture for multiple links. 5G NR LDPC code rates and adaptive modulation scheme are used as they offer variable rates and ultra-reliability. We then solve the continuous optimization problem using a multi-links actor-critic algorithm through the Wolpertinger policy. We evaluate the performance of the proposed scheme by comparing it with various variants of our scheme. The proposed method achieves considerably higher average goodput and lower latency than all the schemes under comparison. The results further demonstrate that our schemes always satisfy uRLLC requirements, whereas other schemes fail to meet most of the time. This happens because we consider multiple vehicular link optimization with code rate optimization. While, for No Coding, reliability can not be satisfied because of not having any coding scheme and for SLO, the agent optimizes the policies for other links without considering the state-action space of other links, where the solution becomes sub-optimal most of the time. Finally, we can conclude that the proposed multi-links actor-critic based DRL framework maximizes the communication rate while respecting uRLLC in vehicular OCC.

CHAPTER 7

Conclusions and Future Work

7.1 Conclusions and Summary

Ensuring uRLLC is essential in AV to provide seamless operation and reliable communication between vehicles. However, existing RF-based communication systems suffer from interference and, therefore, face challenges to meet uRLLC without sacrificing communication performance, i.e., rate and latency. Recently, OCC is considered as one of the promising technologies in vehicular communication as it offers interference-free and LoS communication. Motivated by the advantages of variable rate and ultra-reliability, we employ code rate optimization and adaptive modulation in this thesis. Since the vehicular networks are time-varying and dynamic, solving these problems using a traditional optimization method is challenging. Therefore, we utilize the DRL framework, which can learn from the environment while interacting with unknown environments. We apply the actor-critic framework with Wolpertinger architecture to solve large scale and continuous state-action space problems. Finally, we propose a multi-agent DRL framework based optimization scheme that aims at maximizing the communication rate while selecting the optimal code rate, modulation

scheme and speed of the vehicles.

We summarize the findings of this dissertation as follows:

- In Chapter 3, we analyze the performance of adaptive modulation in vehicular OCC systems. To this aim, we model the latency considering the transmission latency only. We evaluate the BER at various ranges of the AoI and distances. Afterwards, we use a predefined target BER to adaptively adjust the spectral efficiency using the available modulation schemes. In this way, the BER requirement is satisfied. We performed numerous simulations to determine how to adjust the employed modulation scheme and AoIs while satisfying the BER requirements. The results demonstrate that we can achieve 7ms latency by respecting the target BER requirements of 10^{-4} and 10^{-5} when we vary the AoI varied between 0^o to 90^o.

- In Chapter 4, a DRL-based spectral efficiency maximization scheme for a multiple vehicular OCC scenario is studied while meeting BER and latency requirements. To this end, we formulate a sum spectral efficiency maximization problem considering an adaptive modulation scheme subject to BER and latency constraints. We show that the optimization problem is the NP-hard problem and contains non-linear operations. Therefore, to overcome the difficulty of the above challenges, we model the optimization problem as an MDP framework. In this way, we can solve the problem distributively. We also observe that the problem is a constrained problem, where finding the optimal solution is complex and time-consuming. So, we relax the problem to an unconstrained one using the Lagrangian relaxation method. Since we have large state-action spaces, we solve it using the DRL algorithm. We then conduct a performance evaluation to see how the proposed scheme performs over all other schemes under comparison. The results demonstrate that we achieve better sum spectral efficiency and lower average latency compared to all the schemes under comparison. The CDF of latency and BER further show that our system can

satisfy ultra-low latency communication and BER constraints, while the rest of the schemes fail.

- Chapter 5 introduces an actor-critic DRL framework in vehicular OCC by optimizing the code rates, modulation schemes and changing the speed of the vehicles to meet uRLLC requirements. We model the system as a single link vehicular problem. We use 5G NR LDPC code rates as it offers variable rates and ultra-reliability. We employ the Wolpertinger architecture based actor-critic DRL framework to deal with the continuous state-action spaces. The performance of the proposed scheme is verified through simulations in terms of goodput, latency and BER. To show the superiority of the proposed scheme, we compare the performance with two variants of the proposed schemes and RF-base schemes. From the results, we see that our proposed scheme achieves higher average goodput and lower average latency while other schemes fail most of the time. The results further demonstrate that our scheme always satisfies the uRLLC requirements. This happens because we use an interference-free OCC system, which gives low latency and code rate optimization offers ultra-reliability, and DRL provides higher goodput.

- Finally, in Chapter 6, a multi-link DRL based rate maximization scheme is proposed to ensure uRLLC in vehicular OCC. To this end, we choose the optimal code rate, modulation scheme and the speed of vehicles for multiple vehicular links. We employ 5G NR LDPC code rates and an adaptive modulation scheme to support variable rates and ultra-reliability. The large scale and continuous problem is solved through a multi-links actor-critic algorithm based on Wolpertinger architecture. We perform numerous simulations to get an understanding of how our proposed algorithm works and compare it with several variants of our scheme. We observe from the results that the proposed method achieves higher average goodput and lower latency than all

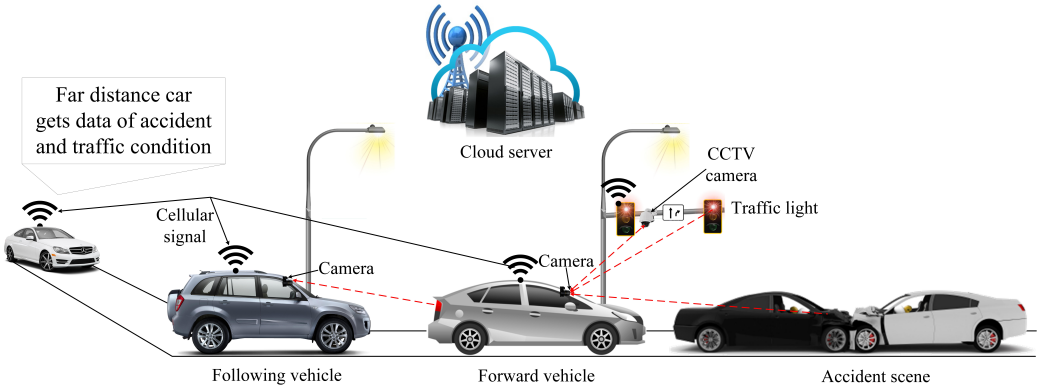

Figure 7.1: Hybrid RF-OCC communication mechanism.

the schemes under comparison. Further, the proposed scheme can meet the uRLLC constraints, whereas other schemes under comparison fail to respect most of the time. This happens because we consider multiple vehicular link optimization with code rate optimization in an interference-free OCC system. While, for No Coding, reliability can not be satisfied because of not having any coding scheme and for SLO, the agent optimizes the policies for other links without considering the state-action space of other links, where the solution becomes sub-optimal most of the time. Finally, we can summarize that the proposed multi-links actor-critic based DRL framework maximizes the communication rate while respecting uRLLC in vehicular OCC.

7.2 Future Directions

We conclude this dissertation by presenting some research directions for future investigation. Through our work in the various Chapters of this thesis, We identified the following possible extensions of our work can be done in the future:

- Throughout the thesis, we only consider V2V communication. But in the urban road environment, there exist various infrastructures, e.g.,

traffic lights and digital signages. Therefore, we want to extend our current work toward vehicle-to-everything communications by introducing V2I communication. In this scenario, there will be communication between vehicles and infrastructures simultaneously. In this way, the agents can take decisions and optimize the performance individually based on the observation of the surrounding environment, which will cover the entire road environment. This will be an interesting problem to solve as the vehicles are moving whereas the infrastructures are stationary. In solving this complicated problem, DRL will be a viable solution by maximizing the communication rate while respecting uRLLC.

- We then want to extend the previous vehicular problem to multi-vehicular problem in a more advance direction, where the vehicles will take a collective decision rather than the decision of individual vehicle. In this manner, the vehicle can communicate with each other more effectively. Here, each vehicle will be able to share their actions or any emergency condition in the road environment. As a result, the vehicle will have clear and collective information about the road environment while satisfying uRLLC.

- We would also like to use hybrid RF-OCC system to communicate with the servers or remote vehicles. Fig. 7.1 illustrates the overall overview of an hybrid system. The communication medium for vehicle to remote vehicle and servers will be performed using RF, whereas the LoS communication will be performed using OCC systems. This will ensure wider coverage area and communication with the servers or remote vehicles. This scheme will be effective if there is any emergency in the road that we need to inform other vehicles or central servers so that they can acts to these situations.

Bibliography

[1] D. Mohr *et al.*, "The road to 2020 and beyond: What's driving the global automotive industry? Mc Kinsey & Company," *Inc. Stuttgart*, 2013.

[2] A. Fernandez, M. Fallgren, and N. Brahmi, "5GCAR scenarios, use cases, requirements and KPIs," *Fifth Generation Communication Automotive Research and innovation, Technical Report D*, vol. 2, Aug. 2017.

[3] S.-h. Sun, J.-l. Hu, Y. Peng, X.-m. Pan, L. Zhao, and J.-y. Fang, "Support for vehicle-to-everything services based on LTE," *IEEE Wireless Communications*, vol. 23, no. 3, pp. 4–8, Jun. 2016.

[4] C. Liu and M. Bennis, "Ultra-reliable and low-latency vehicular transmission: An extreme value theory approach," *IEEE Communications Letters*, vol. 22, no. 6, pp. 1292–1295, Jun. 2018.

[5] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "LTE for vehicular networking: A survey," *IEEE Communications Magazine*, vol. 51, no. 5, pp. 148–157, May 2013.

[6] P. Papadimitratos, A. De La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 84–95, Nov. 2009.

[7] I. Takai, T. Harada, M. Andoh, K. Yasutomi, K. Kagawa, and S. Kawahito, "Optical vehicle-to-vehicle communication system using LED transmitter and camera receiver," *IEEE Photonics Journal*, vol. 6, no. 5, pp. 1–14, Oct. 2014.

[8] T. Yamazato, I. Takai, H. Okada, T. Fujii, T. Yendo, S. Arai, M. Andoh, T. Harada, K. Yasutomi, K. Kagawa, and S. Kawahito, "Image-sensor-based visible light communication for automotive applications," *IEEE Communications Magazine*, vol. 52, no. 7, pp. 88–97, Jul. 2014.

[9] M. I. Ashraf, C.-F. Liu, M. Bennis, and W. Saad, "Towards low-latency and ultra-reliable vehicle-to-vehicle communication," in *Proc. 2017 European Conference on Networks and Communications (EuCNC)*, Oulu, Finland, Jun. 2017, pp. 1–5.

[10] W. Sun, E. G. Ström, F. Brännström, Y. Sui, and K. C. Sou, "D2D-based V2V communications with latency and reliability constraints," in *Proc. 2014 IEEE GC Wkshps*, Austin, TX, USA, Dec. 2014, pp. 1414–1419.

[11] R. S. Sutton and A. G. Barto, *Introduction to reinforcement learning*. MA, USA: MIT press Cambridge, 1998, vol. 135.

[12] H. Li, T. Wei, A. Ren, Q. Zhu, and Y. Wang, "Deep reinforcement learning: Framework, applications, and embedded implementations," in *Proc. 2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Irvine, CA, USA, Nov. 2017, pp. 847–854.

[13] L. Liang, H. Ye, and G. Y. Li, "Spectrum sharing in vehicular networks based on multi-agent reinforcement learning," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2282–2292, Aug. 2019.

[14] M. A. Salahuddin, A. Al-Fuqaha, and M. Guizani, "Reinforcement learning for resource provisioning in the vehicular cloud," *IEEE Wireless Communications*, vol. 23, no. 4, pp. 128–135, Jun. 2016.

[15] D. Hui, S. Sandberg, Y. Blankenship, M. Andersson, and L. Grosjean, "Channel coding in 5G new radio: A tutorial overview and performance comparison with 4G LTE," *IEEE Vehicular Technology Magazine*, vol. 13, no. 4, pp. 60–69, Dec. 2018.

[16] M. Arabaci, I. B. Djordjevic, R. Saunders, and R. M. Marcoccia, "High-rate nonbinary regular quasi-cyclic LDPC codes for optical communications," *Journal of Lightwave Technology*, vol. 27, no. 23, pp. 5261–5267, Aug. 2009.

[17] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971v6*, Jul. 2019.

[18] G. Dulac-Arnold, R. Evans, H. van Hasselt, P. Sunehag, T. Lillicrap, J. Hunt, T. Mann, T. Weber, T. Degris, and B. Coppin, "Deep reinforcement learning in large discrete action spaces," *arXiv preprint arXiv:1512.07679*, 2015.

[19] J. Barbaresso, G. Cordahi, D. E. Garcia, C. Hill, A. Jendzejec, K. Wright, and B. A. Hamilton, "USDOT's intelligent transportation systems (ITS) ITS strategic plan, 2015-2019." United States. Department of Transportation. Intelligent Transportation, Washington, DC, USA, Tech. Rep., 2014.

[20] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *Journal of Network and Computer Applications*, vol. 37, pp. 380–392, Jan. 2014.

[21] M. Saini, A. Alelaiwi, and E. Saddik, "How close are we to realizing a pragmatic VANET solution? a meta-survey," *ACM Computing Surveys*, vol. 48, no. 2, pp. 1–40, Nov. 2015.

[22] S. F. Hasan, X. Ding, N. H. Siddique, and S. Chakraborty, "Measuring disruption in vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 1, pp. 148–159, Jan. 2011.

[23] J. Toutouh and E. Alba, "Light commodity devices for building vehicular ad hoc networks: An experimental study," *Ad Hoc Networks*, vol. 37, pp. 499–511, Feb. 2016.

[24] B. Aslam, P. Wang, and C. C. Zou, "Extension of internet access to VANET via satellite receive–only terminals," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 14, no. 3, pp. 172–190, Dec. 2013.

[25] S. Bitam, A. Mellouk, and S. Zeadally, "VANET-cloud: a generic cloud computing model for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 96–102, Feb. 2015.

[26] W. Travis, A. T. Simmons, and D. M. Bevly, "Corridor navigation with a LiDAR/INS Kalman filter solution," in *Proc. IEEE Intelligent Vehicles Symposium*, Las Vegas, NV, USA, Jun. 2005, pp. 343–348.

[27] M. A. Hossain, A. Islam, N. T. Le, Y. T. Lee, H. W. Lee, and Y. M. Jang, "Performance analysis of smart digital signage system based on software-defined IoT and invisible image sensor communication," *International Journal of Distributed Sensor Networks*, vol. 12, no. 7, pp. 1–14, Jul. 2016.

[28] T. Nguyen, A. Islam, and Y. M. Jang, "Region-of-interest signaling vehicular system using optical camera communications," *IEEE Photonics Journal*, vol. 9, no. 1, pp. 1–20, Feb. 2017.

[29] Satoshi Okada, Tomohiro Yendo, Takaya Yamazato, Toshiaki Fujii, Masayuki Tanimoto, and Yoshikatsu Kimura, "On-vehicle receiver for distant visible light road-to-vehicle communication," in *Proc. 2009 IEEE Intelligent Vehicles Symposium*, Xi'an, China, Jun. 2009, pp. 1033–1038.

[30] T. Yamazato, M. Kinoshita, S. Arai, E. Souke, T. Yendo, T. Fujii, K. Kamakura, and H. Okada, "Vehicle motion and pixel illumination modeling for image sensor based visible light communication," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 9, pp. 1793–1805, Sep. 2015.

[31] S. Haruyama, "Advances in visible light communication technologies," in *Proc. 38th European Conference and Exhibition Optical Communications*, Amsterdam, Netherlands, Sep. 2012, pp. 1–3.

[32] A. Ashok, S. Jain, M. Gruteser, N. Mandayam, W. Yuan, and K. Dana, "Capacity of screen–camera communications under perspective distortions," *Pervasive and Mobile Computing*, vol. 16, pp. 239–250, Jan. 2015.

[33] I. Takai, S. Ito, K. Yasutomi, K. Kagawa, M. Andoh, and S. Kawahito, "LED and CMOS image sensor based optical wireless communication system for automotive applications," *IEEE Photonics Journal*, vol. 5, no. 5, pp. 6 801 418–6 801 418, Oct. 2013.

[34] F. A. Teixeira, V. F. e Silva, J. L. Leoni, D. F. Macedo, and J. M. Nogueira, "Vehicular networks using the IEEE 802.11p standard: An experimental analysis," *Vehicular communications*, vol. 1, no. 2, pp. 91–96, Apr. 2014.

[35] H. B. Eldeeb, M. Elamassie, S. M. Sait, and M. Uysal, "Infrastructure-to-vehicle visible light communications: Channel modelling and performance analysis," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 3, pp. 2240–2250, Jan. 2022.

[36] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and low-latency wireless communication: Tail, risk, and scale," *Proceedings of the IEEE*, vol. 106, no. 10, pp. 1834–1853, Sep. 2018.

[37] 3GPP, "Study on scenarios and requirements for next generation access technologies," *Technical Specification Group Radio Access Network, Technical Report 38.913*, 2016.

[38] Y. Ren, F. Liu, Z. Liu, C. Wang, and Y. Ji, "Power control in D2D-based vehicular communication networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 12, pp. 5547–5562, Dec. 2015.

[39] K. Lee, J. Kim, Y. Park, H. Wang, and D. Hong, "Latency of cellular-based V2X: Perspectives on TTI-proportional latency and TTI-independent latency," *IEEE Access*, vol. 5, pp. 15 800–15 809, Jul. 2017.

[40] P. Popovski, J. J. Nielsen, C. Stefanovic, E. d. Carvalho, E. Strom, K. F. Trillingsgaard, A. Bana, D. M. Kim, R. Kotaba, J. Park, and R. B. Sorensen, "Wireless access for ultra-reliable low-latency communication: Principles and building blocks," *IEEE Network*, vol. 32, no. 2, pp. 16–23, Apr. 2018.

[41] D. Deng, S. Lien, C. Lin, S. Hung, and W. Chen, "Latency control in software-defined mobile-edge vehicular networking," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 87–93, Aug. 2017.

[42] H. Yang, K. Zheng, L. Zhao, K. Zhang, P. Chatzimisios, and Y. Teng, "High reliability and low latency for vehicular networks: Challenges and solutions," *arXiv preprint arXiv:1712.00537*, Dec. 2017.

[43] P. Popovski, "Ultra-reliable communication in 5G wireless systems," in *Proc. 1st International Conference 5G Ubiquitous Connectivity*, Levi, Finland, Nov. 2014, pp. 146–151.

[44] G. Durisi, T. Koch, and P. Popovski, "Toward massive, ultra-reliable, and low-latency wireless communication with short packets," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1711–1726, Aug. 2016.

[45] J. Arnau and M. Kountouris, "Delay performance of MISO wireless communications," in *Proc. 16th International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt 2018)*, Shanghai, China, May 2018, pp. 1–8.

[46] M. M. K. Tareq, O. Semiari, M. A. Salehi, and W. Saad, "Ultra reliable, low latency vehicle-to-infrastructure wireless communications with edge computing," in *Proc. 2018 IEEE Global Communications Conference (GLOBECOM)*, Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–7.

[47] W. Shi and S. Dustdar, "The promise of edge computing," *IEEE Computers*, vol. 49, no. 5, pp. 78–81, May 2016.

[48] M. S. Elbamby, M. Bennis, and W. Saad, "Proactive edge computing in latency-constrained fog networks," in *Proc. European Conference on Networks and Communications (EuCNC)*, Oulu, Finland, Jun. 2017, pp. 1–6.

[49] J. Park and P. Popovski, "Coverage and rate of downlink sequence transmissions with reliability guarantees," *IEEE Wireless Communications Letters*, vol. 6, no. 6, pp. 722–725, Dec. 2017.

[50] M. Sookhak, F. R. Yu, Y. He, H. Talebian, N. Sohrabi Safa, N. Zhao, M. K. Khan, and N. Kumar, "Fog vehicular computing: Augmentation of fog computing using vehicular cloud computing," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 55–64, Sep. 2017.

[51] C. Weng, D. Yu, S. Watanabe, and B.-H. F. Juang, "Recurrent deep neural networks for robust speech recognition," in *Proc. Interna-*

tional Conference on Acoustics, Speech, Signal Processing (ICASSP), Florence, Italy, May 2014, pp. 5532–5536.

[52] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[53] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, Feb. 2017.

[54] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[55] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.

[56] H. Mao, M. Alizadeh, I. Menache, and S. Kandula, "Resource management with deep reinforcement learning," in *Proc. 15th ACM Workshop on Hot Topics in Networks (HotNets)*, New York, NY, USA, Nov. 2016, pp. 50–56.

[57] Z. Xu, Y. Wang, J. Tang, J. Wang, and M. C. Gursoy, "A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs," in *Proc. 2017 IEEE International Conference on Communications (ICC)*, Paris, France, May 2017, pp. 1–6.

[58] Y. He, N. Zhao, and H. Yin, "Integrated networking, caching, and computing for connected vehicles: A deep reinforcement learning approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 44–55, Jan. 2018.

[59] Q. Zheng, K. Zheng, H. Zhang, and V. C. Leung, "Delay-optimal virtualized radio resource scheduling in software-defined vehicular networks via stochastic learning," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 7857–7867, Mar. 2016.

[60] M. Tan, "Multi-agent reinforcement learning: Independent vs. cooperative agents," in *Proc. 10th International Conference Machine Learning*, Jun. 1993, pp. 330–337.

[61] M. Rahmani, M. Bashar, M. J. Dehghani, P. Xiao, R. Tafazolli, and M. Debbahz, "Deep reinforcement learning-based power allocation in uplink cell-free massive MIMO," in *Proc. 2022 IEEE Wireless Communications and Networking Conference (WCNC)*, Austin, TX, USA, Apr. 2022, pp. 1–6.

[62] R. Huang and V. W. Wong, "Joint user scheduling, phase shift control, and beamforming optimization in intelligent reflecting surface-aided systems," *IEEE Transactions on Wireless Communications*, pp. 1–1, Mar. 2022.

[63] L. Lei, T. Liu, K. Zheng, and L. Hanzo, "Deep reinforcement learning aided platoon control relying on V2X information," *IEEE Transactions on Vehicular Technology*, pp. 1–1, Mar. 2022.

[64] H. Gamage, N. Rajatheva, and M. Latva-Aho, "Channel coding for enhanced mobile broadband communication in 5G systems," in *Proc. 2017 European Conference on Networks and Communications (EuCNC)*, Oulu, Finland, Jun. 2017, pp. 1–6.

[65] O. Iscan, D. Lentner, and W. Xu, "A comparison of channel coding schemes for 5G short message transmission," in *Proc. 2016 IEEE Globecom Workshops (GC Wkshps)*, Washington, DC, USA, Dec. 2016, pp. 1–6.

[66] C. Padmaja and B. Malleswari, "Performance analysis of 4G systems with channel coding algorithms," *HELIX*, vol. 8, no. 1, pp. 2742–2746, Jan. 2018.

[67] K. D. Rao, *Channel coding techniques for wireless communications*. Springer, 2015.

[68] B. Feng, S. Gu, J. Jiao, S. Wu, and Q. Zhang, "Novel polar-coded space-time transmit diversity scheme over rician fading MIMO channels," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, pp. 1–10, Dec. 2018.

[69] T. Murata and H. Ochiai, "Performance analysis of CRC codes for systematic and nonsystematic polar codes with list decoding," *Wireless Communications and Mobile Computing*, vol. 2018, no. 7286909, pp. 1–8, May 2018.

[70] S. Pfletschinger, A. Mourad, E. Lopez, D. Declercq, and G. Bacci, "Performance evaluation of non-binary LDPC codes on wireless channels," in *Proc. ICT Mobile Summit*, Spain, Jun. 2009, pp. 1–8.

[71] C. Yoon, J.-E. Oh, M. Cheong, and S.-k. Lee, "A hardware efficient ldpc encoding scheme for exploiting decoder structure and resources," in *Proc. 2007 IEEE 65th Vehicular Technology Conference (VTC2007-Spring)*, Apr. 2007, pp. 2445–2449.

[72] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over GF(q)," *IEEE Transactions on Communications*, vol. 55, no. 4, pp. 633–643, Apr. 2007.

[73] T. Richardson and S. Kudekar, "Design of low-density parity check codes for 5G new radio," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 28–34, Mar. 2018.

[74] Y. Goto, I. Takai, T. Yamazato, H. Okada, T. Fujii, S. Kawahito, S. Arai, T. Yendo, and K. Kamakura, "A new automotive VLC system

using optical communication image sensor," *IEEE Photonics Journal*, vol. 8, no. 3, pp. 1–17, Jun. 2016.

[75] A. Islam, M. T. Hossan, and Y. M. Jang, "Convolutional neural network scheme–based optical camera communication system for intelligent internet of vehicles," *International Journal of Distributed Sensor Networks*, vol. 14, no. 4, pp. 1–15, Apr. 2018.

[76] Peng Deng and M. Kavehrad, "Real-time software-defined single-carrier QAM MIMO visible light communication system," in *Proc. Integrated Communications, Navigation and Surveillance (ICNS)*, Herndon, VA, USA, Apr. 2016, pp. 5A3–1–5A3–11.

[77] P. Luo, M. Zhang, Z. Ghassemlooy, H. Le Minh, H.-M. Tsai, X. Tang, and D. Han, "Experimental demonstration of a 1024-QAM optical camera communication system," *IEEE Photonics Technology Letters*, vol. 28, no. 2, pp. 139–142, Oct. 2015.

[78] S. A. I. Alfarozi, K. Pasupa, H. Hashizume, and K. Woraratpanyaand M. Sugimoto, "Square wave quadrature amplitude modulation for visible light communication using image sensor," *IEEE Access*, vol. 7, pp. 94 806–94 821, Jul. 2019.

[79] J. M. Kahn and J. R. Barry, "Wireless infrared communication," *Proceedings of the IEEE*, vol. 85, no. 2, pp. 265–298, Feb. 1997.

[80] Z. Ghassemlooy, D. Wu, M. A. Khalighi, and X. Tang, "Indoor non-directed optical wireless communications optimization of the lambertian order," *Journal of Electrical and Computer Engineering Innovations*, vol. 1, no. 1, pp. 1–9, May 2013.

[81] T. Komine and M. Nakagawa, "Fundamental analysis for visible-light communication system using LED lights," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 100–107, Feb. 2004.

[82] B. Horn, B. Klaus, and P. Horn, *Robot vision*. MIT press, 1986.

[83] R. Steele and W. T. Webb, "Variable rate QAM for data transmission over Rayleigh fading channel," in *Proceedings of Wireless*, Calgary, Canada, Jul. 1991, pp. 1 –14.

[84] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[85] P. Deng and M. Kavehrad, "Adaptive real-time software defined MIMO visible light communications using spatial multiplexing and spatial diversity," in *Proc. IEEE International Conference Wireless for Space and Extreme Environments (WiSEE)*, Aachen, Germany, Sep. 2016, pp. 111–116.

[86] C. Claus and C. Boutilier, "The dynamics of reinforcement learning in cooperative multiagent systems," in *Proc. 15th National/10th Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*, Madison, Wisconsin, USA, Jul. 1998, pp. 746–752.

[87] R. V. Terres, "Multi-user MISO for visible light communication," Ph.D. dissertation, University of Virginia, Sep. 2015.

[88] A. Islam, L. Musavian, and N. Thomos, "Performance analysis of vehicular optical camera communications: Roadmap to uRLLC," in *Proc. 2019 IEEE Global Communications Conference (GLOBECOM)*, Hawaii, USA, Dec. 2019, pp. 1–6.

[89] P. Deng, "Real-time software-defined adaptive MIMO visible light communications," *Visible Light Communications*, pp. 637–640, Jul. 2017.

[90] D. A. Plaisted, "Some polynomial and integer divisibility problems are NP-HARD," in *Proc. 17th Annual Symposium on Foundations of Computer Science (SFCS)*, Houston, TX, USA, Oct. 1976, pp. 264–267.

[91] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: Algorithms and complexity*. Courier Corporation, 1998.

[92] T. Zinchenko, "Reliability assessment of vehicle-to-vehicle communication," doctoralthesis, Technische Hochschule Wildau, 2014.

[93] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[94] N. Mastronarde and M. van der Schaar, "Joint physical-layer and system-level power management for delay-sensitive wireless communications," *IEEE Transactions on Mobile Computing*, vol. 12, no. 4, pp. 694–709, Feb. 2012.

[95] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[96] N. Salodkar, A. Bhorkar, A. Karandikar, and V. S. Borkar, "An on-line learning algorithm for energy efficient delay constrained scheduling over a fading channel," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 732–742, Apr. 2008.

[97] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO-simulation of urban mobility," *International Journal on Advances in Systems and Measurements*, vol. 5, no. 3&4, Dec. 2012.

[98] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2020.

[99] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, Savannah, GA, Nov. 2016, pp. 265–283.

[100] R. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[101] S. Seo, T. Mudge, Y. Zhu, and C. Chakrabarti, "Design and analysis of LDPC decoders for software defined radio," in *Proc. 2007 IEEE Workshop on Signal Processing Systems*, Shanghai, China, Oct. 2007, pp. 210–215.

[102] K. Zhang, X. Huang, and Z. Wang, "High-throughput layered decoder implementation for quasi-cyclic LDPC codes," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 6, pp. 985–994, Jul. 2009.

[103] T. Zinchenko, "Reliability assessment of vehicle-to-vehicle communication," Ph.D. dissertation, Univ.-Bibl., 2015, [Online]. Available: https://publikationsserver.tu-braunschweig.de/servlets/MCRFileNodeServlet/dbbs_derivate_00036558Diss_Zinc

[104] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. International Conference on Machine Learning*, Beijing, China, Jun. 2014, pp. 387–395.

[105] G. E. Uhlenbeck and L. S. Ornstein, "On the theory of the Brownian motion," *Physical Review*, vol. 36, no. 5, p. 823, 1930.

[106] J. H. Bae, A. Abotabl, H.-P. Lin, K.-B. Song, and J. Lee, "An overview of channel coding for 5G NR cellular communications," *APSIPA Transactions on Signal and Information Processing*, vol. 8, p. e17, Jun. 2019.

[107] A. Islam, L. Musavian, and N. Thomos, "Multi-agent deep reinforcement learning in vehicular OCC," in *Proc. 95th Vehicular Technology Conference (VTC2022-Spring)*, Helsinki, Finland, Jun. 2022, pp. 1–6.

[108] S. Lin and D. J. Costello, *Error control coding*. Prentice hall Scarborough, 2001, vol. 2, no. 4.

[109] Z. Tu and S. Zhang, “Overview of LDPC codes,” in *Proc. 7th IEEE International Conference on Computer and Information Technology (CIT 2007)*, Aizu-Wakamatsu, Japan, Oct. 2007, pp. 469–474.

[110] S.-Y. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, “On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit,” *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, Feb. 2001.

[111] R. Pyndiah, A. Picart, and A. Glavieux, “Performance of block turbo coded 16-QAM and 64-QAM modulations,” in *Proc. 1995 IEEE Global Communications Conference (GLOBECOM)*, Singapore, Nov. 1995, pp. 1039–1043.

APPENDIX A

Stereo Detection

After the camera calibration, the distance from the captured images is computed. Distance information can help to make decisions for the vehicle to pay attention and information extraction. However, all incidents that happen in the real world can be described in three-dimensional (3D) format but the camera generates only two-dimensional (2D) images. Thus, an adaptive convention algorithm is required to measure the distance from a 2D image. There are various approaches to measure the depth of 2D image. The most recent trend is to use a stereo-vision camera, rather than a single camera, in a manner analogous to vision system [71]. A stereo-vision camera consists of two cameras mounted at a fixed position on a single apparatus for (i) synchronizing the focal point and (ii) adjusting the image-focal plane of both cameras. Both cameras capture the same scene but with a slightly shifted FoV, allowing the formation of a stereo-image pair. Distance measurement relies on matching the pixels in the left and right images. The following algorithm is being used to complete the task.

- Image acquisition (i.e., input image from both left and right cameras).
- Image rectification to align epipolar line of two camera images horizontally by using a linear transformation.

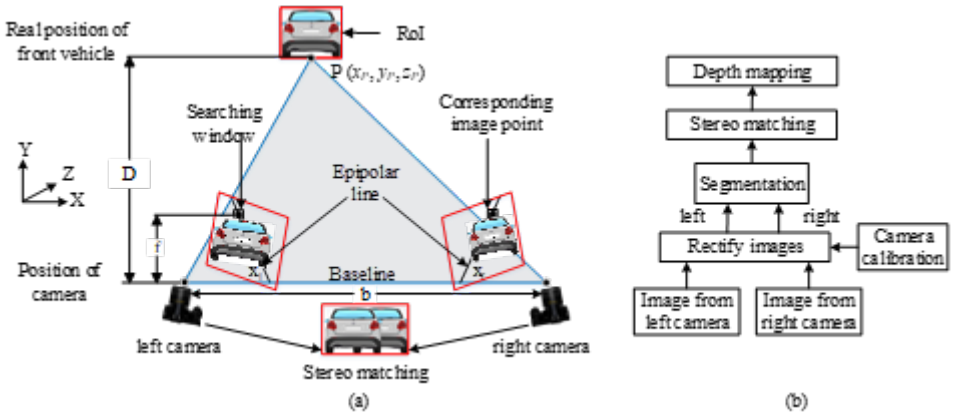

Figure A.1: Distance measurement: (a) using stereo images of stereo camera, (b) system platform algorithm

- Segmentation for detection, recognition, and measurement of objects in images.
- Algorithms for stereo matching for depth calculation. There are different algorithms are being used for stereo matching, such as the sum of absolute differences (SAD), correlation, normalized cross-correlation, and the sum of squared differences (SSD). The SAD algorithm computes the intensity differences for each center pixel (i, j) in a window $W(x, y)$:

$$\text{SAD}(x, y, d) = \sum_{(i,j)\in W(x,y)}^{N} |I_L(i, j) - I_R(i - d, j)| \qquad \text{(A.1)}$$

  where I_L and I_R are pixel-intensity functions of the left and right images, respectively. $W(x, y)$ is a square window that surrounds the position (x, y) of the pixel. The minimum difference value over the frame indicates the best matching pixel, and the position of the minimum defines the disparity of the actual pixel.

Depth map estimation: For stereo cameras with parallel optical axes (see Fig. A.1), focal length f, baseline b, and corresponding image points

(x_l, y_l) and (x_r, y_r), the coordinates of a 3D point $P(x_p, y_p, z_p)$ from 2D image can be determined by the following equations:

$$\frac{z_p}{f} = \frac{x_p}{x_l} = \frac{x_p - b}{x_r} = \frac{y_p}{y_l} = \frac{y_p}{y_r} \tag{A.2}$$

$$x_p = \frac{x_l z}{f} = b + \frac{x_r z}{f} \tag{A.3}$$

$$y_p = \frac{y_l z}{f} = b + \frac{y_r z}{f} \tag{A.4}$$

The depth is calculated from the disparity map using the rectified image from stereo camera. The disparity map (A.5) is determined by the difference between the x-coordinate of the projected 3D coordinate, x_p, onto the left camera image plane and is the x-coordinate of the projection onto the right image plane. Therefore, the disparity can be calculated from the following equation,

$$d = x_l - x_r = f\left(\frac{x_p + \frac{b}{2}}{z_p} - \frac{x_p - \frac{b}{2}}{z_p}\right) = \frac{fb}{z_p} \tag{A.5}$$

$$\text{or, } z_p = \frac{fb}{d} \tag{A.6}$$

APPENDIX B

5G NR LDPC Code

B.1 LDPC Encoder

LDPC codes are the type of linear codes (n, p) which takes p bits information symbol and maps to n bits codeword. In LDPC code, initially, H matrix is constructed which is sparse matrix i.e., having few 1's when compared to 0's [108], [109]. The BER execution of LDPC codes mainly depends on the H matrix. Therefore, different calculation methods are used to compute H that ultimately reduce the error rate and complexity.

The parity check matrix has been constructed through specified column weight w_c and row weight w_r [109]. If specified column and row weights are uniform throughout the H matrix, then the code is termed as "Regular LDPC". In contrast, if column and row weights are not uniform then the code is "Irregular LDPC" [110]. The parameters w_c and w_r are defined as the number of non-zero columns and rows within the H matrix. Also, the parity check matrix is termed as low density if the following two conditions are satisfied, i.e., $w_c << n$ and $w_r << p$. While the coding rate for LDPC is formulated as [110]:

$$\varkappa = 1 - \frac{w_c}{w_r}. \tag{B.1}$$

LDPC encoding is done by matrix multiplication which is given as:

$$E = pG. \tag{B.2}$$

For multiplication, first convert $(n-p) \times n$ parity check matrix into systematic form as $H = [I_{n-p} \mid P_m]$, where I_{n-p} is the identity matrix and P_m is the parity check matrix. Now from parity check matrix, generator matrix is constructed as, $G = \left[P_{p\times(n-p)} \mid I_p\right]$. The resultant generator matrix is used to encode (B.2) into the incoming message.

B.2 LDPC Decoder

The receiver uses standard 2^M-ary QAM to demodulate the incoming message by calculating the log-likelihood ratio (LLR) and the channel decoding is carried out for the LLR values to obtain original information.

The LLR of bit l_i of the received symbol is given by [111]

$$LLR(l_i) = \log\left(\frac{\sum_{\alpha_1 \in S_i^{(1)}} P_r\{x = \alpha_1 \mid y, H\}}{\sum_{\alpha_2 \in S_i^{(0)}} P_r\{x = \alpha_2 \mid y, H\}}\right), \tag{B.3}$$

where where $S_i^{(0)}$ and $S_i^{(1)}$ denote the set of symbols for which, $X = 0$ and $X = 1$, respectively. To reduce implementation complexity, the LLR computation is often simplified as the following min-operation [111]:

$$LLR(l_i) = -\frac{1}{2\epsilon^2}\log\left(\min_{S^{(0)}} \mid y - Hx \mid^2 - \min_{S^{(1)}} \mid y - Hx \mid^2\right). \tag{B.4}$$

Using the $LLR(l_i)$ obtained above, we can derive the analytical expression for the probability of error for the bits, l_i. The probability of error for bit l_i, $P_{b,i}$, is given by [111]

$$P_{b,i} = \frac{1}{2}\left(P_{b,i} \mid_{l_i=1} + P_{b,i} \mid_{l_i=0}\right). \tag{B.5}$$