

University of Windsor

Scholarship at UWindor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2023

Reinforcement Learning-Based Data Rate Congestion Control for Vehicular Ad-Hoc Networks

Gnana Shilpa Nuthalapati
University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Nuthalapati, Gnana Shilpa, "Reinforcement Learning-Based Data Rate Congestion Control for Vehicular Ad-Hoc Networks" (2023). *Electronic Theses and Dissertations*. 9101.
<https://scholar.uwindsor.ca/etd/9101>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Reinforcement Learning-Based Data Rate Congestion Control for Vehicular Ad-Hoc Networks.

By

Gnana Shilpa Nuthalapati

A Thesis

Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2023

©2023 Gnana Shilpa Nuthalapati

Reinforcement Learning-Based Data Rate Congestion Control for Vehicular Ad-Hoc
Networks.

by

Gnana Shilpa Nuthalapati

APPROVED BY:

K. Li
Odette School Of Business

S. Jiang
School of Computer Science

A. Jaekel, Co-Advisor
School of Computer Science

N. Zhang, Co-Advisor
Department of Electrical & Computer Engineering

June 19, 2023

DECLARATION OF CO-AUTHORSHIP

I. Co-Authorship

I hereby declare that this thesis incorporates material that is the result of research conducted under the supervision of Dr. Arunita Jaekel. In all cases, the key ideas, primary contribution, experimental designs, data analysis, and interpretation were performed by the author, and the contribution of the co-author was primarily through providing feedback and the proofreading of the published manuscripts.

I am aware of the University of Windsor Senate Policy on Authorship, and I certify that I have properly acknowledged the contribution of other researchers to my thesis and have obtained written permission from each of the co-author(s) to include the above material(s) in my thesis. I certify that, with the above qualification, this thesis, and the research to which it refers, is the product of my work.

II. General

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyright material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained written permission from the copyright owner(s) to include such material(s) in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Vehicular Ad-Hoc Network(VANET) is an emerging wireless technology vital to the Intelligent Transportation System(ITS) for vehicle-to-vehicle and vehicle-to-infrastructure communication. An ITS is an advanced solution that aims to deliver innovative services pertaining to various transportation modes and traffic management. Its objective is to enhance user awareness, promote safety, and enable more efficient and coordinated utilization of transport networks. ITS aims to mitigate traffic problems and improve the safety of transport by preventing unexpected events. When the vehicle density, i.e., the number of vehicles communicating in a wireless channel, increases, the channel faces congestion resulting in unreliable safety applications. Various decentralized congestion control algorithms have been proposed to effectively decrease channel congestion by controlling transmission parameters such as message rate, transmission power, and data rate. This thesis proposes a data rate-based congestion control technique using the Q-Learning algorithm to maintain the channel load below the target threshold. The congestion problem is formulated as an MDP and solved using a Q-learning algorithm. Q-learning is a model-free Reinforcement Learning algorithm that learns the values of an action within a specific state without relying on an explicit model of the environment. Reinforcement Learning has a set of states and actions and will find the best action for each state. The target is to train the vehicle to select the most appropriate data rate to send out a Basic Safety Message(BSM) by maintaining the channel load below the target threshold value. We use the Q-Learning algorithm with data obtained from a simulated dynamic traffic environment. We define a reward function combining CBR and data rate to maintain the channel load below the target threshold with the least data rate possible. Simulation results show that the proposed algorithm performs better over other techniques such as Transmit Data rate Control(TDRC), Data Rate based Decentralized Congestion Control(DR-DCC) and Data Rate Control Algorithm (DRCA) in low and medium loads and better over TDRC and DR-DCC in heavy load in terms of the Channel Busy Ratio (CBR), packet loss and Beacon Error Rate (BER).

DEDICATION

I dedicate this thesis to my parents and brother for their love, support and encouragement and also to my supervisor for her guidance throughout my research. I also dedicate this thesis to anyone who has ever believed in me or offered me advice or help.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor Dr. Arunita Jaekel, co-supervisor Dr. Ning Zhang and Ph.D. student Xiaofeng Liu, for guidance, support and encouragement throughout my research. I would like to thank my internal reader Dr. Shaoquan Jiang and my external reader Dr. Kevin Li for their support and feedback in improving my thesis.

TABLE OF CONTENTS

DECLARATION OF CO-AUTHORSHIP	III
ABSTRACT	IV
DEDICATION	V
ACKNOWLEDGEMENTS	VI
LIST OF TABLES	IX
LIST OF FIGURES	X
LIST OF ABBREVIATIONS	XI
1 Introduction	1
1.1 Vehicular Ad-Hoc Networks	1
1.2 Motivation	3
1.3 Problem Statement	4
1.4 Solution Outline	5
1.4.1 Contributions	6
1.5 Thesis Organization	6
2 Related Work	7
2.1 Overview of VANET	7
2.1.1 Types of Communication	7
2.1.2 Congestion Control and its Challenges	8
2.2 Overview of Reinforcement Learning	9
2.2.1 Basic Reinforcement Learning Concepts and Terminologies	9
2.2.2 Reinforcement Learning Algorithms	10
2.2.2.1 Model-Free Reinforcement Learning	10
2.2.2.2 Model-Based Reinforcement Learning	11
2.2.2.3 Hybrid Reinforcement Learning	11
2.2.3 RL for Congestion Control in VANET	12
2.3 Literature Review	13
2.3.1 Transmission Power Adaptation Approaches	13
2.3.2 Transmission Rate Control Approaches	15
2.3.3 Data Rate Control Approaches	17
2.3.4 Hybrid Approaches	19
2.3.5 Discussion	21

3	Reinforcement Learning-based Congestion Control	22
3.1	An RL-based Framework for Congestion Control in VANET	22
3.2	Problem Formulation as Markov Decision Process	25
3.3	Q-Learning Algorithm	28
3.4	Comparison with Existing Approaches	35
4	Simulations and Results	37
4.1	Simulation Setup	37
4.2	Comparison with Constant Data Rate Transmissions	40
4.2.1	Comparison of CBR	40
4.2.2	Comparison of Received Packets	43
4.3	Comparison with Existing Congestion Control Techniques	44
4.3.1	Comparison of CBR	45
4.3.2	Comparison of Received Rackets	47
4.3.3	Comparison of Total Lost Packets	49
4.3.4	Comparison of Beacon Error Rate (BER)	51
5	Conclusion and Future Work	54
5.1	Conclusion	54
5.2	Future Work	55
	REFERENCES	56
	VITA AUCTORIS	62

LIST OF TABLES

3.1	Partial Q-Table	34
4.1	Configuration Parameters	39

LIST OF FIGURES

1.1	An example of vehicular ad-hoc network [6]	2
3.1	An RL cycle in VANET	24
4.1	Simulation tools	38
4.2	Comparing CBR with constant data rates (250 Vehicles, 5 Hz)	41
4.3	Comparing CBR with constant data rates (250 Vehicles, 10 Hz) . . .	41
4.4	Comparing CBR with constant data rates (500 Vehicles, 5 Hz)	42
4.5	Comparing CBR with constant data rates (500 Vehicles, 10 Hz) . . .	42
4.6	Comparing received packets with constant data rates (250 Vehicles) .	43
4.7	Comparing received packets with constant data rates (500 Vehicles) .	44
4.8	Comparing CBR with existing methods (250 Vehicles, 5 Hz)	45
4.9	Comparing CBR with existing methods (250 Vehicles, 10 Hz)	46
4.10	Comparing CBR with existing methods (500 Vehicles, 5 Hz)	47
4.11	Comparing CBR with existing methods (500 Vehicles, 10 Hz)	47
4.12	Comparing received packets with existing methods (250 Vehicles) . .	48
4.13	Comparing received packets with existing methods (500 Vehicles) . .	49
4.14	Comparing total lost packets with existing methods (250 Vehicles) . .	50
4.15	Comparing total lost packets with existing methods (500 Vehicles) . .	50
4.16	Comparing BER with existing methods (250 Vehicles)	52
4.17	Comparing BER with existing methods (500 Vehicles)	52

LIST OF ABBREVIATIONS

WHO	World Health Organization
ITS	Intelligent Transportation System
VANET	Vehicular Ad-Hoc Network
MANET	Mobile Ad-Hoc Network
RSU	Road-Side Unit
OBU	On-Board Unit
TA	Trusted Authority
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
V2P	Vehicle-to-Pedestrian
V2X	Vehicle-to-Everything
C-V2X	Cellular Vehicle-to-Everything
BSM	Basic Safety Message
CAMs	Cooperative Awareness Messages
DSRC	Dedicated Short-Range Communication
WAVE	Wireless Access in Vehicular Environment
FCC	Federal Communications Commission

MDP	Markov Decision Process
CBR	Channel Busy Ratio
VD	Vehicle Density
DR-DCC	Data Rate-Decentralized Congestion Control
TDRC	Transmit Data Rate Control
DRCA	Adaptive Data Rate-based Congestion Control
Q-DRCC	Q-learning-based Data rate congestion control
SARSA	State-Action-Reward-State-Action
MBVI	Model-based Value Iteration
MCTS	Monte Carlo Tree Search
MPO	Model-based Policy Optimization
PPO	Proximal Policy Optimization
MBL	Maximum Beacon Load
CCDF	Complementary Cumulative Distribution Function
PULSAR	Periodically Updated Load Sensitive Adaptive Rate Control
AIMD	Additive Increase Multiplicative Decrease
LIMERIC	Linear Message Rate Integration Control
IPD	Inter-Packet Delay
EMBARC	Error Model-Based Adaptive Rate Control
STE	Suspected Tracking Error
BRAEVE	Bsm RAtE control using the Estimated number of VEHICLES in the communication range
CBA	Collision-based beacon rate adaptation scheme

MDPRP	MultiDimensional Power and Rate Policy
CPRC	Combined Power and Rate Control
CACC	Channel-Aware Congestion Control
PCR	Packet Collision Rate
CPMRA	Combined Power and Message Rate Adaptation
CPDRA	Combined Power and Data Rate Adaptation
RL	Reinforcement Learning
BER	Beacon Error Rate
SUMO	Simulation of Urban MObility
OMNeT++	Objective Modular Network Testbed
Veins	Vehicles in Simulation

CHAPTER 1

Introduction

1.1 Vehicular Ad-Hoc Networks

In recent years, we have been facing numerous challenges in terms of transportation, e.g., road safety, security, traffic congestion, environmental impacts, etc. In 2022, World Health Organization (WHO) estimated that approximately 1.3 million die yearly from traffic accidents, affecting between 20 and 50 million more people with non-fatal injuries. Some risk factors contributing to traffic accidents are unsafe road infrastructure and unsafe vehicles on the road [1]. Intelligent Transportation Systems (ITS) have grown in popularity and are crucial in overcoming these challenges. They provide tools that integrate advanced technologies and apply them to transportation to develop solutions that improve people's quality of life [2]. ITS aims to collect real-time information about traffic conditions and flows on roads and send it to control systems like route control, public transport control systems, commercial transport control systems, etc. [3]. The ITS includes various applications, including collision avoidance systems, variable speed limits, emergency vehicle notification systems [4], etc.

Vehicular Ad-Hoc Network (VANET), an element of an Intelligent Transport System, has been introduced for efficient and periodic information sharing between vehicles. VANET [5] has emerged as a promising technology for improving road safety, traffic efficiency, and passenger comfort. VANET is a type of ad-hoc wireless network used for communication between vehicles and the surrounding infrastructure. It is an application of MANET that uses stationary or moving vehicles connected by a

wireless network as nodes to create a mobile network. These nodes send and receive data to and from other vehicles, respectively, in the network.

VANET consists of multiple components, such as Roadside Unit (RSU), Onboard Unit (OBU), and Trusted Authority (TA). The OBU is a GPS-based device embedded in a vehicle that shares information with RSUs and other OBUs in the neighbouring vehicles. The RSU is a fixed device alongside the road, providing connectivity to the vehicles passing it. TA manages the VANET system and is also responsible for the security of VANET [6].

VANET supports several types of communication, including Vehicle-to-Vehicle (V2V) communication, Vehicle-to-Infrastructure (V2I) communication, Vehicle-to-Pedestrian (V2P) communication, and Vehicle-to-Everything (V2X) communication as shown in Figure 1.1.

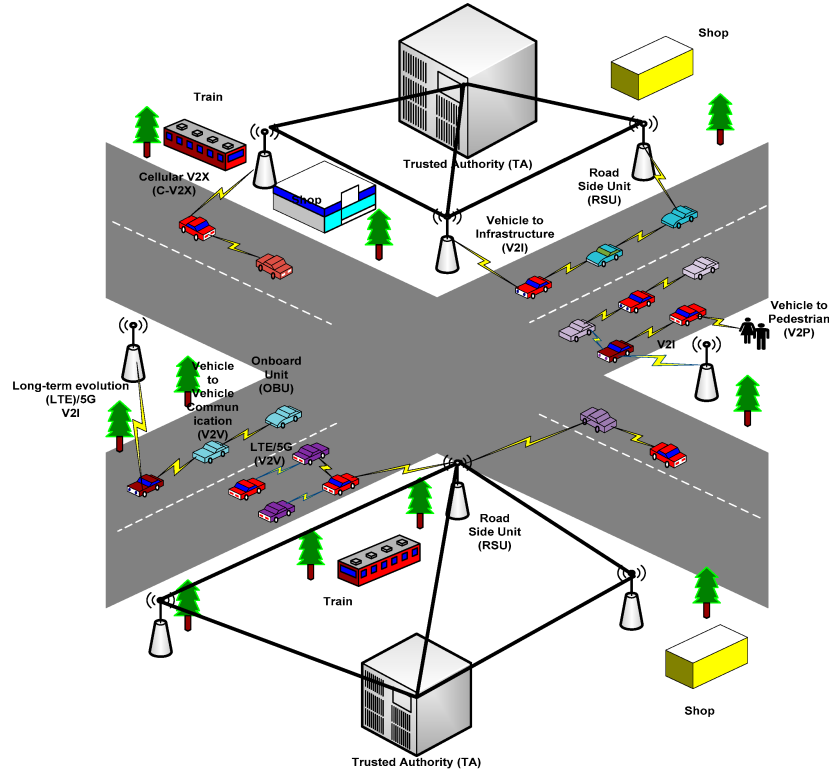


Figure 1.1: An example of vehicular ad-hoc network [6]

VANET applications are categorized as safety and non-safety applications. Safety applications include lane-changing assistance, collision warnings, speed warnings,

emergency warning, etc. Non-safety applications include Internet Access, weather information, location service, dynamic routing, traffic signal control, parking assistance, etc. Many safety applications rely on periodic messages broadcasted by each vehicle containing its status information. These periodic messages are called Basic Safety Messages (BSMs) [7], also known as cooperative awareness messages (CAMs) [8].

To facilitate communication among vehicles and other road infrastructure in VANET, standards like Dedicated Short-Range Communication (DSRC), Wireless Access for Vehicular Environment (WAVE) and IEEE 802.11p have been developed. DSRC communication is used for Vehicle-to-Vehicle and Vehicle-to-Infrastructure communication. The U.S. Federal Communications Commission (FCC) has allocated a 75 MHz spectrum in the 5.9 GHz band for Dedicated Short-Range Communication (DSRC) [7]. This 75 MHz spectrum is further divided into seven channels starting from Channel 172 to Channel 184. Control channel 178 is used for safety power applications. Channel 172 and 184 are reserved for high-power public safety messages [6]. Another 30 MHz spectrum is allocated for Cellular Vehicle-to-Everything (C-V2X) communication [9].

WAVE, a protocol stack built on top of DSRC, offers a set of communication protocols and services. It includes security and privacy features to protect the communication between vehicles and infrastructure. IEEE 802.11p is a wireless communication standard designed for VANET. It operates in the 5.9 GHz band, provides high-speed communication with high reliability and low latency, and supports V2V and V2I communication [7].

1.2 Motivation

Basic Safety Message (BSM) transmission is a crucial component of vehicular communication in DSRC/WAVE (Dedicated Short-Range Communications/Wireless Access in Vehicular Environments) systems. BSMs are messages that contain important safety-critical information, such as the position, speed, and direction of travel of a

vehicle, and they are broadcasted frequently to ensure that all vehicles in the vicinity can receive and use this information to improve safety on the road. The DSRC/WAVE system uses a 5.9 GHz frequency band to facilitate the transmission of BSMs. The frequent broadcast of BSMs can easily cause the channel to become congested as traffic density increases, resulting in lower reception probability and decreased effective transmission range due to limited channel bandwidth.

High channel congestion can lead to the following consequences [10]:

- decrease in successful packet receptions.
- increase in transmission delay.
- decrease in the communication range of a transmitter.

Channel congestion can be reduced by decreasing the frequency of BSM broadcast messages. However, doing so may result in decreased awareness of the vehicle's surroundings. Therefore, it is necessary to maintain a balance between congestion and awareness to ensure efficient communication. In VANET, awareness is the ability of a vehicle to understand the current state of the network. It involves having up-to-date information, in the form of messages, about the state of other vehicles, road infrastructure and any potential hazards in the environment. It helps improve the safety, efficiency and reliability of the network. The proposed work focuses on safety applications in V2V communications by ensuring the reliable and efficient delivery of safety messages.

1.3 Problem Statement

Traditional congestion control methods address the congestion issue by adjusting the transmission parameters, such as data rate, beacon rate (also known as transmission or message rate) and transmission power. To alleviate the congestion in the channel, one or more of the following actions should be taken: i) lower the transmission power, ii) lower the transmission rate or iii) increase the data rate. In this thesis, we focus

on the data rate. It helps in reducing congestion by directly controlling the time needed to transmit each BSM. It is more adaptable to changing network conditions and improves network performance more efficiently by preventing packet loss and reducing delay. The research objective is to maintain the channel load below the threshold, while simultaneously enhancing the vehicles' environmental awareness.

1.4 Solution Outline

The proposed solution to the problem is an RL-based congestion control technique. RL-based congestion control can be an effective way to alleviate congestion and ensure efficient communication between vehicles since it can learn from the environment and optimize the communication process based on real-time traffic conditions. This means that RL-based congestion control can adapt to changing traffic conditions and provide appropriate communication performance for different traffic conditions. We define the data rate selection as a Markov Decision Process (MDP), where each vehicle is considered an agent that interacts with its surroundings and chooses its transmission parameters, the data rate in this instance, based on the existing network conditions. The Q-Learning algorithm, an RL model, trains the vehicles to choose the correct transmission parameters, by observing the dynamic vehicle density and channel busy ratio (CBR). This training is done using the observational data obtained from the initial simulations that were conducted using the 12 different vehicle densities and 7 different data rates. This data serves as the foundation for training the algorithm which creates a Q-table. The Q-table contains Q-values for each state-action pair. Each state has 7 different Q-values corresponding to 7 different actions (data rates). After the Q-table is generated, it is applied in OMNeT++ so that the vehicles can use the Q-values to make decisions regarding the data rate for transmitting a BSM. The vehicle chooses the data rate corresponding to the highest Q-value in a particular state. The vehicle then uses this data rate to transmit a BSM to other vehicles and their surrounding infrastructures in the network.

1.4.1 Contributions

The main contribution of this research is summarized as follows:

- A framework to solve the congestion control problem with the Q-Learning algorithm, an RL method that utilizes discrete state and action space.
- A reward function was defined by combining CBR value and data rate to maintain the channel load under a target threshold with the least data rate possible.
- The Q-learning algorithm was applied with the data directly obtained from a simulated dynamic traffic environment for training purposes. This way, observing the CBR values for different data rates and vehicle densities makes the vehicle's state transitions more realistic.
- Simulation results show that the proposed Q-Learning model can successfully maintain the target channel load under various dynamic traffic scenarios and exhibits a lower packet loss compared to existing methods like Transmit Data Rate Control (TDRC) [11], Data Rate - Decentralized Congestion Control (DR-DCC) [12] and Adaptive Data Rate-based Congestion Control (DRCA) [13].

1.5 Thesis Organization

The remaining outline of this thesis is as follows: chapter 2 provides an overview of the fundamental concepts of VANET, an overview of Reinforcement learning and its different algorithms, along with a literature review of related work in different parameter-based congestion control techniques in VANET. Chapter 3 presents an outline of the proposed methodology followed by the Q-Learning algorithm. Chapter 4 provides an experimental setup and the comparison of results. In the end, chapter 5 gives a conclusion followed by possible future work on the proposed methodology.

CHAPTER 2

Related Work

2.1 Overview of VANET

VANET is a type of wireless network that enables communication between vehicles and roadside infrastructure. It allows vehicles to exchange information about their location, speed, and other relevant data. This information can be used by other vehicles, infrastructure components, and drivers to avoid accidents, optimize traffic flow, and provide real-time information about road conditions. VANET has the potential to significantly improve road safety and the efficiency of intelligent transportation systems. Some of the key applications of VANET include [14] [15]:

- Collision Avoidance: Vehicles can exchange information about their location, speed and direction to help avoid collisions.
- Traffic Management: Vehicles can exchange information about traffic conditions and provide real-time data to help optimize traffic flow.
- Emergency Services: VANET can be used to quickly alert emergency services to accidents or other emergencies.
- Navigation and Routing: Vehicles can receive real-time information about road conditions and use this data to choose the fastest, safest, or more efficient route.

2.1.1 Types of Communication

In VANET, Communication is categorized into four classes as follows [16]:

1. Vehicle-to-Vehicle (V2V): V2V communication enables sharing of information about vehicle position, speed, and other relevant data among vehicles without the need for an intermediary infrastructure. Applications such as collision avoidance, cooperative driving, and traffic monitoring use V2V communication.
2. Vehicle-to-Infrastructure (V2I): V2I communication enables sharing of real-time information like road conditions, traffic congestion, and other relevant data between vehicles and the infrastructure surrounding them, such as traffic lights, roadside units, and other fixed infrastructure. Applications such as traffic management, sign control, and parking assistance use V2I communication.
3. Vehicle-to-Pedestrian (V2P): V2P communication enables information sharing between vehicles and pedestrians to alert them of danger. Applications such as pedestrian safety and traffic management use V2P communication.
4. Vehicle-to-Everything (V2X): V2X communication enables information sharing between vehicles and any other entity, such as vehicles, traffic lights, or road signs.

2.1.2 Congestion Control and its Challenges

In VANET, channel congestion refers to the situation where the number of vehicles trying to access the same communication channel exceeds its capacity, leading to performance degradation and delays in communication. Each vehicle periodically broadcasts BSMs to other vehicles in the network through a communication channel. Due to limited channel bandwidth, when the vehicle density increases, the number of BSMs being broadcasted also increases, leading to congestion in the channel. This channel congestion leading to unreliable delivery of safety messages and packet loss has been a challenging problem for VANET. Therefore, congestion control algorithms have received significant research attention in recent years.

2.2 Overview of Reinforcement Learning

Reinforcement learning is the subfield of artificial intelligence that deals with the problem of how an agent can learn to take actions in an environment to maximize the cumulative reward. The agent interacts with an environment and learns from the feedback provided by the environment in the form of rewards (positive values) or punishments (negative values) for the actions it takes. In general, it learns through trial and error. The goal of the agent is to learn a policy that maps states to actions in a way that maximizes the expected cumulative reward over time. With time, the agent learns to avoid negative rewards and seek positive rewards. There are two kinds of reinforcement learning: Positive Reinforcement and Negative Reinforcement [17] [18]. Positive reinforcement learning occurs when the agent receives a positive reward for taking a particular action, increasing the likelihood of that action being repeated in the future. Negative reinforcement occurs when the agent receives a negative reward for taking a particular action, decreasing the likelihood of that action being repeated in the future. RL applications include robotics for industrial automation, gaming, recommendation systems, data processing, the creation of training systems, natural language processing, image processing, etc.

2.2.1 Basic Reinforcement Learning Concepts and Terminologies

The key components of reinforcement learning are the agent, the environment, the reward, the policy, the state and the action. The agent is responsible for making decisions based on the information it gathers from the environment. The environment is the external system with which the agent interacts and receives feedback, including state observations, rewards, and transitions. The reward comes in the form of feedback that the environment gives the agent following each action, letting them know whether their efforts were successful in accomplishing their objectives. A policy is a mapping between states and actions that the agent learns to take in order to maximize its expected cumulative reward. A policy specifies what action the agent should

take in a given state based on the current state and the agent’s past experiences. A state is a representation of the current situation or environment in which the agent is operating. An action is a decision made by the agent in response to a particular state. In RL, The goal is to find the best policy that maximizes the expected cumulative reward over time. The agent interacts with the environment by observing its current state, taking action based on its policy, receiving a reward from the environment, and transitioning to a new state. The agent learns from its experience and updates its policy accordingly to improve its future decision-making. MDP [19] framework is used to define this decision-making problem. MDP stands for Markov Decision Process. It is defined by a set of states, a set of actions, a reward function and a discount factor. MDPs are widely used in reinforcement learning because they provide a formal framework for modelling complex decision-making problems and can be solved using efficient algorithms. Many real-world problems, such as robotic control, game playing, and financial decision-making, can be modelled as MDPs and solved using RL techniques. Some of the RL algorithms are discussed below.

2.2.2 Reinforcement Learning Algorithms

There are three main types of RL techniques [20]:

2.2.2.1 Model-Free Reinforcement Learning

In this approach, the agent learns an optimal policy without having an explicit model of the environment. The agent learns directly from the observed state, action, and reward data. Some of the model-free reinforcement learning algorithms are:

1. Value-based: Value-based RL algorithms learn a value function that assigns a value to every state or state-action pair. Examples of value-based RL algorithms include:
 - Q-Learning
 - SARSA

2. Policy-based: These methods learn a policy that maps states to actions.

2.2.2.2 Model-Based Reinforcement Learning

In this approach, the agent has an explicit model of the environment, including the rewards and the transition probabilities. The agent uses this model to simulate the environment and learn an optimal policy. Some of the popular algorithms that fall under this category are:

- Model-based Value Iteration (MBVI)
- Monte Carlo Tree Search (MCTS)

2.2.2.3 Hybrid Reinforcement Learning

This approach combines elements of both model-free and model-based RL. The agent uses a model to simulate the environment but also learns directly from observed data. Some of the hybrid RL techniques include:

- Dyna-Q
- Model-based Policy Optimization (MPO)
- Proximal Policy Optimization (PPO)

In this thesis, we use a model-free RL approach for VANET congestion control. Two of the most commonly used RL algorithms in this category are briefly discussed below:

1. Q-Learning: It is a popular model-free reinforcement learning algorithm that learns an action-value function $Q(s,a)$, which estimates the expected long-term reward of taking action \mathbf{a} in state \mathbf{s} and following an optimal policy thereafter. The algorithm updates its estimates of $Q(\mathbf{s},\mathbf{a})$ based on observed state transitions and rewards. Q-learning is a simple and effective algorithm for solving a wide range of reinforcement learning problems.

2. SARSA: SARSA stands for State-Action-Reward-State-Action, which is another popular model-free reinforcement learning algorithm. SARSA is similar to Q-learning, but the main difference is that SARSA is an on-policy algorithm, meaning that it learns the Q-values for the policy that it is currently using to select actions. In contrast, Q-learning is an off-policy algorithm that learns the optimal Q-values regardless of the policy being used. It is applied in situations where the policy needs to be explicitly learned.

2.2.3 RL for Congestion Control in VANET

In VANET, congestion control is a critical problem for safety communication due to the limited wireless channel bandwidth. Congestion control's main goal is to reduce the channel load by adjusting transmission parameters such as power, message rate and data rate. Vehicles need to select the most appropriate action to reduce the channel load in various traffic conditions in a highly dynamic environment where the network conditions are constantly changing. When the vehicle density is low, the channel load tends to be lower, making it optimal to reduce the data rate to achieve a higher BSM delivery rate. Conversely, in high vehicle densities, where the channel is high, the data rate must be adjusted accordingly. This can be viewed as a decision-making challenge as numerous factors, such as vehicle density and CBR, can influence the decision [21]. This decision-making problem can be modelled using Markov Decision Problem (MDP), a mathematical framework used in RL. The congestion problem is formulated as an MDP and solved using an RL algorithm. In this thesis, we use the Q-learning algorithm, a model-free RL algorithm, as it does not require the agent to have knowledge of the underlying model of the environment. It is a popular choice in congestion control because it is well-suited to handle the complex and dynamic nature of the network conditions.

2.3 Literature Review

This section focuses on different transmission parameters used to reduce the congestion in the channel.

2.3.1 Transmission Power Adaptation Approaches

Transmission power-based congestion control is a technique used in wireless communication systems to control congestion in the network. This technique involves adjusting the transmission power of the wireless devices based on the level of congestion in the network. In VANET, transmission power determines the range of communication between vehicles and between vehicles and other road infrastructures. A higher transmission power allows vehicles to communicate with other devices at a longer range, while lower transmission power limits the range of communication. However, a higher transmission power also results in congestion, which can lead to increased packet loss and decreased performance and reliability of the network.

In [22], Torrent-Moreno et al. proposed a Distributed Fair Transmit Power Adjustment for Vehicular Ad Hoc Networks to reduce congestion by adjusting the transmission power based on the number of vehicles in the surrounding area. It aims to keep the channel load under a predefined Maximum Beacon Load (MBL) by choosing an optimal transmission power for beacon transfer. However, predicting the MBL is challenging as vehicle density increases.

In [23], the authors proposed a beacon transmission power control algorithm for VANET based on channel load prediction using the Kalman filter algorithm to keep the channel load within a predefined range and therefore prevent channel congestion. Two threshold values, Maximum and minimum, are predefined. The transmission power is adjusted such that the forecasted channel load is maintained between these two predefined values to avoid channel congestion. This approach also aims to reduce interference and improve the overall network performance by adapting the beacon transmission power to the varying channel conditions.

In [24], the authors proposed a congestion control technique that randomly selects

transmission power from a given probability distribution called Complementary Cumulative Distribution Function (CCDF). To maintain fairness, the mean and variance of the distribution are adjusted for all vehicles in the area. This technique results in an increase in awareness quality and a reduction in channel load.

In [25], the authors proposed a Traffic Density-Based Distributed Congestion Control Strategy for Vehicular Communication, which adapts the transmission power based on the vehicular density in the network. A lower transmission power was used when the vehicle density was high, a medium transmission power was used when the network condition was moderate, and a higher transmission power was used when the vehicle density was low. This approach aims to reduce packet loss, inter-packet delay, beacon error rate and channel busy time and therefore increase the network performance.

In [26], the authors proposed a Speed Based Distributed Congestion Control Scheme for Vehicular Networks in order to reduce channel congestion and beacon error rate by adjusting the transmission power based on the vehicle's driving speed. The faster a vehicle travels, the less dense the network becomes. In this case, higher transmission power can be used as there is more space between the vehicles in the network. At the same power, a lower-speed traffic network may lead to high congestion. As a result, lower transmission power is used.

In [27], Wang et al. (2022) proposed an adaptive beacon message transmission power control scheme for VANET. This method adjusts the transmission power of beacon messages based on the vehicle position prediction error. It increases the transmission power for vehicles with large position prediction errors and reduces it for vehicles with small position prediction errors. The performance is evaluated in terms of packet delivery ratio, network throughput and delay. The results showed that the proposed scheme could effectively improve the network performance and reduce interference between adjacent vehicles. This research has some limitations. Firstly, this technique does not incorporate a mechanism to measure the distance error, which is crucial in determining the transmission power level required to maintain reliable communication among vehicles. Secondly, the paper sets a maximum beacon load

of 70%, which may not be sufficient in dense traffic scenarios. Lastly, this technique is limited to a small-scale simulation, and it is unclear how the algorithm works in large-scale VANET with varying traffic conditions.

2.3.2 Transmission Rate Control Approaches

Another parameter that can be used for congestion control in VANET is the transmission rate. The terms message rate, transmission rate and beacon rate are used interchangeably for the remainder of this thesis. In VANET, the transmission rate determines the rate at which the messages are transmitted from one vehicle to another in the network. A higher transmission rate allows for faster and more reliable communication between vehicles, which is essential for safety applications like collision avoidance and emergency messaging. However, a higher transmission rate also results in congestion, which in turn leads to increased packet loss, delays and reduced network throughput. The standard transmission rate for BSMs is set at 10 packets per second, i.e., 10 Hz.

In [28], the authors introduced a novel Transmission rate adaptation-based method for congestion control called Periodically Updated Load Sensitive Adaptive Rate Control (PULSAR). This method involves measuring the Channel Busy Ratio (CBR) at the end of a fixed time interval called Channel Monitoring and Decision Interval (CMDI) and comparing it to a target threshold value of 0.6. The transmission rate was either decreased if the measured value exceeds the target value or increased otherwise, and this transmission rate increment or decrement is calculated using Additive Increase Multiplicative Decrease (AIMD). The results demonstrate that this approach can effectively control congestion and achieve fairness.

The work in [29] by Bansal et al. proposes a technique called Linear Message Rate Integration Control (LIMERIC) algorithm, a widely recognized transmission rate-based congestion control method. The aim of LIMERIC is to equitably allocate the available channel bandwidth among all vehicles in a DSRC network by dynamically adjusting the rate at which Basic Safety Messages (BSMs) are transmitted. LIMERIC uses linear feedback to adjust the message rate based on the observed net-

work congestion in a specific region to meet a predefined CBR. While LIMERIC is successful in reducing network congestion, it may lead to an increased Inter-Packet Delay (IPD).

The error model-based adaptive rate control (EMBARC) algorithm [30] adapts the transmission rate of V2V communication based on the current channel conditions like channel load and vehicular dynamics. In EMBARC, the transmission rate is a function of the vehicular dynamics (using an STE Component) and the channel load (using a LIMERIC component). It integrates a Suspected Tracking Error (STE) component with LIMERIC [29]. The LIMERIC component computes the transmission rate based on the channel load, using which the next packet is sent. The STE component plays a crucial role in determining the estimated time for the channel to reach its threshold. It ensures that the packet is not transmitted after that threshold has been reached.

In [31], Ogura et al. introduced a rate-based congestion control method called BRAEVE (Bsm RAte control using the Estimated number of VEHicles in the communication range). This method uses the number of observed vehicles in the communication range as a parameter to adjust the transmission rate to transfer a beacon.

In [32], the authors proposed an adaptive congestion control algorithm that is an extended version of the LIMERIC algorithm, which aims to achieve weighted fair message rates. This algorithm controls the total channel load based on a target value.

In the paper [33], the authors proposed CBA: Collision-based beacon rate adaptation scheme. This technique adapts the beacon rate based on the number of detected collisions by a vehicle. When the number of collisions is high, the beacon rate is decreased; when the number of collisions is low, the beacon rate is increased. It consists of two phases: a) a monitoring phase to monitor the number of collisions in the network to detect the presence of congestion, and b) an adaptation phase to adapt the beacon rate according to the local vehicle density and reduce the congestion.

Kim et al., in [34], proposed a Request based Adaptive Beacon Transmission Rate control method which increases the number of successfully transmitted beacons and provides beacon transmission rate fairness between vehicles. A vehicle requests each

nearby vehicle whether to increase or decrease its transmission rate. Based on the current channel condition, each vehicle in the network transmits beacons along with BTR adjustment requests.

In [35], researchers introduced Congestion Aware Message (CAM) for beacon signals on the vehicle environment based on the vehicle ID. This method weighs the randomized back-off numbers selected by the vehicles participating in the back-off procedure using the vehicle ID embedded in each of their own CAMs. This leads to the identification of a vehicle ID-based randomized back-off code, which lowers the risk of a collision due to the identical back-off number. This method is effective in terms of packet delivery ratio, energy consumption, delay, success rate and collision.

The authors, in [36], formulated the beacon rate congestion control problem as a Markov decision process and solved it using approximate reinforcement learning. On-policy control with function approximation is applied to arrive at sensible decisions when choosing an action. This method shows good performance in terms of packet delivery and collision ratios. However, the action space is limited, consisting of only three actions, which limits the vehicle's ability to reduce congestion.

2.3.3 Data Rate Control Approaches

The data rate (also called *throughput*) refers to the amount of data that can be transmitted in a given unit of time. It is often measured in bits per second. There are 7 possible data rates that can be used for V2V communication: 3, 6, 9, 12, 18, 24 and 27 Mbps [13]. A lower data rate allows for the data packets to be transmitted over a more extended period, which increases signal strength and chances of successful packet delivery. However, a lower data rate also results in channel congestion, which in turn leads to increased delays and dropped packets. Using a higher data rate reduces channel congestion, but also decreases awareness and increases the signal-to-interference-plus-noise ratio (SINR) requirement [12]. So, an appropriate data rate has to be selected to send out a beacon such that a balance between congestion and awareness is maintained. There are several data rate-based congestion control techniques to minimize channel congestion in the network.

In [37], the authors concluded that the default data rate for vehicular safety communications should be 6 Mbps.

In [38], the authors introduced a binary data rate congestion control algorithm that adjusts the data rate dynamically based on the channel load. This method uses three states: Relaxed, Active and Restrictive states. It tries to maintain the channel busy ratio (CBR) between 40 and 60% by selecting one of the four data rates: 6, 9, 12 and 18 Mbps. If CBR exceeds 60%, it increases the data rate. If CBR goes below 40%, it decreases the data rate. If CBR lies between 40 and 60%, no data rate change is needed. This way, the congestion can be minimized.

The authors in [12] introduced a data rate-based congestion control scheme that follows the Decentralized Congestion Control (DCC) framework called Data Rate-DCC or DR-DCC. The algorithm increases the data rate to reduce the CBR. The transition from one state to another was based on CBR measurements for every T seconds, where the states correspond to levels in DCC. The algorithm increases the level, if the CBR is higher than the mean threshold C_t while maintaining the same level when the CBR is lower than the mean threshold C_t and greater than C_{min} . They decreased the data rate level whenever the CBR is below a specific min threshold C_{min} .

The authors in [39] introduced a packet-count-based decentralized data rate congestion control algorithm (PDR-DCC). It uses a packet count P_c along with CBR measurements of every second to select the optimal data rate for each vehicle. Results show that channel load was maintained near the desired target, and larger awareness was achieved.

In [11], the authors implemented a multi-state active DCC algorithm which adapts data rate according to the available channel load. They considered three states based on CBR values. A data rate 12 Mbps is used for the message transmission if CBR is less than 0.2, which is a relaxed state, a data rate 9 Mbps is used for the message transmission if CBR is between 0.2 and 0.43, which is the active state 1, and a data rate 6 Mbps is used if it's between 0.43 and 0.6 which is called as an active state 2 and 3 Mbps is used otherwise. It is to be noted that this scheme can switch between only four data rates. This mechanism is implemented in a V2V propagation simulator

called *GEMV*².

In [13], the authors proposed Adaptive Data Rate-based Congestion Control in Vehicular Ad Hoc Network, which dynamically selects an appropriate data rate for each BSM transmission, based on the current Channel Busy Ratio (CBR). It attempts to maintain the CBR between two specified thresholds: cbr_{low} and cbr_{high} . When the CBR is below the desired threshold, the proposed algorithm calculates the appropriate data rate and starts transmitting the message directly using this data rate, rather than going through intermediate levels. This technique was able to bring the channel congestion to the desired level leading to lower packet loss and improved packet delivery ratio.

2.3.4 Hybrid Approaches

A number of recent approaches have been proposed that use a combination of many parameters instead of using a single parameter to effectively control congestion. The ultimate goal of congestion control is to improve vehicle awareness. While decreasing the transmission power or transmission rate may reduce congestion, it also reduces vehicle awareness. On the other hand, increasing awareness with high transmission power or rate leads to increased congestion. Some of the hybrid congestion control approaches that use multiple transmission parameters are discussed in this section.

In [40], Aznar-Poveda et al. proposed a new approach called MDPRP (Multi-Dimensional Power and Rate Policy) that uses Q-learning to jointly control the transmission rate and power in VANET. They modelled the congestion problem as an MDP and solved it using the Q-learning technique. This MDP characterization is denoted as MDPRP (indicating Rate and Power). This method not only keeps the channel load under the desired level but also offers favourable results in terms of packet delivery ratio.

The authors in [41] proposed a new approach called Combined Power and Rate Control (CPRC) which adjusts both rate and power in a single algorithm. CPRC introduces cooperative behaviour by allowing vehicles that are not directly involved in a dangerous situation to reduce their transmission power and to allow vehicles,

involved in a potentially dangerous situation, to increase the transmission rate without having the channel load exceed the predefined target value. These vehicles, which reduce the transmission power as a consequence of the transmission rate increase of other vehicles, can increase their transmission rate.

In [42], the authors proposed an approach for adjusting the transmission power, which switches between high and low-powered transmissions and can be used with and without existing transmission rate control algorithms. This algorithm maintained two separate transmission power levels and transmission rates such that close-by vehicles send packets at high transmission rates, while far-off vehicles send packets at high transmission power. This approach showed good performance in terms of both awareness levels and packet error rates.

In [43], the authors presented a channel-aware congestion control algorithm (CACC) that adjusts both the data rate and transmission power. They took the received signal strength (RSS) into account when detecting packet loss to determine the channel conditions like fading or congestion. CACC decreases the data rate in the case of severe fading or when a desirable packet collision rate (PCR) is reached to increase the packet delivery rate. It also adjusts the transmission power to maintain a desirable packet error rate. It decreases the transmission power to reduce the transmission range, thereby decreasing the number of packet collisions.

In [44], the authors proposed a new congestion control technique called Random Transmission Power Control to reduce the channel load by randomly selecting transmission power based on given probability distribution. The results demonstrated that this technique could improve Awareness Quality and reduce channel congestion.

In [45], the authors present two DCC mechanisms that adapt message rate and data rate combined with transmission power control mechanisms called Combined power and message rate adaptation (CPMRA) and Combined power and data rate adaptation (CPDRA). In the CPMRA mechanism, to keep the congestion within predetermined limits, transmission power and message rate are adjusted based on the current CBR value. If CBR is below the threshold, CPMRA increases the message rate, and if CBR exceeds the threshold value, the message rate is decreased. The

power control algorithm is applied at the ego node based on the CBR value and current neighbours and environmental conditions. In CPDRA, the data rate and transmission power are adjusted based on the current congestion status. The data rate values are selected depending on the current channel congestion. As the CBR value increases, the data rate is reduced depending on the threshold CBR value and DCC state to reduce the collisions between safety messages. To prevent data loss, the data rate adaptation mechanism modifies the communication bandwidth. The ego node modified its transmission power in accordance with the CPMRA algorithm's power control mechanism and CBR values of nearby nodes.

2.3.5 Discussion

To reduce channel congestion, researchers have developed a number of congestion control strategies. The network's awareness may be negatively impacted by transmission rate or power-based congestion control techniques. So, this thesis focuses on data rate as the control parameter to alleviate channel congestion by maintaining vehicles' awareness. Recently, RL has gained considerable attention in solving various problems in VANET, including congestion. At present, there is little research on RL-based solutions for congestion control in VANET. This study demonstrates how to use RL-based methodology to reduce congestion by adjusting the data rate.

This thesis proposes a Q-learning-based congestion control algorithm to maintain the channel load under a specified target threshold. The performance of the proposed method is compared in terms of the Channel Busy Ratio (CBR), the number of packets received, the number of total packets lost and the Beacon Error Rate (BER).

CHAPTER 3

Reinforcement Learning-based Congestion Control

3.1 An RL-based Framework for Congestion Control in VANET

VANET has emerged as a promising technology for enabling communication among vehicles and with infrastructure, thereby improving road safety and traffic efficiency. However, the dynamic nature of vehicular networks can lead to congestion, making it a critical challenge for safety communications in VANET due to the limited wireless channel bandwidth. Congestion control aims to maintain the channel load, which is frequently measured using the CBR, under a predetermined target value. This value is called Maximum Beaconing Load, whose optimal value is considered to be approximately 60 or 70 percent of channel capacity, which allows for the remaining percentage of the channel to be available for other necessary services [40]. A heavier load may result in increased packet loss and disrupt the safety application operation. To overcome this problem, congestion control is used. Congestion can be controlled by increasing the amount of data being transmitted at a particular time period, i.e., increasing the data rate. However, when increasing the data rate, there is decreased awareness about the network, resulting in vehicles being less informed about their surroundings. So, the vehicles must balance congestion control and awareness by adapting the most suitable data rate to send a BSM. In order to manage congestion

without sacrificing situational awareness, it is suitable to choose a higher data rate when the density of vehicles is high, while a lower data rate is more appropriate in low-density situations. This can be referred to as a decision-making process where the vehicle determines the appropriate data rate to utilize when transmitting a BSM. The vehicle should take into account factors, such as current vehicle density and current CBR, before making the right decision of choosing an appropriate data rate to send a BSM. This implies that the vehicles' decision-making should solely rely on the current circumstances, disregarding previous conditions. This is based on Markov property. It states that "the future is independent of the past, given the present". Therefore, we can formulate our congestion control problem as an MDP. MDP is a mathematical framework used to model decision-making problems, and RL can be a suitable mechanism for finding solutions to an MDP.

RL-based congestion control in VANET involves utilizing reinforcement learning algorithms to dynamically adjust the transmission parameter, specifically the data rate, in response to changing traffic conditions and network congestion. The objective is to optimize the utilization of available network resources and maintain efficient communication within the VANET. By employing RL techniques, the agent or the learner learns from its interactions with the environment, which includes monitoring network conditions, traffic patterns, and congestion levels. The RL agent then makes decisions on adjusting the data rate based on its learned policies to control congestion. RL has 5 elements: an agent, an environment, an action, a state and a reward. The agent, which in our context is the vehicle, is the entity responsible for decision-making or learning from its environment to achieve the best possible behaviour or outcome. The agent does not receive direct guidance on which actions to choose rather, it must engage in exploration and exploitation to determine the actions that result in the greatest rewards. The primary challenge in reinforcement learning is finding the right balance between exploration and exploitation. The agent must exploit its current knowledge to maximize immediate rewards while also exploring to improve future decision-making. The outcomes of these actions can impact not only immediate rewards but also future rewards. These two crucial aspects, trial-and-error learning

and delayed rewards, are the defining characteristics that set reinforcement learning apart [46]. Here, the vehicle is responsible to select the appropriate data rate based on the current state of the environment and receives the rewards after each decision. The environment is the external surroundings or the conditions that an agent interacts with, which in our context are the wireless channel, other vehicles, pedestrians, etc. It is responsible for providing the agent with observations and rewards based on its actions. An action refers to the specific move or decision made by an agent in response to the environment's current state. The set of possible actions that an agent can take in a given state is called action space. In this thesis, the actions are the possible data rates that a vehicle can use to send a BSM. A state is a collection of all the relevant information that the agent can observe about the current state of the environment. The state can be discrete or continuous, depending on the nature of the environment. In this work, the state represents the channel status and vehicle density. A reward is a scalar value that represents the feedback the vehicle receives from the wireless channel after choosing a data rate in a particular state to send a message to other vehicles in the network.

Figure 3.1 shows the RL cycle for the proposed approach for congestion control in V2V communication.

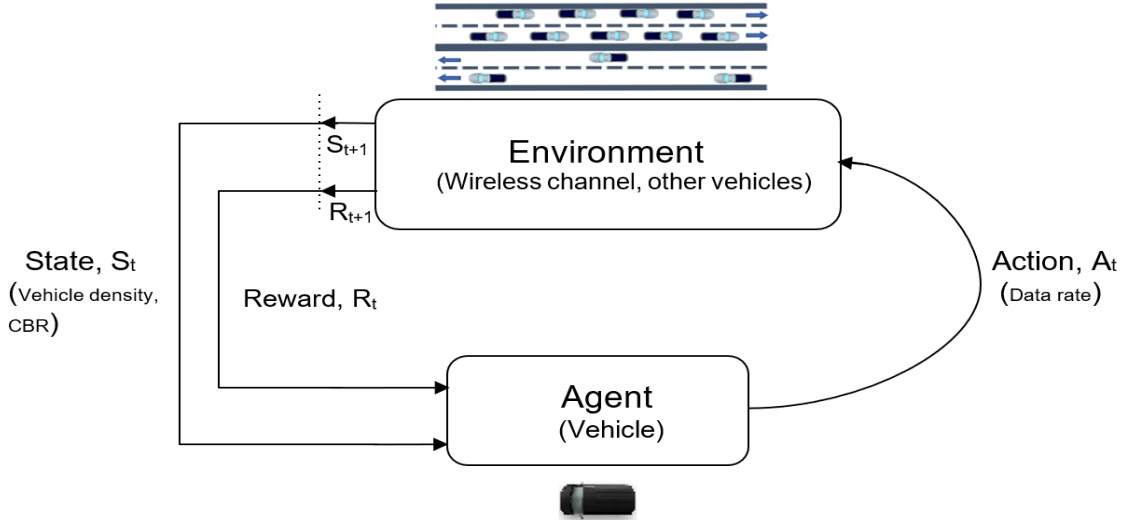


Figure 3.1: An RL cycle in VANET

In figure 3.1, the vehicle (agent) interacts with the environment (wireless channel

or other vehicles) at each state at time steps $t = 0, 1, 2, \dots$. The vehicle receives some representation, such as channel status and vehicle density, of the environment's state, $S_t \in S$ and selects an action, $A_t \in A(s)$. After the time step t , the vehicle will receive a numerical reward, $R_{t+1} \in R$ and moves to a new state, S_{t+1} . RL aims to maximize the long-term reward rather than the immediate reward.

3.2 Problem Formulation as Markov Decision Process

MDPs are useful for congestion control as they provide a mathematical framework to model the complex and dynamic nature of the environment. MDPs can represent the interactions between network nodes and the flow of data, where the state is the current congestion level, and the action is the data rate to send a BSM. The MDP model can be used to formulate the decision-making process for congestion control. RL algorithms can be applied to the MDP model to learn the optimal policy, i.e., the policy that keeps the channel load below a target threshold by selecting the appropriate data rate. Therefore, we design the MDPs to solve the congestion problem using RL. The following are the key elements which are used to solve the MDP for congestion control:

- The *environment*: It is the world that the agent lives in and interacts with. The environment is characterized by uncertainty, and the agent can influence it by taking action. However, it cannot change the rules or dynamics of the environment by those actions. In this thesis, the environment includes the wireless channel or the other vehicles that the agent is communicating with. The uncertainty is the dynamic traffic flow, such as dynamic vehicle density, and velocity. The environment is responsible for providing feedback to the agent on the effectiveness of its actions, such as whether the congestion level has been reduced or not.
- The *agent*: In our scenario, the vehicle acts as the agent, which is responsible

for making decisions on which actions to take. The agent's goal is to learn an optimal policy for congestion control. The agent continuously interacts with the environment, receives the current state (channel load and vehicle density) of the environment and takes action, i.e., adjusting the data rate, to reduce the congestion. The agent, then, receives rewards based on their actions.

- The *goal*: The agent's goal is to learn how to make decisions that maximize the cumulative reward it receives from the environment. To make decisions is to choose an optimal action for each state. In this thesis, the goal is to maximize the reward for actions that maintain the CBR below 0.6. This value is defined as the Maximum Beacon Load (MBL) and is considered the optimal value according to several works [40], [36], [21].
- The *action*: In reinforcement learning, an action is a decision or a choice that an agent makes at a particular state. The actions can include adjusting transmission power, beacon rate and a data rate that a vehicle can use to transfer the beacon messages. This thesis considers only the data rate. We defined the action space, which is the set of possible actions that an agent can take in a given state, as 7 discrete data rates, and they are 3 Mbps, 6 Mbps, 9 Mbps, 12 Mbps, 18 Mbps, 24 Mbps and 27 Mbps. The agent selects one action from the action space at each state.
- The *state*: A state is a representation of the environment at a particular time, which captures all the relevant information that the agent needs to make decisions about what action to take next. This information includes factors that affect the wireless channel status, such as vehicle density, CBR, or the number of neighbouring vehicles, etc. The state space is defined as a 2-tuple, including the CBR and vehicle density. It is denoted as $s = (CBR, VD)$, $CBR \in \mathbb{R}^+$, $0 \leq CBR \leq 1$, and $VD \in \mathbb{N}$, $1 \leq VD \leq maxVD$. CBR is the Channel Busy Ratio, a metric to evaluate the amount of time that a wireless channel is sensed busy, and it lies between 0 and 1. The vehicle density is determined by the number of vehicles within a 100 m radius. $maxVD$ is the maximum vehicle density used,

and it is set to 50 in this thesis. Each vehicle density has 7 CBR values corresponding to 7 different data rates. It is crucial to note that the vehicle density cannot be known based on the action taken but can only be known based on the number of BSMs received from the neighbouring vehicles. As a result, the state space is made up of 350 individual states, with the vehicle selecting a new data rate from 7 possible data rates for each state and updating its state based on the information, in the form of BSMs, obtained from its neighbours.

- The *reward*: A reward is a numerical value that the agent receives on performing an action a in a state s . This reward can either be positive or negative based on the actions of the agent. In RL, the primary focus is on maximizing the cumulative reward (all the rewards) the agent has received from the environment, rather than just the immediate reward received at the current state (also called immediate reward). This total sum of rewards is called returns. The rewards are calculated based on the vehicle's goals and observations from the environment. A reward function is used to calculate the rewards, and it is defined as follows:

$$r(CBR, DR) = (CBR/DR) * sign(\eta - CBR) \quad (1)$$

where *sign* is the signum function shifted by the target value η . The objective of the proposed approach is to maintain the CBR below a specific threshold η while also trying to minimize the total packet loss, and Beacon Error Rate (BER) and maximize the number of BSMs received. Actions that result in a CBR value higher than η will have a negative reward, and this helps in speeding up the learning process [21]. In this thesis, we set $\eta = 0.6$ as the target channel load. A lower data rate is not encouraged as it can yield a negative reward if the CBR is high. Similarly, a higher data rate is also not encouraged because it would yield a lower reward. This reward function can help to select an ideal data rate for each state that provides the highest possible reward.

3.3 Q-Learning Algorithm

Q-Learning is an off-policy and model-free reinforcement learning algorithm used to solve MDP problems. In the context of congestion control in VANET, Q-learning can be used to determine the best action that needs to be taken based on the current state. It is known as off-policy because it learns from actions that are not currently part of the policy, such as taking random actions, and therefore does not require a policy. The goal of Q-Learning is to learn a policy that maximizes the total reward. The 'Q' in Q-learning stands for Quality. Quality, in this case, represents the quality of a particular action taken in a particular state, i.e., the expected reward for taking that action in that state and following the optimal policy thereafter. Traditional MDP models are less effective than Q-learning since it does not require complete knowledge of the environment's transition probabilities. Transition probabilities describe the probability of transitioning to state s' given an action a . It is denoted as a function $P(s, a, s')$ that gives the probability of ending up in s' given current state s and action a . These probabilities can be quite difficult or impossible to obtain due to the complex and dynamic nature of traffic flow, which makes it difficult to accurately model and predict the transitions between states. Another advantage of Q-learning is that it can handle environments with a very large number of states or actions, making it a more practical choice in some situations. In this work, we first implemented the Q-learning algorithm to generate the Q-table, which was later used for policy application to select the most suitable data rate for each BSM transmission. In order to implement the Q-learning algorithm effectively, the observational data obtained from the initial simulations are used. This data serves as the foundation for training the algorithm and enables the generation of a Q-table. The Q-table itself is a valuable resource as it contains the best policies for each state, representing the optimal strategies for selecting data rates for transmitting BSMs. These policies, stored in the Q-table, offer insights into the most favourable data rates for efficient and effective BSM transmissions by vehicles. The entire process, including the implementation of the Q-learning algorithm, and the generation of the Q-table, is thoroughly demonstrated in

the Algorithm 1. After the Q-table is generated, it serves as a crucial resource for the vehicles to make informed decisions regarding the data rate for the BSM transmission. The Algorithm 2 provides a detailed explanation of how the vehicle utilizes the Q-table to determine the most suitable data rate for each BSM transmission.

The following presents the details of the Algorithms.

Algorithm 1: This algorithm presents a summary of Q-learning-based Data rate congestion control (Q-DRCC) approach. To begin, all state-action pairs are set to an initial value of 0. The vehicle chooses an action a_t , at each time step t , observes and gathers information from the environment, receives a reward r_t , and moves to a new state s_{t+1} . The Q-value of the current state-action pair is then updated using the following Equation (2):

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(R_{t+1} + \gamma \max_a Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \quad (2)$$

where α is the learning rate, and $0 \leq \alpha \leq 1$. The learning rate determines how much of the newly learned value will be used. γ is the discount rate, where $0 \leq \gamma \leq 1$. It determines the importance of future rewards. A value of $\gamma = 0$ indicates that greater importance is given to immediate rewards, whereas a value of $\gamma = 1$ indicates greater importance is given to future rewards. A discount factor of 0 cannot learn effectively since it only considers immediate rewards, while a value of 1 may lead to infinite rewards. In this work, we set the discount factor as 0.99 and the learning rate as 0.01. $Q(s_t, a_t)$ represents the estimated action-value function or Q-value for the current state-action pair $Q(s, a)$, $\max_a Q(s_{t+1}, a_{t+1})$ is the maximum Q-value over all possible actions a_t in the next state s_t and $r(s_t, a_t)$ is the reward when an agent takes action a in state s at time step t .

Algorithm 1 Q-Learning-Based Data rate Congestion Control (Q-DRCC)

Parameters: step size $\alpha \in (0, 1]$, number of episodes**Result:** Q-table with values of each state-action pair

- 1: Initialize \mathcal{S} , the set of states (which contains one state for each data rate)
 - 2: Set $\mathcal{A}(s)$ to be the set of actions that can be taken in state s , which consists of the seven possible data rates regardless of the value of s .
 - 3: Initialize the Q-table Q , where $Q(s, a) = 0$ for all $s \in \mathcal{S}, a \in \mathcal{A}(s)$
 - 4: **for** each episode **do**
 - 5: Set s to be a random state in \mathcal{S}
 - 6: **for** each step of episode **do**
 - 7: Choose action a from $\mathcal{A}(s)$
 - 8: Compute the reward $r(s, a)$ using Equation (1), where CBR is calculated using Equation (3), with the vehicle density as input
 - 9: Observe the next state s'
 - 10: Update $Q(s, a)$ using equation (2) and the obtained reward
 - 11: Take action a and move to the corresponding state
 - 12: Set s to be the new state
 - 13: **end for**
 - 14: **end for**
-

A total of 12 different traffic models with different vehicle densities: 1, 3, 5, 11, 13, 17, 19, 21, 27, 29, 37, and 45, were created to generate the Q-table by implementing Algorithm 1. The range of vehicle densities varied from 0 to $maxVD$, within a radius of 100 m. For this particular scenario, $maxVD$ was set to 50. A total of 84 simulations were run for each traffic model using the data rates 3, 6, 9, 12, 18, 24 and 27 Mbps. The purpose was to collect observation data, specifically the CBR, for each action at every state. Later, this initial simulation data collected was utilized to create best-fit curve functions. These functions that were derived are used to create Equation (3), which enables the prediction of CBR values for different combinations of data rate and vehicle density that were specifically used during the initial simulations. This function is used to predict the CBR values of the remaining vehicle density and data rate pairs that were not included in the initial simulations. To prevent values from exceeding 1 due to negligible changes in CBR at high densities, a maximum return value of 0.92 was imposed. The Q-learning algorithm was then executed to generate a Q-table, which stores Q-values for each state-action pair. In this context, the state represents the combination of vehicle density and the CBR value, while the action

corresponds to the selected data rate.

The below equation (3) incorporates the variables VD (vehicle density), DR (data rate), and estCBR (estimated Channel Busy Ratio). This equation is employed by the Q-learning algorithm to estimate the CBR for different combinations of vehicle density and data rate. In the Q-table, each row represents a specific combination of vehicle density, ranging from 0 to 50, and the data rate (3, 6, 9, 12, 18, 24 and 27 Mbps) used by neighbouring vehicles. Whereas, each column represents the data rate available to the current vehicle with its corresponding Q-value.

Steps 1 and 2 in Algorithm 1 involve defining the state and action space, respectively. In step 3, the Q-table is initialized, with all values set to 0. To prevent the algorithm from running indefinitely, the number of episodes is set to 80,000. Setting the number of episodes too low hinders the agent's ability to learn effectively. Similarly, selecting a high number of episodes is not reasonable, as the difference in results beyond 80,000 episodes is negligible. Many research papers have adopted 80,000 episodes as a commonly used value [21][36]. In each episode of the algorithm, the vehicle executes steps 5 through 13 in order to update the Q-table. To facilitate the learning process, the algorithm assigns specific values to the discount factor γ and learning rate α . In this work, γ is set to 0.99, representing the significance given to future rewards, while α is set to 0.01. At the end of 50,000 iterations, the final Q-table is saved into a file for subsequent use in Algorithm 2.

$$\begin{aligned}
 estCBR(VD, DR) = & \left\{ \left\{ \begin{array}{ll} 0.0088 * VD + 0.6576 & \text{if } VD \leq 6 \\ 0.1662 \ln(VD) + 0.4049 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 3 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0139 * VD + 0.4023 & \text{if } VD \leq 11 \\ 0.2225 \ln(VD) + 0.1099 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 6 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0154 * VD + 0.2597 & \text{if } VD \leq 11 \\ 0.2242 \ln(VD) - 0.0043 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 9 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0154 * VD + 0.1764 & \text{if } VD \leq 11 \text{ OR } VD \geq 33 \\ 0.2114 \ln(VD) - 0.0543 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 12 \quad \text{if } BR = 5 \\
 & \left\{ \left\{ \begin{array}{ll} 0.014 * VD + 0.101 & \text{if } VD \leq 11 \text{ OR } VD \geq 33 \\ 0.1825 \ln(VD) - 0.0826 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 18 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0122 * VD + 0.0635 & \text{if } VD \leq 11 \text{ OR } VD \geq 33 \\ 0.1547 \ln(VD) - 0.0848 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 24 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0116 * VD + 0.0568 & \text{if } VD \leq 11 \text{ OR } VD \geq 33 \\ 0.1463 \ln(VD) - 0.0823 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 27 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0034 * VD + 0.8465 & \text{if } VD \leq 8 \\ 0.0765 \ln(VD) + 0.7167 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 3 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0079 * VD + 0.6553 & \text{if } VD \leq 6 \\ 0.1513 \ln(VD) + 0.423 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 6 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0107 * VD + 0.5152 & \text{if } VD \leq 5 \\ 0.1876 \ln(VD) + 0.2467 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 9 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0122 * VD + 0.4147 & \text{if } VD \leq 6 \\ 0.2013 \ln(VD) + 0.1431 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 12 \quad \text{if } BR = 10 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0134 * VD + 0.2918 & \text{if } VD \leq 5 \\ 0.2028 \ln(VD) + 0.0409 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 18 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0135 * VD + 0.213 & \text{if } VD \leq 5 \\ 0.1938 \ln(VD) - 0.012 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 24 \\
 & \left\{ \left\{ \begin{array}{ll} 0.0134 * VD + 0.1913 & \text{if } VD \leq 5 \\ 0.1894 \ln(VD) - 0.0241 & \text{otherwise} \end{array} \right. \right. & \text{if } DR = 27
 \end{aligned}
 \tag{3}$$

Algorithm 2 Policy Application of Q-DRCC in OMNeT++

```

1: curCBR = Get current CBR
2: curVD = Get current vehicle density
3: if curVD > maxVD then
4:   curVD = maxVD
5: end if
6: bestDatarate = 6 (default value to ensure there is always a valid output)
7: index = 1 (default value to ensure there is always a valid output)
8: maxVal = -1
9: level = 1 (default value to ensure there is always a valid output)
10: Initialize datarates[] = 3, 6, 9, 12, 18, 24, 27
11: for i = 0 to length[datarates] do
12:   if estCBR(curVD, datarates[i]) using equation (3) ≤ curCBR then
13:     index = i
14:     break
15:   end if
16: end for
17: for i = 0 to 6 do
18:   qVal = Get the entry at index i of the row in the Q-table corresponding to
        curVD and index
19:   if qVal > maxVal then
20:     maxVal = qVal
21:     level = i
22:   end if
23: end for
24: bestDatarate = datarates[level]
25: Send beacon using bestDatarate

```

Algorithm 2: After determining an optimal policy using Algorithm 1, a vehicle can utilize this policy to select the appropriate data rate by employing Algorithm 2. The algorithm consists of several steps to assess the vehicle's environment and make data rate decisions.

In the first two steps of Algorithm 2, the vehicle gathers information about the current state of the environment by measuring the CBR and vehicle density. If the measured vehicle density exceeds the maximum allowed value, which is denoted as *maxVD*, it is capped at *maxVD* in steps 3 and 4. Steps 6 to 9 involve initializing the variables *bestDatarate*, *maxVal*, *index* and *level*. *bestDatarate* is initially set to 6 Mbps, as this is considered the optimal data rate. Both *index* and *level* are set to 1, and *maxVal* is set to a very small number, -1, to ensure that always a higher value

is encountered in the Q-table. In Step 10, we initialize the `datarate[]` array with the following values: 3, 6, 9, 12, 18, 24 and 27. These values represent the available data rate options for the vehicle to choose from to send a BSM to other vehicles and their surroundings in the vehicular network.

Table 3.1: Partial Q-Table

Transmission rate (BSMs/s)	Vehicle Density	Data rate (Mbps)	Estimated CBR	Data rate (Mbps)						
				3	6	9	12	18	24	27
5	1	3	0.6664	-16	63	27	29	28	26	67
5	1	27	0.0684	74	55	50	42	36	37	76
5	5	3	0.7016	18	81	34	32	35	38	83
5	5	27	0.1148	88	52	47	45	45	38	88
5	10	3	0.7876	29	89	42	44	43	45	92
5	10	27	0.1728	95	47	50	46	44	48	95
5	50	3	0.92	102	123	101	101	101	101	101
5	50	27	0.6368	131	100	100	100	100	100	101
10	1	3	0.8499	38	85	43	47	48	52	96
10	1	27	0.2047	99	47	43	49	51	47	98
10	5	3	0.8635	37	89	48	45	42	44	97
10	5	27	0.2583	100	44	48	50	46	53	99
10	10	3	0.8928	45	94	41	44	42	49	98
10	10	27	0.412	100	50	49	50	49	49	99
10	50	3	0.92	100	120	100	101	100	100	100
10	50	27	0.7168	130	100	100	100	100	100	101

In steps 11 to 16 of Algorithm 2, the vehicle evaluates a possible data rate from surrounding vehicles. In line 12, equation (3) is used to determine the combination of vehicle density and data rates that results in the CBR value, which is less than or equal to the current CBR value. This calculation is performed to determine the current state of the wireless channel, represented by a two-tuple consisting of the vehicle density and CBR. It is to find the data rate used in the previous transmission by considering

the current vehicle density and all seven data rate levels. Each estimated CBR value is compared to the current CBR value, and if an estimated CBR is lower, it indicates that the corresponding data rate, which resulted in that estimated CBR, was used for the previous Basic Safety Message (BSM) transmission. This process is conducted to identify the appropriate row (state) in the Q-table, which guides the selection of the optimal data rate. Steps 17 to 23 involve the process of selecting the optimal data rate based on the provided Q-table. Each row in the Q-table represents a state of the environment, including vehicle density, the estimated data rate of neighbouring vehicles, and the corresponding estimated CBR. The Q-table contains values for each possible data rate that the vehicle can employ. The data rate with the highest value in a given row represents the optimal action or policy for that specific state.

Table 3.1 shows selected rows from the Q-table rather than the entire table due to limited space. In step 18, the algorithm accesses the specific row in the Q-table that matches the current state of the vehicle. It proceeds to compare each value in the row and keeps track of the maximum value encountered. This process ensures that the vehicle selects the data rate associated with the highest Q-value, indicating the optimal choice based on the provided Q-table. For instance, for a vehicle density is 1, a vehicle sending 10 BSM/s at the rate of 3 Mbps with an estimated CBR of 0.849, the optimal data rate is 27 Mbps with the highest Q-value of 96. Similarly, for vehicle density 10, a vehicle sending 5 BSM/s at the rate of 3 Mbps with an estimated CBR of 0.7876, the optimal data rate is 27 Mbps with the highest Q-value of 92.

3.4 Comparison with Existing Approaches

The proposed algorithm differs from existing traditional methods in several key aspects.

- The proposed method incorporates Q-learning, a popular RL technique, to make data rate decisions. This approach allows the agent/vehicle to learn and adapt to the changing network conditions based on past experiences and rewards. In contrast, some existing approaches rely on fixed and predefined rules for data

rate selection, lacking the ability to dynamically adjust to varying conditions.

- The existing data rate-based congestion control schemes [11][12][13][37][38][39] discussed in Section 2.3.3 did not consider vehicle density while selecting the data rate. Whereas, the proposed method considers vehicle density, CBR and the estimated data rate used by the neighbouring vehicles to select the optimal data rate.
- The proposed algorithm employs a Q-table that stores Q-values for each state-action pair offering an advantage in terms of reading speed. By organizing the Q-values in a tabular format, the algorithm can quickly access and retrieve the Q-values associated with specific state-action pairs. It enables efficient and effective decision-making in contrast to other existing solutions that involve additional processing tasks to compute the optimal data rate [40]. Overall, the utilization of the Q-table enhances the algorithm's performance and speed in learning and decision-making tasks.

CHAPTER 4

Simulations and Results

Evaluating the performance of congestion control algorithms for VANET in real-world scenarios poses challenges due to safety reasons, cost, and resource requirements. Therefore, a simulation environment is employed to evaluate the performance of the proposed approach. In this chapter, section 4.1 discusses experimental setup, including simulation tools used and the parameters involved. The following section 4.2 presents a comparison between the proposed approach and constant data rate transmissions, while section 4.3 compares the proposed approach with existing congestion control techniques such as DRCA, TDRC, and DR-DCC. The comparison involves evaluating various performance aspects, including the Channel Busy Ratio (CBR), number of received packets, number of total lost packets, and Beacon Error Rate (BER).

4.1 Simulation Setup

In this section, the simulation setup and the parameters used for the simulations are discussed. Simulation of Urban MObility (SUMO) [47] as the traffic simulator, Objective Modular Network Testbed (OMNeT++) [48] as the network simulator, and Vehicles in Simulation (Veins) [49], which combines the network and traffic simulator to simulate the vehicle communication, are used to run the simulations.



Figure 4.1: Simulation tools

A 4 km square-shaped highway with 1 km on each side was considered on which the vehicles were positioned for the simulations to test the performance of the Q-learning algorithm. We ran four different simulations considering the beacon rate of 5 Hz and 10 Hz each for 250 and 500 vehicles driving on the 4 km square road. The simulations were run over a duration of 100 s. Since the vehicles operate within a square-shaped highway and remain within the network without exiting, the duration of 100 s was sufficient for our purpose. Whether we ran the simulations for 100 s or 1000 s, it did not significantly impact the results, given the continuous movement of the vehicles within the network. The configuration parameters that are used for the experiment are listed in Table 4.1.

The performance of the proposed Q-DRCC approach is evaluated by comparing it with constant data rates and other existing techniques using the dynamic traffic model mentioned above. The following approaches are considered for comparison with the proposed approach.

1. 6 Mbps: BSMs are transmitted at 6 Mbps. Optimal Data rate Selection proposed in [37]
2. 12 Mbps: BSMs are transmitted at 12 Mbps.
3. 18 Mbps: BSMs are transmitted at 18 Mbps.
4. 24 Mbps: BSMs are transmitted at 24 Mbps.
5. 27 Mbps: BSMs are transmitted at 27 Mbps.
6. TDRC: Transmit Data rate based Decentralized Congestion Control [11].

7. DR-DCC: Data Rate Based Decentralized Congestion Control [12].
8. DRCA: Adaptive Data rate Control Algorithm [13].

Table 4.1: Configuration Parameters

Parameter Name	Parameter Value
Fixed Transmission Power	20 mW
BSM Size	4096 Bits / 512 Bytes
Transmission rate	5 BSM/s and 10 BSM/s
Data Rate	3, 6, 9, 12, 18, 24 and 27 Mbps
Highway Length	4 km
Number of lanes	4
Maximum Vehicle Density	50 vehicles (per 100 m)
Minimum Vehicle Density	1 vehicle (per 100 m)
Number of vehicles on the road	250 and 500
Simulation Time	100 seconds

The approaches used for the comparison included the usage of two different transmission rates: 5 Hz and 10 Hz. By considering both transmission rates, insights into the performance of the proposed approach across a range of transmission rates were obtained. By analyzing the proposed approach's performance with different transmission levels and assessing its effectiveness in various scenarios, we gained a better understanding of its capabilities and limitations. The results from both 5 Hz and 10 Hz transmissions helped us draw more informed conclusions about the overall performance of the approach. A fixed transmission power of 20 mW was used for all the approaches. The metrics used to evaluate the performance are the CBR, the number of received packets, the number of total lost packets, and the Beacon Error Rate (BER). In VANET, Channel Busy Ratio (CBR) is the ratio between the time the channel is sensed as busy and the total observation time. It provides an indication

of the channel utilization and congestion level within the network. The CBR can be calculated by measuring the time duration during which the channel is busy and dividing it by the total observation time. Mathematically, it can be expressed as:

$$CBR = (\text{Busy Time} / \text{Total Observation Time}) * 100\%$$

These metrics were used to assess and compare the performance of the approaches in terms of channel utilization, packet reception and packet loss. The upcoming sections will cover the evaluation of different methods and their comparison in terms of performance.

4.2 Comparison with Constant Data Rate Transmissions

In this section, the proposed approach will be compared with different constant data rates: 6 Mbps, 12 Mbps, 18 Mbps, 24 Mbps and 27 Mbps.

4.2.1 Comparison of CBR

The following figures show the average CBR for the constant data rate transmissions and the proposed approach. Figures 4.2, 4.3, 4.4, and 4.5 illustrate the average CBR for scenarios 250 vehicles transmitting 5 BSMs/s, 250 vehicles transmitting 10 BSMs/s, 500 vehicles transmitting 5 BSMs/s and 500 vehicles transmitting 10 BSMs/s, respectively. Figure 4.2 demonstrates that the average CBR of the proposed approach remains below the predefined threshold of 0.6, lower than 6 Mbps and higher than other constant data rate transmissions. Figure 4.3 demonstrates that the average CBR of the proposed approach remains below the threshold, lower than 6 Mbps and 12 Mbps, but higher than other data rate transmissions: 18, 24 and 27 Mbps. Similarly, Figure 4.4 shows that the average CBR of the proposed approach remains below the threshold, lower than 6 Mbps and 12 Mbps, but higher than other constant data rate transmissions: 18, 24 and 27 Mbps.

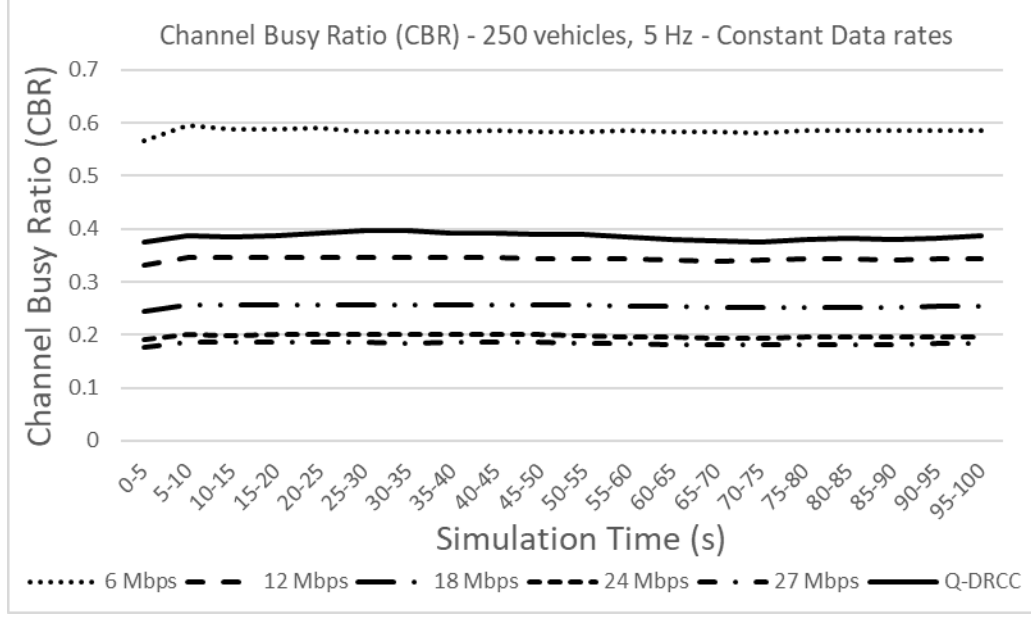


Figure 4.2: Comparing CBR with constant data rates (250 Vehicles, 5 Hz)

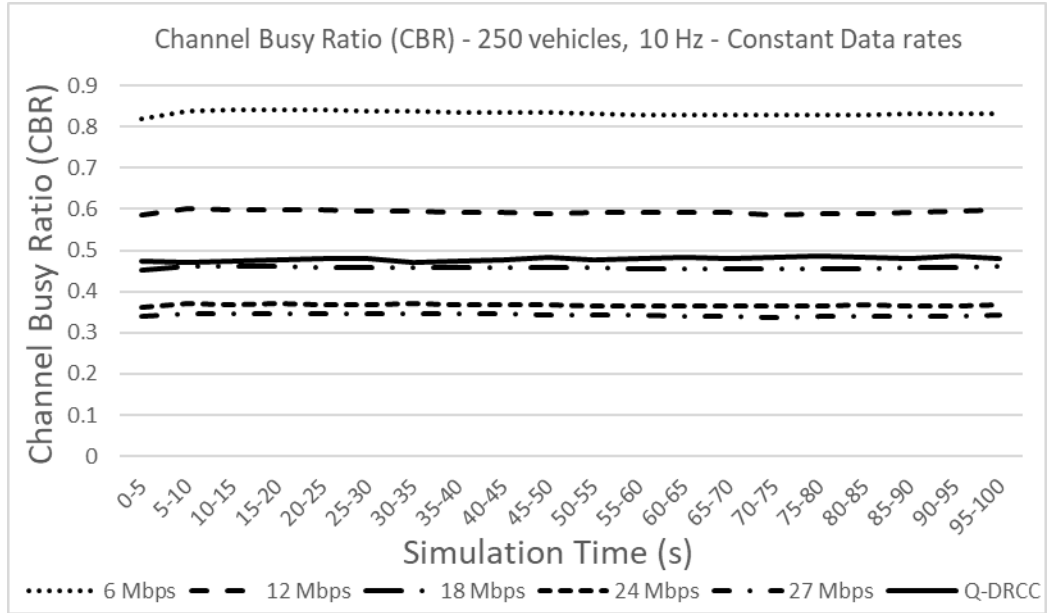


Figure 4.3: Comparing CBR with constant data rates (250 Vehicles, 10 Hz)

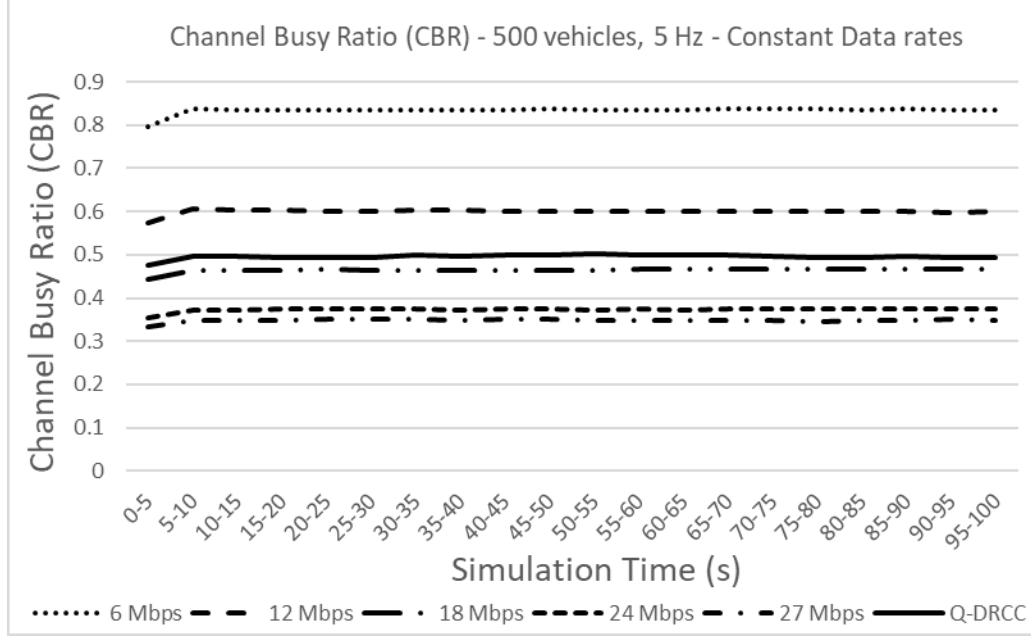


Figure 4.4: Comparing CBR with constant data rates (500 Vehicles, 5 Hz)

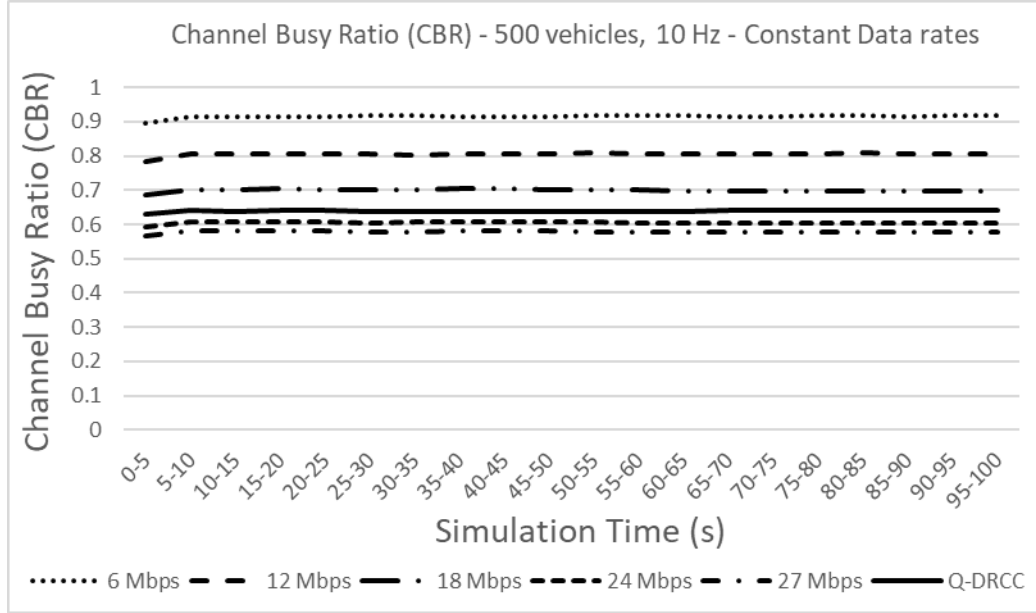


Figure 4.5: Comparing CBR with constant data rates (500 Vehicles, 10 Hz)

Figure 4.5 shows that the average CBR of the proposed approach remains close to the threshold, lower than 6 Mbps, 12 Mbps and 18 Mbps, but higher compared to 24 and 27 Mbps transmissions. Despite the fact that the CBR of the proposed approach was not consistently lower than all constant data rate transmissions, it was

consistently kept near the threshold. In cases where the CBR was not lower than the constant data rates, a balance was maintained between congestion and awareness by ensuring a higher number of received packets which is discussed in section 4.2.2.

4.2.2 Comparison of Received Packets

This section will discuss the number of BSMs received by the vehicles. The figures 4.6 and 4.7 show the number of received packets for scenarios 250 vehicles and 500 vehicles, respectively, transmitting 5 BSMs/s and 10 BSMs/s each.

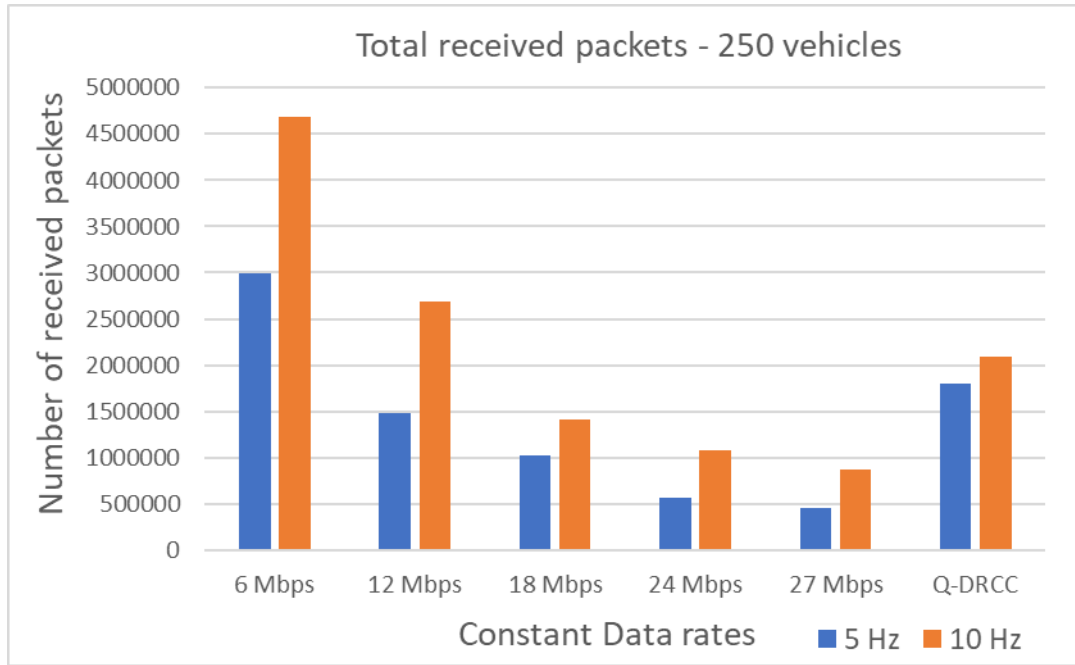


Figure 4.6: Comparing received packets with constant data rates (250 Vehicles)

Figure 4.6 demonstrates that the proposed Q-DRCC approach achieves a higher number of received packets compared to 12 Mbps, 18 Mbps, 24 Mbps and 27 Mbps data rate transmissions at a transmission rate of 5 Hz and a higher number of received packets compared to 18 Mbps, 24 Mbps and 27 Mbps at a transmission rate of 10 Hz. However, it is lower compared to 6 Mbps data rate transmission at 5 Hz and 6 Mbps and 12 Mbps data rate transmissions at 10 Hz. Similarly, Figure 4.7 shows that the proposed Q-DRCC approach results in a higher number of received packets compared to 18 Mbps, 24 Mbps and 27 Mbps, but lower than the number of received packets

achieved with 6 Mbps and 12 Mbps data rate transmissions when the transmission rate is 5 Hz. Whereas when the transmission rate is 10 Hz, it is higher compared to 24 Mbps and 27 Mbps but lower than 6, 12, and 18 Mbps transmissions. This decrease in received packets can be attributed to the reduction in the Channel Busy Ratio (CBR). Therefore, it is reasonable for our proposed approach to have fewer received packets when compared to certain data rate transmissions discussed above.

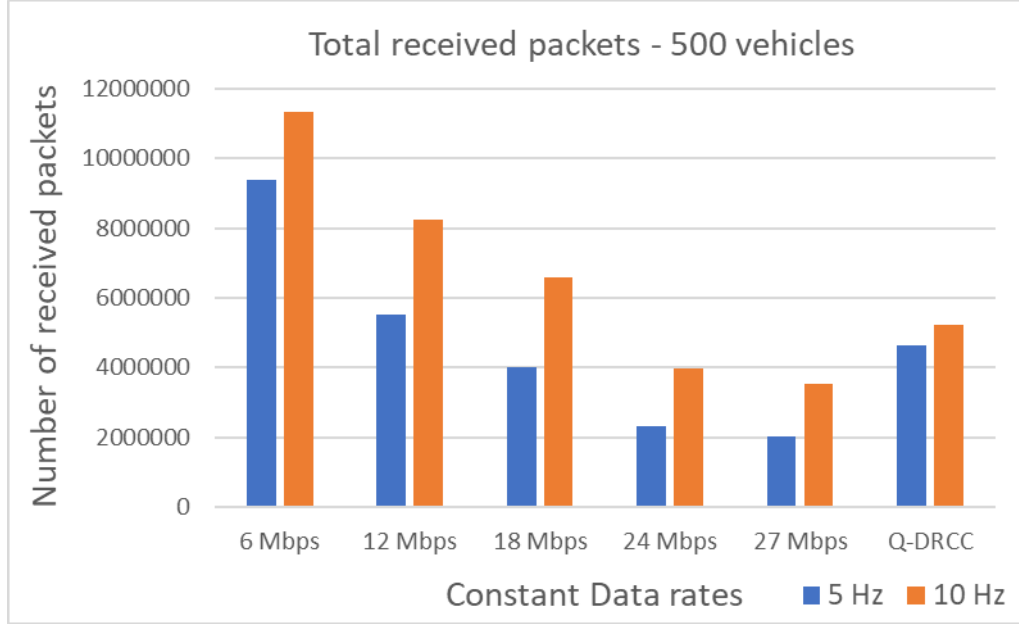


Figure 4.7: Comparing received packets with constant data rates (500 Vehicles)

4.3 Comparison with Existing Congestion Control Techniques

In this section, the proposed Q-DRCC approach will be compared with existing data rate-based congestion control techniques like TDRC, DR-DCC and DRCA. The metrics used to evaluate the performance are the Channel Busy Ratio (CBR), the number of received packets, the number of total lost packets, and the Beacon Error Rate (BER). These metrics are already discussed in detail in the previous section 4.1.

4.3.1 Comparison of CBR

This section shows the average CBR for the proposed approach and the existing algorithms. Figures 4.8, 4.9, 4.10, and 4.11 illustrate the average CBR for 250 vehicles transmitting 5 BSMs/s, 250 vehicles transmitting 10 BSMs/s, 500 vehicles transmitting 5 BSMs/s and 500 vehicles transmitting 10 BSMs/s, respectively. Figure 4.8, 4.9 and 4.10 demonstrates that the average CBR of the proposed Q-DRCC approach remains below the predefined threshold of 0.6 and is lower compared to the three existing algorithms discussed in this thesis: TDRC, DR-DCC and DRCA. However, in Figure 4.11, the average CBR of the proposed Q-DRCC approach remains near the threshold, lower than TDRC and DR-DCC, but slightly higher than DRCA.

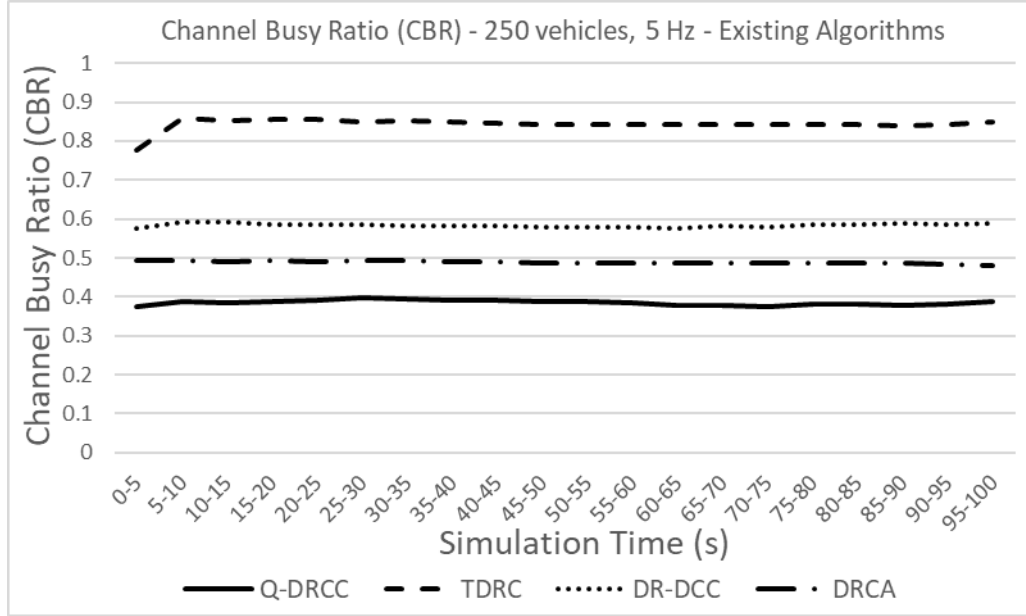


Figure 4.8: Comparing CBR with existing methods (250 Vehicles, 5 Hz)

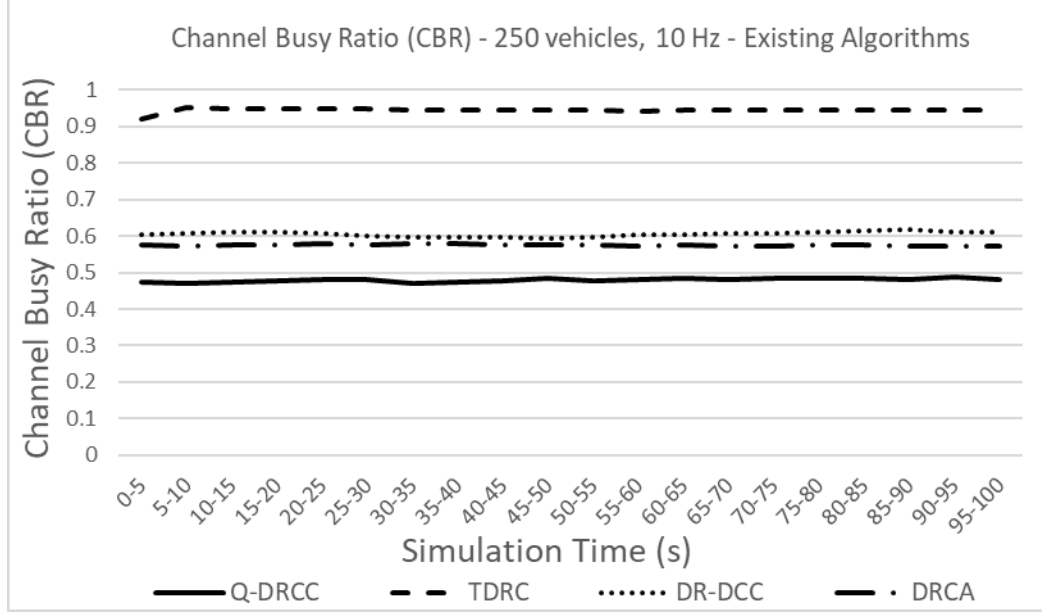


Figure 4.9: Comparing CBR with existing methods (250 Vehicles, 10 Hz)

These findings indicate that the proposed approach excels in both lower and higher vehicle densities, 250 and 500 vehicles, respectively, at a transmission rate of 5 Hz and in lower vehicle density (250 vehicles) at a transmission rate of 10 Hz. However, at a higher vehicle density (500 vehicles) with a higher transmission rate of 10 Hz, the proposed approach was able to outperform TDRC and DR-DCC but the DRCA algorithm. The proposed Q-DRCC approach showed favourable results in three out of the four cases that were discussed above.

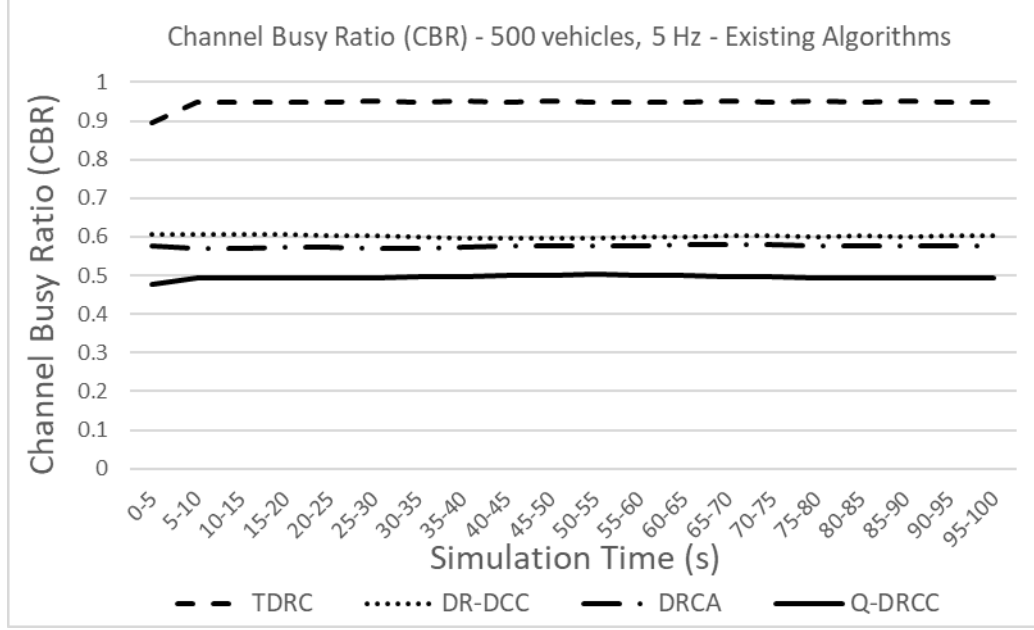


Figure 4.10: Comparing CBR with existing methods (500 Vehicles, 5 Hz)

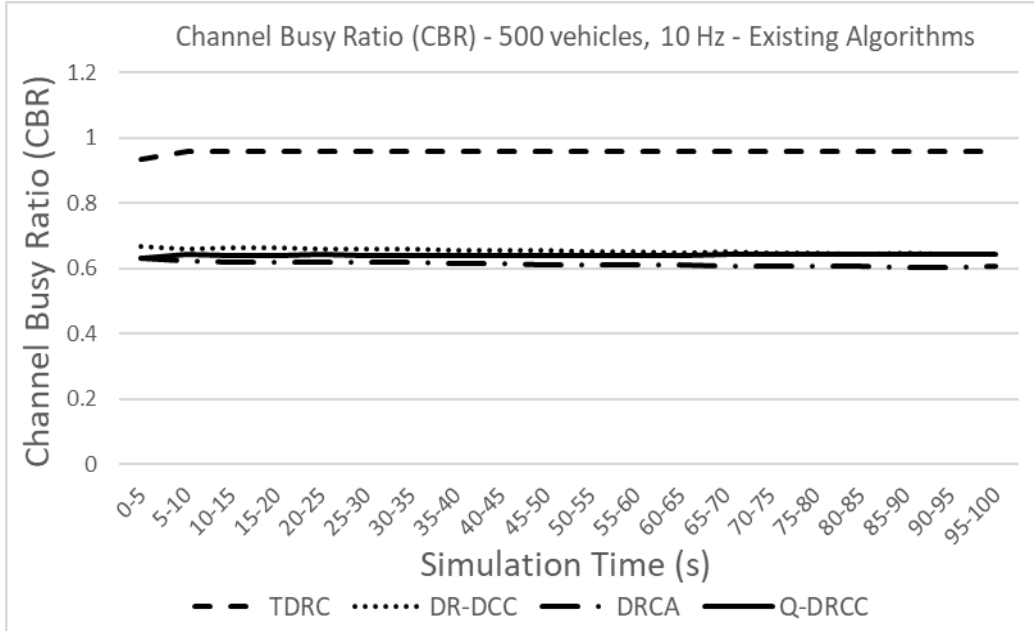


Figure 4.11: Comparing CBR with existing methods (500 Vehicles, 10 Hz)

4.3.2 Comparison of Received Rackets

This section will discuss the number of packets received by the vehicle communicating with its surroundings. The figures 4.12 and 4.13 show the number of received packets

for scenarios 250 vehicles and 500 vehicles, respectively, transmitting 5 BSMs/s and 10 BSMs/s each. In Figure 4.12, the proposed Q-DRCC approach achieved a lower number of received packets compared to existing algorithms: TDRC, DR-DCC, and DRCA for a transmission rate of both 5 Hz and 10 Hz.

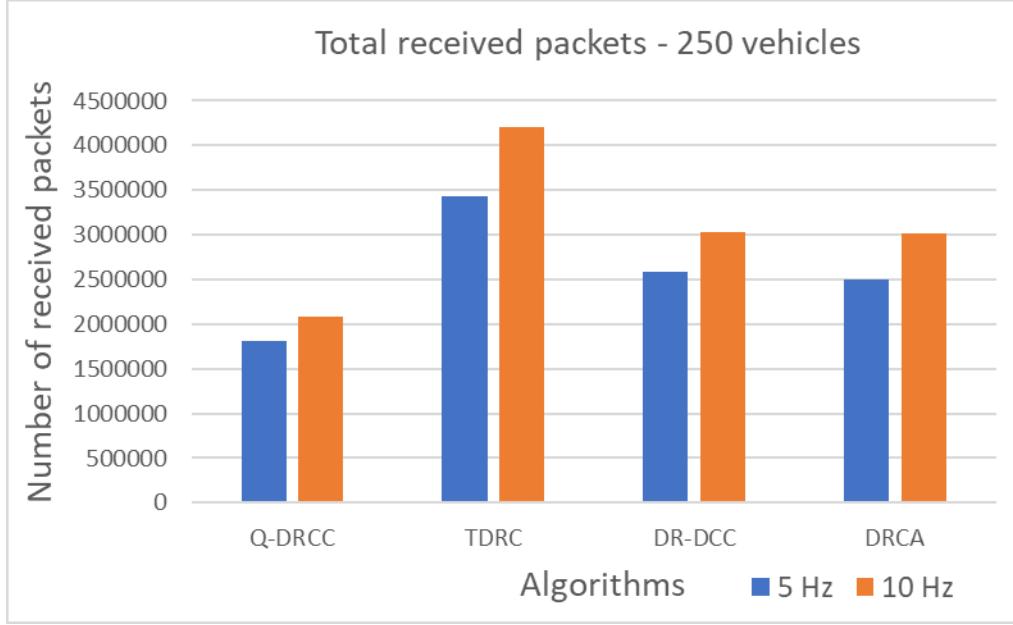


Figure 4.12: Comparing received packets with existing methods (250 Vehicles)

Similarly, Figure 4.13 shows that the proposed Q-DRCC approach results in a lower number of received packets compared to all existing algorithms for a transmission rate of 5 Hz and a higher number of received packets compared to DRCA, but lower compared to TDRC and DR-DCC for a transmission rate of 10 Hz. This decrease in received packets can be attributed to the decrease in the Channel Busy Ratio (CBR). Thus, it is reasonable for our proposed approach to have a lower count of received packets when compared to TDRC and DR-DCC, as the CBR is lower in our proposed algorithm compared to these two algorithms.

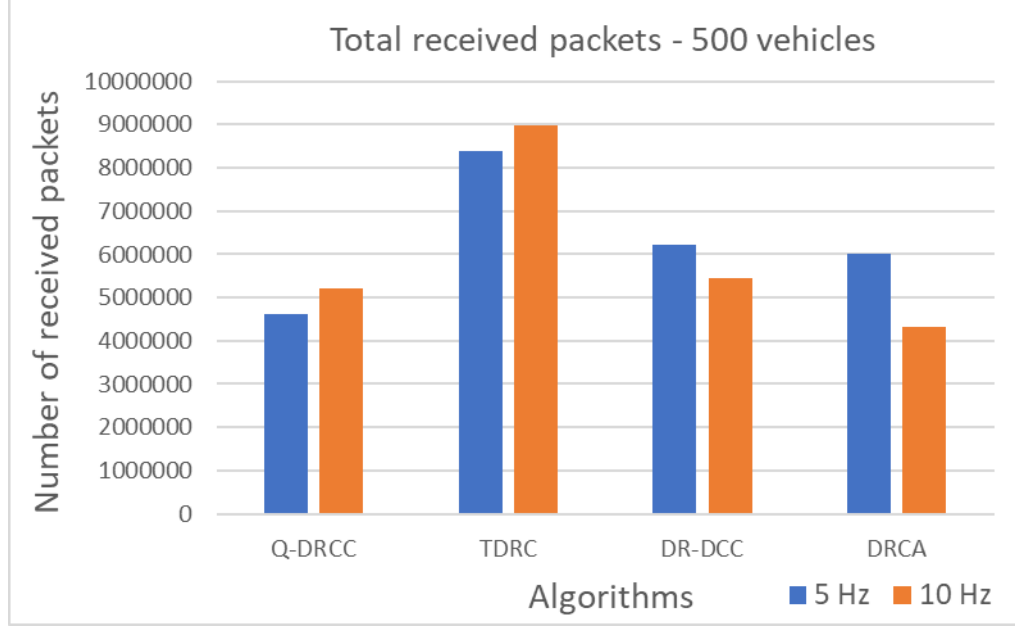


Figure 4.13: Comparing received packets with existing methods (500 Vehicles)

4.3.3 Comparison of Total Lost Packets

This section will discuss the number of total packets lost by the vehicles participating in V2V communication. In VANET, the lost packets refer to the packets that are not successfully received by the recipients. The packet loss can occur due to various reasons, including interference, collisions or network congestion. The lost packets can impact the reliability and effectiveness of communication among vehicles. The figures 4.14 and 4.15 show the number of total lost packets for scenarios 250 vehicles and 500 vehicles, respectively. Figure 4.14 shows that the total number of lost packets for the proposed Q-DRCC approach is less compared to the existing techniques for both 5 Hz and 10 Hz transmission rates when the number of vehicles involved in the communication are 250. Similarly, Figure 4.15 shows that when the number of vehicles involved in the communication is 500, the proposed approach has a lower number of lost packets compared to all existing approaches when the transmission rate is 5 Hz but slightly higher compared to DR-DCC and DRCA when the transmission rate is 10 Hz. However, the difference in the number of total packets lost between the proposed approach Q-DRCC and the existing approaches DR-DCC and DRCA is very less.

Therefore, we can say that the proposed approach shows very good results in terms of total packet loss.

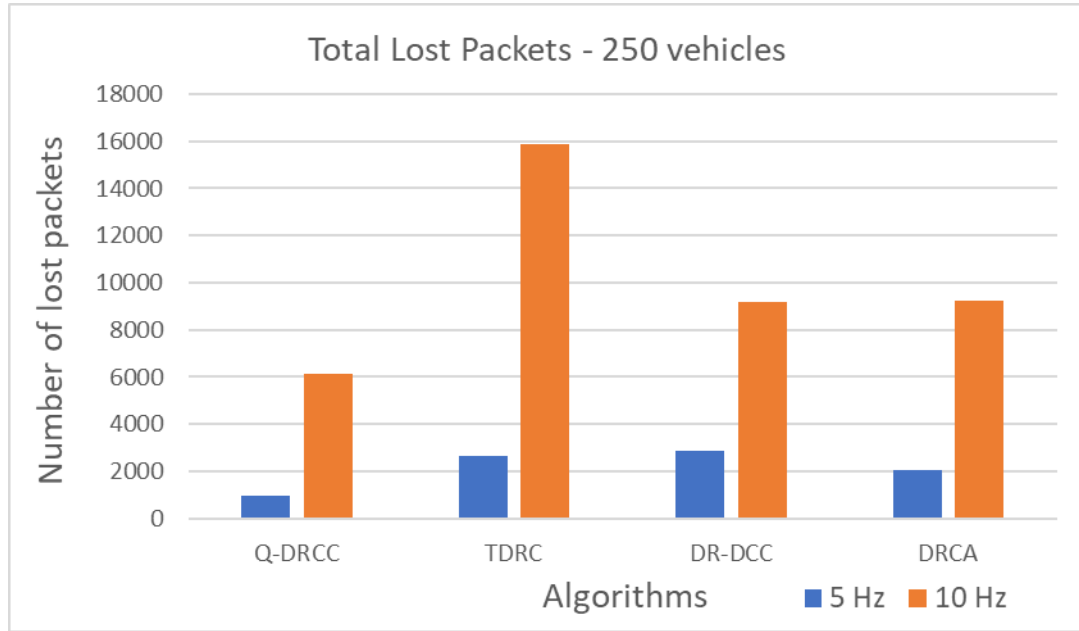


Figure 4.14: Comparing total lost packets with existing methods (250 Vehicles)

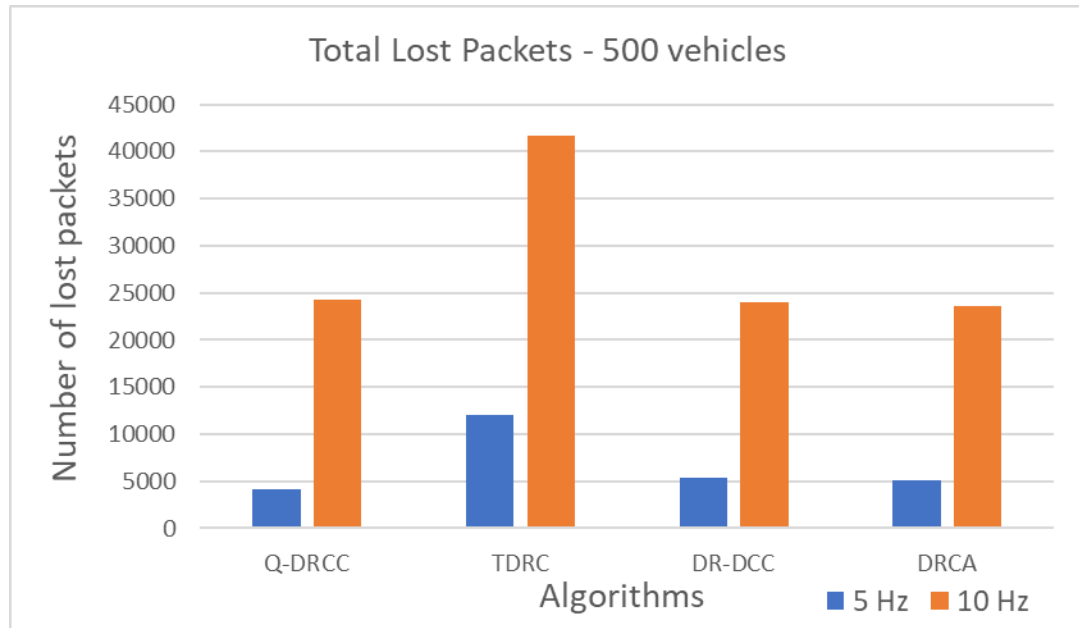


Figure 4.15: Comparing total lost packets with existing methods (500 Vehicles)

4.3.4 Comparison of Beacon Error Rate (BER)

This section will discuss the Beacon Error Rate (BER). In VANET, the Beacon Error Rate (BER) refers to the measurement of the percentage of sent packets that are lost during transmission. It represents the ratio of the number of beacon packets that fail to reach their intended recipients to the total number of beacon packets sent. It can be mathematically represented as:

$$BER = \frac{\text{lost packet}}{\text{sent packets}}$$

The figures 4.16 and 4.17 show the Beacon Error Rate (BER) for scenarios 250 vehicles and 500 vehicles, respectively. Figure 4.16 shows that the proposed Q-DRCC approach has lower BER compared to existing techniques for both 5 Hz and 10 Hz transmission rates when the number of vehicles involved in the communication is 250. Similarly, Figure 4.17 shows that when the number of vehicles involved in the communication is 500, the proposed approach has a lower BER compared to all existing approaches when the transmission rate is 5 Hz but slightly higher compared to DR-DCC and DRCA when the transmission rate is 10 Hz. However, the difference in the BER between the proposed approach Q-DRCC and the existing approaches DR-DCC and DRCA is very less. Therefore, we can say that the proposed approach shows very good results in terms of BER.

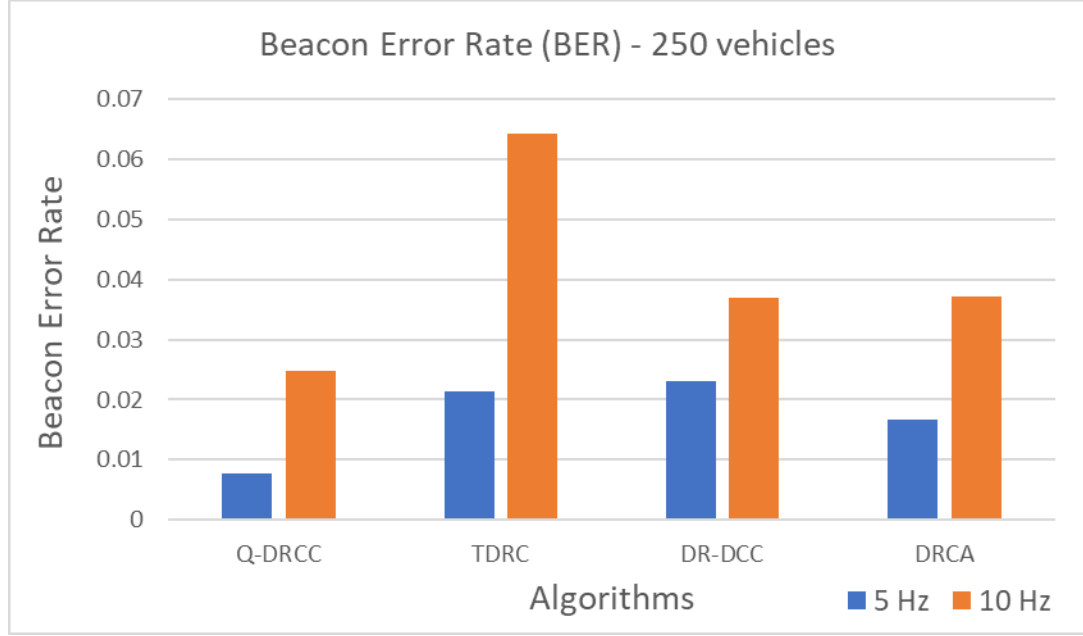


Figure 4.16: Comparing BER with existing methods (250 Vehicles)

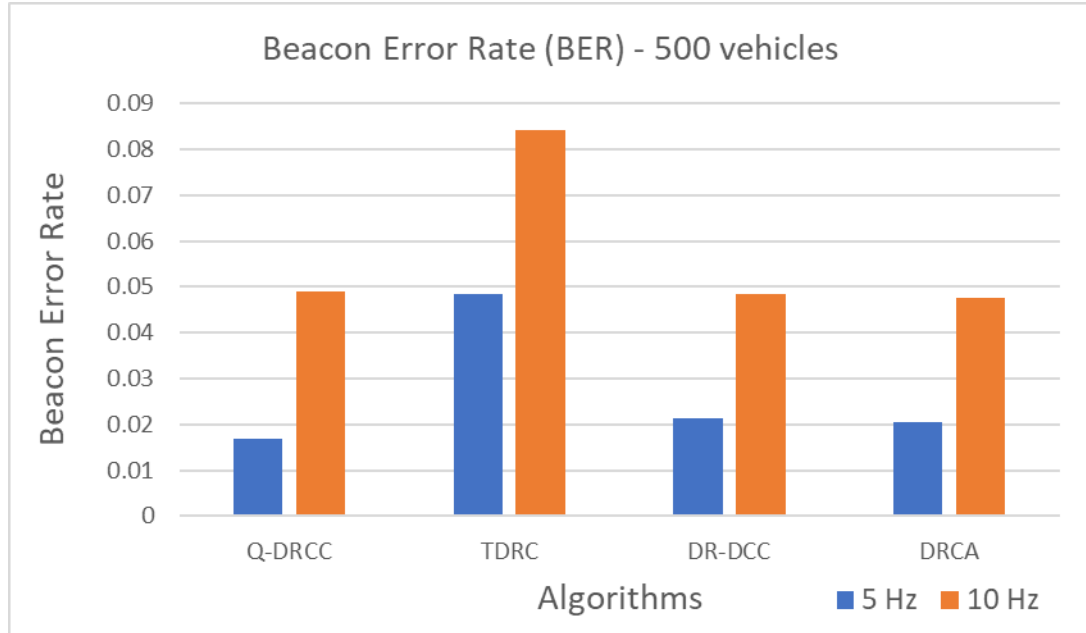


Figure 4.17: Comparing BER with existing methods (500 Vehicles)

When considering the case of 500 vehicles with a transmission rate of 5 Hz, the CBR measures around 0.5, indicating that the channel load is effectively managed and maintained below the desired threshold of 0.6. However, when the transmission rate is 10 Hz, the CBR exceeds 0.6 but still is maintained near the threshold of 0.6. This

can be attributed to the increased frequency of message transmissions. The higher message delivery rate leads to a higher CBR value. Comparing the two transmission rates, it becomes evident that the CBR is influenced not only by the network density and the number of participating vehicles but also by the transmission rate. The higher transmission rate of 10 Hz increases the channel utilization and message delivery, resulting in a CBR that exceeds the desired threshold. In contrast, when analyzing the scenario with 250 vehicles, as shown in Figures 4.2, 4.3, 4.8, and 4.9, the CBR is significantly lower than the desired level of keeping the channel load below and close to the threshold of 0.6, measuring around 0.3 and 0.4 for 5Hz and 10 Hz transmission rates, respectively. This is primarily due to the low network density, which refers to the fewer number of vehicles actively engaged in V2V communication within the channel. With fewer vehicles present, there is a decrease in the number of messages being transmitted, resulting in lower channel utilization and subsequently leading to a lower CBR value.

To summarize, Q-DRCC successfully controls the channel load to remain below the predefined threshold of 0.6, as defined in the reward function 1, in three of the four cases discussed, outperforming the existing data rate-based congestion control approaches in terms of packet loss, and error rate, leading to enhanced network performance, as observed through metrics such as total lost packets and BER.

CHAPTER 5

Conclusion and Future Work

5.1 Conclusion

In VANET, congestion control for V2V communication represents a critical challenge. To address this issue, reinforcement learning presents a novel approach for training vehicles to optimize their transmission parameters when exchanging messages. This thesis introduces a novel congestion control technique in VANET called Q-Learning-based Data rate congestion control (Q-DRCC), which uses RL principles. This thesis contains the design of Markov Decision Process (MDP) elements and the implementation of the Q-learning algorithm to generate a Q-table. Subsequently, extensive simulations were conducted using a dynamic traffic model to evaluate the performance of Q-DRCC. The simulations and evaluations demonstrated the efficiency of Q-DRCC in effectively managing the channel load, ensuring it remains below the predefined threshold and thereby minimizing packet loss. These positive outcomes were observed through the analysis of average CBR, number of received packets, number of total lost packets, and BER, as discussed in Chapter 4. The results obtained from the simulations strongly indicate that Q-DRCC represents a promising solution for enhancing the efficiency of channel load management in wireless networks, specifically within the context of VANET.

5.2 Future Work

In future work, there is potential to adjust multiple parameters concurrently, including transmission power, and transmission rate, in addition to data rate, to improve the efficiency of the Q-table. Security and privacy can also be considered while implementing the algorithm to protect against attacks such as jamming, spoofing or eavesdropping while preserving user privacy. In addition, the reward function can also be improved to enhance the learning process, leading to more efficient and effective congestion control in VANETs. By pursuing these research directions, the proposed congestion control technique can be further improved, and optimized, resulting in efficient and reliable V2V communication in VANET.

REFERENCES

- [1] World Health Organization (WHO). *Road traffic injuries*. 2022. URL: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>.
- [2] Steve Sutcliffe Johann Andersen. “Intelligent Transport Systems (ITS) - An Overview.” In: *IFAC Proceedings Volumes* 33.18 (2000), pp. 99–106.
- [3] Jarašūnienė A. “Research into intelligent transport systems (ITS) technologies and efficiency.” In: *TRANSPORT* 22.2 (2007), pp. 61–67.
- [4] Wikipedia contributors. *Intelligent transportation system* — *Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Intelligent_transportation_system&oldid=1147171738. [Online; accessed 8-June-2023]. 2023.
- [5] Abir Mchergui, Tarek Moulahi, and Sherali Zeadally. “Survey on Artificial Intelligence (AI) techniques for Vehicular Ad-hoc Networks (VANETs).” In: *Vehicular Communications* 34 (2022), p. 100403.
- [6] Muhammad Sameer Sheikh, Jun Liang, and Manzoor Ahmed Khan. “A Comprehensive Survey on VANET Security Services in Traffic Management System.” In: *Wireless Communications and Mobile Computing* 2019 (2019), pp. 1–23.
- [7] John B Kenney. “Dedicated short-range communications (DSRC) standards in the United States.” In: *Proceedings of the IEEE* 99.7 (2011), pp. 1162–1182.
- [8] K. Kunert A. Böhm M. Jonsson and A. Vinel. “Context-Aware Retransmission Scheme for Increased Reliability in Platooning Applications.” In: *Communication Technologies for Vehicles, Springer* (2014), pp. 30–42.

- [9] John Eggerton. *FCC to split up 5.9 ghz*. 2019. URL: <https://www.nexttv.com/news/fcc%02to-split-up-5-9-ghz>.
- [10] Xiaofeng Liu and Arunita Jaekel. “Congestion control in V2V safety communication: Problem, analysis, approaches.” In: *Electronics* 8.5 (2019), p. 540.
- [11] Ashish Patil et al. “Transmit data rate control based decentralized congestion control mechanism for VANETs.” In: *IEEE 2019 International Conference on Data Science and Communication (IconDSC)* (2019), pp. 1–5.
- [12] C. B. Math et al. “Data Rate based Congestion Control in V2V communication for traffic safety applications.” In: *IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)* (2015), pp. 1–6.
- [13] S. Jayachandran and A. Jaekel. “Adaptive Data Rate Based Congestion Control in Vehicular Ad Hoc Networks (VANET).” In: *Proceedings of the Ad Hoc Networks and Tools for IT: 13th EAI International Conference, ADHOCNETS 2021, Virtual Event* (2022), pp. 144–157.
- [14] Hakim Badis and Abderrezak Rachedi. *Vehicular Ad Hoc Network*. 2015. URL: <https://www.sciencedirect.com/topics/computer-science/vehicular-ad-hoc-network>.
- [15] Kaur bal. *Difference between MANET and VANET*. 2023. URL: <https://www.geeksforgeeks.org/difference-between-manet-and-vanet/>.
- [16] X. Wang, S. Mao, and M.X. Gong. “An overview of 3GPP cellular vehicle-to-everything standards.” In: *GetMobile: Mobile Computing and Communications* 21.3 (2017), pp. 19–25.
- [17] Prateek Bajaj. *Reinforcement Learning*. 2023. URL: <https://www.geeksforgeeks.org/what-is-reinforcement-learning/>.
- [18] Ahmad Hammoudeh. “A concise introduction to reinforcement learning.” In: *Princess Suamaya University for Technology: Amman, Jordan* (2018).

- [19] R. Nian, J. Liu, and B. Huang. “A review on reinforcement learning: Introduction and applications in industrial process control.” In: *Computers Chemical Engineering* 139 (2020), p. 106886.
- [20] Robert Moni. *Reinforcement Learning Algorithms: An intuitive overview*. 2019. URL: <https://smartlabai.medium.com/reinforcement-learning-algorithms-an-intuitive-overview-904e2dff5bbc>.
- [21] Xiaofeng Liu, Ben St Amour, and Arunita Jaekel. “A Reinforcement Learning-Based Congestion Control Approach for V2V Communication in VANET.” In: *Applied Sciences* 13.6 (2023), p. 3640.
- [22] Mare Torrent-Moreno, Paolo Santi, and Hannes Hartenstein. “Distributed fair transmit power adjustment for vehicular ad hoc networks.” In: *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks* 2 (2006), pp. 479–488.
- [23] Yuanfu Mo et al. “A beacon transmission power control algorithm based on wireless channel load forecasting in VANETs.” In: *PloS one* 10.11 (2015), e0142775.
- [24] Bernhard Kloiber, Jérôme Härri, and Thomas Strang. “Dice the TX Power - Improving Awareness Quality in VANETs by Random Transmit Power Selection.” In: *IEEE vehicular networking conference (VNC)* (2012), pp. 56–63.
- [25] Akinlade Oluwaseyi et al. “Traffic density based distributed congestion control strategy for vehicular communication.” In: *IEEE 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)* (2019), pp. 195–197.
- [26] Caitlin Facchina and Arunita Jaekel. “Speed Based Distributed Congestion Control Scheme for Vehicular Networks.” In: *2020 IEEE Symposium on Computers and Communications (ISCC)* (2020), pp. 1–4.
- [27] Wang M et al. “Research on adaptive beacon message transmission power in VANETs.” In: *Journal of Ambient Intelligence and Humanized Computing* (2022), pp. 1–3.

- [28] Tessa Tielert et al. “Design methodology and evaluation of rate adaptation based congestion control for vehicle safety communications.” In: *IEEE vehicular networking conference (VNC)* (2011), pp. 116–123.
- [29] Gaurav Bansal, John B. Kenney, and Charles E. Rohrs. “LIMERIC: A linear adaptive message rate algorithm for DSRC congestion control.” In: *IEEE Transactions on Vehicular Technology* 62.9 (2013), pp. 4182–4197.
- [30] Gaurav Bansal et al. “EMBARC: Error model based adaptive rate control for vehicle-to-vehicle communications.” In: *Proceeding of the tenth ACM international workshop on Vehicular inter-networking, systems, and applications* (2013), pp. 41–50.
- [31] Kazumine Ogura, Jiro Katto, and Mineo Takai. “BRAEVE: Stable and adaptive BSM rate control over IEEE802. 11p vehicular networks.” In: *2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)* (2013), pp. 745–748.
- [32] Gaurav Bansal and John B. Kenney. “Achieving weighted-fairness in message rate-based congestion control for DSRC systems.” In: *IEEE 5th International Symposium on Wireless Vehicular Communications (WiVeC)* (2013), pp. 1–5.
- [33] Nader Chaabouni, Abdelhakim Hafid, and Pratap Kumar Sahu. “A collision-based beacon rate adaptation scheme (CBA) for VANETs.” In: *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)* (2013), pp. 1–6.
- [34] G. Kim et al. “A request based adaptive beacon rate control scheme for vehicular ad-hoc networks.” In: *IEEE Eighth International Conference on Ubiquitous and Future Networks (ICUFN)* (2016), pp. 67–69.
- [35] Mahendrakumar Subramaniam et al. “A Traffic Density-Based Congestion Control Method for VANETs.” In: *Wireless Communications and Mobile Computing* (2022).

- [36] J. Aznar-Poveda et al. “Approximate reinforcement learning to control beaconing congestion in distributed networks.” In: *Scientific Reports* 12.1 (2022), p. 142.
- [37] Daniel Jiang, Qi Chen, and Luca Delgross. “Optimal data rate selection for vehicle safety communications.” In: *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking* (2008), pp. 30–38.
- [38] Seungnam Yang, Hyogon Kim, and Seungho Kuk. “Less is more: need to simplify ETSI distributed congestion control algorithm.” In: *Electronics Letters* 50.4 (2014), pp. 279–281.
- [39] C.B. Math et al. “Fair decentralized data-rate congestion control for V2V communications.” In: *IEEE 24th International Conference on Telecommunications (ICT)* (2017), pp. 1–7.
- [40] J. Aznar-Poveda et al. “MDPRP: A Q-learning approach for the joint control of beaconing rate and transmission power in VANETs.” In: *IEEE Access* 9 (2021), pp. 10166–10178.
- [41] Roberto Baldessari et al. “Joining forces for vanets: A combined transmit power and rate control algorithm.” In: *6th international workshop on intelligent transportation (WIT)* (2010).
- [42] Jordan T. Willis, Arunita Jaekel, and Ikjot Saini. “Decentralized congestion control algorithm for vehicular networks using oscillating transmission power.” In: *IEEE Wireless Telecommunications Symposium (WTS)* (2017), pp. 1–5.
- [43] Byeong-Moon Cho, Min-Seong Jang, and Kyung-Joon Park. “Channel-aware congestion control in vehicular cyber-physical systems.” In: *IEEE Access* 8 (2020), pp. 73193–73203.
- [44] Bernhard Kloiber, Jérôme Härri, and Thomas Strang. “Dice the tx power—improving awareness quality in vanets by random transmit power selection.” In: *IEEE vehicular networking conference (VNC)* (2012), pp. 56–63.

- [45] M. Deeksha et al. “Multistate active combined power and message/data rate adaptive decentralized congestion control mechanisms for vehicular ad hoc networks.” In: *Journal of Physics: Conference Series* 2161.1 (2022), p. 012018.
- [46] R.S. Sutton and A.G. Barto. *Reinforcement Learning: An introduction*. 2018.
- [47] *SUMO*. URL: <https://www.eclipse.org/sumo/>.
- [48] *OMNET*. URL: <https://omnetpp.org/>.
- [49] *Veins*. URL: <https://veins.car2x.org/>.

VITA AUCTORIS

NAME: Gnana Shilpa Nuthalapati

PLACE OF BIRTH: Chennai, Tamil Nadu, India

EDUCATION: Bachelor of Engineering in Computer Science, Sree Sastha Institute Of Engineering and Technology, Chennai, Tamil Nadu, India. (2017)

M.Sc. Computer Science, University of Windsor, Windsor, Ontario, Canada. (2023)