

THE NEXT GENERATION INTELLIGENT
TRANSPORTATION SYSTEM: CONNECTED, SAFE AND
GREEN

RIBAL ATALLAH

A THESIS
IN
THE CONCORDIA INSTITUTE
FOR
INFORMATION SYSTEMS ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
CONCORDIA UNIVERSITY
MONTRÉAL, QUÉBEC, CANADA

APRIL 2017

© RIBAL ATALLAH, 2017

CONCORDIA UNIVERSITY
Engineering and Computer Science

This is to certify that the thesis prepared

By: **Mr. Ribal Atallah**
Entitled: **The Next Generation Intelligent Transportation System:
Connected, Safe and Green**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Information and Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____	Chair
Dr. Long Bao Le _____	External Examiner
Dr. Mustafa Mehmet Ali _____	External to Program
Dr. Lingyu Wang _____	Examiner
Dr. Roch Glitho _____	Examiner
Dr. Chadi Assi _____	Supervisor
Dr. Maurice Khabbaz _____	Co-supervisor

Approved _____
Chair of Department or Graduate Program Director

Abstract

The Next Generation Intelligent Transportation System: Connected,
Safe and Green

Ribal Atallah, Ph.D.

Concordia University, 2017

Modern Intelligent Transportation Systems (ITSs) employ communication technologies in order to ameliorate the passenger's commuting experience. Vehicular Networking lies at the core of inaugurating an efficient transportation system and aims at transforming vehicles into smart mobile entities that are able to sense their surroundings, collect information about the environment and communicate with each other as well as with Roadside Units (RSUs) deployed alongside roadways. As such, the novel communication paradigm of vehicular networking gave birth to an ITS that embraces a wide variety of applications including but not limited to: traffic management, passenger and road safety, environment monitoring and road surveillance, hot-spot guidance, Drive Thru Internet access, remote region connectivity, and so forth. Furthermore, with the rapid development of computation and communication technologies, the Internet of Vehicles (IoV) promises huge commercial interest and research value, thereby attracting a significant industrial and academic attention.

This thesis studies and analyses fundamentally challenging problems in the context of vehicular environments and proposes new techniques targeting the improvement of the performance of ITSs envisioned to play a remarkable role in the IoV era. Unlike existing wireless mobile networks, vehicular networks possess unique characteristics, including high node mobility and a rapidly-changing topology, which should be carefully accounted for. Four major problems from the pool of existing vehicular networking persisting challenges will be addressed in this thesis, namely: *a)* establishing a connectivity path in a highly dynamic Vehicular Ad Hoc Network, *b)* examining the performance of Vehicle-to-Infrastructure communication Medium Access Control schemes, *c)* addressing the scheduling problem of a vehicular networking scenario

encompassing an energy-limited RSU by exploiting machine learning techniques, particularly reinforcement learning, to train an agent to make appropriate decisions and develop a scheduling policy that prolongs the network's operational status and allows for acceptable Quality-of-Service levels and *d)* overcoming the limitations of reinforcement learning techniques in high-dimensional input scenarios by exploiting recent advances in deep learning in an effort to satisfy the driver's well-being as well as his demand for continuous connectivity in a green, balanced, connected and efficient vehicular network. These problems will be extensively studied throughout this thesis, followed by discussions that highlight open research directions worth further investigations.

Acknowledgments

I would like to express my deepest gratitude and acknowledgement to all the great people who supported me along the way towards completing my Ph.D. studies.

First and foremost, I am sincerely grateful to my Ph.D. supervisor Prof. Chadi Assi, for his consistent motivation, exemplary supervision, great guidance, and unconditional support. Prof. Assi, you are a humble, dedicated, inspiring, and caring advisor. Thank you.

Secondly, I would also like to extend my deepest appreciation to my Ph.D. co-supervisor Prof. Maurice Khabbaz for his endless encouragement and constant advisement. Every second along this journey, I truly enjoyed our beneficial discussions, research collaboration, and continuous exchange of knowledge. Thank you.

I gratefully acknowledge my Ph.D. committee members, Prof. Mustafa Mehmet Ali, Prof. Lingyu Wang and Prof. Roch Glitho, for their constructive comments and insightful suggestions, which helped me improve the quality of this thesis. Also, it is a pleasure to truly acknowledge Prof. for accepting to serve as a delegate in my Ph.D. thesis examining committee.

Furthermore, I am greatly thankful to all of my colleagues in the research lab at Concordia University for providing me with the warm and friendly atmosphere.

Last but not least, my most tender thanks go to those people who are the closest to my heart: my father Fares Atallah, my mother Vera, my fiancée Diane, my brother Firas and his family (my sister-in-law Nicole, my nephews Joey-Fares and Lucas, and my nieces Angela and Melissa) and my sister Zeina. You always believed in me and I am always stronger on your shoulders. You are the reason why I keep pushing and aiming higher. I can never put my feelings and sincere gratitude to you all into words. I love you all. Diane, you invaded my life and stole my heart and made me feel at home here, in Montreal. I love you loads.

Trust in the Lord with all your heart, and lean not on your own understanding, in all your ways acknowledge Him, and He shall direct your paths.

Contents

List of Figures	xi
List of Tables	xiii
List of Acronyms	xiv
1 Introduction	1
1.1 Overview and Motivation	1
1.2 Thesis Contributions	5
1.2.1 Connecting Vehicles Residing in Dark Areas	6
1.2.2 A Vehicle’s Perspective of MAC Schemes	7
1.2.3 Optimizing Downlink Traffic Scheduling - The Single RSU Case	8
1.2.4 Optimizing the Operation of a Connected Vehicular Network - The Multiple RSU Case	10
1.3 Thesis Organization	11
2 Preliminaries and Literature Review	13
2.1 The WAVE Protocol Suite	13
2.1.1 WAVE Protocol Stacks	13
2.1.2 The IEEE 802.11p MAC Sublayer:	17
2.1.3 The IEEE 1609.4 Standard:	18
2.1.4 Selective Literature Survey:	20
2.2 Scheduling-Based Access Methods	24
2.2.1 Overview:	24
2.2.2 Selective Literature Survey:	26
2.3 Markov Decision Processes	29

2.3.1	MDP Mathematical Definition	30
2.3.2	Optimality Equations	31
2.3.3	Solving Model-Free MDPs	34
3	Multi-hop Vehicular Connectivity Path: A Feasibility Study	42
3.1	Introduction	42
3.2	Related Work	44
3.2.1	Literature Review:	44
3.2.2	Novel Contributions:	47
3.3	Multi-Hop V2I Communication Scenario	50
3.4	Vehicular Traffic Model	51
3.5	Modelling and Analysis of Path Availability	55
3.5.1	Availability of Relay Nodes:	55
3.5.2	Path Availability:	56
3.5.3	Analysis of Number of Hops:	58
3.6	Delay Analysis:	62
3.6.1	Preliminaries:	62
3.6.2	End-to-end Delivery Delay Upper Bound:	64
3.7	Throughput Analysis	68
3.7.1	Collision and Channel Busy Probabilities:	69
3.7.2	Throughput Expression:	71
3.8	Numerical and Simulation Results	71
3.8.1	Simulation Setup:	71
3.8.2	Delay Simulation vs Theoretical Results:	72
3.8.3	Throughput Discussion:	74
3.9	Conclusion	76
4	A Vehicle’s Perspective of MAC Schemes	78
4.1	Introduction	78
4.2	Related Work and Contribution	80
4.2.1	Selective Literature Survey	80
4.2.2	Distinguishing Features of This Present Work	83
4.3	Vehicular Traffic Model	85
4.4	Description of The Proposed V2I MAC Schemes	87

4.4.1	Random Vehicle Selection Scheme	88
4.4.2	Least Residual Time Scheme	89
4.5	Modelling and Analysis of a Vehicle’s OBUB Under RVS and LRT . .	90
4.5.1	Model Definition	90
4.5.2	Basic Assumptions	90
4.5.3	RVS Model Resolution	91
4.5.4	LRT Model Resolution	95
4.5.5	OBUB’s Queueing Model Characterization	98
4.6	Simulations and Numerical Analysis	99
4.7	Concluding Remarks	104
5	Optimizing Downlink Traffic Scheduling - The Single RSU case	106
5.1	Introduction	106
5.1.1	Problem Statement and Motivation	109
5.2	Related Work	112
5.2.1	Scheduling-Based Access Methods:	112
5.2.2	Energy-Aware Vehicular Networks:	113
5.2.3	Novel Contributions:	114
5.3	Vehicular Traffic Model	117
5.4	PEARL Theoretical Formulation	119
5.4.1	Overview:	119
5.4.2	PEARL Operation:	121
5.4.3	Finite Horizon Markov Decision Process:	121
5.4.4	MDP Solution Approach:	127
5.5	Simulation Results	127
5.5.1	Simulation Setup	127
5.5.2	Simulation Results	131
5.6	Conclusion and Future Research Direction	136
6	Optimizing Downlink Traffic Scheduling - The Multi RSU case	137
6.1	Introduction	137
6.2	Problem Statement and Novel Contributions	140
6.3	Related Work	142
6.3.1	IoV Enabling Technologies:	142

6.3.2	V2I Scheduling-Based Access Methods:	143
6.3.3	Deep Reinforcement Learning:	144
6.4	V2I Communication Scenario	147
6.5	Vehicular Traffic Model	149
6.6	Markov Decision Process Model	150
6.6.1	Input From the Environment:	151
6.6.2	Immediate Single Step Costs	152
6.6.3	Proposed Solution Approach:	154
6.7	Simulation and Results Discussion	155
6.7.1	The Learning Phase:	155
6.7.2	Simulation Setup:	156
6.7.3	Simulation Results:	157
6.8	Conclusion and Future Research Direction	164
7	Discussion and Future Directions	165
7.1	Conclusions	165
7.2	Future Work	167
7.2.1	Secure Communications in a Vehicular Environment	168
7.2.2	UAV-Aided Multi-Hop Connectivity	168
7.2.3	UAV-Aided Temporary V2I Services	169
	Bibliography	169

List of Figures

1.1	The Internet of Vehicles	3
1.2	Exploiting V2V communications to connect vehicles in dark areas. . .	6
1.3	Energy-Limited Multi-RSU Vehicular Network	10
2.1	WAVE Protocol Stack.	14
2.2	The 75 MHz frequency spectrum allocated by FCC to support vehicular networking applications	16
2.3	IEEE 1609.4 channel coordination schemes.	19
3.1	Connecting Vehicles in Dark Areas.	44
3.2	Vehicles Forming Clusters Along d_{SD}	53
3.3	Probability of having a relay node on a per-hop basis.	54
3.4	Min and Max Number of V2V Communication Hops	56
3.5	Distance Travelled by a Single Hop	56
3.6	Verification and validation of $F_Y(y)$ and $F_{d_h}(\delta)$	57
3.7	Probability of path availability.	60
3.8	Events Changing the Vehicular Network's Topology	63
3.9	IEEE 802.11p Ranges and Collision Events.	68
3.10	Packet Delays.	72
3.11	Delivery Delay Versus d_{SD}	74
3.12	Collision Probability and Throughput for Different Vehicular Densities and Carrier Sensing Ranges	75
4.1	Vehicle-to-Infrastructure Communications in a Vehicular Network. . .	86
4.2	C.D.F of the packet service time for different vehicle flow rates. . . .	93
4.3	Instantaneous Position and Residual Times of Vehicles i and k.	95
4.4	Theoretical mean packet service time of RVS and LRT plotted versus their respective simulated counterparts.	98
4.5	System's Response Time.	100

4.6	Queue Lengths and Throughput Comparison	101
5.1	Solar Powered RSU in an Energy-Limited VANET	108
5.2	V2I Scheduling Algorithms Performance	111
5.3	PEARL's Operation	120
5.4	PEARL Performance Evaluation	128
5.5	QoS under Variable Request Size	130
5.6	Per-vehicle and Network Throughputs	134
6.1	Energy-Limited Multi-RSU Vehicular Network	139
6.2	Performance Evaluation and Comparisons	158
6.3	IoT-GWs' Lifetime	159
6.4	Performance Evaluation for $\rho = 6$ (veh/km)	163
7.1	Enhanced Connectivity Using UAVs	168
7.2	UAVs as Mobile RSUs	169

List of Tables

5.2	PEARL Simulation Input Parameters	130
6.2	Simulation Input Parameters	155

List of Acronyms

IoT	Internet of Things
5G	Fifth Generation
IoV	Internet of Vehicles
V2V	Vehicle-to-vehicle
V2I	Vehicle-to-infrastructure
V2S	Vehicle-to-sensor
ITS	Intelligent Transportation System
FCC	Federal Communications Commission
DSRC	Dedicated Short-Range Communication
MAC	Medium Access Control
WAVE	Wireless Access in Vehicular Environments
MDP	Markov Decision Process
RL	Reinforcement Learning
RSU	RoadSide Unit
VANET	Vehicular Adhoc NETWORK
PEARL	Protocol for Energy-efficient Adaptive scheduling using Reinforcement Learning
QoS	Quality-of-Service
DQN	Deep Q-Network
IPv6	Internet Protocol version 6
WSMP	WAVE Short Message Protocol
OSI	Open Systems Interconnections
PHY	PHYsical Layer
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
WSM	WAVE Short Message
WME	WAVE Management Entity

WLAN	Wireless Local Area Network
FCC	Federal Communications Commission
PLME	Physical Layer Management Entity
MLME	MAC Layer Management Entity
CCH	Control CHannel
SCH	Service CHannel
LLC	Logical Link Control
EDCA	Enhanced Distributed Channel Access
AC	Access Class
EDCAF	Enhanced Distributed Channel Access Function
AIFS	Arbitration Inter-Frame Space
DCF	Distributed Coordination Function
WBSS	WAVE Basic Service Set
AP	Access Point
WSA	WAVE Service Advertisement
CCI	Control CHannels interval
SCI	Service CHannel interval
GI	Guard Interval
FER	Frame Error Rate
AIFSN	PAGE 22
DORA	Dynamic Optimal Random Access
SIG	Stationary Internet Gateway
DTI	Drive-Thru Internet
SR	Service Request
VI	Value Iteration
PI	Policy Iteration
DP	Dynamic Programming
SGD	Stochastic Gradient Descent
SUMO	Simulation for Urban MObility
OBU	OnBoard Unit
CLV	Cluster Leader Vehicle
CTV	Cluster Tail Vehicle
p.d.f	Probability Distribution Function

p.m.f	Probability Mass Function
c.d.f	Cumulative Distribution Function
MSE	Mean Squared Error
ACK	Acknowledgement
RVS	Random Vehicle Selection
LRT	Least Residual residence Time
OBUB	OnBoard Unit Buffer
SDS	Selective Downlink Scheduling
SFTM	Simple Free-flow Traffic Model
AR	Access Request
HoL	Head-of-Line
VOIP	Voice Over IP
RMS	Rate Monotonic Scheduling
GPC	Greedy Power Conservation

Chapter 1

Introduction

1.1 Overview and Motivation

To cruise towards the 5G technology, intelligence, communication capabilities and processing power will need to be diffused across networks and mobile devices, empowering even the smallest of connected devices to do heavy computational tasks and run rich content and services. Soon enough, the Internet of Things (IoT) paradigm, which is a key enabling technology for the next generation 5G network, will become an absolute reality in modern wireless communications. At this point, an enormous number of “things” is being (and will continue to be) connected to the Internet at an unprecedented rate realizing the concept of IoT. Unquestionably, the IoT will remarkably impact people’s everyday life. The Internet of Vehicles (IoV) emerges as a result of the fusion between the mobile Internet and the IoT. IoV technology, illustrated in Figure 1.1, refers to highly dynamic mobile communication systems that enable communication among vehicles and between vehicles and other wireless units (possibly mobile or stationary) using either V2V (vehicle-to-vehicle), V2I (vehicle-to-infrastructure) or V2S (vehicle-to-sensor) or a combination of these several types of

interactions. IoV enables information exchange between vehicles and their surroundings (*e.g.*, other vehicles, roadside units, portable devices carried by proximity users, etc.). Moreover, IoV features the processing, computing, sharing and secure release of information onto intelligent platforms, allowing these platforms to effectively guide and supervise the vehicles' behaviour, and provision them with a variety of multimedia and mobile services. IoV leverages road objects (*e.g.* traffic lights, cameras, speed sensors, etc.) with the ability to sense, process and exchange information related to the safety and comfort of passengers. When conventional vehicles are supplemented with the latest IoV technology, the feasibility of vehicle dynamics monitoring, intelligent navigation, fleet management, and value-added services become endless. For this purpose, the transportation research community is working collaboratively to build an end-to-end full-fledge Intelligent Transportation System (ITS) that enhances the user experience, reduces operational costs and promotes a safe driving environment. A revolutionary transportation experience in the IoV era realizes several benefits, including, but not limited to: *a)* greater efficiency achieved through the reduction of fuel consumption through fuel-saving assistance that accounts for the driving distance, road conditions and driving patterns, *b)* increased safety using remote vehicle diagnostics that promote the responsiveness of service centers to driver drowsiness, vehicle theft, accidents as well as maintenance requests, *c)* higher reliability resulting from the reduction of vehicle downtime as well as expensive unplanned repairs following the use of vehicle performance tracking systems that send maintenance notifications, and *d)* enhanced quality of experience achieved through the support of infotainment services and on-the-fly access to information systems for the purpose of recuperating some knowledge (*e.g.* about weather and roads conditions) or identifying hot spots (*e.g.* rest stops, restaurants, parking spots, etc.).

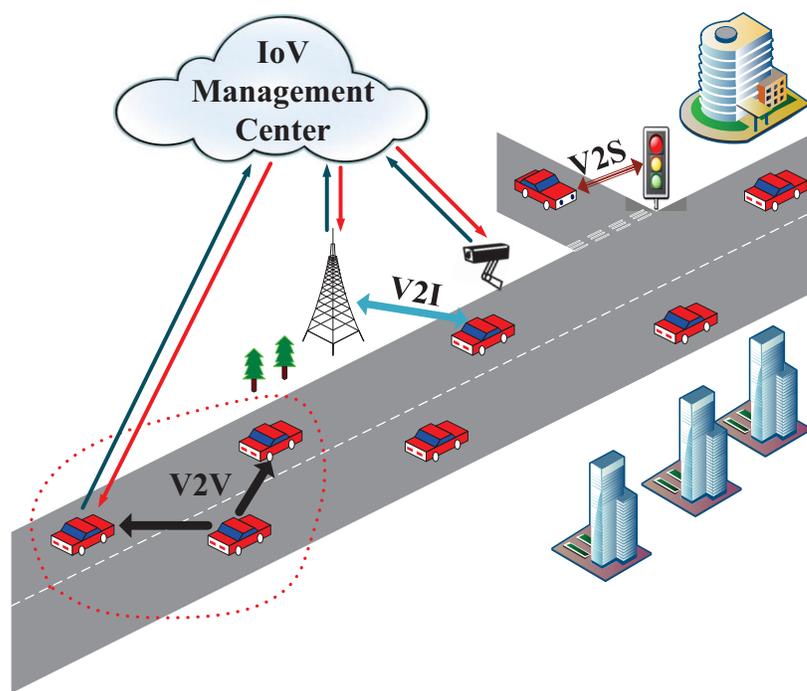


Figure 1.1: The Internet of Vehicles

During the past decade, vehicular networking has emerged as a highly active research field, which serves to establish a fully operational ITS that supports a myriad of applications related to vehicles, vehicular traffic, drivers, passengers and pedestrians. Investments of the world's leading vehicle manufacturers and public transport authorities have set to be the key drivers of the remarkably increasing popularity of this newly emerging field. A close collaboration with numerous academic and industrial parties has led to the development of various strategic plans through which novel ITS services were defined and offered (*e.g.*, navigation safety, traffic management, infotainment, etc.). At first, a few applications (*e.g.*, learning management systems, electronic toll collection, etc.) were operated using a 26 MHz bandwidth within the 900 MHz spectrum band. Conversely, being quite tiny and, at the time, remarkably defiled, that spectrum band, obviously, could not have been used to support the ideated revolutionary ITS evolution. A historical breakthrough was when the Federal Communications Commission (FCC) has allocated a 75 MHz additional bandwidth over the 5.85 – 5.925 GHz spectrum range for the purpose of sustaining Dedicated Short-Range Communication (DSRC) based ITS services, [1]. Ever since, governmental authorities, academic researchers and numerous third parties have been collectively investing significant efforts and resources for the common objective of expediting the inauguration of a full-fledged ITS.

The notable research enthusiasm to establish a revolutionary and efficient vehicular network is primarily due to the applications and services as well as their potential benefits and associated challenges. In fact, the major challenges restraining the fast and proper inauguration of an ITS are numerous, including: *a)* vehicles' high mobility, *b)* highly dynamic nature of the vehicular network, *c)* real-time nature of applications and *d)* a multitude of system and application-related requirements. Such challenges and opportunities serve as the background of the widespread interest in vehicular

networking by governmental, industrial, and academic bodies. The inception of an operational vehicular network that lives up to today’s expectations influenced the research industry to devote additional forces in testing, analysing, and optimizing the various services offered by an ITS. Official reports as well as highly reputable magazines (*e.g.* [2]) are highlighting the significant role of vehicles in extending the IoT. In point of fact, vehicles will be a *major element* of the expanding IoT, with one in five vehicles having wireless communication capabilities by 2020, accounting for more than a quarter of a billion of the cars navigating along global roads. This is especially true since, according to Gartner Inc. (a leading information technology research and advisory company), the connected vehicle is already a reality, and in-vehicle wireless connectivity is rapidly spreading from luxury models and premium brands to high-volume mid-market models. Consequently, “*smart transportation is not our future, it is our present*” [3]. The journey of establishing an operational intelligent transportation system has begun, and it shall continue until a competent, efficient and IoT-supportive vehicular network becomes a plain reality.

1.2 Thesis Contributions

This thesis aims to supplement the existing and undergoing research efforts towards the proper inauguration of a full-fledged ITS. The first contribution of this thesis manifests itself in a brief, yet comprehensive overview of the state-of-the-art Medium Access Control (MAC) protocol standards for Wireless Access in Vehicular Environments (WAVE), namely: a) the IEEE 802.11p protocol and b) the IEEE 1609.4 protocol. Precisely, Chapter 2 of this thesis presents an in-depth review of the WAVE protocol stack followed by a survey of a selection of existing work addressing numerous problems and limitations of the current WAVE MAC protocols. Chapter 2 then presents preliminary background on discrete time stochastic control using

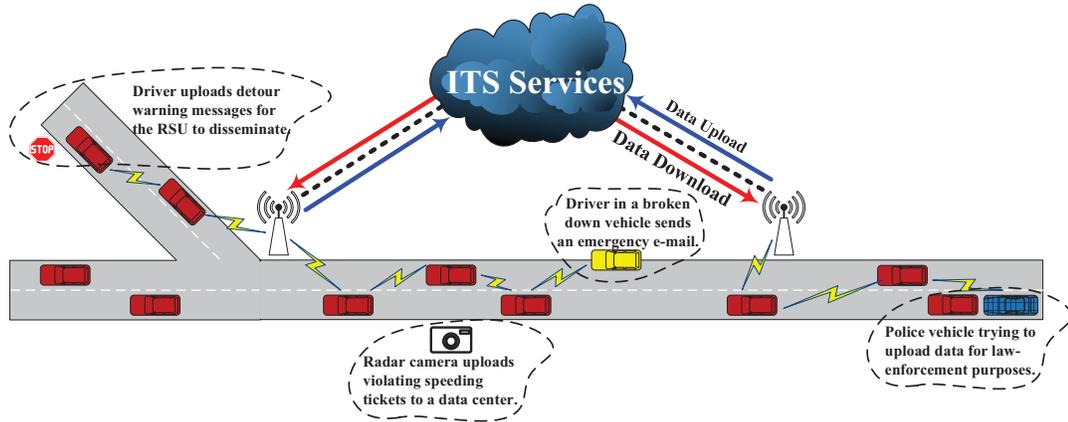


Figure 1.2: Exploiting V2V communications to connect vehicles in dark areas.

reinforcement learning. In particular, we present the basic components of Markov Decision Processes (MDPs) and introduce the mathematical and notational subtleties that will be used throughout this thesis. Finally, Chapter 2 lays out a detailed discussion on Reinforcement Learning (RL) and deep RL methods, which will be exploited in Chapters 5 and 6 of this thesis.

Next, Chapters 3 through 6 will mainly focus on three fundamental problems from the pool of existing challenges faced in vehicular networking. These problems are summarized next, and presented in details in their dedicated chapters of this thesis.

1.2.1 Connecting Vehicles Residing in Dark Areas

In a typical vehicular network, a vehicle may benefit from various ITS services during its residence time within the communication range of a RoadSide Unit (RSU) deployed alongside a roadway. However, upon its departure from the RSU’s communication range, this vehicle enters a dark area and loses all means of communication with the RSU. As a result, provisioning connectivity to vehicles residing in dark areas of a roadway is remarkably important for the safety of commuting passengers [4].

Under several circumstances (*e.g.* examples are illustrated in Figure 1.2) vehicles residing in dark areas of a long road segment desire to establish a communication with a roadside unit. For this purpose, the research industry has delved into the promotion and feasibility of a fully-connected vehicular network. Observe that, establishing a connectivity path between faraway vehicles and the RSU is possible through a combination of V2I together with multi-hop V2V communications among cooperative relay vehicles residing between the source vehicle and the RSU. This thesis investigates the necessary conditions for establishing a connectivity path between an isolated source vehicle S and a far away destination RSU D that is deployed along a roadway segment that is operating under free-flow traffic conditions. In this context, Chapter 3 of this thesis examines the availability of intermediate vehicles serving as relay nodes, and establishes an analytical framework for the purpose of estimating the probability of having an available connectivity path between S and D . In reality, a vehicular network is not stationary with time. Consequently, the events which alter the network's topology are also examined and their effect on a packet's average end-to-end delivery delay is considered. Finally, a mathematical study is conducted for examining the achieved multi-hop throughput.

1.2.2 A Vehicle's Perspective of MAC Schemes

After characterizing the properties of a vehicular connectivity link between a vehicle and a distant RSU, the forwarded message waits in the queue of its carrying vehicle until it is transmitted to the corresponding RSU. Observe that more than one vehicle may be present within the coverage range of a RSU, and multiple ones of these vehicles may request to communicate with that RSU (*i.e.*, V2I Communication) at a time. This gives rise to a multiple access problem whose resolution is, indeed, challenging. The performance of this multiple access system depends on the length of

the roadway segment covered by the RSU and is also highly affected by the vehicular traffic behaviour; precisely, the time-dependent number of simultaneously present vehicles together with these vehicles' respective residence times within the RSU's coverage range. Furthermore, other parameters such as the RSU's data transmission rate affect the amount of data that vehicles can download whereas the vehicles' respective data transmission rates determine the amount of data each of these vehicles can individually upload. Indeed, all of the vehicular traffic, wireless network settings and configurations as well as the data communication capabilities of individual vehicles exhibit a crucial interplay through which they collectively contribute to the performance of this V2I communication system.

Motivated by the above, Chapter 4 of this thesis is dedicated for the modelling and performance analysis of two novel, yet simple, MAC schemes for V2I communications. The fundamental contribution of Chapter 4 lies in the presentation of a modelling approach that is distinguishable from the existing work in the literature as it evaluates the performance of the considered system as seen from the angle of any arbitrary vehicle residing within the coverage range of a RSU. Chapter 4 lays out the detailed traffic scenario, mathematical formulation and reported results of this addressed problem.

1.2.3 Optimizing Downlink Traffic Scheduling - The Single RSU Case

Similar to all typical wireless networks, in a vehicular network, an efficient operation is the ultimate goal. Given the highly mobile facet of a vehicular network, continuous connectivity has become a crucial research challenge that is recently receiving significant attention. In order to increase the connectivity of a Vehicular Ad-hoc Network (VANET), a possible solution is to deploy multiple RSUs on long roadway

segments. Another possible solution is to increase the coverage range of VANETs' wireless nodes (*i.e.*, vehicles and RSUs). It is true that extending the coverage range of vehicles' radios may widen the connectivity area of a network, however, this solution is associated with several drawbacks such as smaller data rates and increased packet collision which may result in elevated delivery delays [5]. In both proposed solutions, the RSU is consuming larger amounts of energy, which is not always procurable. In fact, a long-lasting source of energy is provided to vehicles by their electric motor, however, it is considerably expensive to provide grid-power connection to RSUs in certain areas. This is a significant barrier to the operation of a vehicular network. Furthermore, and according to the U.S. Department of Transportation [6], it is expected that 40% of all rural free-way roadside infrastructure would be solar powered by year 2050 due to the unavailability of a power-grid connection. Consequently, since most of the RSUs will be battery-powered, it is important to schedule their operation efficiently in order to achieve minimum energy consumption. Energy-efficient scheduling at the RSU is one way to overcome the restrictions imposed by an energy-limited vehicular environment.

Chapter 5 of this thesis addresses the problem of energy-limited RSUs in a vehicular network. A Markov Decision Process is formulated and solved using a reinforcement learning technique, namely the Q-learning algorithm. The resolution is a Protocol for Energy-efficient Adaptive scheduling using Reinforcement Learning (PEARL), which is proposed for the purpose of increasing the number of downloaded bits per unit time as well as avoiding the undesired event of a vehicle departing from the RSU's communication range with an incomplete service request. PEARL's objective is to augment the RSU with the required artificial intelligence to realize and hence exploit an optimal scheduling policy which will guarantee the operation of the vehicular network during the RSU's discharge cycle while fulfilling the largest number

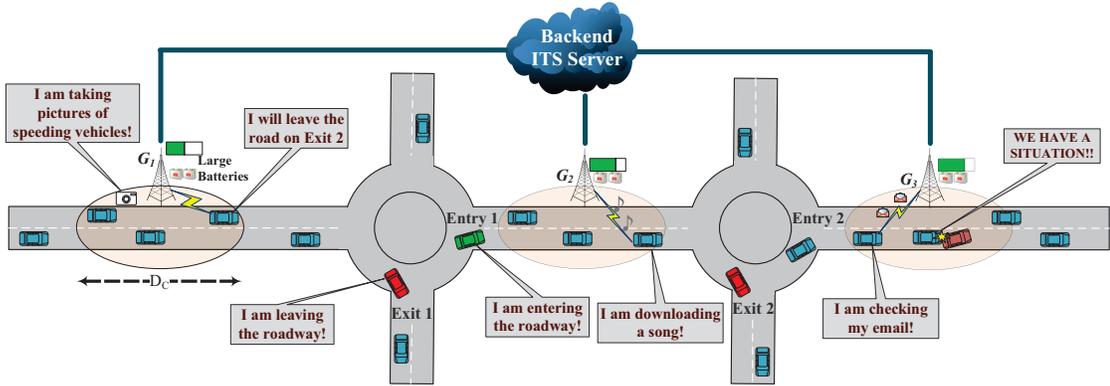


Figure 1.3: Energy-Limited Multi-RSU Vehicular Network

of service requests.

1.2.4 Optimizing the Operation of a Connected Vehicular Network - The Multiple RSU Case

The proper inauguration of a full-fledged, smart and efficient ITS is foreseen to support the legitimate realization of the next generation 5G network by providing several benefits including easier content sharing and efficient computation offloading. As such, the need to establish an intelligent and connected vehicular environment, similar to the one illustrated in Figure 1.3, has become an emerging research priority. Hence, in an effort to satisfy the driver's well-being as well as his demand for continuous connectivity, Chapter 6 of this thesis addresses both safety and Quality-of-Service (QoS) concerns in a green, balanced, connected and efficient vehicular network with multiple RSUs. Using the recent advances in training deep neural networks, this thesis exploits the deep reinforcement learning model, namely deep Q-network, which learns a scheduling policy from high-dimensional inputs corresponding to the current characteristics of the underlying network. Chapter 6 of this thesis develops an artificial DQN agent that derives efficient representations of the environment, learns from

past experience, and progresses towards the realization of a successful scheduling policy in order to extend the lifetime of a green, safe and connected vehicular network and achieve acceptable levels of QoS.

1.3 Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 presents a brief overview of the composition of the WAVE protocol suite followed by a thorough review of the IEEE 802.11p and IEEE 1609.4 MAC protocols as well as a survey of a selection of existing work that address the limitations of the current WAVE MAC protocols [7]. Chapter 2 also lays out preliminary mathematical background on MDPs and their resolution using machine learning techniques. Chapter 3 of this thesis investigates the possibility of establishing a multi-hop connectivity path between an isolated source vehicle and a faraway gateway RSU [4]. The average end-to-end delivery delay was analysed after carefully examining the events that alter the network's topology [8]. Also, a Markovian framework is established for the purpose of evaluating the per-vehicle and network throughputs [9]. Chapter 4 examines the performance of a V2I communication scenario from a vehicle's perspective. A mathematical framework is established with the objective of modelling a vehicle's onboard unit's buffer and evaluate its performance under two proposed MAC algorithms in terms of several quality-of-service metrics [10] and [11]. Chapter 5 of this thesis examines a vehicular network whose RSU dispossesses a permanent power source, but is instead equipped with a large battery, which is periodically recharged. A reinforcement learning technique is proposed for the purpose of optimizing the RSU's downlink traffic scheduling during a discharge period [12]. Chapter 6 considers a vehicular network whose operation is dictated by a deep reinforcement learning agent, which strives to optimize

the operation a tandem of connected RSUs. The large-scale nature of the underlying problem motivates the instantiation of an intelligent agent that will *a)* learn a scheduling policy from high-dimensional inputs using end-to-end deep reinforcement learning, *b)* derive efficient representations of the environment, and *c)* progress towards the realization of a successful scheduling policy which meets multiple objectives. Chapter 7 concludes the thesis and highlights a collection of open research directions for future consideration.

Chapter 2

Preliminaries and Literature

Review

The first section of this chapter presents a brief overview of the constituents of the WAVE protocol suite followed by a thorough review of the IEEE 802.11p and IEEE 1609.4 MAC protocols. Then, a survey of a selection of existing work addressing the limitations of the current WAVE MAC protocols is presented. The second section of this chapter lays out a comprehensive literature review on scheduling-based access methods. Finally, the last section presents a preliminary discussion of stochastic control of Markov Decision Processes using Machine Learning techniques.

2.1 The WAVE Protocol Suite

2.1.1 WAVE Protocol Stacks

As illustrated in Figure 2.1, the WAVE architecture incorporates two major protocol stacks: *a*) Internet Protocol version 6 (IPv6), and *b*) WAVE Short Message Protocol (WSMP). The discussed protocols in this subsection are, where applicable,

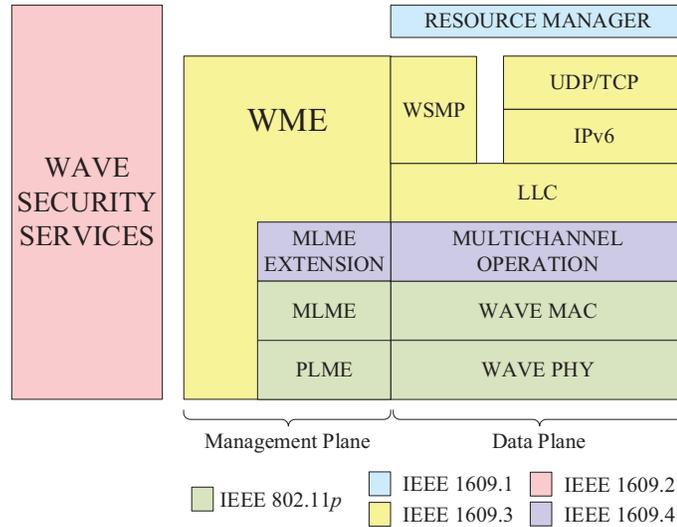


Figure 2.1: WAVE Protocol Stack.

loosely correlated to their Open Systems Interconnections (OSI) counterparts. In terms of the OSI model, both of the WAVE stacks are built on top of the same PHY and data link layers. In contrast, the network and transport layers differ from one stack to another. Note that none of the session, presentation and application layers are defined within either one of the two WAVE protocol stacks. However, two novel components that do not fit within the OSI layered structure are brought in, namely: *a)* Resource manager, and *b)* WAVE Security Services. These are handled respectively by the IEEE 1609.1 and the IEEE 1609.2 protocols.

The two WAVE protocol stacks are defined for the purpose of supporting critical, high-priority, and delay-sensitive communications as well as the typical Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) data exchanges. Safety applications in vehicular environments impose stringent requisites in terms of communication delay and error probability. However, they involve single-hop transmissions of short and complexity-minimal messages. The IEEE 1609 working group

defined the WSMP that allows for efficient WAVE Short Message (WSM) transmissions together with direct radio-resource parameters tuning in such a way to ensure reliability-maximal and timely message deliveries. The WSMP sets the minimum length of a WSM to 5 bytes, and imposes no constraint on the maximum length that would rarely exceed 20 bytes [13]. The efficiency of WSMP is quite valuable because channel congestion is a major concern in DSRC. Nevertheless, WSMP is not designed to handle classical Internet applications that persuade additional private-sector investments aiming at spreading and reducing the WAVE systems' setup and implementation costs; hence the integration of IPv6. At levels that parallel those of the third and fourth OSI layers, the IEEE 1609.3 protocol covers the specifications as well as the functionalities of WSMP and elucidates the adequate incorporation of the orthodox IPv6, UDP and TCP within the WAVE systems. Furthermore, this protocol lays out a compendium of management operations that serve the provision of network services. These are docketed as the WAVE Management Entity (WME) in Figure 2.1.

As mentioned earlier, the WAVE protocol suite is moulded on top of the IEEE 802.11 standard covering layer one in its entirety as well as a part of layer two of both of the WAVE protocol stacks as depicted in Figure 2.1. It was established that building the WAVE architecture on top of this standard has several advantages, of which the most important are: *a)* the IEEE 802.11 is a stable standard adopted by the wide majority of wireless networking experts, *b)* it supports interoperability between nodes produced by different manufacturers, *c)* it allows for the co-existence of emerging vehicular networks with other legacy wireless networks and *d)* it promotes chipset design synergism and hence contributes to achieving higher-level economies. Nonetheless, the unique characteristics that distinguish a vehicular networking environment from an archetypal 802.11-based Wireless Local Area Network (WLAN)

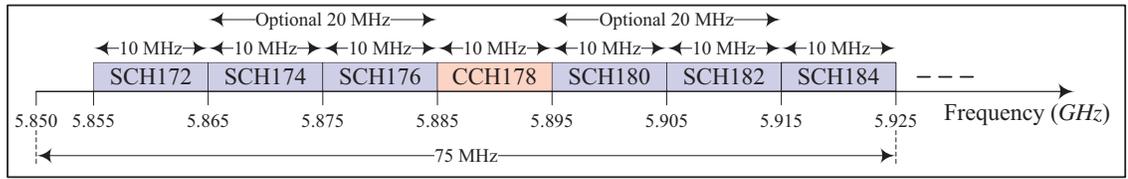


Figure 2.2: The 75 MHz frequency spectrum allocated by FCC to support vehicular networking applications

environment include but are not limited to: *a)* exiguity of central coordination, *b)* wireless link instability, *c)* remarkable path disruptions and *d)* highly dynamic topology. These differences have enforced the strict requirement for an amendment to the IEEE 802.11 standard. This amendment is, today, referred to as the IEEE 802.11p. It incorporates the data transmission functionalities as well as essential PHY and MAC layer management operations, namely: *a)* the Physical Layer Management Entity (PLME) and *b)* the MAC Layer Management Entity (MLME).

WAVE operations may be carried out over multiple channels. In particular, as explained in more details in a subsequent section and illustrated in Figure 2.2, the WAVE spectrum is subdivided into one Control CHannel (CCH) and multiple Service CHannels (SCHs). It follows that a WAVE device may have to continuously alternate between the CCH and either one of the SCHs. For reasons as such, the IEEE 1609.4 protocol has been founded and placed in one part of a layer of the WAVE protocol stack. This layer corresponds to the second OSI layer. The IEEE 1609.4 protocol is specifically tailored to govern the WAVE multichannel operations and it includes several management extensions, namely the MLME extensions block shown in Figure 2.1.

The remaining part of layer two is populated by the Logical Link Control (LLC) protocol which is based on the IEEE 802.2. For a further description of the WAVE standards, the reader is referred to [13], which describes, in details, the recent IEEE standards for vehicular environments.

The next two subsections are dedicated to present a brief, yet comprehensive overview of the IEEE 802.11p and IEEE 1609.4 MAC sublayers.

2.1.2 The IEEE 802.11p MAC Sublayer:

At the core of the IEEE 802.11p MAC sublayer lies an improved version of the IEEE 802.11e's Enhanced Distributed Channel Access (EDCA) mechanism. Essentially, EDCA is a prioritized contention-based channel access mechanism. It supports four Quality-of-Service-based (QoS) priority classes referred to as Access Classes (ACs). At the MAC sublayer, a data packet queue is associated to each AC. Each of these queues exactly behaves as an independent station and contends for channel access following the EDCA Function (EDCAF) with station-specific parameter values. Following the access regulations dictated by EDCA, higher-priority traffic is more likely to be transmitted than lower-priority traffic. Under EDCA, data packet transmission priorities are realized through the selection of different AIFS, minimum and maximum contention window values for each AC. The highest channel access priority is given to the AC with the smallest AIFS value whereas the lowest priority is given to the AC with the largest AIFS value. Furthermore, a small contention window being assigned to a high-priority AC secures that AC has a higher channel access probability than its lower priority counterparts. Information on how to determine the AIFS values as well as the minimum and maximum contention window sizes corresponding to each AC can be found in [14]. For the purpose of collision avoidance, a node engages in a backoff procedure before initiating any packet transmission. This procedure is exactly the same as that followed under the Distributed Coordination Function (DCF) with the only difference that, here, it applies for each one of the four internal AC queues.

As opposed to the typical IEEE 802.11, the IEEE 802.11p defines a WAVE Basic

Service Set (WBSS) as a unique identifier for delineating a communication zone governed by a certain wireless service provider's Access Point¹ (AP). In order to initiate a WBSS, an AP periodically diffuses WAVE Service Advertisement (WSA) beacons periodically over the CCH. In-range nodes may associate with the advertised WBSS. However, the association procedure described by the IEEE 802.11p differs from that of the typical IEEE 802.11 standard in that it excludes any form of active scanning or authentication phases. Nodes monitoring the CCH become aware of the AP's presence together with the services offered by the wireless provide to which that AP belongs. Consequently, if required, nodes join the in-range WBSS by simply switching to the advertised Service CHannel (SCH).

2.1.3 The IEEE 1609.4 Standard:

Allowing for multi-channel operations in vehicular environments is a remarkably challenging task. This is especially true since, the proper functionality of these operations comes as the result of efficient coordination and synchronization. The IEEE 802.11p by itself does not support multi-channel operations. In order to enable this feature, the IEEE 1609.4 protocol has been standardized. The underlying fundamental protocol design objective is to enable upper layer data transfers to take place over multiple channels while maintaining complete lower-layer transparency. In particular, this standard defines extensions to the IEEE 802.11p's MAC layer in order to allow for multi-channel coordination, switching and routing operations.

As illustrated in Figure 2.3, the IEEE 1609.4 presents four channel coordination schemes for single-radio devices, namely:

1. *Continuous Access*: Radio devices remain continuously tuned to a single of

¹In vehicular environments, an AP may be fixed (*e.g.* a RSU) or mobile (*e.g.* a wireless device mounted over a mobile vehicle).

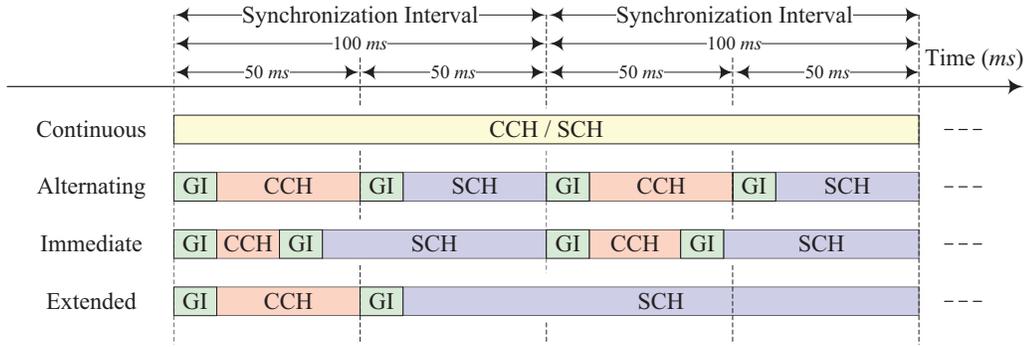


Figure 2.3: IEEE 1609.4 channel coordination schemes.

either one of the channels (*i.e.* CCH or SCH). No channel coordination is required in this case.

2. *Alternating Access*: Radio devices periodically alternate between the CCH and either one of the SCHs. Precisely, a time-division-multiplex-like scheme is adopted where a channel access cycle is subdivided into two time intervals, namely: *a*) a CCH interval (CCI) and *b*) an SCH interval (SCI). Both intervals are of equal length of 50 (*msec*); hence composing a nominal synchronization interval of 100 (*msec*). Note that the start of each CCI and SCI, a Guard Interval (GI) of 5 (*msec*) is used during which devices that are switching channels remain silent. This design is specifically tailored to satisfy the stringent requirements of the majority of emergency applications.
3. *Immediate Access*: Radio devices do not abide by the nominal synchronization intervals and immediately switch from CCH to SCH without having to wait for the beginning of the SCH interval.
4. *Extended Access*: Radio devices relentlessly communicate over either one of the SCHs without reverting back to the CCH.

2.1.4 Selective Literature Survey:

The authors of [15] performed an analytical study to examine the influence of various parameters, such as the normalized throughput and packet delivery delay, on the system performance. A 2D Markov chain was established to model the backoff procedure of the IEEE 802.11p's EDCA function. With the assumption that the channel is ideal, the results showed that, as the number of vehicles increase inside the SRU's range, the normalized throughput decreases. Furthermore, the packet delivery delay of higher priority packets was acceptable whereas, for lower priority ACs, the packet delivery delay was very high.

The authors of [16] argued that the results obtained in [15] were only applicable in the saturated case, which is not relevant in a WAVE environment. Hence, in [16], the authors eliminated the assumptions of traffic saturation and ideal channel, and analysed the performance of the IEEE 802.11p's EDCA function at the level of the CCH. Their reported results showed that the throughput of the highest-priority ACs was not affected even when the system reached saturation. However, the Frame Error Rate (FER) as well as the delays experienced by the highest-priority packets increased drastically as the traffic load increased beyond 60%. The work in [15] and [16] showed that the EDCA priority-based mechanism deployed in the IEEE 802.11p standard does not guarantee an acceptable QoS, especially for lower-priority packets.

In [17], the authors aimed at investigating the efficiency of the IEEE 802.11p's broadcast scheme in the context of a safety-related application scenario where vehicles typically exchanged two types of messages being: status and emergency messages. A Markov chain was established for the purpose of evaluating the protocol's performance in terms of several metrics, namely: *a*) the successful reception rate, *b*) the transmission probability, *c*) the collision probability and *d*) the network-level and nodal-level throughputs.

The work of [18] followed almost the same spirit of that of [17]. However, in this work, the author focused on examining the performance of each of the four AC queues. The reported results revealed the proportionality of the number of successfully received messages to the number of collisions. Nonetheless, it appeared that collisions have lower impact on lower-priority AC queues than they have on their higher-priority counterparts. This is mainly due to the fact that lower-priority AC queues are often subject to larger contention window sizes and their rate of packet transmission attempts is much less than that of the higher-priority AC queues. Following the insights provided in [18], it is important to point out a non-trivial drawback that the IEEE 802.11p has; this being the fact that, in the context of dense vehicular network scenarios, this protocol will fail to secure an acceptable QoS level for multiple high-priority messages, [19]. This is especially true since these messages will experience severe collisions. The traffic shaping algorithms sound like a promising approach to work around this problem especially if the vehicular density and the number of successfully received messages per AC are accounted for when designing a shaping function.

In [20], the authors pointed out that the typical static backoff schemes poorly adapt to the dynamic number of contending vehicles and hence significantly alter the IEEE 802.11p's throughput performance at both the overall network and individual node levels. To work around this problem, the authors propose two solutions of which one is centralized and the other is distributed. Under the centralized scheme, the optimal transmission probability is computed as a function of the exact number of contending vehicles assumed to be known a priori. Under the distributed scheme, each vehicle individually sets its backoff window based on an estimation of the channel conditions.

In [21], the authors extensively presented the problem of multi-channel access in

vehicular environments. Furthermore, the authors layout a brainstorm-like discussion on technical approaches that may lead to the resolution of each one of the identified problems.

The authors of [22] established a mathematical framework with the objective of modelling and evaluating the performance of the IEEE 802.11p multi-channel broadcasting mechanism in terms of the packet delivery probability. Their study captured the dynamics of the channel switching operations in the context of a networking scenario characterized by a variable number of vehicles. The authors highlighted the severe impact of the WAVE channel switching on the communication's reliability and provided design guidelines that aim at improving the performance of the WAVE's broadcast mechanism. These guidelines seem to be technically sound and generally valid.

Recall that, in the basic time division concept defined in IEEE 1609.4, time is segmented into *sync periods*, consisting of one CCI and one SCI, which are, by default, 50 ms each. Many studies have been laid out to examine the efficiency of this time division. For instance, in [23], the authors studied the synergy of time allocations on CCH and SCH with EDCA differentiation parameters using IEEE 802.11p. Using a probabilistic analysis as well as a queueing theory approach, the authors highlighted the unfairness among traffic classes caused by different AIFSN values for different traffic categories. Furthermore, the results shed the light on the drawbacks of having an inactive channel time (i.e., the CCH and SCH being inactive in SCI and CCI respectively). The authors argued that the severity of these problems can be diminished by wisely selecting the values of the channel duty cycles as well as slighter EDCA differentiation parameters.

The work of [24] revolved around prioritized broadcasting in IEEE 802.11p-based

multi-channel vehicular networks. In particular, the authors have developed an analytical framework whose objective was to capture the dynamics of beaconing and service advertisement broadcast mechanisms which take place over the CCH.

The system's model in [24] was developed under the assumption of non-saturation conditions. Thus, as opposed to the typical 802.11-based wireless network analysis (*e.g.*, [25]), that model relaxed the orthodox assumption of nodal independence whose validity stands only under network saturation conditions. Instead, the authors of [24] presented the exact mathematical analysis describing the model's underlying three-dimensional stochastic process. Moreover, their model accounted for the unique features of multi-channel-based WAVE communications, namely: *a)* the alternate channel switching, *b)* the criticality of the routine control traffic broadcasted over the CCH, *c)* the inherent EDCA's differentiated access priorities and *d)* the short validity of the information carried by beacons and service advertisement messages. The system's performance was evaluated in terms of the frame delivery probability. This was especially reasonable given the short frame validity deadline which renders the queueing delay a marginally relevant measure of the system's performance.

The principle finding of [24] was that the access priority differentiation mechanism described in the standard's documentation does not serve the QoS improvement objective. Indeed, this is mainly due to the short contention window values which, in turn, aggravate internal collisions and lead to a severely degraded system's performance. Apparently, increasing the backoff intervals seems to be an appropriate countermeasure to suppress the inimical impact of synchronized frame collisions and reduce the frame loss probability. However, increasing the backoff intervals beyond a certain threshold disallows the expiry of backoff counters ahead of CCI's end. This, in turn, increases frame losses. At this level, increasing the data rate could impede frame losses. However, a maximum rate of 6 (*Mbps*) alters the system's robustness

against interference. Finally, the use of smaller frame sizes helps in avoiding channel-engendered errors and improving the frame loss probability. Nonetheless, according to [26], the frame size is reduced only to repudiate overheads.

Vehicular networking communication protocols are still under careful realization and the practical full-fledged deployment of vehicular networks is still at its early stages. Furthermore, WAVE protocol standards lack any mechanism that classifies vehicular nodes under multiple priority classes and regulates the channel access in such a way to adequately distribute the available bandwidth and possibly other resources among these nodes. The limitations and shortcomings of the current WAVE protocols have been established as key drivers for numerous investigations on the potential suitability of scheduling-based access technologies that may be exploited to support the proper functionality of vehicular networks. The next section sheds the light on a selection of scheduling-based MAC methods in the context of a V2I communication scenario.

2.2 Scheduling-Based Access Methods

2.2.1 Overview:

On a microscopic scale, the IEEE 802.11p's EDCA priority-based mechanism suffers from several loopholes rendering it often prone to failure in ensuring fair access to all of a node's internal AC queues. This is especially true in cases where nodes accumulate heavy high-priority data traffic in a network with medium to high traffic intensity. In such cases, the lower priority AC queues suffer from a stringent starvation problem.

On a macroscopic scale, the WAVE protocol standards lack any mechanism that

classifies vehicular nodes under multiple priority classes and regulates the channel access in such a way to adequately distribute the available bandwidth and possibly other resources among these nodes. For instance, in the context of V2I communications, an arriving vehicle to a RSU resides within the range of that RSU for a limited period of time, which depends on its speed. Observe that multiple vehicles may co-exist within the RSU's range at a time. Whenever more than one of these vehicles require access to the RSU, that RSU may engage in a decision-making process whereby, based on some criterion, it selects which vehicle to serve first.

Definition: *In the context of vehicular networking, scheduling is a decision-making process whose outcome is an efficient, ultimately optimal, joint channel access regulation and resource allocation policy that has to be adopted by vehicular nodes for the purpose of realizing one or several objectives concurrently.*

For instance, if the objective is to maximize the number of served vehicles, then vehicles have to be served in ascending order of their residence times. Alternatively, if the objective is to reduce the system's response time (*i.e.* the amount of time a vehicle has to wait before it gets served) then vehicles have to be served in ascending order of their nominal service times. If an optimal trade off between throughput and response time has to be achieved, an earlier-deadline-first policy has to be adopted where a vehicle's deadline is possibly determined by a cost function consisting of a weighted combination of that vehicle's nominal service time (*i.e.* the amount of time a vehicle requires to successfully complete service) and its residence time.

In addition to the above, there exist a multitude of other remarkably challenging objectives whose realization was made possible through the design of appropriate scheduling algorithms. The next subsection presents a survey on a prime selection of existing scheduling algorithms that were specifically tailored for vehicular networks.

2.2.2 Selective Literature Survey:

In the context of a V2I communication scenario, the authors of [27] aimed at maximizing the delivery ratio of non-safety messages. This was achieved through their design of a control-theoretical packet scheduling strategy that mitigates the joint impact of the rapidly changing propagation conditions as well as the short connection lifetime. The core of their design was inspired by guidelines which, in turn, were dictated by the resolution of an optimal packet transmission scheduling problem. This problem enclosed stringent constraints imposed by the unique characteristics of VANETs. The authors argue that packet prioritization is necessary to achieve service differentiation in VANETs with heterogeneous information flows. Their strategy enables QoS differentiation through the allocation of a fixed amount of bandwidth for each access class. The typical EDCA mechanism of the WAVE standard can be easily augmented with the above-proposed control-based scheduling scheme in order to further support delay-sensitive safety messages.

The proposed scheme in [27] suffers from a serious fairness/starvation problem. This problem arises as a result of the low-priority access queues being subject to service denials whenever their higher priority counterparts are not empty. In this regard, an efficient channel access protocol design has to account for this type of starvation through ensuring fair channel access among all queue classes. In addition to the above, service advertisement and channel coordination have been overlooked in [27]. In this regard, it is important to attract the reader's attention that these additional constraints further increase the problem's complexity. This is especially true since advertising service and coordinating channel assignments require the transmission of additional control messages and hence wasting precious portions of the CCI.

The work of [28] revolved around a study of the random access problem in a

Drive-Thru Internet (DTI) scenario. The aim was to design efficient resource allocation schemes that fully utilize the limited communication opportunities subject to time-varying channel contention and capacity levels. Formulated as a finite-horizon decision problem, the model enclosed the Dynamic Optimal Random Access (DORA) algorithm that lead to establishing an optimal access policy for a single RSU. Results showed that, depending on the applications' QoS requirements, the vehicle can achieve different levels of trade-off between the uploaded file size and the paid fee to the RSU by enabling a self-incurred penalty. Furthermore, the authors considered the case of multiple consecutive RSUs along the highway. DORA outperformed three other heuristic schemes and achieved minimal total cost and highest upload ratio. The major challenges the authors faced throughout their design were related to both scalability, *i.e.*, the dynamic number of vehicles involved in the channel contention process, and time, *i.e.*, vehicles resided for a limited time period within the RSU's coverage range and hence suffered from limited data upload opportunities.

The authors of [29] examined the V2I wireless access for streaming applications in a public transportation system. The authors formulated an optimization problem with the objective of providing a cost-minimal wireless connectivity that satisfies the end-users QoS requirements. For this purpose, a hierarchical optimization framework was established to determine an optimal policy indicating whether or not it is convenient for a vehicle to request bandwidth reservation from the Stationary Internet Gateway (SIG). The proposed mathematical model studied the system's performance variation as a function of the following factors: *a)* the streaming application's requirements, *b)* the vehicular mobility and *c)* the channel quality. The work in [29] considered both the user-centric and network-centric point of views to provide a unified model for optimizing wireless access in a V2I communication scenario.

The authors of [30] investigated the impact of vehicular traffic characteristics (*i.e.*

traffic flow and density, the vehicle speeds, etc.) on the throughput performance of DTI. They resorted to Markov Reward Processes for the purpose of developing an analytical model to quantitatively characterize the per-vehicle-per-RSU downloaded data volume. In particular, they derived a distribution for the number of bytes downloaded per RSU. Their model accounted for the RSU's transmission rate and coverage area. Nonetheless, it was founded on top of the restrictive ideal MAC operation conditions and fair distribution assumptions. Such assumptions delineate the surrealism of the reported results in [30].

In [31], the authors proposed a basic low-complexity V2I access scheme called $D*S$ where the RSU stored the Service Requests (SRs) and the request with the least $D*S$ was served first. D is the SR's deadline and S is the data size to be uploaded to the SIG. $D * S$ showed better performance when compared to three other access schemes namely *a)* FCFS, *b)* Earliest Deadline First, and *c)* Smallest Datasize First. The authors then worked on improving the performance of their proposed algorithm by using broadcasting techniques, and hence, serving more requests simultaneously. Furthermore, in an attempt to study the tradeoffs between the service ratio and the data quality, the authors extend their algorithm and propose a Two-Step scheme where two priority queues were used, *i.e.*, one for upload requests and the other for download requests. The results presented therein showed that the Two-Step scheduling scheme is adaptive to different workload scenarios. The authors then studied the uplink MAC performance of a DTI scenario in [32]. Both the contention nature of the uplink and the realistic traffic model were taken into consideration. An analytical framework was developed to quantify the uplink performance of DTI in an IEEE 802.11p environment in terms of packet collisions and uplink capacity. Furthermore, for the purpose of maintaining optimal system performance, the authors explored the adjustment of transmission power as a means of admission control by the roadside

unit.

Finally, an identified common drawback of all of the above surveyed work is the fact that the failure to account for safety messages. Consequently, the surveyed scheduling algorithms require fundamental refinements in order to support high-priority delay-sensitive and safety-related applications. One possible approach consists of augmenting the RSU with an additional queue for safety messages which, in turn, will be granted preemption capabilities and hence the supremacy of being served prior to other non-safety messages.

2.3 Markov Decision Processes

This section introduces the basic components of a Markov Decision Process (MDP) and discusses the mathematical and notational subtleties that will be used throughout this thesis, particularly in Chapters 5 and 6. MDPs provide a mathematical framework for modelling decision making in situations where outcomes are influenced by a decision maker. MDPs are particularly useful for studying a wide range of optimization problems that can be solved via dynamic programming and reinforcement learning.

More precisely, an MDP is a discrete time stochastic control process where decisions are made at points of time referred to as decision epochs. Let T denote the set of decision epochs. In this thesis, T is a discrete finite set of time steps where time is divided into N decision epochs. Let x define the state of the system. At time step t_n ($n = 0, 1, 2, \dots, N$), the process is in state x_n , and the decision maker may choose any action a_n from the set of admissible actions in x_n , A_n . The process responds at the next time step t_{n+1} by randomly moving into a new state x_{n+1} , and giving the decision maker a corresponding reward $r(x_n, a_n, x_{n+1})$.

It is important to mention that, the probability that the process moves into its

new state x_{n+1} is influenced by the chosen action a_n at t_n . Specifically, it is given by the state transition probability $P(x_{n+1}|x_n, a_n)$. As such, the next state x_{n+1} depends on the current state x_n and the decision maker's action a_n . Note that, given x_n and a_n , x_{n+1} is conditionally independent of all previous states and actions; in other words, the state transitions of an MDP process satisfies the Markov property. The formal mathematical definition of a Markov Decision Process is laid out next.

2.3.1 MDP Mathematical Definition

A MDP model with a planning horizon N is a 5-tuple $(E, A, P(x'|x, a), r(x, a, x'), \phi)$ where:

- E is the state space where, at any time t_n , the system state $x_n \in E$.
- A is the action space. The set of admissible actions whenever the system is in state x is $A_x \subset A$.
- $P(x'|x, a)$ is a transition probability, for reaching state x' when taking action a from state x .
- $r(x, a, x')$ is the immediate reward (or expected immediate reward) received after transitioning from state x to state x' due to action a .
- ϕ is a set of specific parameters for some of the different kinds of learning settings. For instance, ϕ includes $\gamma \in [0, 1]$, which is the discount factor representing the difference in importance between future rewards and present rewards.

An MDP is fully determined by the above 5 quantities that completely characterize the environment. A policy, contingency plan or strategy specifies the decision rule to be used at all decision epochs. It provides the decision maker or agent with guidelines for action selection under any possible system state in order to control

(or partially control) the MDP environment. The core problem of MDPs is to find a policy $\pi : E \rightarrow A$ that maps the set of states to the set of actions. Whenever the agent is following the policy π , the action at t_n is $a_n = \pi(x_n)$, and thus, the immediate step reward can be written as $r_n = r(x_n, \pi(x_n))$. The goal is to find a policy π that will maximize some cumulative function of the immediate rewards, typically the expected discounted sum over a potentially infinite horizon, given by:

$$R = \sum_{n=0}^{\infty} \gamma^n r(x_n, \pi(x_n)) \quad (2.1)$$

Recall that, γ is a discount factor which is set between 0 and 1. $\gamma = 0$ will make the agent short-sighted by only considering current rewards, while a factor approaching 1 will make it strive for a future high reward. The next subsections discuss various ways to solve an MDP and find an optimal policy.

2.3.2 Optimality Equations

In order to solve an MDP, an optimal policy $\pi^* : E \rightarrow A$ should control the agent's decisions to maximize the expected sum of rewards over the planning horizon. Let Π denote the set of all admissible policies. π^* is obtained by solving the following problem:

$$\pi^* = \max_{\pi \in \Pi} \mathbb{E}_{\pi} \left\{ \sum_{n=0}^{\infty} \gamma^n r(x_n, \pi(x_n)) \right\} \quad (2.2)$$

Now, in order to solve the above equation, the agent will estimate *how good* it is to be in a given state when following policy π . Let $V_{\pi}(x)$ be the value of being in state x_n when following policy π . Informally, the value of a state x under a policy π , denoted by $V_{\pi}(x)$, is the total expected returns when starting in x and following π

thereafter. Hence, $V_\pi(x)$ is given by:

$$V_\pi(x) = \mathbb{E}_\pi [R_n | x_n = x] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k (R_{n+k} | x_n = x) \right] \quad (2.3)$$

where R_n is the total expected discounted sum starting from time t_n . V_π is called the state-value function for policy π . According to [33], for a given admissible policy $\pi \in \Pi$, the value function V_π satisfies the following Bellman recursive equation:

$$V_\pi(x = x_n) = r(x_n, \pi(x_n)) + \gamma \sum_{x_{n+1}} \left\{ P(x_{n+1} | x_n, \pi(x_n)) V_\pi(x_{n+1}) \right\} \quad (2.4)$$

The optimal policy π^* gives the optimal value function V^* , defined by:

$$V^*(x) = \max_{\pi \in \Pi} V_\pi(x), \forall x \in E \quad (2.5)$$

The optimal policy associated with optimal value function given in Equation (2.5) achieves the maximum reward expression laid out in Equation (2.1) [34]. However, in order to solve Equation (2.5), the knowledge of the transition probability function, $P(x_{n+1} | x_n, \pi(x_n))$ is required. In these cases, techniques such as Value Iteration (VI) and Policy Iteration (PI) may be used in order to find the optimal policy [34]. However, note that, the formulated MDPs in this thesis are model-free (*i.e.*, unaware of the state transition probability). Classical Dynamic Programming (DP) techniques (*e.g.* VI and PI) may not be exploited, thus, this thesis will not discuss these methods. The reader is kindly referred to [33] and [34] for detailed discussions on DP methods to solve an MDPs.

Now, a reinforcement learning method, Q-learning, presents itself as a simple way to find the optimal policy by experiencing the consequences of actions without the requirement of an established transition function $P(x_{n+1} | x_n, \pi(x_n))$. Therefore, let $Q_\pi(x, a)$ be value of taking action a in state x under a policy π . Informally, $Q_\pi(x, a)$

is the expected return starting from x , taking the action a , and thereafter following policy π . $Q_\pi(x, a)$ is given by:

$$Q_\pi(x, a) = \mathbb{E}_\pi \left[R_n | x_n = x, a_n = a \right] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k (R_{n+k} | x_n = x, a_n = a) \right] \quad (2.6)$$

According to [33], for a given admissible policy $\pi \in \Pi$, the action-value function Q_π satisfies the following Bellman recursive equation:

$$Q_\pi(x, a) = \mathbb{E}_\pi \left[r(x_n, a_n, x_{n+1}) + \gamma Q_\pi(x_{n+1}, a_{n+1}) | x_n = x, a_n = a \right] \quad (2.7)$$

The optimal policy π^* gives the optimal action-value function Q^* , defined by:

$$Q^*(x, a) = \mathbb{E}_{\pi^*} \left[r(x_n, a_n, x_{n+1}) + \gamma V^*(x_{n+1}) | x_n = x, a_n = a \right] \quad (2.8)$$

$$= \mathbb{E}_{\pi^*} \left[r(x_n, a_n, x_{n+1}) + \gamma \max_{a_{n+1}} Q^*(x_{n+1}, a_{n+1}) | x_n = x, a_n = a \right] \quad (2.9)$$

Now, optimal action selection can be simply put as:

$$\pi^*(x) = \arg \max_a Q^*(x, a) \quad (2.10)$$

That is, the best action is the action that has the highest expected action-value function based on all the possible next states resulting from taking that action. The next subsections discuss two methods to solve Equation (2.10) in model-free environments; particularly, reinforcement learning and deep reinforcement learning.

2.3.3 Solving Model-Free MDPs

Reinforcement Learning

Q-learning is a model-free Reinforcement Learning (RL) technique which is widely used to find an optimal policy for any given finite MDP. It works by learning an action-value function that eventually achieves the optimal reward of taking a given action in a given state and following the optimal policy thereafter.

Since the Q-function makes the action explicit, the Q-values can be estimated using the following online incremental update stochastic Q-learning algorithm:

$$Q(x_n, a_n) := Q(x_n, a_n) + \alpha(n) \left[r(x_n, a_n) + \gamma \max_{a_{n+1}} Q(x_{n+1}, a_{n+1}) - Q(x_n, a_n) \right] \quad (2.11)$$

Note that, the choice of the value of γ becomes very crucial for the convergence of the Q-values presented above. In fact, $\gamma = 0$ will make the agent short-sighted by only considering current rewards, while a factor approaching 1 will make it strive for a future high reward. Furthermore, $\alpha(n)$ is the step-size learning rate which is set between 0 and 1. Note that, whenever $\alpha(n) = 0$ the Q-values are not updated and hence nothing is learnt. However, setting a high value for $\alpha(n)$ means that learning occurs quickly. The learning rate satisfies the following conditions:

$$\begin{aligned} \sum_n \alpha(n) &= \infty, \\ \sum_n \alpha(n)^2 &< \infty. \end{aligned} \quad (2.12)$$

The first condition ensures that the algorithm does not prematurely converge, whereas the second condition ensures that the noise in the algorithm asymptotically

vanishes [33]. Most often, the step sizes are simply chosen to be $\alpha(n) = 1/n$. The convergence of the underlying algorithm has been widely analysed and proven. The optimal Q-values as well as the optimal policy are obtained upon the convergence of the following algorithm. Note that, several studies have addressed the problem of improving the convergence speed of Q-learning algorithms, where recommendations for faster convergence were empirically derived. For instance, the use of bootstrapping methods may accelerate the convergence of reinforcement learning algorithms. However, bootstrapping requires a priori initial knowledge of the action-value function using supplied initial policies, which is outside the scope of this thesis.

When the action-value function converges to the true values of $Q(x, a)$, $\forall(x, a) \in E \times A$, the decision maker chooses an optimal action a_n given by:

$$a_n = \arg \max_a Q(x_n, a) \tag{2.13}$$

During the agent learning phase, if it chooses the action using equation (2.13) all the time, the resulting algorithm would not be guaranteed to lead to an optimal solution [35]. The problem is that the factors $Q(x, a)$ might underestimate the value of a state-action pair. As a result actions that map the system to an underestimated state will never be chosen, and the decision maker may end up ignoring actions that might be quite rewarding. Hence, a rule that forces the agent to explore states and actions that may not look attractive is essential in the learning phase. One of the simplest ways of overcoming this issue is to modify the decision policy for choosing an action during the learning phase using the ϵ -greedy policy. Using this policy, with probability ϵ , the agent chooses an action at random from the set of admissible actions at t_n . And with probability $1 - \epsilon$, the agent chooses the action according to equation (2.13), in which case the agent is exploiting its current knowledge of the value of each state-action pair. The ϵ -greedy policy is simple and intuitive, and

produces a guarantee that the agent will visit every (reachable) state and action infinitely often. Note that, the ϵ -greedy method works well for some problems, but not for others. Solving the problem of when to explore and when to exploit is known as the exploration versus exploitation problem, which is still considered one of the most challenging problems in a learning environment.

The Q-learning algorithm used in this thesis to find the optimal policy π^* is presented below:

Algorithm 2.1 Q-Learning Method to Compute π^*

- 1: For each $(\mathbf{x}, \mathbf{a}) \in \mathbf{E} \times \mathbf{A}$, initialize $Q(\mathbf{x}, \mathbf{a}) = \mathbf{0}$
 - 2: Repeat (for each learning episode)
 - 3: **for all** $n \in T$ **do**
 - 4: Observe \mathbf{x}_n
 - 5: Select action \mathbf{a}_n using ϵ -greedy method
 - 6: Execute \mathbf{a}_n and observe $r(\mathbf{x}_n, \mathbf{a}_n, \mathbf{x}_n + \mathbf{1})$ and $\mathbf{x}_n + \mathbf{1}$
 - 7: Update $Q(\mathbf{x}_n, \mathbf{a}_n)$ (according to Equation (2.11)).
 - 8: $\mathbf{x}_n \leftarrow \mathbf{x}_n + \mathbf{1}$
 - 9: **Until** \mathbf{x} is terminal
-

The Q-learning method of reinforcement learning assumes that the estimates of state-action values could always be represented as a look-up table with one entry for each state-action pair. Unfortunately, this strategy is only applicable in a very limited number of problem, and is often infeasible. In fact, in a high-dimensional state space scenario, not only does this approach present unrealistic memory requirements, but also an agent is not able to visit all states or state-action pairs, and thus the time needed to fill the look-up table becomes increasingly problematic. In the case of a very large or continuous state-space, the problem becomes intractable. This is known as the *curse of dimensionality* in reinforcement learning.

The literature encloses numerous solutions which address this problem (*e.g.*, linear function approximation [36], hierarchical representations [37], state aggregation [38], etc.). These methods greatly rely on the system state representations, thus making

the agent not fully autonomous and reducing its flexibility. The use of non-linear function approximation techniques was relinquished as these methods turned out to be unstable and non-converging when used to represent the action-value function [36]. Only recently, the authors of [39] presented the Deep Q-Network (DQN) algorithm and tested it in a challenging framework composed of several Atari games. DQN achieved dramatically better results than earlier approaches and professional human players and showed a robust ability to learn representations from very high-dimensional input.

Chapter 6 of this thesis examines a variant of DQN in the context of a V2I communication scenario in a connected vehicular network with multiple RSUs. Hence, the next subsection lays out fundamental background about deep reinforcement learning.

Deep Reinforcement Learning

So far, the previous section has established that an MDP is a discrete-time stochastic control process, where, at each time step, the process is in some state x_n and the decision maker chooses a feasible action a_n . Accordingly, the process then moves to a new state x_{n+1} and awards the decision maker a corresponding reward $r(x_n, a_n, x_{n+1})$. The probability that the process moves into its new state x_{n+1} is influenced by the chosen action. It is defined by the state transition function $P(x_{n+1}|x_n, a_n)$, which satisfies the Markov Property. The core problem of MDPs is to find an optimal policy for the decision maker, defined by a function π^* that specifies the action that the decision maker will choose when in state x in order to maximize a cumulative function of the random rewards.

A classical MDP can be solved by value iteration or policy iteration [34] to determine the optimal policy; however, these two methods assume that the decision maker accurately knows the transition function and the reward for all states in the

environment. Whereas, in practice, the decision maker may not have an explicit representation of the transition and reward functions. Fortunately, there is a way to learn these functions; the decision maker trades learning time for a priori knowledge through a form of reinforcement learning known as Q-learning. Q-learning is a form of model-free learning, which teaches the decision maker how to behave in an MDP when the transition and/or reward functions are unknown. The previous subsection highlighted the major setback of Q-learning, a phenomenon commonly known as the curse of dimensionality. This thesis exploits function approximation techniques to overcome this widely known limitation of RL in large state space scenarios. In particular, neural networks are exceptionally good at coming up with good features for high dimensional input data. In fact, the action-value function can be represented with a neural network, which takes the current system state and action as input and outputs the corresponding Q-value. This technique is commonly known as deep reinforcement learning.

A neural network with weights θ , referred to as a Q-network [39], is a non-linear function approximator which approximates the action-value function $Q(x, a)$ by $Q(x, a; \theta)$. A Q-network can be trained in order to learn the parameters θ of the action-value function $Q(x, a; \theta)$ by minimizing a sequence of loss functions, where the i^{th} loss function $L_i(\theta_i)$ is given by:

$$L_i(\theta_i) = \mathbb{E} \left[r(x_n, a_n) + \max_{a_{n+1}} \left\{ Q(x_{n+1}, a_{n+1}; \theta_{i-1}) \right\} - Q(x_n, a_n; \theta_i) \right]^2 \quad (2.14)$$

Note that, θ_i are the neural network's parameters at the i^{th} update, and the parameters from the previous update, θ_{i-1} are held fixed when optimizing the loss function $L_i(\theta_i)$. Thus, the term $\left\{ r(x_n, a_n) + \max_{a_{n+1}} Q(x_{n+1}, a_{n+1}; \theta_{i-1}) \right\}$ is the target for iteration i , which depends on the neural network's parameters from the last update. Hence,

the objective is to find the neural network’s set of weights which make the above cost expression as small as possible. This is done using an algorithm known as gradient descent, which repeatedly computes the gradient $\nabla_{\theta_i} L_i(\theta_i)$, and updates the neural network’s weights in order to reach a global minimum. Hence, differentiating the loss function with respect to the neural network’s parameters at iteration i , θ_i gives the following gradient:

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E} \left[\left(r(x_n, a_n) + \max_{a_{n+1}} \{ Q(x_{n+1}, a_{n+1}; \theta_{i-1}) \} - Q(x_n, a_n; \theta_i) \right) \nabla_{\theta_i} Q(x_n, a_n; \theta_i) \right] \quad (2.15)$$

The use of batch methods, which utilize the full training set to compute the next update to parameters at each iteration tend to converge very well to local optima. However, often in practice, computing the cost and gradient for the entire training set is extremely slow and sometimes intractable on a single machine, especially when the training dataset is large. Therefore, rather than computing the full expectations in the above gradient equation, it is computationally desirable to optimize the loss function using the Stochastic Gradient Descent (SGD) method. SGD updates the neural network’s parameters after seeing only a single or a few training examples. The use of SGD in the neural network setting is motivated by the high cost of running back propagation over the full training set.

In practice, reinforcement learning methods tend to diverge when used with non-linear function approximators such as a neural network. In order to avoid the divergence of deep reinforcement learning algorithms, three techniques were introduced in [39], namely:

1. Experience Replay: The parameters of the neural network are updated by performing a SGD step on random samples of past experience rather than the most recent samples of experience. This reduces the correlations between successive updates applied to the network, hence breaking the strong correlations of the

training data and reducing the variance of the updates.

2. Fixed Target Network: The neural network’s parameters used to compute the target for the i^{th} iteration are held fixed for intervals of several thousand SGD steps, then they are updated with the current realized parameters. A target Q-network reduces the correlations between the target and the obtained Q-values, thus making the problem less non-stationary.
3. Reward Normalization: This technique limits the scale of the error derivatives and ensures that gradients are well conditioned. This approach eliminates the instability in error back-propagation, such that no outlier update can have too much impact on the learning. However, in this setting, the learning agent can no longer differentiate between small and large rewards. As such, normalizing the rewards adaptively to sensible range increases the robustness of the derived gradients and stabilizes the deep learning process.

As a result, deep reinforcement learning is used to minimize the following loss function:

$$L_i(\theta_i) = \mathbb{E}_{x_n, a_n, r_n, x_{n+1} \sim \mathbb{D}} \left[r(x_n, a_n) + \max_{a_{n+1}} \left\{ Q(x_{n+1}, a_{n+1}; \theta^{-1}) \right\} - Q(x_n, a_n; \theta_i) \right]^2 \quad (2.16)$$

where \mathbb{D} is the experience replay memory and the symbol \sim is used to denote that a minibatch of transitions (x_n, a_n, r_n, x_{n+1}) is sampled from \mathbb{D} . θ^{-1} are the parameters of the target Q-network. By using the above three techniques, the convergence of the underlying deep reinforcement learning algorithm has been empirically proven in [39] and [40]. On the other hand, the drawback of using the experience replay is the substantial memory requirements.

Deep Q-Network (DQN) is an off-policy algorithm as it learns an optimal action $a_n = \max_a Q(x_n, a_n; \theta)$ while choosing random actions to ensure adequate exploration

of the state space. Recall that, a common problem often faced by autonomous learning agents is the trade-off between acting to gain information and acting to gain rewards. As such, the ϵ -greedy strategy is also used herein for active exploration. The neural network training process using a deep reinforcement learning algorithm, which will serve to approximate the state-action function is presented below:

Algorithm 2.2 Deep Q-Learning with Experience Replay and Fixed Target Network

- 1: Initialize replay memory \mathbb{D} to capacity C
 - 2: Initialize Q-network with random weights θ
 - 3: Initialize target Q-network with random weights $\theta^- = \theta$
 - 4: **for** episode = 1, M **do**
 - 5: Collect network characteristics to realize state x_0
 - 6: **for** $n = 0, T$ **do**
 - 7: $a_n = \operatorname{argmax}_a Q(x_n, a_n; \theta)$ with probability $1 - \epsilon$.
 - 8: Otherwise, action in a_n is randomly selected.
 - 9: Execute a_n and observe r_n and x_{n+1}
 - 10: Store transition (x_n, a_n, r_n, x_{n+1}) in \mathbb{D}
 - 11: Sample random minibatch of transitions (x_n, a_n, r_n, x_{n+1}) from \mathbb{D}
 - 12: Set the target to r_n if episode terminates at $n + 1$, otherwise, target is $r_n + \max_{a_{n+1}} Q(x_{n+1}, a_{n+1}; \theta^-)$
 - 13: Perform a SGD update on the current Q-network parameters θ
 - 14: Every C steps, set the target Q-network parameters $\theta^- = \theta$
 - 15: **end**
 - 16: **end**
-

Chapter 3

Multi-hop Vehicular Connectivity Path: A Feasibility Study

3.1 Introduction

The transportation research industry has long anticipated the deployment of a full-fledged vehicular network that will help reduce accidents, facilitate eco-friendly driving, provide accurate real-time traffic information, and offer entertainment and leisure services to commuting passengers. Today, technology advancements have augmented vehicles with intelligent devices allowing them to behave as mobile sensors and/or data relays and hence, rendering them indispensable major components of ITSs that, in turn, provision users with adequate access to modern information and safety-related services during their commute on roads and highways. At the core of these ITSs are VANETs that incorporate two types of communication, namely: *a)* V2V and *b)* V2I, which support numerous attractive applications related to safety (*e.g.*, collision detection and lane change warning), driving assistance (*e.g.*, online navigation and smart parking), as well as infotainment (*e.g.*, mobile office and media streaming).

In a typical VANET, a vehicle residing within the range of a RSU may directly communicate with that RSU using V2I communications and, hence, exploit a variety of services that happen to be offered by that RSU. However, upon its departure from the RSU's coverage range, the vehicle enters a dark area and loses all means of communication with the RSU. As illustrated in Figure 3.1, under several circumstances, vehicles residing in dark areas of a roadway require to communicate with an RSU. Thus far, the literature encloses numerous work revolving around maintaining connectivity in vehicular environments (*e.g.* [4], [41]). In particular, the work of [41] addresses the optimal placement of RSUs along a highway in order to increase the connectivity of vehicles navigating along that highway. However, it is possible that the number of RSUs required to achieve such an objective may be large especially if long inter-city highways are considered. At this level, it is important to mention that, the cost of deploying a sufficient number of RSUs in order to fully connect a roadway is remarkably elevated [6]. This renders solutions such as the one proposed in [41] practically limited. Alternatively, establishing a connectivity path between faraway vehicles and the RSU is possible through the exploitation of cooperative relay vehicles as illustrated in Figure 3.1. This chapter is dedicated for the investigation of the necessary conditions to be satisfied for establishing a connectivity path between an isolated vehicle S and a far away RSU D . An analytical framework is established for the purpose of estimating the probability of the existence of a set of vehicles in tandem allowing for the establishment of a communication path between S and D . In reality, a vehicular network is not stationary with time. Consequently, the events which alter the network's topology are examined and their effect on a packet's average end-to-end delivery delay is considered. Finally, a mathematical study is conducted for examining the achieved multi-hop throughput.

The remainder of this chapter is organized as follows. Section 3.2 summarizes the

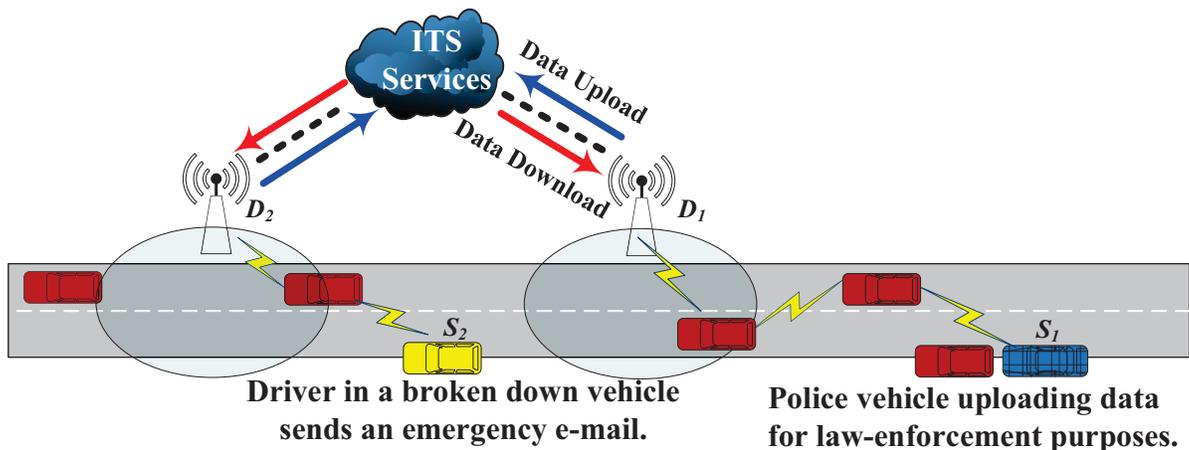


Figure 3.1: Connecting Vehicles in Dark Areas.

related work and distinguishes the work presented herein from the existing studies in the literature. Section 3.3 lays out a brief description of the considered multi-hop V2I communication scenario. In Section 3.4, the vehicular traffic model is laid out. The proposed path connectivity model is formulated in Section 3.5. The packet's end-to-end delivery delay analysis is laid out in Section 3.6. The multi-hop throughput analysis is conducted in Section 3.7. Results are reported in Section 3.8, and finally, concluding remarks are summarized in Section 3.9.

3.2 Related Work

3.2.1 Literature Review:

The authors of [42] presented a performance modelling of message dissemination in a vehicular network with two priority classes of traffic and accounted for the interference induced by hidden terminals in a multi-hop communication scenario. The authors assume a one-dimensional highway segment where vehicles are dispersed according to a Poisson process. Hence the nodes are uniformly distributed over the

considered highway segment. Once a message arrives to the VANET, it will be transmitted in the channel following a randomly chosen back-off counter, which is considered a part of the messages' exponential inter-arrival times. The transmission time of a message is assumed to follow the exponential distribution also. The highway segment is partitioned into activity and inactivity regions, and the total length of activity regions as well as interference sub-regions were presented given the total number of concurrent transmissions. Next, the authors determined the distribution of the number of concurrent transmissions of low priority messages in the network. Since the number of transmissions follows a Poisson distribution and their transmission times are exponentially distributed, then the number of transmissions was modelled as a birth-death process and the average number of concurrent transmissions was then calculated.

The work of [42] examined the probability that a destination node is in an interference region. This probability may be interpreted as the percentage of destination node population that cannot receive the disseminated message correctly due to interference. Hence, it was concluded that all nodes within the transmission range of a sender experience the same probability of being blocked, which turned out to be independent from their distances to the sending node. The authors of [42] also studied higher priority message dissemination using the contention-based forwarding technique.

In [43] an analytical model was established with the objectives of studying the effects of traffic flow, vehicles speeds and transmission range on network connectivity. Therein, queueing theoretical techniques were employed to study the connectivity between different vehicle pairs. The reported results showed that the network connectivity can be quite poor under sparse vehicular traffic conditions. In [44], it was observed that links in VANETs are established and disrupted unexpectedly. In fact,

this is due to the dynamic topology changes together with the short transmission ranges mandated by the FCC and the DSRC standards. In a similar context, the work of [45] revolved around three networking scenarios distinguished by considering three different vehicular densities mainly representing light, medium and heavy vehicular traffic. Under each density, a probabilistic analysis was conducted with the objective of deriving the distribution of the length of distance headway between consecutive vehicles along a highway.

The authors of [46] studied the steady-state statistical properties of continuous communication path availability in VANETs. The authors examined a network of highways characterized by arbitrary topologies where vehicles arrive to the considered network through any entry point following a Poisson process and move along a path according to a mobility model with a state-dependent mean speed. They highlighted the effects of mobility and traffic arrival process on the continuous communication path availability and packet delay. Particularly, their results showed that the mean durations of path availability and mean packet delay increased with the increasing transmission range.

In [47], a store-and-forward framework was proposed for VANETs with extra storage using buses and taxis as message carriers. Therein, the optimal link strategy was derived in terms of which types of vehicles to piggyback and how long the message should wait for them. The authors of [41] presented an analytical framework for the purpose of determining the maximum separation distance between adjacent RSUs in order to guarantee that the vehicle-to-RSU data delivery delay does not exceed a certain threshold. Motivated by the mathematical findings in [42] and the shortcomings of [41], the authors of [4] considered a roadway segment experiencing Free-flow traffic and examined the physical availability of a connectivity path between a far away vehicle S and a destination RSU D . Finally, in [48] a mathematical model was

presented with the objective of finding the packet delivery ratio when a vehicle forwards its data to another vehicle ahead of it. In that model, the author assumed that the packet undergoes a fixed number of V2V communication hops until it reaches its destination. The authors of [49] presented a theoretical model which studies the spatial propagation of information in a vehicular network, however, they did not consider any real-world communication aspect, (*e.g.* access protocol, bandwidth constraints).

3.2.2 Novel Contributions:

This chapter presents a stochastic model that aims at investigating the feasibility and availability of a connectivity path between a source vehicle S and a remote RSU D . The novel contributions are detailed as follow:

- Throughout the development of the analytical framework presented herein, the relatively restrictive assumptions made in [41] and [48] are relaxed. Precisely, first, the highway segment considered in this work is assumed to experience free-flow vehicular traffic conditions under which, vehicle densities are relatively higher than those considered in [41]. This, however, does not mitigate inter-vehicular connectivity intermittence. Instead, as opposed to the scenario investigated in [41], V2V communication is, herein, still partially feasible and has a relatively higher utility in linking completely isolated nodes to distant connected infrastructure. As a result, the assumption that only a single vehicle is present in an RSU's transmission range is completely eliminated.
- As opposed to [41] where vehicles were assumed to navigate at either one of a high or low speed, the free-flow traffic model (*i.e.*, [50]) is adopted herein where per-vehicle speed is a random variable with a known distribution.
- In [48], a packet was allowed to undergo only a constant number of V2V hops

until it reaches its destination. The mathematical framework developed in this work examines the distance travelled by a packet in a single hop and characterizes the number of experienced communication hops with technically, no upper bound limitations.

- As opposed to the work presented in [48], the traffic model adopted in this study eliminates the limiting assumption of a stationary network. Herein, the dynamic nature of vehicular networks together with the frequently alternating turn of random events leading to topology changes are carefully examined and considered throughout the derivation of the packet end-to-end delivery delay.
- Finally, realistic mobility traces were generated from SUMO [51], and used as mobility input for an event-driven simulator in order to evaluate the performance of our proposed model under real-world traffic conditions.

In what follows, a concise but yet comprehensive description of the system dynamics is presented and a mathematical framework is established for the purpose of:

1. Determining the availability of a connectivity path between a source vehicle and a destination RSU.
2. Evaluating a packet's end-to-end delivery delay as well as the multi-hop throughput of the proposed system.

Extensive simulations are conducted in order to legitimize the validity of the proposed model.

List Of Symbols

Symbol	Description
ρ	Vehicular density (veh/km)
λ_s	Vehicle arrival rate to the considered segment (veh/s)
d_{SD}	Distance between source vehicle S and RSU D (m)
N	Number of vehicles in d_{SD}
S	Inter-vehicular separation distance (m)
R	Vehicle's communication range (m)
W	Distance between consecutive vehicles within the same cluster (m)
Z	Length of a cluster (m)
C	Number of nodes in a cluster
G	Distance separating two consecutive clusters (m)
P_r	Probability that a vehicle has at least one relay vehicle within R
P_{SD}	Probability of an available V2V connectivity path between S and D
K	Number of hops a packet undergoes from S to D
d_h	Distance travelled by a packet in a single V2V hop
E_1	Event 1 when CLV_i approaches CTV_{i+1}
E_2	Event 2 where a faster vehicle overtakes CLV_i in cluster i
E_3	Event 3 where CLV_i enters D 's communication range
t_i^j	Average time required for event E_j to occur when the packet is carried by CLV_i
$V_{n,i}$	Speed of n^{th} vehicle in the i^{th} cluster
Δ_v	average difference in speeds between two vehicles
p_n	Probability that CLV is faster than the n^{th} vehicle in the next cluster
d^i	Distance between the vehicle carrying the packet and D
$T(d^i)$	Average end-to-end delay
τ	Transmission probability
P_b	Busy channel probability

P_c	Collision probability
r_i	Interference range
r_c	Carrier sensing range
N_c	Number of vehicles in carrier sensing range
P_{data}	Probability of data collision
N_D	Number of vehicles in data collision range
P_{ack}	Probability of ACK collision
N_A	Number of vehicles in ACK collision range
S_v	Single node throughput
P_t	Probability of a successful transmission
L_{data}	Length of transmitted packet
S_e	End-to-end per-vehicle throughput

3.3 Multi-Hop V2I Communication Scenario

This study considers a network model similar to the one illustrated in Figure 3.1, where a packet may undergo multiple V2V communication hops until eventually, it reaches a vehicle which will transmit it to the RSU within its communication range. Each vehicle is equipped with an OnBoard Unit (OBU) through which it communicates with other vehicles as well as RSUs using the WAVE protocol suite [52], according to which, the procedure to set-up a connection between two nodes of a vehicular network is simple as active scanning, association, and authentication procedures are no longer necessary. In fact, each node in a vehicular network (*i.e.*, a vehicle or a RSU) periodically broadcasts on the CCH announcement beacon messages that contain information identifying its offered applications (in the case of the RSU) and information about the speed, location and direction of travel (in the case of a vehicle). A vehicle monitors the CCH, coordinates with the RSU or neighbouring

vehicles and then simply switches to a SCH in order to establish a communication link. Herein, it is assumed that an arbitrary vehicle desires to establish a connection with either the RSU (if it is within its communication range) or another vehicle within its coverage range that will serve as an intermediate store-carry-and-forward vehicle. Note that, if a transmitting vehicle has more than one vehicle inside its communication range, it will select the farthest one as its potential next hop. In fact, a vehicle may select any in-range vehicle to forward data packets to. However, selecting the farthest vehicle allows the packet to travel a larger distance within a single hop. Transmitting vehicles will take advantage of the disseminated control information (here, other vehicles' respective locations) in order to determine which of the inrange vehicles is indeed the farthest one. It is worthwhile noting that, several studies have argued that selecting the farthest reachable vehicle results in the fastest data delivery in routing scenarios [53]. The case where the selected vehicle is unwilling to cooperate is outside the scope of this current study and is left for future work (discussed in Chapter 7.2).

Whenever a source or an intermediate vehicle has no reachable relay vehicles within its communication range, the packet waits in the queue of the carrying vehicle until a transmission opportunity arises, possibly as a result of a change in the vehicular network topology. The events leading to the vehicular network's topology changes are further discussed in Section 3.6.1.

3.4 Vehicular Traffic Model

Consider a multi-lane unidirectional roadway segment of length d_{SD} separating a source vehicle S and a destination RSU D . This segment operates under free-flow traffic conditions. According to [43], [42] and [50], vehicle arrivals to a particular lane l of the considered segment follow a Poisson process with parameter λ_l . Consequently,

the vehicles' arrival process to the entire segment is also a Poisson process with parameter $\lambda_s = \sum_{l=1}^L \lambda_l$, where L is the number of lanes ([54]).

The per-vehicle speeds are i.i.d. random variables in the range $[V_{\min}; V_{\max}]$. These speeds are drawn from a truncated Normal distribution with mean $E[V]$ and standard deviation σ_V . It is assumed that vehicles maintain their respective speeds constant during their entire navigation period over the considered roadway segment [42],[50]. As such, the number of vehicles N residing within the segment of length d_{SD} follows a Poisson distribution whose p.m.f. is given by:

$$f_{N|d_{SD}}(n) = \frac{(\rho d_{SD})^n}{n!} e^{-\rho d_{SD}}, \quad n \geq 0 \quad (3.1)$$

where $\rho = \frac{\lambda_s}{E[V]}$ is the vehicular density in vehicles per meter and $E[V]$ is the average vehicle speeds.

The inter-vehicular separation distance, denoted herein by S , was examined in [45] and [55], and both studies showed that S follows an exponential distribution whose p.d.f. is given by:

$$f_S(s) = \lambda_s e^{-\lambda_s s}, \quad s \geq 0 \quad (3.2)$$

where $\frac{1}{\lambda_s}$ is the average inter-vehicular distance.

Following the illustrative example presented in Figure 3.2, vehicles residing within the segment of length d_{SD} form several clusters. The distance between consecutive vehicles within the same cluster is smaller than the communication range R of a vehicle, and is denoted by W . The distance separating the first and last vehicle in a single cluster i is denoted by Z_i . And finally, the distance separating two consecutive clusters i and $i + 1$ is denoted by G_i . Let C denote the number of nodes that form a cluster. According to [42], C follows the geometric distribution and its p.m.f. is

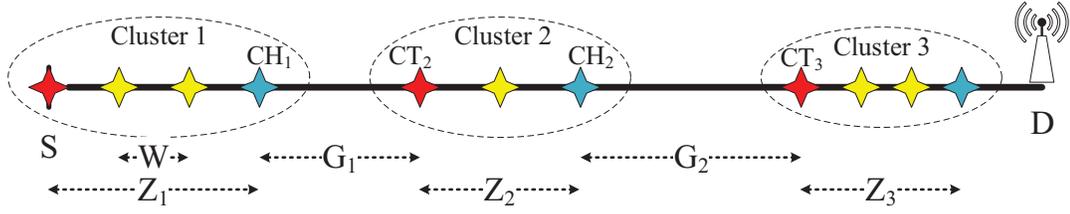


Figure 3.2: Vehicles Forming Clusters Along d_{SD} .

given by:

$$f_C(c) = P_e(1 - P_e)^c, \quad c \geq 0 \quad (3.3)$$

where $P_e = e^{-\lambda_s R}$. Consequently, the average number of nodes forming a cluster is $\bar{C} = (1 - P_e)/P_e$.

The intra-cluster distance, denoted by W , is the distance between two consecutive vehicles within the same cluster. Its p.d.f. $f_W(w)$ has been presented in [42] and is given by:

$$f_W(w) = f_S(w|w < R) = \frac{f_S(w)}{1 - P_e}, \quad 0 \leq w \leq R \quad (3.4)$$

The mean intra-cluster distance is therefore:

$$\bar{W} = \frac{1 - P_e(1 + \lambda_s R)}{(1 - P_e)\lambda_s} \quad (3.5)$$

Similarly, the inter-cluster distance, denoted by G , is the distance separating the farthest vehicle in one cluster (referred to herein Cluster Leader Vehicle (CLV)) and the closest upcoming vehicle in the next cluster (referred to herein as Cluster Tail Vehicle (CTV)). Its p.d.f. is $f_G(g)$. Following [56]:

$$f_G(g) = \frac{f_S(g)}{P_e}, \quad g > R \quad (3.6)$$

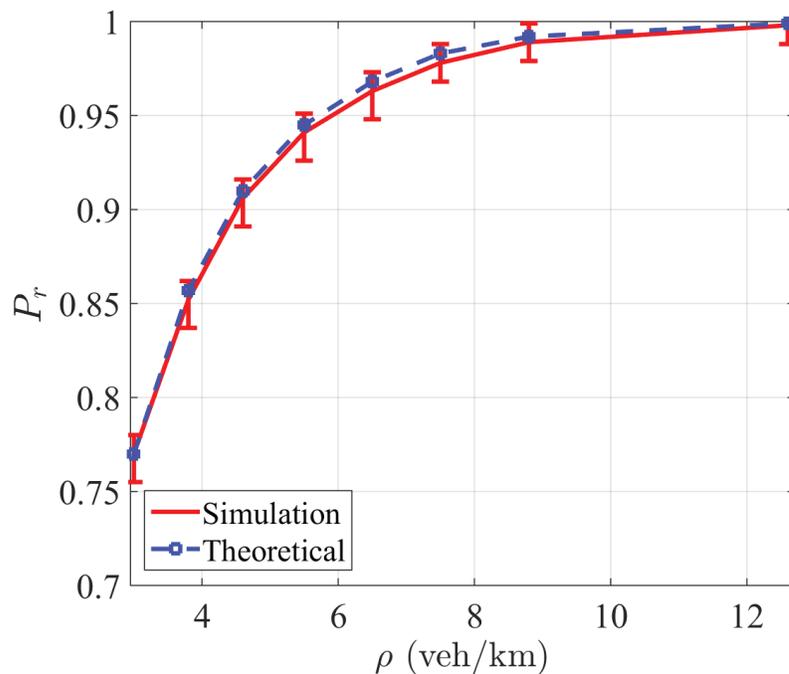


Figure 3.3: Probability of having a relay node on a per-hop basis.

The mean inter-cluster distance is given by:

$$\bar{G} = R + 1/\lambda_s \quad (3.7)$$

Finally, let Z denote the length of a cluster. Its average value \bar{Z} has been established in [42], and is given by:

$$\bar{Z} = \frac{1 - P_e(1 + \lambda_s R)}{\lambda_s P_e} \quad (3.8)$$

The properties of the considered vehicular traffic model have been characterized. Note that, the case where a transmitting vehicle wishes to exploit vehicles travelling in the opposite direction in order to establish a connectivity path is outside the scope of this study. In the next section, a stochastic model is developed to examine the feasibility of establishing an end-to-end path between S and D through intermediate relay nodes.

3.5 Modelling and Analysis of Path Availability

3.5.1 Availability of Relay Nodes:

Assume that all vehicles are equipped with identical wireless communication devices each of which having a transmission range of R meters. A vehicle i is able to relay packets to a subsequent vehicle along d_{SD} , say vehicle j , if the distance separating these two vehicles does not exceed R .

Let P_r denote the probability that a vehicle i has at least one vehicle within its range to which it can forward data packets. Note that, the vehicle receiving the data may not be i 's direct predecessor. Truly, that latter may be fast enough to pass slower vehicles and get out of i 's communication range. Hence, vehicle i may relay its data packet to any vehicle, which previously entered the highway segment and is still within i 's communication distance. Hence, the probability of having a relay vehicle within the communication range of vehicle i is given by the following equation:

$$P_r = 1 - \prod_{x=1}^{\infty} (1 - F_{Y_x}(R)) \quad (3.9)$$

where Y_x is an Erlang- x random variable which denotes the sum of x inter-vehicular distances. Its c.d.f. is given by [57]:

$$F_{Y_x}(y) = 1 - \sum_{i=0}^{x-1} \frac{1}{i!} e^{-\frac{y}{\lambda_s}} \left(-\frac{y}{\lambda_s} \right)^i \quad (3.10)$$

Equation 3.9 is verified in Figure 3.3 where the theoretical curve falls within the 95 % confidence interval of its simulated counterpart. The simulation framework is presented in details in Section 3.8. The results show that whenever the vehicular density ρ increases, the probability that a vehicle i has a relay vehicle within a distance of R meters increases as well. This is due to the fact that, as ρ increases, the average

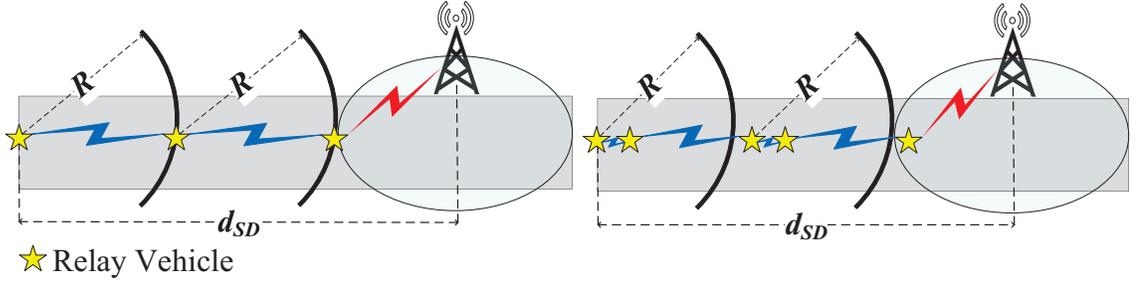


Figure 3.4: Min and Max Number of V2V Communication Hops

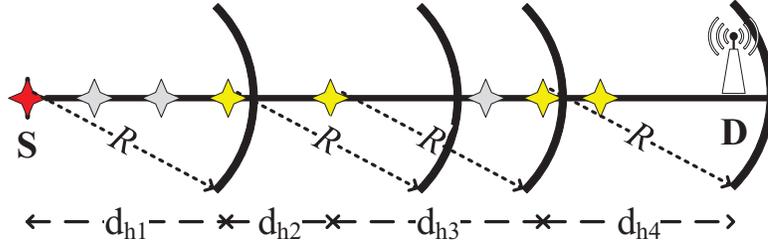


Figure 3.5: Distance Travelled by a Single Hop

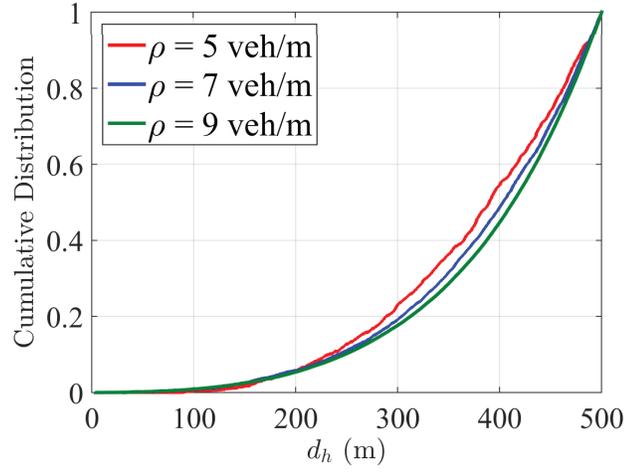
inter-vehicular distance decreases, and the probability that at least one vehicle is within a distance of R meters of vehicle i increases. Note that, a packet sent from S experiences several hops along its path until it reaches D . The availability of an end-to-end path between S and D is presented next.

3.5.2 Path Availability:

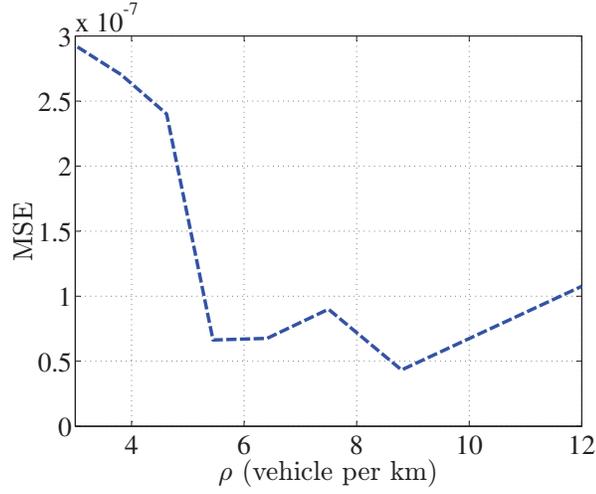
Let K denote the total number of hops a packet undergoes from the moment it is released from S until it reaches D . Denote by $f_K(k)$ the p.m.f. of K . The probability of having an available path from S to D is given by:

$$P_{SD} = \sum_{n=0}^{N_{max}} \sum_{k=K_{min}}^{K_{max}} \left\{ [P_r]^k f_K(k) \right\} \cdot f_{N|d_{SD}}(n) \quad (3.11)$$

where K_{min} and K_{max} are the minimum and maximum number of experienced V2V hops. According to Figure 3.4, a packet reaches its destination in K_{min} hops whenever



(a) $F_{d_h}(\delta)$.



(b) MSE for $F_Y(y)$.

Figure 3.6: Verification and validation of $F_Y(y)$ and $F_{d_h}(\delta)$.

it travels the transmission distance of a vehicle in a single hop. Consequently, $K_{min} = \lfloor \frac{d_{SD}}{R} \rfloor$. However, that packet may need a number of K_{max} hops to reach its destination whenever that same transmission distance is covered in two V2V hops, as illustrated in Figure 3.4. As such, $K_{max} = \lfloor \frac{2d_{SD}}{R} \rfloor$. Next, the analysis and characterization of K is presented.

3.5.3 Analysis of Number of Hops:

C1. Distance Travelled in a Single Hop:

Let d_h denote the distance travelled by a certain packet in a single hop. Since all vehicles are equipped with similar radio devices whose communication radius is equal to R , then $0 \leq d_h \leq R$. Recall from Section 3.3 that a source vehicle transmits the carried data to the farthest vehicle within its communication distance.

As illustrated in Figure 3.5, d_h is the length of the roadway segment a packet travels in a single V2V hop. Note that, the value of d_h is highly correlated to the vehicle selection process. To this end, it is important to highlight the fact that this process is not stationary. It varies from one hop to another and is also highly correlated to the time instant a packet is being transmitted as well as the vehicular traffic status at that time. Under such circumstances, the traditional approach in determining the statistical distribution of d_h appears to be remarkably complex. An attempt to work around this complexity consists of resorting to simulations for the purpose of collecting a large number of sample values of d_h and hence approximate its p.d.f. or c.d.f. based on these collected samples. The simulated c.d.f. of d_h for three different vehicular densities is shown in Figure 3.6(a). One can directly notice the diverging shape of the c.d.f. of d_h , which does not resemble any known distribution that may be used for approximation purposes. However, knowing that d_h is bound to the interval $[0; R]$, defining the random variable $Y = R - d_h$ seemed to be appropriate especially that the resulting c.d.f. of Y converges to 1 as Y increases, as illustrated in Figure 3.6(a).

Now, a closed form expression for the c.d.f. of Y shall be presented. A further examination of the collected samples of Y showed that the squared coefficient of variation of Y is larger than 1 for all collected samples under all considered Free-flow vehicular traffic densities. This indicated the feasibility of approximating the

p.d.f. of Y by a sum of two exponential distributions. As such, it is proposed (and justified) herein to approximate the p.d.f. of Y , $f_Y(y)$, by the sum of two exponential distributions whose parameters are determined using the Least-Squares-Fitting criterion. This approximation has the advantages of: *i*) being highly accurate for all investigated vehicular traffic densities and *ii*) presenting a relatively simple closed-form expression for $f_Y(y)$ which will serve the purpose of tractability of the mathematical analysis presented hereafter. The approximated p.d.f. of Y is:

$$f_Y(y) = \alpha e^{-\phi y} + \beta e^{-\psi y}, \quad 0 \leq y \leq R \quad (3.12)$$

where α, ϕ, β and ψ are the approximation parameters. It follows that the c.d.f. of Y , denoted by $F_Y(y)$, is:

$$F_Y(y) = \frac{\alpha}{\phi} + \frac{\beta}{\psi} - \left(\frac{\alpha}{\phi} e^{-\phi y} + \frac{\beta}{\psi} e^{-\psi y} \right) \quad (3.13)$$

The validity and accuracy of the adopted approximation is verified in Figure 3.6(b) where the Mean Squared Error (MSE) is computed between the simulated c.d.f. of Y and its theoretically approximated counterpart for different vehicular densities. As that figure shows, the largest MSE is of the order of 10^{-7} which is a tangible proof of the accuracy of Equation (3.13). Thorough analysis reveals that $F_Y(y)$ grows faster for increased vehicular densities; hence indicating that, as ρ increases, more vehicles are present within the communication range of the transmitting vehicle, and the probability of having one of them close to the edge of the communication range R increases. Consequently, the distance travelled by a single hop increases, and hence, $Y = R - d_h$ decreases. Note that:

$$F_Y(y) = Pr[R - d_h \leq y] = 1 - F_{d_h}(R - y) \quad (3.14)$$

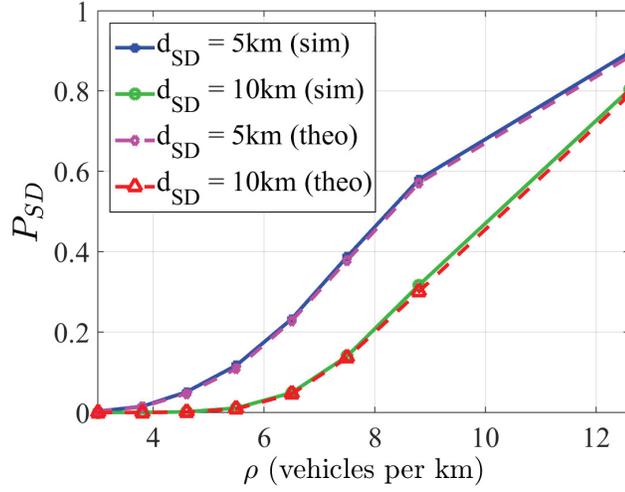


Figure 3.7: Probability of path availability.

As such, the c.d.f. of d_h is: $F_{d_h}(R - y) = 1 - F_Y(y)$.

Let $\delta = R - y$, then $F_{d_h}(\delta)$ is:

$$F_{d_h}(\delta) = 1 - \frac{\alpha}{\phi} - \frac{\beta}{\psi} + \left(\frac{\alpha}{\phi} e^{-\phi(R-\delta)} + \frac{\beta}{\psi} e^{-\psi(R-\delta)} \right) \quad (3.15)$$

As a result, the p.d.f. of d_h is given by:

$$f_{d_h}(\delta) = \alpha e^{-\phi(R-\delta)} + \beta e^{-\psi(R-\delta)}, \quad 0 \leq \delta \leq R \quad (3.16)$$

The expected value of d_h , denoted by $E[d_h]$ can be easily evaluated and is presented in the following equation:

$$E[d_h] = \frac{R\alpha}{\phi} + \frac{R\beta}{\psi} - \frac{\alpha}{\phi^2}(1 - e^{-\phi R}) - \frac{\beta}{\psi^2}(1 - e^{-\psi R}) \quad (3.17)$$

C2. Number of Experienced Hops:

Observe that, thus far, the only remaining unknown required to complete Equation (3.11) is $f_K(k)$. In this subsection, the characterization of $f_K(k)$ is investigated. The characterization of $f_K(k)$ relies on the average distance travelled within each of the

individual hops made by arriving packets to D . This latter average distance varies from one packet to another. Consequently, presenting a closed form expression for $f_K(k)$ is notably complex. For this purpose, numerical values of $f_K(k)$ are directly extracted from simulations. Hence, the obtained $f_K(k)$ can now be used with other theoretical expressions of $P_r(n)$ and $f_{N|d_{SD}}(n)$ in order to evaluate P_{SD} . In what follows, the resulting values of this evaluation are referred to as the theoretical values. The integrity of the adopted approach is validated in Figure 3.7 which concurrently plots the theoretical and simulation results for P_{SD} for two different values of d_{SD} . It is clear that, it is more likely to have an available path for the data to be relayed from S to D for smaller separation distance. This is especially true since, for $d_{SD} = 10$ km, a packet sent from S undergoes more V2V hops than that sent from a source 5 km away from the destination. Furthermore, Figure 3.7 indicates that the probability of having an available path from S to D increases as the vehicular density increases. This is due to the fact that a path is available if and only if there exists a relay vehicle for the source S as well as for each and every intermediate relay vehicle along d_{SD} , which is shown in Figure 3.3 and complies with the latter analysis.

After characterizing the probability of establishing a connectivity path between a vehicle S residing at a distance of d_{SD} from an RSU D , the next objective of this manuscript is to examine the delay experienced by a packet released from S until it reaches its destination D . An end-to-end delivery delay upper bound is presented in the next section.

3.6 Delay Analysis:

3.6.1 Preliminaries:

Figure 3.7 shows that a connectivity path may not always be available between a source vehicle S and a destination RSU D . For instance, under a vehicular density of 10 vehicles per km, a vehicle residing at a distance of 10 km away from RSU D has a chance of 45 % to have a connectivity path through intermediate vehicles to D . In fact, a connectivity path between S and D may be broken at multiple locations along d_{SD} as shown in Figure 3.2. Whenever a link is broken, a packet cannot be forwarded to the next vehicle, but instead, remains in the CLV's buffer, say CLV_i , until a packet transmission opportunity arises due to a change in the vehicular network topology. Given the highly dynamic nature of a vehicular network, numerous events may alter the topology of the network [58]. This study considers the events that lead to forwarding data from the carrying CLV to another vehicle ahead or to the destination RSU. These events are illustrated in Figure 3.8, and described as follows:

- Event E_1 illustrated in Figure 3.8(a): After $(t_1 - t_0)$ seconds, CLV_i approaches the CTV of the forthcoming cluster, CTV_{i+1} , such that the distance separating them becomes less than R . In this case, CLV_i transfers the packets it holds to the next cluster.
- Event E_2 illustrated in Figure 3.8(b): After $(t_1 - t_0)$ seconds, a faster vehicle passes the CLV_i and becomes the new CLV. Consequently, the new CLV receives the packets carried by the old CLV.
- Event E_3 illustrated in Figure 3.8(c): After $(t_1 - t_0)$ seconds, CLV_i enters D 's communication range and uploads the packets it is carrying.

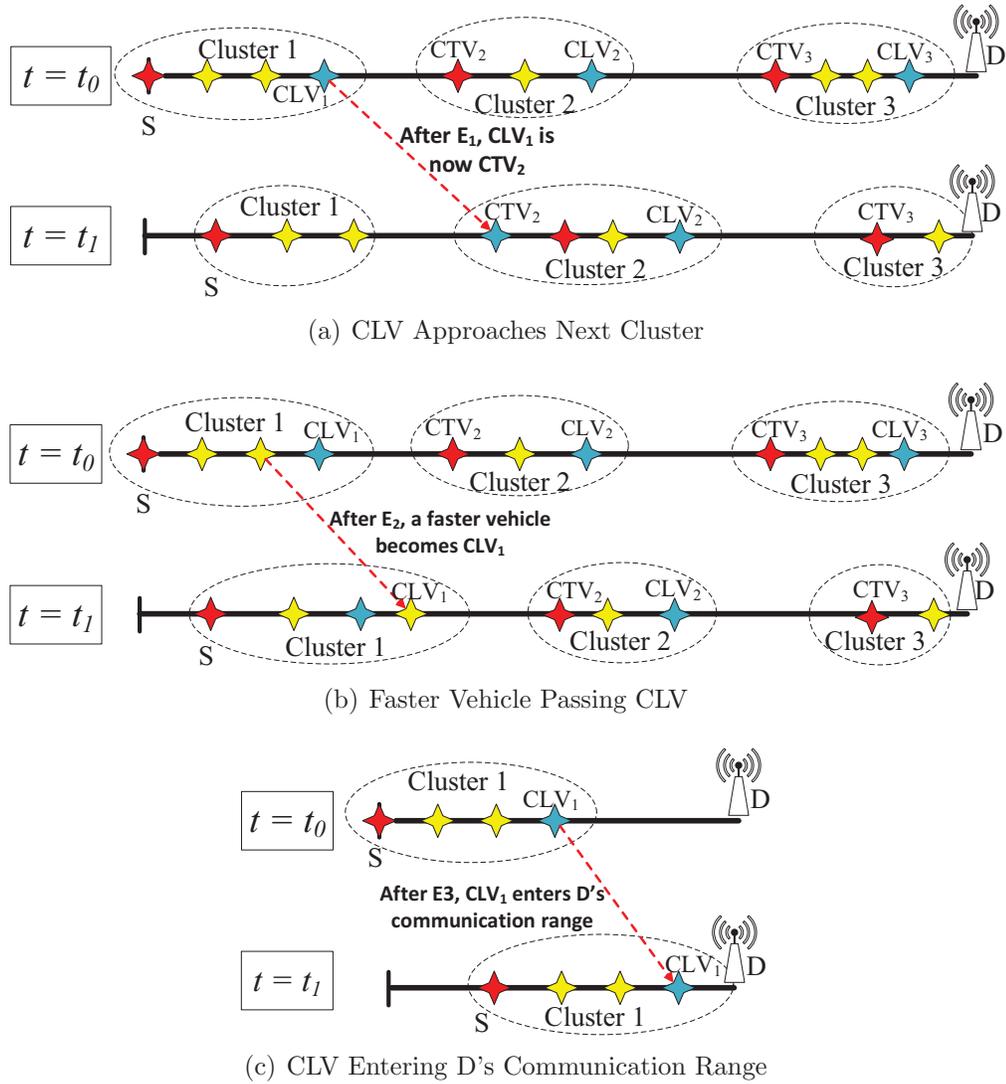


Figure 3.8: Events Changing the Vehicular Network's Topology

Note that the number of broken links along the considered segment of length d_{SD} is $B = \lfloor \frac{N}{C} \rfloor$, where N is the total number of vehicles in d_{SD} and C is the size of a formed cluster.

In this section, we are interested in the delay experienced by a packet released from a source vehicle S to a destination RSU D . That delay is composed of:

1. The time a packet spends in a CLV waiting for the occurrence of any of the

above-mentioned three events. Note that, according to [45], the network’s topology may change frequently under low vehicular densities, the occurrence of the events E_1 , E_2 or E_3 may take place within seconds.

2. The time needed for a packet to be transmitted between vehicles within the same cluster. Note that, once a connectivity link is established, a source vehicle may transmit a bulk of packets to the next intermediate vehicle, since the lifetime of that connectivity link is in the order of seconds [45]. Now, the time needed for a bulk of packets to be delivered to the next hop is negligible compared to the time a packet spends in a CLV waiting for a release opportunity. As a result, this delay component will be neglected when computing the packet end-to-end delivery delay.

In what follows, we assume that a packet arriving to an intermediate vehicle’s buffer will be transmitted first to the next relay vehicle. Therefore, the packet’s queueing delay is only the time a packet spends in a vehicle’s buffer until a transmission opportunity arises.

3.6.2 End-to-end Delivery Delay Upper Bound:

This section is dedicated to find a tight upper bound for the average time a packet needs until it is delivered to a destination RSU D knowing that it has been released from a source vehicle S at a distance d_{SD} from D . Note that, whenever a connectivity path is available between S and D , the packet’s end-to-end delivery delay is its transmission time over the radio, which is, as previously mentioned, neglected in this study. In this case, the delay is zero, which is, in fact, the packet’s end-to-end delivery delay lower bound.

As illustrated in Figure 3.2, a packet released from S or any one of the CTVs, say CTV_i travels an average distance of \bar{Z} and reaches CLV_i where it encounters

a link disconnection. Note that, a packet is expected to encounter B broken links throughout its journey from S to D . When a packet encounters a broken link, it waits for either of the events E_1 , E_2 or E_3 to occur. Note that, event E_1 happens first if CLV_i is faster than a vehicle in the forthcoming cluster and is able to decrease the distance separating them to a value smaller than R before it reaches D . Event E_2 happens first if CLV_i is passed by a faster vehicle preceding it before it enters the communication range of either D or the CTV of the next cluster, CTV_{i+1} . Event E_3 happens first if the time needed for CLV_i to arrive to D is not sufficient for a modification in the network topology given the speeds and locations of the vehicles residing within the considered highway segment of length d_{SD} .

Let t_j^i denote the average time needed for event E_j to occur when the packet is carried by CLV_i . Based on the above definitions of events E_1 , E_2 and E_3 , their respective incurred delays on the packet end-to-end delivery delay is presented next. Let Δ_v denote the relative average speed between two vehicles. According to [48], Δ_v follows the Normal distribution with mean 0 and variance $2\sigma_v^2$. Event E_1 occurs when the CLV vehicle carrying the packets approaches the CTV of the next cluster. Note that, the CTV of the next cluster, (*i.e.*, CTV_{i+1}), may have changed from the time the packet reaches CLV_i until the time that latter approaches the next cluster. This is especially true since, vehicles have different speeds, and the network topology is highly dynamic.

Let $V_{n,i}$ represent the speed of the n^{th} vehicle which is residing in the i^{th} cluster. Note that, $V_{1,1}$ is the source vehicle S , $V_{C,1}$ is the CLV of the first cluster whose size is C , $V_{1,2}$ is the CTV of the second cluster, and so forth. Furthermore, let p_n denote the probability that the CLV is faster than the n^{th} vehicle in the next cluster, and let $p_n' = (1 - p_n)$.

For CLV_i to be able to forward its packets to the n^{th} vehicle in the next cluster,

it has to travel an additional distance of $(n - 1)W + G - R$ compared to the next relay vehicle (recall that all the vehicles ahead of CLV_i are moving towards D with a certain speed). In order to cover a larger distance, CLV_i has to be faster than the vehicle it is approaching. Therefore, the time t_1^i needed for CLV_i to approach the forthcoming cluster, which contains C vehicles, is given by Equation (3.18).

$$\begin{aligned}
t_1^i &= \frac{G - R}{E[\Delta_v | \Delta_v > 0]} p_1 + \frac{W + G - R}{E[\Delta_v | \Delta_v > 0]} p_1' p_2 + \frac{2W + G - R}{E[\Delta_v | \Delta_v > 0]} p_1' p_2' p_3 + \\
&\dots \\
&+ \frac{(C - 1)W + G - R}{E[\Delta_v | \Delta_v > 0]} p_1' p_2' \dots p_{C-1}' p_C
\end{aligned} \tag{3.18}$$

According to [48], $p_n = 1/2$ for all values of n . Consequently, Equation (3.18) can be written as:

$$t_1^i = \frac{\sum_{x=1}^C [(x - 1)W + G - R] p_1^x}{E[\Delta_v | \Delta_v > 0]} \tag{3.19}$$

where $E[\Delta_v | \Delta_v > 0]$ is given by:

$$E[\Delta_v | \Delta_v > 0] = \int_{\Delta_v} v \frac{f_{\Delta_v}(v)}{Pr[\Delta_v > 0]} dv \tag{3.20}$$

where Δ_v is bounded to $[-(V_{max} - V_{min}); V_{max} - V_{min}]$.

The time needed for event E_2 to occur is determined next through similar analysis used for computing t_1^i . Note that any vehicle which belongs to the same cluster as the CLV may pass this latter and become the new CLV. Let q_n denote the probability that the n^{th} vehicle in the cluster is faster than the CLV. Similar to p_n , the probabilities q_n are identical and equal for all values of n . Therefore, the time needed for event E_2

to occur is:

$$t_2^i = \frac{\sum_{x=1}^{C-1} (C-x)Wq_1^{C-x}}{E[\Delta_v | \Delta_v > 0]} \quad (3.21)$$

Finally, the time needed for event E_3 to occur is presented. Let d^i denote the distance between the vehicle carrying the packet and the destination RSU D , where $i = 1, 2, \dots, B$ and $d^1 = d_{SD}$. t_3^i is the time required for vehicle CLV_i to travel a distance of $d^i - Z - R$. As such, CLV_i enters the communication range of RSU D and transfers the packets it is carrying. Consequently, t_3^i is given by:

$$t_3^i = \frac{d^i - Z - R}{E[V]} \quad (3.22)$$

Note that, if either of the events E_1 or E_2 takes place before E_3 , then the same delay analysis is repeated, however, at this point, the source vehicle becomes the vehicle carrying the packet, which now resides at a distance d^{i+1} from D .

$$d^{i+1} = \begin{cases} d^i - Z - (t_1^i \times E[V | \Delta_v > 0]) & \text{if } t_1^i < t_2^i \\ d^i - Z - (t_2^i \times E[V | \Delta_v > 0]) & \text{if } t_1^i > t_2^i \end{cases} \quad (3.23)$$

Finally, let $T(d^i)$ denote the average end-to-end packet delivery delay experienced by a packet released from a distance d^i from D , where $i = 1, 2, \dots, B$. It is given by:

$$T(d^i) = \begin{cases} t_1^i + T(d^{i+1}) & \text{if } t_1^i < (t_2^i, t_3^i) \\ t_2^i + T(d^{i+1}) & \text{if } t_2^i < (t_1^i, t_3^i) \\ t_3^i & \text{if } t_3^i < (t_1^i, t_2^i) \end{cases} \quad (3.24)$$

Equation (3.24) is a recursive equation which reflects the repeated analysis of the packet delivery delay only in the two cases where that packet is released from a CLV either to a passing faster vehicle or to the next cluster.

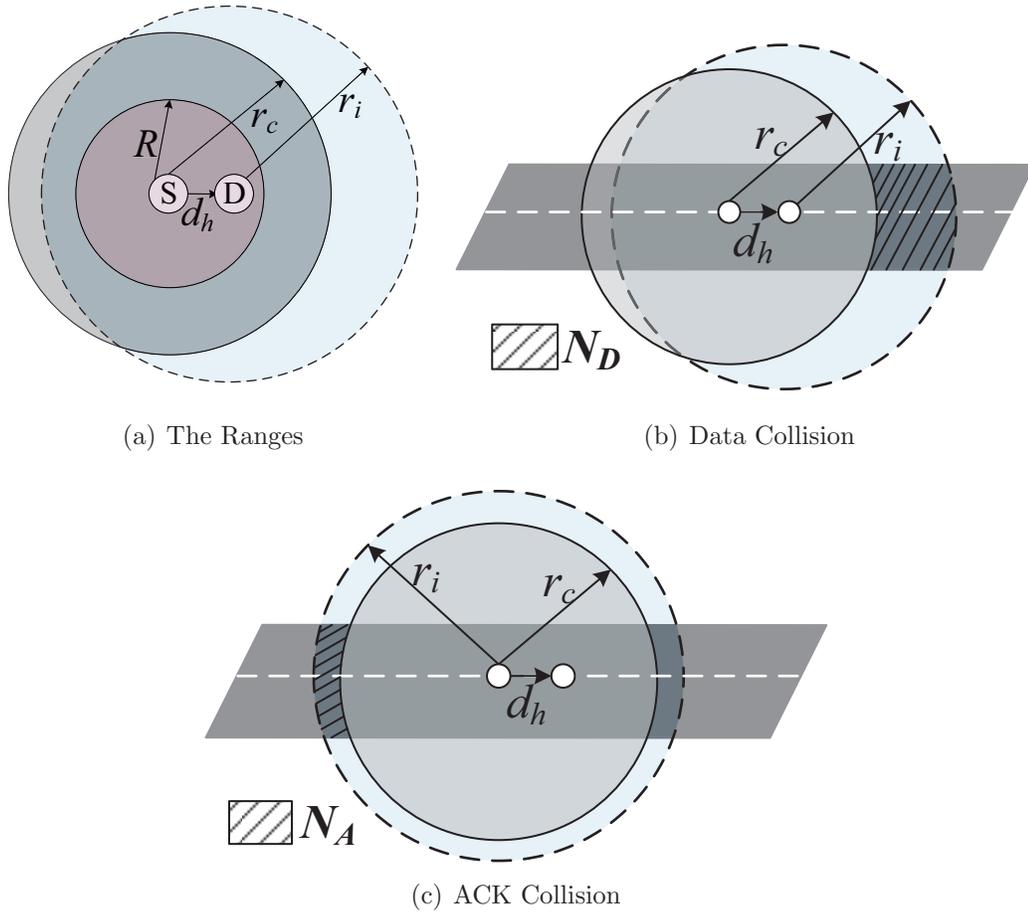


Figure 3.9: IEEE 802.11p Ranges and Collision Events.

3.7 Throughput Analysis

This section is dedicated to study the per-hop throughput and the per-vehicle end-to-end throughput. The conducted throughput analysis herein accounts for the respective probabilities of relay availability, P_r , and channel availability. Here, a vehicle transmits its data to the next vehicle in a random time slot for a duration of T_{avg} (*i.e.*, the expected time interval between the beginning instants of two consecutive slots, [59]). The access algorithm adopted herein is the IEEE 802.11p ([14]), where vehicles contend for channel access.

Let P_b denote the busy channel probability, which is the probability that a channel

is sensed busy at the beginning of a time slot. Note that, in our scenario, if a vehicle's backoff counter reaches zero, a vehicle initiates a transmission only if there exists a relay vehicle within its communication range. Using a 2-dimensional Markov Chain, the transmission probability τ can be easily calculated as a function of the busy channel probability P_b and the collision probability P_c .

3.7.1 Collision and Channel Busy Probabilities:

In order to characterize the collision and channel busy probabilities, P_c and P_b , the different ranges according to IEEE 802.11p have to be revisited. Figure 9(a) illustrates the transmission range R of a source vehicle, the carrier sensing range r_c and the interference range r_i . Recall that, in the connectivity model studied herein, d_h denotes the distance travelled by a packet in a single hop. A source vehicle S may transmit its data to any vehicle within a distance of R meters away from it. Whenever S is transmitting, all vehicles within a distance of r_c from S sense the medium busy, and hence defer their transmissions. The busy probability is given by the following equation:

$$P_b = 1 - (1 - \tau)^{N_C} \quad (3.25)$$

where $N_C = \rho r_c$ is the number of vehicles residing in the carrier sensing range of the transmitting vehicle.

During the data transmission from S to D , if a vehicle residing outside the carrier sensing range of S but within the interference range of D attempts to transmit, a collision will take place. This is known as the hidden terminal problem, which is accounted for herein. In this adopted communication model, two events may cause a collision:

1. Data collision caused by an attempted transmission by a vehicle in the shaded

area in Figure 9(b). The probability of data collision is given by:

$$P_{data} = 1 - (1 - \tau)^{N_D} \quad (3.26)$$

where N_D is the total number of vehicles present within the shaded area in Figure 9(b). It is given by:

$$N_D = \int_0^R \rho(\delta + r_i - r_c) f_{d_h}(\delta) d\delta \quad (3.27)$$

2. ACK collision that takes place when the receiving vehicle is sending an acknowledgement packet to the transmitting vehicle and a vehicle within the shaded area in Figure 9(c) attempts to transmit. The probability of an ACK collision, denoted by P_{ack} , is given by the following equation:

$$P_{ack} = 1 - (1 - \tau)^{N_A} \quad (3.28)$$

where N_A is the total number of vehicles present within the shaded area in Figure 9(c). It is given by:

$$N_A = \rho(r_i - r_c) \quad (3.29)$$

Consequently, the probability of a collision P_c is:

$$P_c = 1 - (1 - P_{data})(1 - P_{ack}) \quad (3.30)$$

3.7.2 Throughput Expression:

Let S_v denote the single node throughput defined as the number of payload bits that can be transmitted by a vehicle per unit time. S_v is calculated as follows:

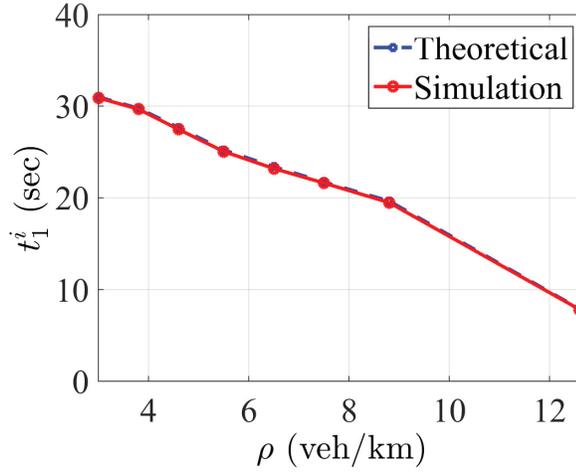
$$S_v = \frac{P_t L_{data}}{T_{avg}} \quad (3.31)$$

where $P_t = \tau(1 - P_c)$ is the probability of a successful transmission, L_{data} is the length of the transmitted packet. T_{avg} depends on σ , T_s and T_c which are borrowed from the IEEE 802.11p standard [14]. The end-to-end per-vehicle throughput can be expressed as $S_e = S_v \cdot P_{SD}$, where P_{SD} is given in Equation (3.11).

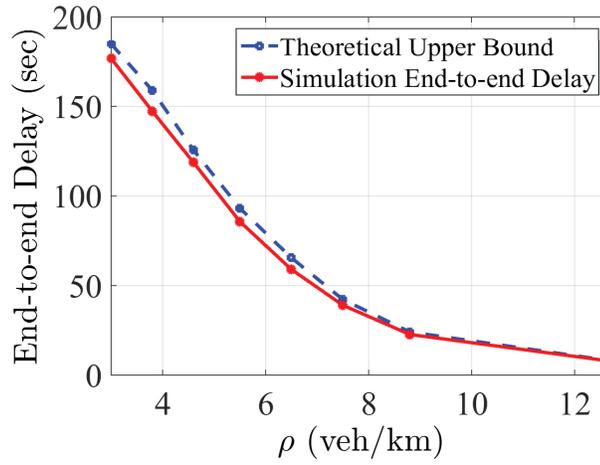
3.8 Numerical and Simulation Results

3.8.1 Simulation Setup:

The VEINS simulator [60] is used in order to validate the expressions established in Sections 3.4, 3.5 and 3.6. Vehicular nodes arrive at one unidirectional highway segment of length 30 km from one of three lanes. A transmitting vehicle forwards the carried packets to the next relay vehicle residing within its communication range until the packet arrives to its final destination, which is the RSU D . The simulator's input parameters are, namely: *a*) $\rho \in [3; 12]$ (veh/km), *b*) $R = 500$ (m), *c*) $r_i = 3.16R$ (*e.g.* [59]), *d*) $E[P] = 1000$ (bytes) and *e*) $R_c = 27$ (Mbps). All results reported herein were averaged out over five runs of the simulator, each of which spanned a time interval large enough to allow for the arrival and departure of 10^7 vehicles; hence ensuring that a confidence interval of 95% is realized.



(a) Time for E_1 .



(b) End-to-end Delivery Delay

Figure 3.10: Packet Delays.

3.8.2 Delay Simulation vs Theoretical Results:

Figure 3.10(a) plots the theoretical versus the simulation curves of t_1^i being the time needed for event E_1 to occur given that the packet is carried by CLV_i . This figure constitutes a tangible proof of the accuracy of Equation (3.19). It is clear from Figure 3.10(a) that the time required for a CLV to catch up with the next cluster decreases as the vehicular density increases. In fact, as ρ increases, the intervehicular distance as well as the inter-cluster distance decrease. Therefore, a CLV will have to

cover a smaller distance to forward the packet to the next cluster for larger vehicular densities. Consequently, as illustrated in Figure 3.10(a), t_1^i decreases as ρ increases. It is important to note that, Figure 3.10(a) proves that a broken communication link is re-established quickly under higher density conditions since more vehicles are present in a particular roadway segment, and their respective separation distance is small. On the other hand, it takes a lot more time for a broken communication link to be re-established under free-flow traffic conditions. Figure 3.10(b) plots the average packet end-to-end delivery delay obtained from simulations versus the established theoretical upper bound where the distance separating the source S and the destination D is 10 km. This result proves the validity of the Equation (3.24) since the simulated average packet end-to-end delivery delay does not exceed the theoretical upper bound laid out in Section 3.6 under all considered vehicular densities. Furthermore, as illustrated in Figure 3.10(b), a packet's average end-to-end delay decreases tremendously as the vehicular density increases. Recall that, under heavier traffic conditions, more vehicles are present within the segment of length d_{SD} , and thus, a packet encounters a smaller number of broken links throughout its journey from S to D . As a result, the time a packet spends waiting for a broken link to be re-established decreases, and hence, its total end-to-end delivery delay decreases as well. Figure 3.11 plots the packet end-to-end delivery delay versus d_{SD} under three vehicular traffic densities. Figure 3.11 shows that the packet's end to end delivery delay increases monotonically as the separation distance d_{SD} increases under a fixed vehicular density. This is expected since, as presented in Equation (3.24), under the same traffic conditions, the same delay analysis may be performed, independent from d_{SD} .

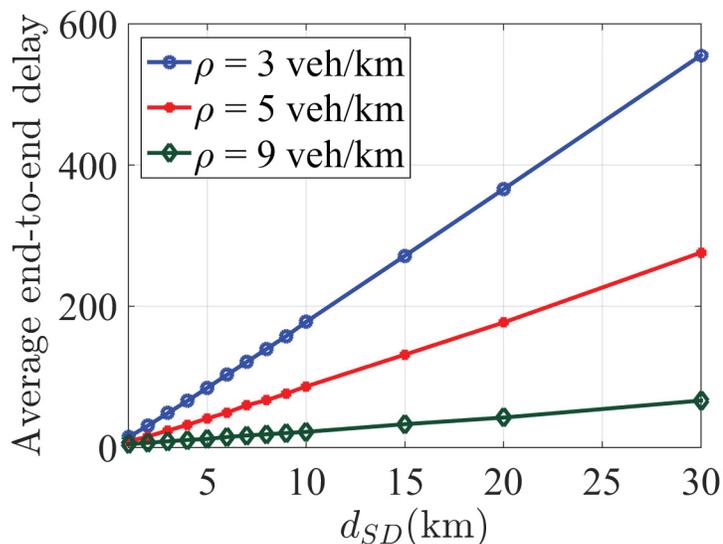


Figure 3.11: Delivery Delay Versus d_{SD}

3.8.3 Throughput Discussion:

Figure 3.12(a) plots the collision probability as a function of the vehicular density for two different carrier sensing ranges. P_c is directly proportional to ρ because whenever more vehicles are present within a certain highway segment, then more vehicles may initiate a transmission and hence increase the probability of a collision. However, as the carrier sensing range increases, less vehicles reside in the interference range of a transmission, and therefore, collisions are less probable. Figure 3.12(b) plots the per-vehicle throughput and shows that S_v drops as ρ increases regardless of the carrier sensing range. This is a direct consequence of the increased collision probability for higher values of vehicular density. However, notice that S_v decreases less significantly for larger carrier sensing ranges. The per-vehicle end-to-end throughput is illustrated in Figure 3.12(c). It is clear that the end-to-end throughput is null for a vehicular density less than 50 vehicles per km. This is due to that fact that, under such a low vehicular density, the existence of an available path from S to D is very limited, as illustrated in Figure 3.7. The end-to-end throughput increases as the probability of an available path increases for higher vehicular densities. However, S_e decreases

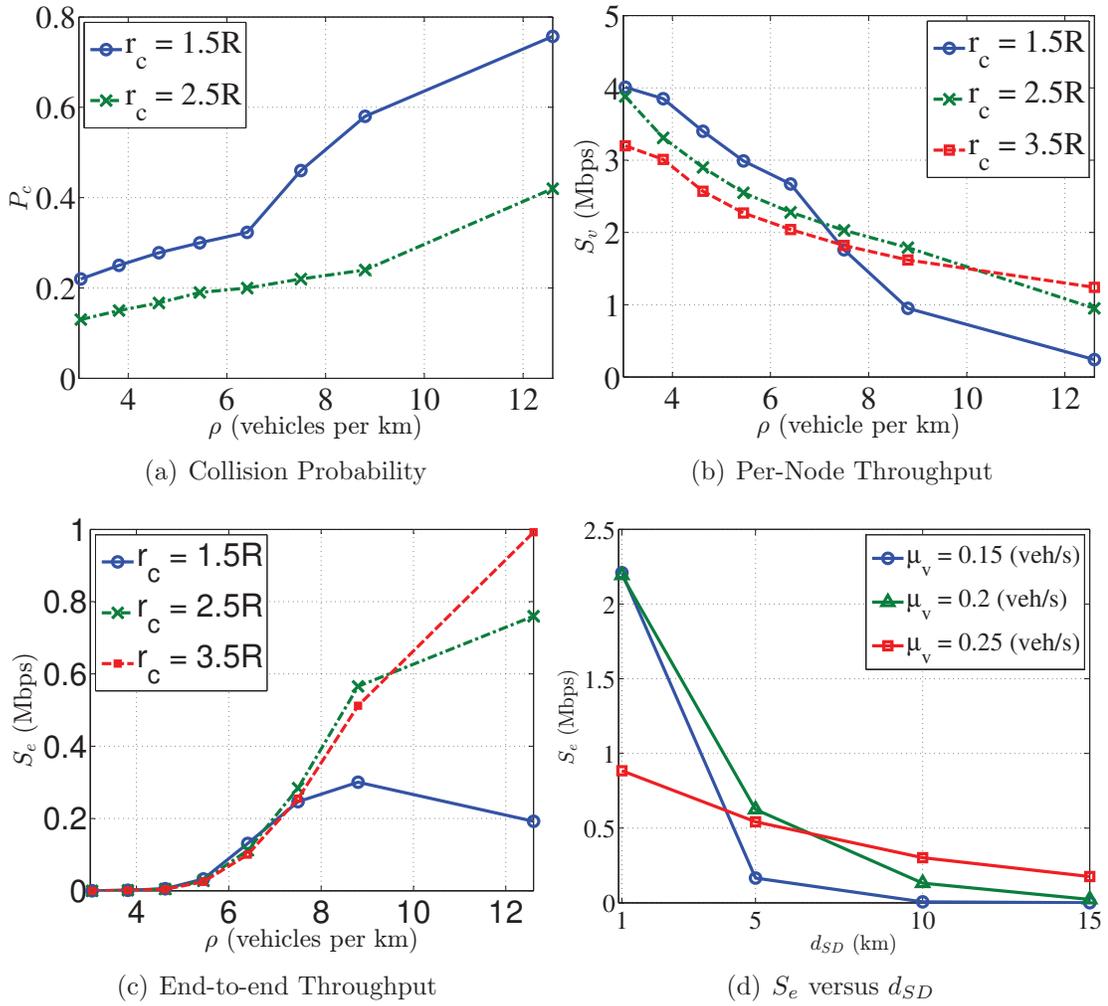


Figure 3.12: Collision Probability and Throughput for Different Vehicular Densities and Carrier Sensing Ranges

again for large vehicular densities as a result of the increased collision probability under such traffic conditions. Figure 3.12(d) plots the end-to-end throughput as a function of the distance separating the source vehicle S and the destination RSU D for three different vehicle arrival rates. This result clearly shows that the obtained end-to-end throughput decreases as S becomes farther from D . For small values of d_{SD} , the probability of having an available path P_{SD} is large, therefore, S_e records high values. However, even for such a small distance separating S and D , S_e is low for $\rho = 0.25$ vehicles per second. This is due to the large collision probability a transmitted packet is subject to under this high vehicular density. As d_{SD} increases, S_e decreases much faster for low vehicle arrival rates. This is due to the fact that, as d_{SD} increases, P_{SD} decreases, especially when μ_v and ρ are small. This shows that two major factors influence the end-to-end throughput, (P_c for small d_{SD} and P_{SD} for larger d_{SD}).

3.9 Conclusion

In this chapter, the probability of establishing a connectivity path between vehicles residing in dark areas of a roadway and a remote RSU is presented. In this context, this study examines the availability of intermediate vehicles serving as relay nodes. In addition, knowing that a packet undergoes multiple V2V hops in order to reach its final destination, the number of hops is characterized given the distance travelled by a packet on a per-hop basis. The conducted analyses were validated through extensive simulations. The results show that the probability of having an available path is directly proportional to the vehicular density, and inversely proportional to the distance between the source vehicle and the destination RSU. Next, the average end-to-end delivery delay was analysed after carefully examining the events that alter the network's topology. A tight theoretical upper bound was established and validated

through simulations. Also, a Markov framework is established for the purpose of evaluating the per-vehicle and network throughputs. Results show the significant impact of the path availability for large separation distances between source and destination as well as the collision probability under high vehicular densities.

According to [61], the ITS can play a significant role in offloading the dense cellular infrastructure. As such, the availability and reliability of a connectivity path between an arbitrary vehicle and the closest RSU becomes remarkably important and worth further investigation.

In the study considered herein, the connectivity path is established using multiple V2V hops where a single V2V link is formed between a vehicle and the farthest one within its communication range. Note that, such a link may be vulnerable given the dynamic topology changes of a vehicular network. Therefore, optimization methods may be exploited in order to choose the highest reliability/lifetime connectivity path between vehicles and distant RSUs.

Another interesting research idea arises as multiple RSUs are deployed in tandem along a roadway, each of which providing its own set of services. A vehicle now may choose not to transmit its packets to the closest RSU, but instead, attempts to adopt the best promising connectivity path in terms of availability and delay. Machine learning techniques present themselves as strong candidates which may be utilized in order to optimize the selection of the best available connectivity path.

Chapter 4

A Vehicle's Perspective of MAC Schemes

4.1 Introduction

V2I Communication is the wireless exchange of critical, safety, and operational data between vehicles and highway infrastructure, intended primarily to avoid accidents and enable a wide range of other safety, mobility, and environmental benefits. V2I communications have been rapidly advancing and paved their way to becoming among the fundamental contributors to transportation intelligence. Over the past few years, the performance of V2I communication systems has received significant attention. Indeed, the literature encloses numerous seminal publications revolving around the mathematical modelling and performance analysis of such systems.

This chapter's fundamental contributions can be summarized as follows:

1. The presentation of a modelling approach that differs from the existing work in the literature as it evaluates the performance of the V2I communication system as seen from the angle of any arbitrary vehicle residing within the coverage

range of a RSU. Observe that, by looking at the V2I scheme from the RSU's perspective, several events may occur and complicate the theoretical modelling of the RSU's buffer. For instance, and according to [62], vehicles' Service Requests (SRs) queue into the RSU's buffer until opportunities arise for them to be served. However, in that case, upon the departure of a vehicle from the RSU's range, all of its associated queueing SRs at the RSU will be discarded and any of its SRs receiving service will be subject to service force-termination. Given the elevated complexity of capturing the dynamics of a V2I system from the RSU's point of view, this study deviates to viewing the access scheme from the vehicle's perspective.

2. The proposal of two Medium Access Control (MAC) schemes, namely: *a*) Random Vehicle Selection (RVS) and *b*) Least Residual residence Time (LRT). Under RVS, a RSU will grant access to a single vehicle being uniformly selected among all of the vehicles present within that RSU's coverage range. The presentation of RVS herein has the purpose of clearly describing the channel assignment and access regulation mechanisms underlying a V2I communication system. In contrast, LRT implements vehicle prioritization based on the observation that faster vehicles will reside within the RSU's range for shorter periods of time than slower vehicles. Under such conditions, servicing the faster vehicles first is expected to increase bandwidth utilization efficiency and, hence, improve the system's throughput performance.
3. The development of mathematical frameworks leading to the formulation of single-server queueing models to represent the V2I system's operation and performance under both RVS and LRT. Together with their remarkable simplicity and the analytical tractability of their respective solutions, a distinguishing feature of these models is their ability to capture the system's dynamics from a

vehicle’s perspective with an accuracy that bypasses that of their much more complex existing counterparts. This is especially true since these models are built on top of a vehicular mobility model that accounts for fundamental macroscopic parameters characterized in Vehicular Traffic Engineering and Theory. In particular, these models lead to the development of closed-form solutions for: *a)* the per-vehicle throughput towards the time vehicles depart from the RSU’s range as well as *b)* the packet service time and *c)* the system’s response time.

The remainder of this chapter is organized as follows. Section 4.2 surveys a selection of the existing publications and highlights the major contributions that distinguish the work presented in this chapter from its existing counterparts. Section 4.3 lays out the traffic model adopted in this study. Section 4.4 presents a detailed description of two novel V2I MAC access methods. Section 4.5 is dedicated for the modelling and analysis of the vehicle’s OnBoard Unit Buffer’s (OBUB) queueing system under the two proposed MAC schemes. The performance evaluation results are reported in Section 4.6, and finally, concluding remarks are pointed out in Section 4.7.

4.2 Related Work and Contribution

4.2.1 Selective Literature Survey

The authors of [28] indicated that the DTI application of V2I communications suffered from a random access problem. Indeed, a close observation of a DTI system reveals the highly probable existence of multiple vehicles within the range of a RSU. This, together with the fact that more than one of these vehicles may simultaneously require Internet access gives rise to a joint random access and spectrum

allocation problem whose resolution is challenging. To this end, the authors developed the Dynamic Optimal Random Access (DORA) algorithm with the objective of maximizing the channel utilization subject to time-varying contention severity and capacity levels. DORA was developed in the context of a DTI scenario consisting of vehicles communicating with a single RSU. It incorporated self-incurred channel access fees and accounted for the Quality-of-Service (QoS) levels required by different applications. A finite-horizon Markov Decision Process (MDP)-based model was formulated to capture the system's dynamics as well as to evaluate its performance when operating under DORA.

In a similar scenario to [28], the work of [29] examined the V2I wireless access for streaming applications in a public transportation system. The authors formulated an optimization problem with the objective of providing a cost-minimal wireless connectivity that satisfies the end-users QoS requirements. For this purpose, a hierarchical optimization framework was established to determine an optimal policy indicating whether or not it is expedient for a vehicle to request bandwidth reservation from the RSU. The proposed mathematical model studied the system's performance variation as a function of the following factors: *a)* the streaming application's requirements, *b)* the vehicular mobility and *c)* the channel quality. The work in [29] considered both the user-centric and network-centric point of views to provide a unified model for optimizing wireless access in a V2I communication scenario.

The authors of [30] modelled the vehicular data download process using a series of transient Markov Reward Processes. Their objective was to characterize the distribution of a vehicle's downloaded data volume throughout its residence time within a RSU's range. The authors computed the influence of traffic density, vehicle speed, and RSUs transmission range on the amount of downloaded data.

In [31], the authors proposed a basic low-complexity V2I access scheme called $D*S$

where the RSU stored the Service Requests (SRs) and the request with the least $D*S$ was served first. D is the SR's deadline and S is the data size to be uploaded to the RSU. $D*S$ showed better performance when compared to three other access schemes namely *a)* FCFS, *b)* Earliest Deadline First, and *c)* Smallest Datasize First. The authors then worked on improving the performance of their proposed algorithm by using broadcasting techniques, and hence, serving more requests simultaneously. Furthermore, in an attempt to study the tradeoffs between the service ratio and the data quality, the authors extend their algorithm and propose a Two-Step scheme where two priority queues were used, *i.e.*, one for upload requests and the other for download requests. The results presented therein showed that the Two-Step scheduling scheme is adaptive to different workload scenarios. The authors then studied the uplink MAC performance of a DTI scenario in [32]. Both the contention nature of the uplink and the realistic traffic model were taken into consideration. An analytical framework was developed to quantify the uplink performance of DTI in an IEEE 802.11p environment in terms of packet collisions and uplink capacity. Furthermore, for the purpose of maintaining optimal system performance, the authors explored the adjustment of transmission power as a means of admission control by the roadside unit.

The work of [62] revolved around modelling the RSU as a multi-server queue whose customers are Service Requests (SRs) generated by newly incoming vehicles into the RSU's range. SRs will queue into the RSU's buffer until either they get served or they renege from the queue due to the departure of their initiating vehicles from the RSU's range. Upon the departure of a vehicle from the RSU's range, all of its associated SRs will be either discarded if they are still queueing in the RSU's buffer or force-terminated in case they are being served. The proposed complex model in [62] was partially simplified through approximations.

The authors of [63] aimed to improve the video quality and reduce its playback delay in a DTI network scenario. For this purpose, the authors propose a Selective Downlink Scheduling (SDS) algorithm whose main objective is to exploit information on the vehicles' positions in order to maximize the amount of data each vehicle receives before leaving the roadside unit's coverage. Their priority-based SDS algorithm is deployed at the roadside unit to coordinate the transmission of packets according to their importance, playback deadline as well as real-time information of vehicles such as the velocity, link quality and residence time within the range of the roadside unit.

Finally, To guarantee near-absolute service differentiation in VANETs supporting multimedia applications with different QoS requirements, a control-theoretic packet scheduling algorithm was proposed in [27]. This algorithm relies on the polling-based contention-free access method of the IEEE 802.11e standard and adapts resource allocation to many factors such as queue length and vehicle residence time.

4.2.2 Distinguishing Features of This Present Work

The proposed RVS scheme is a random access scheme that achieves a fair distribution of a RSU's bandwidth among all vehicles simultaneously present within that RSU's coverage range. However, as opposed to the study conducted in [32], the underlying RVS is a centralized channel allocation mechanism that allows the RSU to perform uniform vehicle selection and grant channel access to a single vehicle at a time resulting in contention-free data transmission for both download and upload. Hence, the amount of time wasted for channel contention under DORA is exploited for effective data transmission under RVS. This, obviously, allows RVS to expectedly outperform DORA in terms of throughput. The second MAC scheme proposed in this section, namely LRT, assigns vehicles different priority levels based on their residual residence time within the RSU's coverage range. In other words, a vehicle's

continuously decreasing residual residence time within the RSU’s range is interpreted, herein, as an increasing priority indicator. The vehicle with the highest priority (*i.e.* the shortest residual residence time) is served first. LRT boosts the system’s throughput performance beyond that of RVS especially under free-flow vehicular traffic where the proportion of fast vehicles with short residence times within the RSU’s range is higher than that of slow vehicles residing within the RSU’s range for larger time periods. Now, two queueing models are developed herein for the purpose of representing the DTI system operating under both RVS and LRT. A distinguishing feature of these models is their remarkable simplicity as compared to the complex optimization-based analytical frameworks developed in [28] and [29]; let alone the complex multi-server queueing model with reneging and force-termination developed in [62] and the dual priority queues adopted in [31]. Nonetheless, the presented models herein have the ability of accurately capturing the DTI system’s dynamics in a way that parallels those developed in [28, 29, 62]. Also, our mathematical analysis is more realistic than some of its earlier-published counterparts (*e.g.* [30]) as we relax one of [30]’s highly restrictive assumption which considers that all vehicles navigate at the same constant speed.

List Of Symbols

Symbol	Description
μ_v	Vehicle arrival rate to considered segment (veh/s)
v_i	Speed of an arriving vehicle i (m/s)
d_C	RSU’s communication area (m)
D_i	Vehicle i ’s residence time within d_C (s)
τ	Length of time slot (s)

J_i	Vehicle i 's discrete residence time within d_C
N_j	Number of vehicles present in d_C during j^{th} time slot
P_{n_j}	p.m.f. of number of vehicles n_j during j^{th} time slot
λ_p	Per-vehicle packet arrival rate
L_p	Packet length
R_D	Data rate
T_S	Packet service time
T_R	Packet response time
t_k	Beginning of k^{th} time slot
$T_i(t_k)$	Remaining residence time for vehicle i at time t_k
I_{ki}	Discrete time between the arrival of vehicle k and vehicle i
a_i	Time of vehicle i 's arrival
p_{ki}	Probability that vehicle k has a less remaining residence time than vehicle i
p_k	Probability that vehicle k has the least remaining residence time

4.3 Vehicular Traffic Model

This study adopts a discrete-time variant of the free-flow traffic model presented in [50]. That model examined the free-flow traffic conditions that were then characterized by homogeneous and uninterrupted light-to-medium vehicular traffic flowing over a one-dimensional roadway segment of fixed length. The inter-arrival time of vehicles under free-flow traffic conditions is exponentially distributed. As such, vehicle arrivals from a single lane follow a Poisson process. When multiple lanes are considered, the vehicle arrivals from each of these lanes follow independent and identically distributed Poisson processes. It follows that the overall vehicle arrival process from all lanes is the sum of i.i.d. Poisson processes, which is also a Poisson process with rate μ_v .

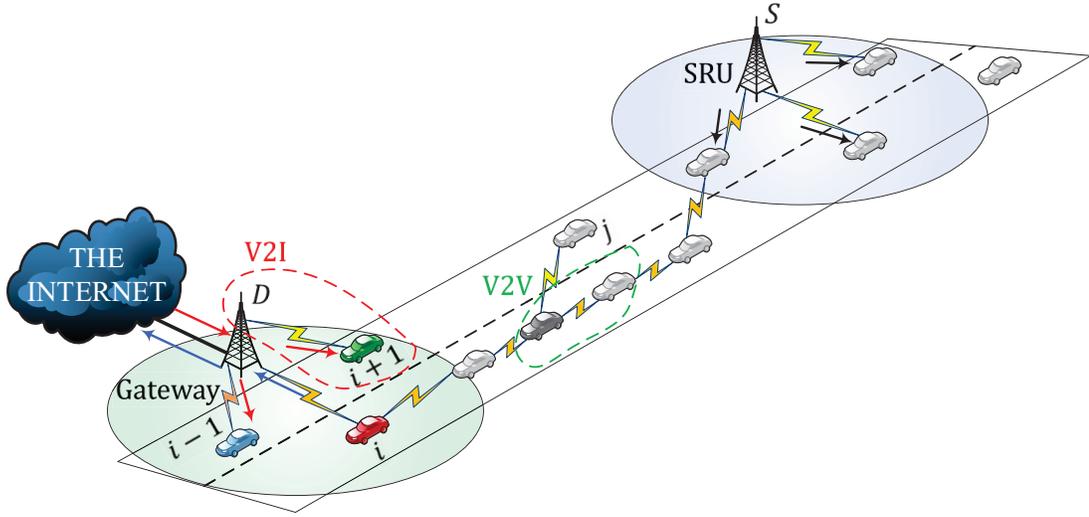


Figure 4.1: Vehicle-to-Infrastructure Communications in a Vehicular Network.

The speeds of arriving vehicles are independent and identically distributed in the range $[V_{\min}; V_{\max}]$. These speeds are drawn from a truncated Normal distribution having an average \bar{V} , a standard deviation σ_V . Moreover, it is assumed herein that, vehicles maintain their speeds constant during their entire navigation period in the RSU's communication range. As such, the residence time of arbitrary vehicle i with speed v_i within the range of G is a random variable defined as $D_i = \frac{d_C}{v_i}$ and whose probability density function $f_D(t)$ is expressed as:

$$f_D(t) = \frac{\xi d_C \sigma_V^{-1}}{t^2 \sqrt{2\pi}} \exp \left[- \left(\frac{\frac{d_C}{t} - \bar{V}}{\sigma_V \sqrt{2}} \right)^2 \right], \frac{d_C}{V_{\max}} \leq t \leq \frac{d_C}{V_{\min}} \quad (4.1)$$

where ξ is a normalization constant. In the vehicular traffic model presented herein, the time axis is subdivided into slots of length τ each.

Let J_i be the discrete version of the vehicle's residence within any RSU, and let d_C be the length of the segment that falls within the communication area of that RSU.

The p.m.f. of J_i has been derived in [64], and is given by:

$$f_{J_i}(j) = \frac{\xi}{2} \left[\operatorname{erf} \left(\frac{\frac{d_C}{j\tau} - \bar{V}}{\sigma_V \sqrt{2}} \right) - \operatorname{erf} \left(\frac{\frac{d_C}{(j-1)\tau} - \bar{V}}{\sigma_V \sqrt{2}} \right) \right] \quad (4.2)$$

where $J_{\min} \leq j \leq J_{\max}$.

Now, denote by N_j the number of vehicles present within the coverage range of G during time slot j . It has been established in [50] that the number of vehicles within a segment of finite length follows the Poisson distribution:

$$P_{n_j} = \Pr[N_j = n_j] = \frac{[\mu_v j \tau \cdot q(j)]^{n_j}}{n_j!} e^{-\mu_v j \tau \cdot q(j)} \quad (4.3)$$

where $n_j \in [0; N_{\max}]$, N_{\max} is the maximum vehicle capacity of the considered roadway segment and $q(j) = \frac{1}{j} \sum_{k=0}^j [1 - F_{J_i}(k)]$ and $F_{J_i}(k)$ is the cumulative distribution function (c.d.f.) of J_i .

4.4 Description of The Proposed V2I MAC Schemes

Figure 4.1 illustrates a V2I scenario where a RSU D is privileged with an Internet connection through minimal networking infrastructure. D has a coverage range that spans a segment of the road of length d_C along which it is deployed. Vehicles navigating along the road at different speeds enter D 's range at random times. An arriving vehicle wishing to transmit packets queueing at its OnBoard Unit's Buffer (OBUB) will announce to D its presence as well as its need for channel access through the transmission of an Access Request (AR) including its arrival time and speed. ARs are very small messages that are swiftly transmitted only once (*i.e.* upon arrival) with merely zero collision probability. Recall that time is subdivided into mini-slots. Hence, once aware of a vehicle's presence and need for channel access, the RSU D

computes and records that vehicle's initial residence time in terms of the number of mini-slots during which that vehicle will reside within its coverage range. With each time slot that passes, the vehicle's residence time is decremented by one until, eventually, the vehicle departs from D 's range. This is how the RSU keeps track of the vehicle's residual residence time within its range. Note that packets generated by the vehicles (respectively by the passengers commuting onboard using their smart mobile devices) remain buffered at these vehicles' respective OBUBs in anticipation of possible release opportunities to the RSU D . In turn, D may route the received packets over the Internet. Now, access provisioning will occur following the rules dictated by the adopted of either one of the below proposed MAC schemes.

4.4.1 Random Vehicle Selection Scheme

In the context of the above-described scenario, it is possible that multiple vehicles be simultaneously present within the range of D . Consequently, under the Random Vehicle Selection (RVS) scheme, the RSU will uniformly select only one of these vehicles during one time slot and grant that vehicle access to the communication channel during that time slot only. In the next time slot the RSU will perform another random selection and so forth. As such, all vehicles become equally likely to be selected and granted access to the channel. This ensures a fair distribution of the RSU's bandwidth among all present vehicles. Particularly, it is important to highlight that a vehicle, which has been selected during a certain time slot j may be reselected again with a certain probability during a time slot k where $k > j$ and provided that the vehicle still resides within the RSU's coverage range during time slot k . Note, however, that the number of vehicles residing within the range of a RSU is not fixed. As a matter of fact, some new vehicles may arrive and some other vehicles residing within the RSU's range may depart. Hence the probability of vehicle selection may

vary from one time slot to another.

4.4.2 Least Residual Time Scheme

In light of the adopted vehicular mobility model, an in-depth observation of the free-flow vehicular traffic's behaviour reveals a larger proportion of fast vehicles as compared to that of slow vehicles. Fast vehicles reside within d_C for a smaller period of time compared to slow vehicles. Therefore, servicing the faster vehicles first is expected to result in a better throughput performance. Motivated by this observation, the Least Residual residence Time (LRT) is proposed for prioritizing faster vehicles. As a matter of fact, as time slots are ticking, the decreasing vehicle residence time is interpreted, herein, as an increasing priority indicator instructing the RSU to select, among all vehicles present within its range, the vehicle with the highest priority (*i.e.* the least residual residence time) and hence grant channel access to that vehicle. Such a vehicle is repeatedly selected from one time slot to another for as long as it remains the highest priority vehicle. That is, the choice of vehicle is altered in two cases, namely: *a*) another much faster vehicle takes over or *b*) the currently selected vehicle departs from the RSU's range.

Finally, it is important to highlight that under both RVS and LRT the RSU G is the sole channel access arbitrator. As such, no contention will take place, and packet transmission will occur in a collision-free environment.

4.5 Modelling and Analysis of a Vehicle's OBUB Under RVS and LRT

4.5.1 Model Definition

Following the description laid out in Section 4.4, the OBUB of an arbitrary vehicle i witnesses the arrival of packets at random times. These packets queue at the OBUB and wait until they get released to the RSU D . Upon the occurrence of a packet release opportunity (*i.e.* D grants access to vehicle i), only a single packet (*i.e.* the packet occupying the Head-of-Line (HoL) position of the OBUB) is transmitted to D , which, in turn, routes that packet over the Internet.

A close observation of the above-described system dynamics reveals the possibility of developing a simple single-server queueing model to represent the OBUB of an arbitrary vehicle and characterize its performance. This proposed queueing model is, however, non-traditional especially that it differs from its classical counterparts (*e.g.* [57]) by the fact that the OBUB has no physical server. Consequently, in order to parallel traditional queueing models, the proposed model herein abstracts the existence of an imaginary server located at the front position of the OBUB. That is, a packet that reaches the HoL position of OBUB is considered as being admitted into service. That packet's service time is denoted by T_S , and defined as the amount of time the packet occupies the OBUB's HoL position before it gets released to the RSU D .

4.5.2 Basic Assumptions

The resolution of the OBUB model described in the previous section is founded on top of the following basic assumptions that have been borrowed from [62] and [22]:

- A1: Each vehicle is equipped with an infinite size OBUB.
- A2: The packet arrivals to each vehicle's OBUB follow a Poisson process with rate λ_p .
- A3: The packet length L_p is fixed.
- A4: The data rate R_D of a vehicle's OBU is constant.
- A5: Throughout its residence time within the RSU's range, a vehicle always has a packet to transmit.
- A6: Vehicle selection is performed independently from one time slot to another.

The next subsections are dedicated for the resolution of the described model under the two proposed MAC schemes, namely RVS and LRT.

4.5.3 RVS Model Resolution

Consider an arbitrary tagged vehicle i with P packets queueing at its OBUB. That vehicle's overall residence within the range of D will span a number of J_i time slots. Assume that, a packet P_z ($z > 0$) queueing within vehicle i 's OBUB has just been admitted into service (*i.e.* has just reached vehicle i 's OBUB's HoL position) at a certain time $t_j = (j - 1)\tau$ where $1 \leq j \leq J_i$. Note that, t_j represents the beginning of the j^{th} time slot. Recall from Section 4.4 that, since a vehicle which is granted access to the channel transmits only a single packet at a time during a single time slot, then the channel becomes available at the beginning of each time slot. Recall that, N_j denotes the total number of vehicles present within the range of D at time t_j . Then, under RVS, the RSU D uniformly selects one of these N_j vehicles and grants it access to the channel. It follows that, with a probability of $p_j = \frac{1}{N_j}$, vehicle i is granted access to the channel. If vehicle i has been selected to transmit, the service time

of packet P_z is $S_z = 1$ since that packet has been released to D immediately after being admitted into service, and it takes exactly one time slot to transmit a single packet. Otherwise, with a probability of $1 - p_j$, another vehicle, say vehicle k ($k \neq i$) is selected. In turn, that vehicle will transmit a single packet and return the channel to D for a subsequent selection process to take place at time $t_{j+1} = t_j + \tau$. However, at time t_{j+1} the number of vehicles residing within the range of D may have changed. As a matter of fact, some new vehicles may have arrived while some others may have departed. Let N_{j+1} denote the number of vehicles residing within the range of D at time t_{j+1} . Hence, D will uniformly grant access to one of these N_{j+1} vehicles. Under such conditions, with a probability of $p_{j+1} = \frac{1}{N_{k+1}}$, access is granted to vehicle i in which case the service time of packet P_z will be $S_z = 2$. Otherwise, with a probability of $1 - p_{j+1}$ any vehicle other than vehicle i is granted access and so forth. Without loss of generality, assume that vehicle i is granted access to transmit packet P_z at time $t_s = s\tau$ ($1 < s \leq J_i$). Consequently, the total number of time slots during which packet P_z occupies the HoL position of the OBUB before being released to the RSU D is $S_z = s$. Here, note that, right after the release of packet P_z , packet P_{z+1} would immediately advance one position to the front and, hence, be admitted into service. Then, the exact same packet release process described above will apply to packet P_{z+1} . As a matter of fact, this packet release process applies to all packets queueing at vehicle i 's OBUB. Hence, in the sequel, for generality purposes, the subscript z

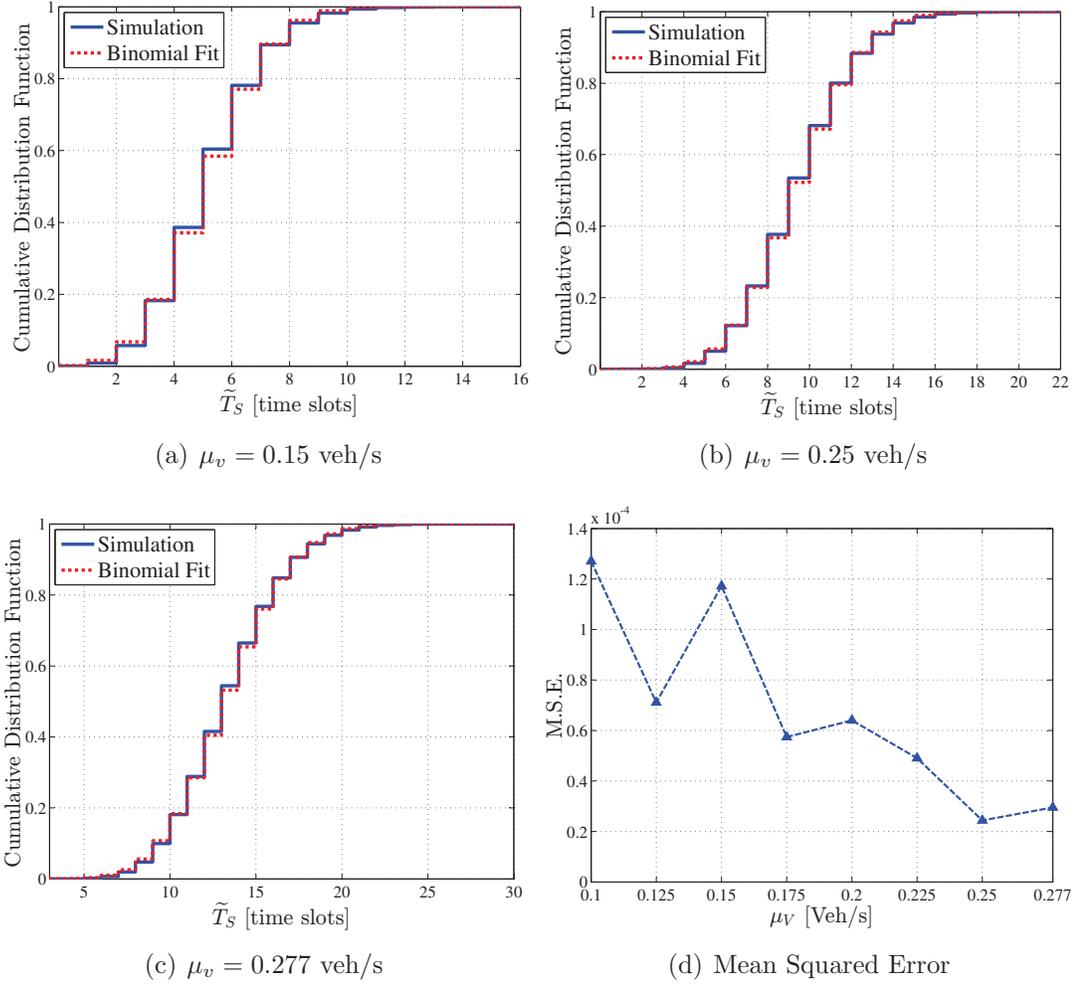


Figure 4.2: C.D.F of the packet service time for different vehicle flow rates.

will be dropped from S_z . The conditional p.m.f of S is given by:

$$g_{S|\{N_1, \dots, N_s\}}(s) = \Pr \left[S = s \mid N_1 = n_1, \dots, N_s = n_s \right] = \begin{cases} p_s & , s = 1 \\ p_s \prod_{j=1}^{s-1} (1 - p_j) & , s > 1 \\ 0 & , \text{otherwise} \end{cases} \quad (4.4)$$

where $p_j = \frac{1}{N_j}$ and N_j represents the number of vehicles present within the RSU's

coverage range during time slot j ($1 \leq j \leq s$). The packet service time p.m.f. unconditioned on $\{N_s, N_{s-1}, \dots, N_2, N_1\}$ is given in equation (4.5).

$$g_S(s) = \Pr[S = s] = \sum_{n_s=0}^{N_{\max}} \cdots \sum_{n_1=0}^{N_{\max}} g_{S|\{N_s, \dots, N_1\}}(s) \times \Pr[N_s = n_s, \dots, N_1 = n_1] \quad (4.5)$$

Denote by $G_S(\sigma) = \sum_{s=0}^{\sigma} g_S(s)$ the cumulative distribution function (c.d.f.) of S . Deriving a closed-form expression for $G_S(\sigma)$ is difficult and evaluating it numerically is complex and computationally exhaustive. This complexity stems from the fact that S strongly depends on the time slot K where a packet is admitted into service as well as the strong correlation between the number of vehicles in consecutive time slots. Fortunately, there exists a simple, yet highly accurate, approximation technique that allows for working around this problem. In particular, we resort to the collection of a large number of 10^8 samples of S in scenarios characterized by different values for μ_v that span the entire range of Free-flow vehicular traffic rates. Results show that $G_S(\sigma)$ may be approximated using a Binomial distribution $\tilde{G}_S(\sigma; N, p)$ whose parameters N and p can be easily computed numerically. Figures 4.2(a) through 4.2(c) concurrently plot the simulated version of the packet service time c.d.f. with its theoretical counterpart based on the presented model. Figure 4.2(d) plots the Mean Squared Error (MSE) between the theoretical foundation and the simulation results. Figures 4.2(a) through 4.2(d) constitute tangible proofs of the accuracy of the proposed model. This is especially true since the maximum MSE is of the order of 10^{-5} .

To this end, the approximated average packet service time is given by:

$$\tilde{T}_S = \tilde{S} \times \tau = \left[\sum_{\sigma=0}^{J_i} \left[1 - \tilde{G}_S(\sigma) \right] \right] \times \tau \quad (4.6)$$

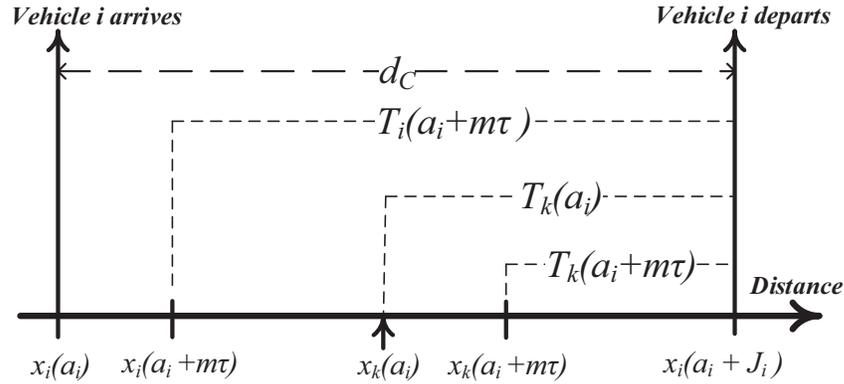


Figure 4.3: Instantaneous Position and Residual Times of Vehicles i and k .

where \tilde{S} denotes the approximated version of $E[S]$.

4.5.4 LRT Model Resolution

In what follows, the probability that an arbitrary vehicle, say i , has the least remaining residence time among all the vehicles within D 's range is first derived. Then, this probability is used through the characterization of the packet service time, and hence the system's response time.

Figure 4.3 illustrates an axis of distance on which the positions of two vehicles i and k are shown. The system is observed at the time of arrival of vehicle i to the RSU's range and, without loss of generality, this time is assumed to be time a_i . The position of vehicle i at that time is denoted by $x_i(a_i)$ and its residual residence time, in terms of time slots, is $T_i(a_i) = J_i$. Now, upon its arrival, vehicle i finds a number of vehicles residing within D 's coverage range. D has to select one of these vehicles and grant it access to the communication channel. For this purpose, the residual residence times for all vehicles within D 's range must be determined and compared. Suppose that, upon the arrival of vehicle i , the position of an arbitrary one of these vehicles, say vehicle k , is $x_k(a_i)$ as illustrated in Figure 4.3. This assumes that vehicle

k has arrived to d_C before vehicle i . Therefore, the residual residence time of vehicle k within D 's range can be expressed as $T_k(a_i) = J_k - I_{ki}$, where I_{ki} represents the discrete time between the arrival of vehicle k and the arrival of vehicle i . Note that, the number of slots delimited by the arrival of two consecutive vehicles is drawn from a geometric distribution. However, vehicle i may not be the vehicle arriving directly after vehicle k . In other words, there may have been several vehicle arrivals following the arrival of vehicle k but preceding the arrival of vehicle i . Consequently, the time that elapsed from the arrival of vehicle k until the arrival of vehicle i is:

$$I_{ki} = \sum_{x=k}^{i-1} I_{(x+1)x} = I_{(k+1)k} + I_{(k+2)(k+1)} + \cdots + I_{i(i-1)} \quad (4.7)$$

It has been proven in [50] that the number of time slots that elapse between the arrival of the two vehicles i and k is the sum of $k - i$ geometrically distributed random variables. As such, I_{ki} has a negative binomial distribution whose p.m.f. is given by:

$$f_I(l) = \binom{l-1}{k-i-1} p^{k-i-1} (1-p)^{l-k+i} \quad (4.8)$$

where $p = \mu_v \tau$ is the probability of vehicle arrival in a single time slot, and $l = k - i, k - i + 1, \dots$.

Up to this point, we have examined the system at the time of arrival of vehicle i and characterized the residual residence times of vehicles i and k as well as the time that elapsed between their arrivals to the segment of length d_C . Now, as illustrated in Figure 4.3, observe the system after $M = m$ time slots, and derive the probability that vehicle k has a smaller remaining residence time than that of vehicle i .

At $M = m$ time slots after its arrival, the residual residence time of vehicle i is $T_i(a_i + m\tau) = J_i - m$, while the residual residence time of vehicle k is $T_k(a_i + m\tau) = J_k - I_{ki} - m$.

Now, knowing I_{ki} and m , let p_c^k denote the conditional probability that vehicle k has a smaller remaining residence time than that of i . p_c^k is given by:

$$\begin{aligned}
p_c^k &= Pr[T_k(a_i + m\tau) < T_i(a_i + m\tau) | I_{ki} = l, M = m] \\
&= Pr[J_k - I_{ki} - m < J_i - m | I_{ki} = l, M = m] \\
&= Pr[J_k - I_{ki} < J_i | I_{ki} = l, M = m]
\end{aligned} \tag{4.9}$$

Observe that J_i , J_k and I_{ki} are independent of M . As such, Equation 4.9 reduces to:

$$p_c^k = Pr[J_k - I_{ki} < J_i | I_{ki} = l] \tag{4.10}$$

Let p_{ki} denote the absolute probability that, at the beginning of any time slot, vehicle k has a lower remaining residence time than that of vehicle i . Using Baye's theorem, p_{ki} can be expressed as:

$$\begin{aligned}
p_{ki} &= Pr[T_k(M) < T_i(M)] \\
&= Pr[J_k - I_{ki} - M < J_i - M] \\
&= Pr[J_k - I_{ki} < J_i] \\
&= \sum_{l=0}^{\infty} Pr[J_k - I_{ki} < J_i | I_{ki} = l] f_I(l)
\end{aligned} \tag{4.11}$$

Now, let p_k denote the probability that the selected vehicle k has the LRT among all the vehicles that are residing inside D 's range. Therefore, p_k is given by:

$$p_k = \sum_{i=0}^{\infty} p_{ki} \cdot Pr(N_i = n_i) \tag{4.12}$$

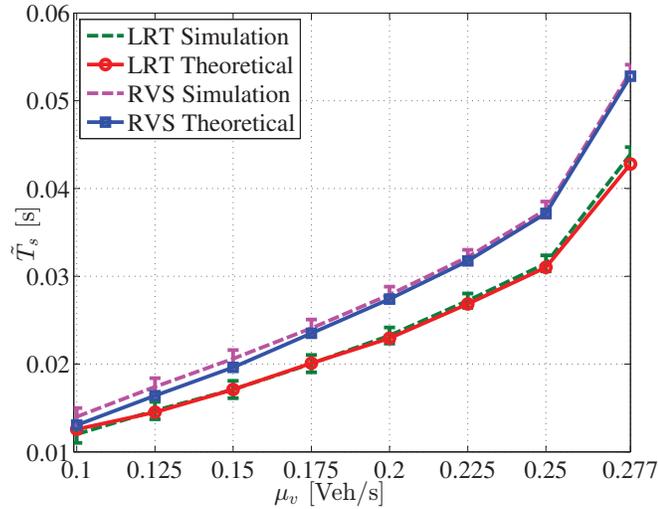


Figure 4.4: Theoretical mean packet service time of RVS and LRT plotted versus their respective simulated counterparts.

where $Pr(N_i = n_i)$ is the probability mass function of the number of vehicles within the segment of length d_C . p_k is the absolute probability that any arbitrarily chosen vehicle in d_C will be granted channel access in order to upload its HoL packet to D . Therefore, by knowing p_k , we may proceed to characterizing the performance metrics of the OBUB's queueing system.

4.5.5 OBUB's Queueing Model Characterization

In light of the above, the vehicle's OBUB can be represented using an $Geo(p)/G_S/1$ queueing model. Note that the characteristic parameters (*i.e.* the average customer queueing delay, the average number of customers in the system, etc.) are well known and have been fully derived in [57]. Nonetheless, the equations in [57] do not directly apply to the above-established model. This is especially true since, herein, the service position is the HoL position of the OBUB. It must be noted that a packet occupying that position is considered as being a buffered packet. Consequently, the average packet's service time is a factor contributing to the overall response time (*i.e.* the average amount of time a packet is buffered at the OBUB before being released to

the RSU D). The average response time is therefore given by:

$$\tilde{T}_R = \tilde{T}_S + \frac{\lambda_p S^2 \tau^2}{2(1 - \lambda_p T_S)} \quad (4.13)$$

where S^2 denotes the second moment of S . Finally, the per-vehicle-average throughput is given by:

$$T_p^v = \frac{L_p}{\tilde{T}_S} \quad (4.14)$$

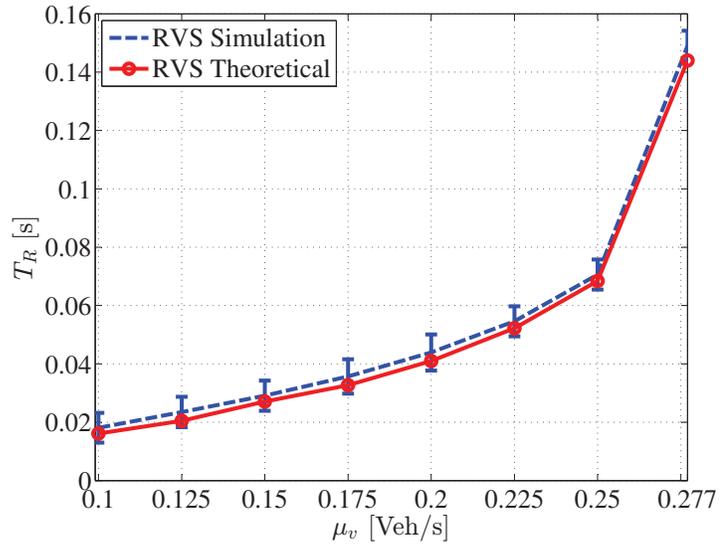
where L_p is the packet length in bits.

4.6 Simulations and Numerical Analysis

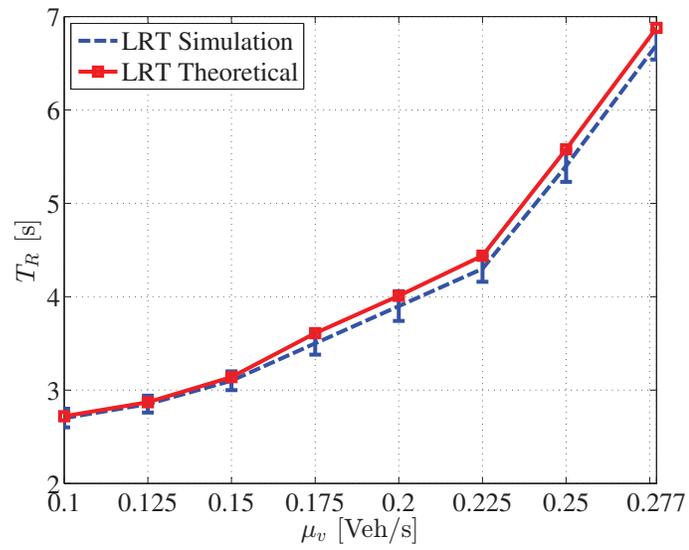
A discrete-event simulator was developed for the purpose of validating the proposed model in Section 4.5 and evaluate the performance of RVS in terms of: *a*) the average packet service time, *b*) the average response time and *c*) the average per-vehicle throughput. The above-listed performance metrics were evaluated for a total of 10^6 vehicles and averaged out over multiple runs of the simulator to ensure that a 95% confidence interval is realized. The simulator's input parameters are as follows: *a*) $\lambda_p = 10$ (pkts/s), *b*) $P = 12000$ (bits), *c*) $D_R = 3$ (Mbps), *d*) $\mu_v \in [0.1; 0.277]$ (veh/s), *e*) $V_{\min} = 2.78$ (m/s), *f*) $V_{\max} = 50$ (m/s), and *g*) $d_C = 1000$ (m).

Figure 4.4 plots the theoretical curves of the average packet service times T_S together with their simulated counterparts, under both RVS and LRT selection algorithms. This figure constitutes a tangible proof of the validity of the theoretical model presented in Section 4.5 as well as the accuracy of the adopted approximations. This is especially true given the almost perfect match between the theoretical and simulated curves.

Furthermore, Figure 4.4 shows that, under both selection methods, the average

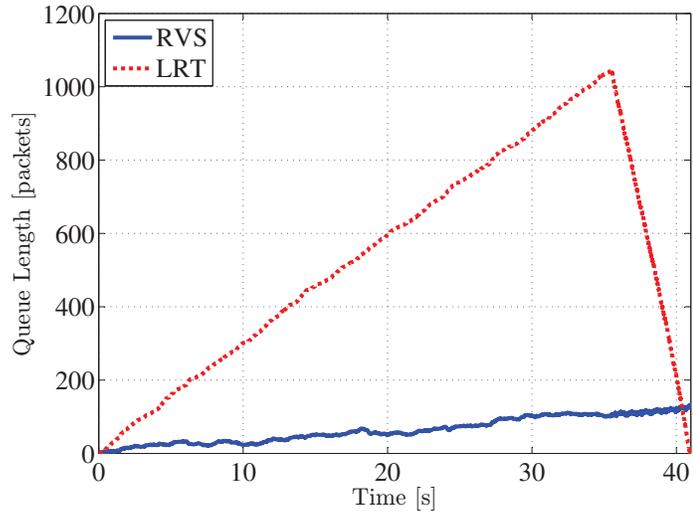


(a) Under RVS

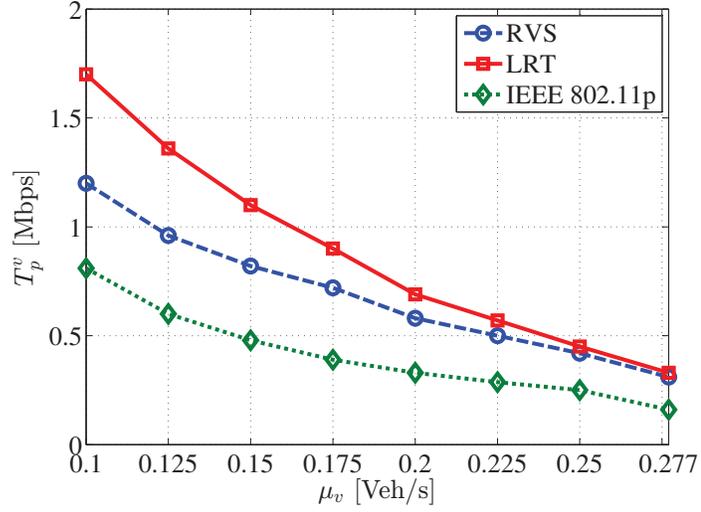


(b) Under LRT

Figure 4.5: System's Response Time.



(a) OBUB's Queue Length



(b) Mean Per-Vehicle Throughput

Figure 4.6: Queue Lengths and Throughput Comparison

packet service time, T_S , increases as a function of the vehicle flow-rate, μ_v . In fact, an increase in μ_v is accompanied by a decrease in the average vehicle speed and, hence, an increase in the vehicular density over the considered roadway segment. This result was obtained and well verified in [50]: the number of vehicles within the range of the RSU will increase as μ_v increases. Given that, under RVS, the RSU uniformly grants channel access to vehicles residing within its range, the likelihood of selecting a certain tagged vehicle will decrease when μ_v increases and more vehicles are present within D 's range. Consequently, a packet occupying the HoL position of that vehicle's OBUB will experience a larger service period. On the other hand, under LRT, as μ_v increases and more vehicles are present inside D 's range, the probability that a certain vehicle has the lowest residual residence time decreases. As a result, the packet residing at the HoL position will wait a larger amount of time before being released to D , and hence, the average packet service time increases.

Figure 4.4 also indicates that a RSU deploying the LRT selection method results in lower average packet service times in comparison with the RVS scheme. This is due to the fact that, under LRT, once a vehicle i has the lowest residual residence time, it gains channel access and keeps on transmitting its packets until either: *a*) a faster vehicle, say j , now has a lower residual residence time, *b*) it completely empties its buffer or *c*) it goes out of range. In all of these cases, there exists a large number of packets that will be released to the RSU as soon as they reach the HoL position of the queue. Hence, their recorded service time is, in this case, zero.

Figures 4.5 reveal that the average system response time, T_R , under both selection algorithms, is also an increasing function of μ_v . Surprisingly, the results came out different than what was expected. This is especially since one may intuitively believe that the decrease in the average vehicle speeds accompanying the increase in μ_v will cause vehicles to spend a larger amount of time within the RSU's range. Consequently,

a particularly tagged vehicle is expected to be able to clear out a larger amount of packets from its OBUB and, hence, following *Little's Theorem* (refer to [57]), the system's response time (*i.e.* the average packet queueing delay) will decrease. Although the latter argument is true, it is in reality being opposed by the decreased probability that a vehicle is selected and granted access to the channel. This follows directly from the increased number of vehicles residing within the range of the RSU as a function of μ_v ; hence, an increase in T_S . Knowing that T_S is an integral part of T_R , it directly affects T_R . As a result, T_R increases.

When comparing Figures 4.5(a) and 4.5(b), one will directly notice the substantial difference in the system's response time under both selection algorithms. Truly, under LRT, the system's response time is roughly 30 times slower than that under RVS. Intuitively, one would expect that, under LRT, the obtained lower T_S shown in Figure 4.4 will also lead to an improvement in T_R . Surprisingly, as indicated in Figure 4.5, this is not the case. The figure shows that RVS evidently outperforms LRT in terms of T_R . This could be well understood by looking at Figure 4.6(a). Recall that, under RVS, all vehicles present within the RSU's range are equally likely to be selected at all times. As such, the number of packets that accumulate within a particular vehicle's buffer will remain relatively small. In contrast, under LRT, the RSU may not immediately grant access to a newly arriving vehicle unless that vehicle is much faster than all its predecessors and hence will exhibit the shortest residence time. Accordingly, from the time a vehicle arrives until the time it gets prioritized, packets will be accumulating inside its buffer, which is evident in Figure 4.6(a). As such, the LRT version of T_R will increase much more rapidly than its RVS counterpart. However, once this particular vehicle holds access to the channel, it transmits a large number of packets from its queue up until either it no longer has the least remaining residence time in range of D , or it departs from D 's communication range.

The per-vehicle throughput performance of the two proposed schemes, RVS and LRT, was compared with that achieved under the IEEE 802.11p standard in order to highlight the contribution of this current work. Figure 4.6(b) shows that both proposed schemes herein outperform the IEEE 802.11p standard in terms of the achieved per-vehicle throughput. The WAVE protocol implements a contention-based access algorithm where a collision occurs when two or more vehicles attempt to upload their packets concurrently, which results in throughput deterioration. Under RVS and LRT, only the vehicle selected by the RSU may upload its packets, hence collisions are eliminated.

Figure 4.6(b) also shows that the LRT selection method outperforms its RVS counterpart in terms of T_p^v . This is due to fact that, under LRT, once a vehicle has the lowest remaining residence time among all the vehicles in d_C , it may have sufficient time to clear out a large number of its buffering packets. This results in a larger per-vehicle throughput than that recorded under the random selection scheme.

4.7 Concluding Remarks

This chapter presented two novel channel access schemes for V2I communications. Under both selection algorithms, a vehicle's OBUB was modelled as a $Geo(p)/G_S/1$ queueing system, which is characterized by a packet service time that has a general distribution. A simulation framework was established to verify the proposed models' validity, evaluate their performance, and finally, compare them in terms of the average packet service time, average system response time as well as the per-vehicle throughput. Results show that LRT outperforms RVS in terms of average service time. However, under LRT, packets experienced larger queueing delays. As a result LRT performed worse than RVS in terms of the system's response time. Finally, it is worthwhile noting that both proposed access schemes performed better than the

IEEE 802.11p standard in terms of the per-vehicle throughput.

It is clear that the RVS algorithm ensures fair access to the vehicles residing within the range of the RSU, whereas LRT achieves a higher per-vehicle throughput. Consequently, it is worthwhile noting that the use of RVS is efficient whenever the application requires fair access to vehicles such as music download or email, whereas LRT may be used for applications requiring higher throughput such as VOIP applications.

As the V2I communication systems have attracted increasingly more attention over the past years, this work presents a novel modelling perspective to evaluate the performance of MAC schemes in a vehicular environment. This work establishes a solid background for future research studies that wishes to optimize V2I communications from a vehicle's perspective. Service providers are encouraged to take into consideration the vehicle's interpretation of MAC schemes in order to properly plan and dimension their service infrastructure and offerings and provide a consistent user experience.

Chapter 5

Optimizing Downlink Traffic

Scheduling - The Single RSU case

5.1 Introduction

The road traffic crashes and consequent injuries and fatalities, traditionally regarded as random and unavoidable accidents, are recently recognized as a preventable public health problem. Indeed, as more countries (*e.g.* USA and Canada) are taking remarkable measures to improve their road safety situation, the downward trend for the number of fatalities and serious injuries due to motor vehicle crashes continues, dropping between 7 and 10% yearly between 2010 and 2014 [65]. Certainly, VANET communications offer safety related services such as road accident alerting, traffic jam broadcast, and road condition warnings. However, on the other hand, through V2I communications, mobile users are able to obtain a number of non-safety Internet services such as web browsing, video streaming, file downloading, and online gaming. As such, a multi-objective RSU scheduling problem arises whose aim is to meet the diverse QoS requirements of various non-safety applications while preserving a safe driving environment. At this point, it is important to mention that the unavailability

of a power-grid connection and the highly elevated cost of equipping RSUs with a permanent power source set a crucial barrier to the operation of a vehicular network.

Indeed, it has been reported that energy consumption of mobile networks is growing at a staggering rate [66]. The U.S. Department of Energy is actively engaged in working with industry, researchers, and governmental sector partners through the National Renewable Energy Laboratory (NERL) in order to provide effective measures to reduce the energy use, emissions, and overall transportation system efficiency [67]. Furthermore, from the operators' perspective, energy efficiency not only has great ecological benefits, but also has significant economic benefits because of the large electricity bill resulting from the huge energy consumption of a wireless base station [68]. Following the emerging need for energy-efficient wireless communications as well as the fact that grid-power connection is sometimes unavailable for RSUs, [69], it becomes more desirable to equip the RSUs with large batteries rechargeable through renewable energy sources such as solar and wind power [70] and [71]. Hence, it becomes remarkably necessary to schedule the RSUs' operation in such a way that efficiently exploits the available energy and extends the lifetime of the underlying vehicular network. In the case where the RSU uses transmit power control to maintain constant bit rate reception, the power consumed to transmit to closer vehicles is significantly less than that consumed when transmitting to farther ones. As a result, it seems like a RSU operating under a strict energy conservation mode tends to serve the vehicles residing in low energy consumption zones of its coverage range (*i.e.*, the closest vehicles). It is true that such a strategy would preserve the maximum amount of RSU's available energy, however, the increasing number of vehicles leaving the RSU's communication range with incomplete service requests instigate unsatisfied users, which is a clear indicator of an unacceptable QoS. This chapter examines a vehicular network where a RSU is deprived from a permanent grid-power

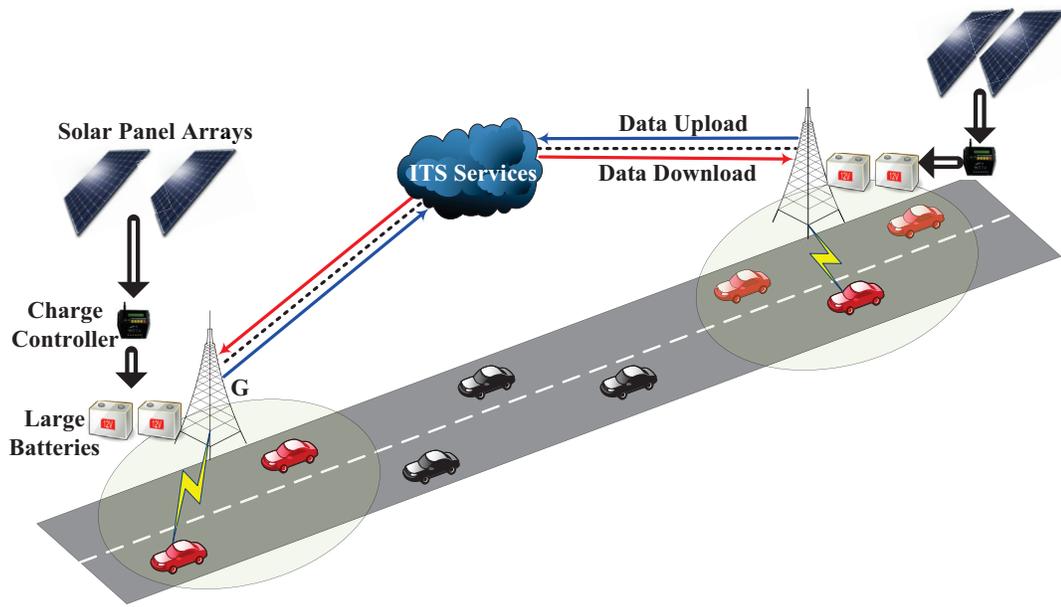


Figure 5.1: Solar Powered RSU in an Energy-Limited VANET

connection, but instead, the RSU is equipped with large batteries which are periodically recharged. An example of such scenario is illustrated in Figure 5.1, where the RSU batteries are recharged using solar energy harvesting techniques.

In a highly mobile vehicular network, changes in the network's topology and data-traffic load are frequent. To ensure an energy-efficient RSU performance, a scheduling protocol must change its vehicle selection policy to account for the aforementioned changes in network conditions. In other words, the RSU should implement an intelligent and adaptive scheduling algorithm which efficiently exploits battery power during times where the RSU is delineated with a limited amount of energy. In fact, a RSU leveraged with a smart identity has the ability to adapt its scheduling policy to diverse network scenarios, which persuade an energy-efficient RSU operation. It is important to mention that the limited energy constraint arises in a multitude of scenarios such as solar-powered battery discharge during night time as illustrated in Figure 5.1, insufficient amounts of harvested energy to support the RSU operation, high cost of grid-power connection, etc. A RSU exercising a Reinforcement Learning

(RL)-based scheduling protocol learns how to adapt to the persisting topology and load changes of a vehicular network and hence, admits vehicles to service in such a way that limits the RSU power consumption while maintaining an acceptable QoS for the arriving vehicles.

5.1.1 Problem Statement and Motivation

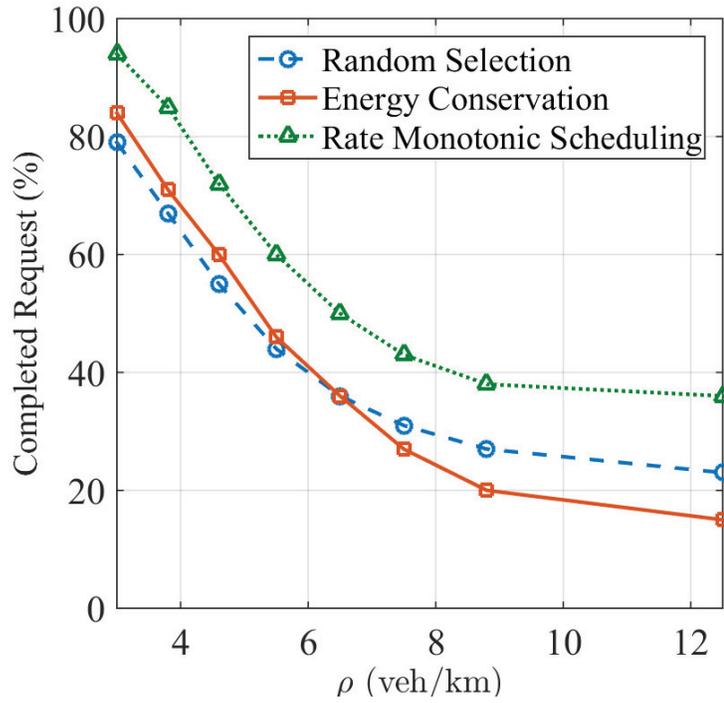
In this chapter, a Protocol for Energy-efficient Adaptive scheduling using Reinforcement Learning (PEARL) is proposed for the purpose of optimizing the operation of a RSU during its battery discharge period. The operation of PEARL is summarized as follow:

1. Collects traffic and network information for a sufficient amount of time to realize the service request load as well as the number of vehicles residing within the RSU's communication range at equilibrium.
2. Observes the environment and constructs PEARL's state representation of the system.
3. Engages in an exploration phase which allows PEARL to learn an optimal scheduling policy that maximizes a designated reward expression.
4. Exploits the realized optimal scheduling policy at the beginning of each time slot and admits a single vehicle to service.

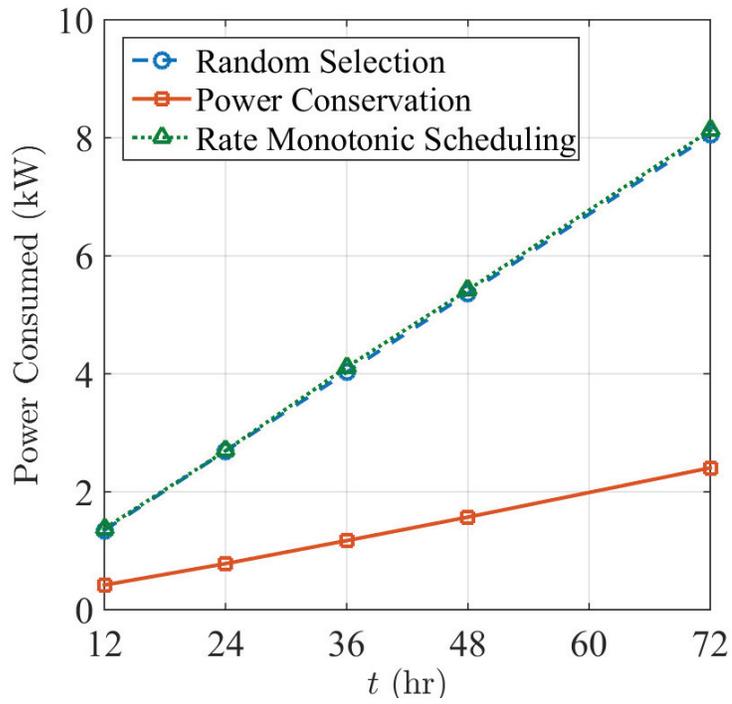
A Markov Decision Process model is formulated for the purpose of efficiently utilizing the available RSU energy while maintaining the vehicular network's operation and acceptable QoS. A reinforcement learning approach, in particular, the Q-learning algorithm, is proposed in order to grant the RSU the required artificial intelligence to realize an optimal scheduling policy which minimizes energy consumption and

achieves acceptable levels of QoS. The need for a competent, smart, and green vehicular environment motivates the establishment of an intelligent scheduling policy which meets multiple objectives. Figure 5.5.2 plots the energy consumption as well as the percentage of completed service requests in a downlink vehicular networking scenario similar to the one illustrated in Figure 5.1. Figure 5.2(a) shows that, whenever the RSU is operating under the energy conservation mode (*i.e.*, serving only the vehicles residing in low energy consumption zones), the percentage of the completed number of requests deteriorates dramatically as the vehicular density ρ increases. The random vehicle selection as well as the rate monotonic scheduling algorithms both outperform the greedy energy conservation method in terms of the completed request percentage, especially when the network load increases and more vehicles are present within the communication range of the energy-limited RSU. On the other hand, Figure 5.2(b) plots the power capacity required to maintain network operation when the discharge period varies between 12 and 72 hours. It is clear that, the RSU power consumption is significantly small when the network is operating under an energy conservative scheduling policy compared to the other two scheduling disciplines. As such, it becomes remarkably important to find an optimal scheduling policy that realizes the two objectives of minimal energy consumption with the largest fraction of completed requests.

The remainder of this chapter is structured as follows. Section II summarizes the related work and distinguishes the work presented herein from the existing studies in the literature. Section III presents the adopted vehicular traffic model. Section IV lays out the complete theoretical formulation of PEARL. The performance of PEARL is examined and compared to three RSU scheduling heuristics in Section V. Section VI concludes the chapter and describes a wealth of open future directions.



(a) Completed Request Percentage



(b) Power Consumption

Figure 5.2: V2I Scheduling Algorithms Performance

5.2 Related Work

In the context of vehicular networking, scheduling is a decision-making process whose outcome is an efficient, ultimately optimal, joint channel access regulation and resource allocation policy that has to be adopted by vehicles as well as RSUs for the purpose of realizing one or several objectives concurrently. In order to establish a fully-operational ITS, there exist a multitude of remarkably challenging objectives whose realization is possible through the design of appropriate scheduling algorithms. The next subsection lays out a selection of scheduling-based access methods which supplement the RSU with an intelligent identity allowing it to make vehicle selection decisions that contribute to realizing a desired objective.

5.2.1 Scheduling-Based Access Methods:

The authors of [28] indicated that the DTI application of V2I communications suffered from a random access problem and hence, developed the Dynamic Optimal Random Access (DORA) algorithm with the objective of maximizing the channel utilization subject to time-varying contention severity and capacity levels. The authors of [29] examined the V2I wireless access for streaming applications in a public transportation system. The authors formulated an optimization problem with the objective of providing a cost-minimal wireless connectivity that satisfies the end-users QoS requirements. The authors of [30] modelled the vehicular data download process using a series of transient Markov Reward Processes. Their objective was to characterize the distribution of a vehicle's downloaded data volume throughout its residence time within a RSU's range. In [72], the authors proposed a basic low-complexity V2I access scheme called $D * S$ where the RSU stored the Service Requests (SRs) and the request with the least $D * S$ was served first. D is the SR's deadline and S is the data size to be uploaded to the RSU. The authors then studied the uplink MAC

performance of a DTI scenario in [73]. The authors of [11] proposed two complexity-minimal V2I access schemes and modelled the vehicle's on-board unit buffer's queue as an $M/G/1$ queueing system and captured the V2I system's performance from a vehicle's perspective. The algorithms proposed in [28, 29, 30, 72, 73, 11] overlooked the RSU energy consumption pertaining to the adopted scheduling discipline. Given the increasing concern over the energy consumption in wireless networks as well as the highly likely unavailability of permanent power sources in vehicular networks, the conventional design approaches may not be feasible to green communications and should be revisited. The next subsection surveys related research work that proposed RSU scheduling methods which addressed the energy limitation in a vehicular environment.

5.2.2 Energy-Aware Vehicular Networks:

The work of [74] focused on using solar cell energy harvesting to provide an alternative power source for stationary RSUs. The goal was to design an efficient and adaptive energy-harvesting module which could be used with different types of embedded RSUs. A power management scheduler was deployed which predicts traffic status based on the historical data, and switches the RSU between ON (active) and OFF (idle or power saving) states. The authors of [75] addressed the problem of scheduling for energy efficient RSU. Therein, the objective was to minimize the long term power consumption subject to satisfying the communication requests associated with the passing vehicles. The authors first formulated lower bounds for total energy needed by a RSU in order to serve a finite set of vehicular arrival demands. Lower bounds were obtained by assuming that the total number of arriving vehicles and their respective speeds and associated requests are made available a priori to the RSU. Then, the authors proposed three online scheduling algorithms which

used vehicles' locations and speeds as inputs for a linear optimization problem which dynamically scheduled communication activity. However, under the three proposed algorithms, the scheduler is interrupted whenever a vehicle arrives to the RSU. More interruptions occur as the vehicle flow rate increases, and as such, the efficacy of the proposed algorithms becomes questionable.

The authors of [76] presented an energy-efficient scheduling scheme in the presence of multiple RSUs deployed along a highway, which are inter-connected using cellular communication links. The authors considered the case of a unicast RSU-to-vehicle communication scenario only. Integer linear programming bounds were derived for the normalized minimum and maximum energy usage of a single RSU, and then four online scheduling algorithms were proposed and evaluated in terms of total RSU energy consumption. The reported results in [76] showed that, in order to achieve near optimal energy consumption, online scheduling algorithms require some a priori information which may not be always available. In [77], the authors addressed the same problem as [76] and introduced the concept of a Virtual Control Node (VCN), which is considered connected to all RSUs through physical wires. The authors then derived a temporal graph that shows the connected nodes in the network, and used to find the minimum number of active RSUs needed to maintain a fully connected network. The reported results therein showed that the RSU's transmission range has a great impact on the total number of active RSUs required.

5.2.3 Novel Contributions:

The following points highlight the identifying contributions of this work:

- Unlike the work presented in [28, 29, 30, 72, 73, 11], this work realizes a scheduling policy that recognizes that the RSU is equipped with a limited-lifetime power source.

- An illiterate energy-limited RSU G is deployed alongside a road segment where arriving vehicles request access to the Internet infrastructure. G is not provided with any information related to the vehicle arrival process or the network expected load. G 's operation is dictated by PEARL, which explores the evolution of the vehicular network and exploits an adaptive dynamic policy in order to limit its energy consumption while retaining an acceptable QoS.
- PEARL implements a reinforcement learning algorithm which maximizes the long term system rewards conceded by the total number of downloaded packets as well as the number of completed service requests. PEARL considers that the event of the departure of a vehicle with incomplete download request is an undesired event which induces remarkable penalties on the system's performance.
- This study develops, analyses, and evaluates an iterative algorithm that finds an optimal RSU scheduling policy using the history of interactions with the environment. By applying the Q-learning method of RL, the RSU maintains a state-action-rewards table which is updated after each scheduling decision in order to tune the scheduling policy towards maximum returns.

This work establishes the first step in introducing artificial intelligence and reinforcement learning scheduling methods to vehicular environments for the purpose of conserving the RSU's battery while providing a competent QoS. The next sections lay out the vehicular traffic model as well as PEARL's theoretical formulation.

List Of Symbols

Symbol	Description
ρ	Vehicular density (veh/km)

μ_V	Vehicle arrival rate to considered segment (veh/s)
T	Discharge period (hours)
τ	Length of a time slot (s)
N	Number of time slots in a discharge period
t_n	Beginning time of time slot n
S_n	Set of inrange vehicles at the beginning of the n^{th} time slot
i	Index of an inrange vehicle
v_i	Speed of vehicle i (m/s)
V_{min}	Minimum vehicle speed (m/s)
V_{max}	Maximum vehicle speed (m/s)
\bar{V}	Average vehicle speed (m/s)
σ_V	Vehicles speeds standard deviation
D_C	Communication distance of the RSU (m)
J_i	Vehicle i 's discrete sojourn time in the coverage range of the RSU
H_i	Vehicle i 's download service request size
R_i^n	Vehicle i 's remaining discrete residence time in RSU's coverage range
P_c^n	Power consumed by RSU in the n^{th} time slot
B_c	RSU transmission data rate
d_i	Distance between RSU and vehicle i (m)
P_t	Total power capacity of RSU's battery
β_n	Exact Number of vehicles residing within RSU's range at t_n
β_e	Expected Number of vehicles residing within RSU's range at t_n
w_i^n	Weight of vehicle i at t_n
w_{max}	Maximum vehicle weight
x_n	System state at t_n
E	State space

A	Action space
A_n	Set of admissible actions at time t_n
a_n	RSU's action at time t_n
r_n	Single-step reward at received by RSU at t_n
P_n	Remaining RSU battery power at time t_n
T_n	Time until next recharge starting from time t_n
H_i^n	Remaining request size for vehicle i at time t_n
$\alpha(n)$	Step-size learning rate
γ	Discount factor

5.3 Vehicular Traffic Model

This study adopts a discrete-time free-flow traffic model characterized by homogeneous and uninterrupted light vehicular traffic flowing over a one-dimensional roadway segment of fixed length (*i.e.*, D_C in Figure 5.1). The inter-arrival time of vehicles under free-flow traffic conditions is exponentially distributed. As such, vehicle arrivals from a single lane follow a Poisson process [50]. When multiple lanes are considered, vehicle arrivals from each of these lanes follow independent and identically distributed Poisson processes. It follows that the overall vehicle arrival process from all lanes is the sum of i.i.d. Poisson processes, which is also a Poisson process with rate μ_V vehicles per second. In this work, time is divided into time slots of length τ . Hence, in the vehicular traffic model presented herein, the total discharge period T is divided into N time slots (referred to as the planning horizon later), each of length τ seconds. Let S_n be the set of vehicles residing within the communication range of G at the beginning of the n^{th} time slot (where $n = 1, 2, \dots, N$). G schedules to serve one vehicle i , where $i \in S_n$, at the beginning of the n^{th} time slot.

According to [78], under free-flow traffic condition, the speed v_i of an arbitrary

arriving vehicle i is a Normally distributed random variable whose probability density function (p.d.f.) is given by:

$$f_{v_i}(v_i) = \frac{1}{\sigma_v \sqrt{2\pi}} e^{\left[-\left(\frac{v_i - \bar{v}}{\sigma_v \sqrt{2}}\right)^2\right]} \quad (5.1)$$

The authors of [50] assumed justifiably that $v_i \in [V_{min}; V_{max}]$, and accordingly, the speeds of arriving vehicles follow a truncated version of $f_{v_i}(v_i)$, which is given by:

$$\widehat{f}_{v_i}(v_i) = \frac{2f_{v_i}(v_i)}{\operatorname{erf}\left(\frac{V_{max} - \bar{v}}{\sigma_v \sqrt{2}}\right) - \operatorname{erf}\left(\frac{V_{min} - \bar{v}}{\sigma_v \sqrt{2}}\right)} \quad (5.2)$$

Furthermore, since a vehicle's speed is maintained constant during the vehicle's navigation period within the RSU's coverage range [78]. Let J_i be the discrete version of the vehicle's residence within any RSU, and let D_C be the length of the segment that falls within the communication area of that IoT-GW. The p.m.f. of J_i has been derived in Chapter 4 and presented in Equation 4.2.

An arriving vehicle communicates its speed and download requirements as soon as it enters the coverage range of the RSU G . Consequently, G keeps record of all vehicles within its range as well as their associated service requirements. In this work, a vehicle i 's download service request is a uniformly distributed Random Variable H_i between H_{min} and H_{max} .

In the case where the RSU uses transmit power control to maintain constant bit rate reception, the power consumed to serve closer vehicles is significantly less than that consumed when serving farther ones. In fact, the RSU's power consumption increases exponentially as the receiving vehicle moves farther [59]. Moreover, less power is required to serve a vehicle at a specific distance under a lower data rate. Denote by d_i the distance between vehicle i and the RSU.

The next section lays out the mathematical formulation of PEARL being a RL-based scheduling algorithm deployed at the RSU for the purpose of regulating its energy consumption.

5.4 PEARL Theoretical Formulation

5.4.1 Overview:

The main objective of PEARL is to build a RSU that, at the beginning of each time slot, selects a vehicle to serve in such a way that maximizes its long term reward. Recall that, the RSU's reward is a performance metric for the total number of downloaded bits as well as the number of fulfilled vehicle requests per discharge period. PEARL is a well-trained agent that, given the traffic characteristics, the RSU's power budget, and the total discharge period, is able to achieve highest reward returns. In the exploration/learning phase, PEARL is tuned and enhanced as the vehicular network's operation evolves and the RSU explores the various observations, actions, and associated rewards. During this phase, and since PEARL is first unaware of the network conditions and state transitions, it follows an epsilon-greedy exploration method ([33]) which allows the agent to keep exploring the evolution of the underlying network for the purpose of fine tuning its scheduling policy. Once the exploration phase is completed, the RSU now exploits PEARL's optimal scheduling policy during its recharge period. It is worthwhile mentioning that, PEARL learns an optimal scheduling policy for different traffic conditions, which is devised whenever the RSU is operating in an energy-limited mode.

This current section is dedicated to present the various constituents of PEARL's properties and characteristics.

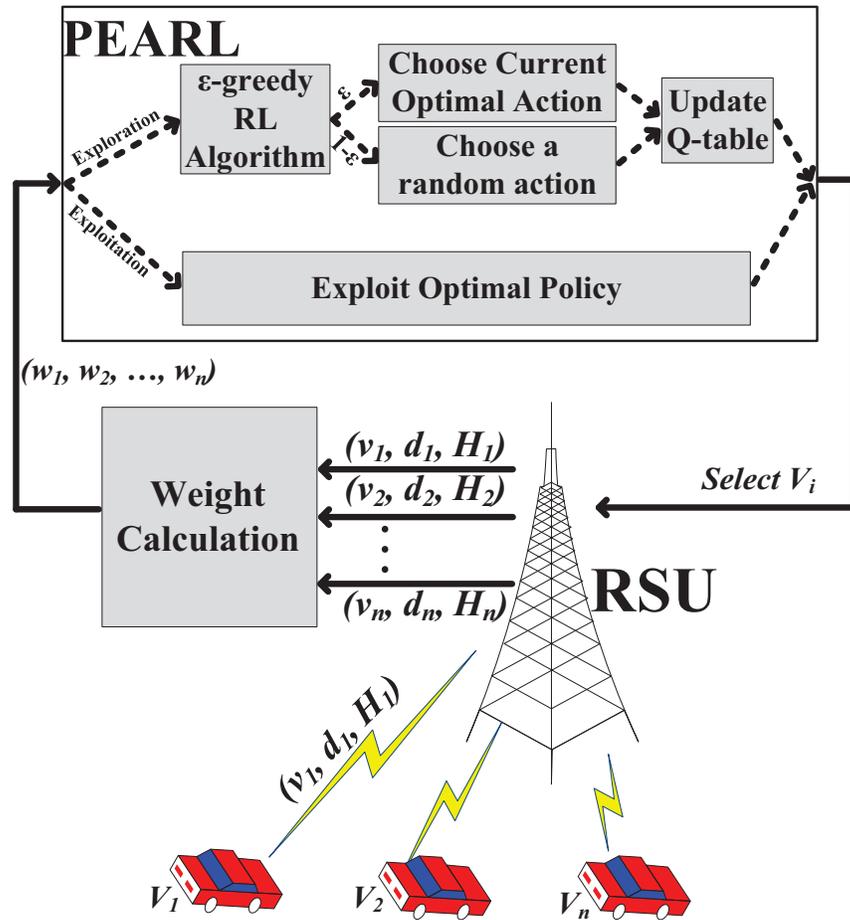


Figure 5.3: PEARL's Operation

5.4.2 PEARL Operation:

PEARL is a discrete-time learning protocol that develops an optimal scheduling policy of an energy-constrained RSU. At the beginning of each time slot, the vehicles residing within RSU's communication range broadcast beacon messages which the RSU collects in order to realize the network information corresponding to the number of vehicles within its range along with their respective speeds, locations, and remaining request size. The RSU then consults the PEARL agent in order to make a decision regarding which vehicle is granted service next. PEARL calculates each vehicle's weight according to the received information, and hence, establishes the observed system state. PEARL then engages in an exploration versus exploitation decision imposed by the ϵ -greedy method. In the exploration method, PEARL selects a random vehicle, which allows for the exploration of various system states. This operation is illustrated in Figure 5.3. On the other hand, in the exploitation method, PEARL endorses the choice of the vehicle which most contributes to the maximization of the long term system reward.

5.4.3 Finite Horizon Markov Decision Process:

This subsection lays out a precise definition of the PEARL's finite horizon Markov Decision Process. The underlying Markov model, its characteristics and its input data are presented in mathematical terms.

Preliminaries

Suppose that, at time $t_n = 0$, G 's battery is fully recharged, and its total power capacity is P_t . Also, assume that the time between two recharge periods is known, and the length of the discharge period is T seconds divided into N time slots, each of length τ . Let t_n be the beginning of the n^{th} time slot where $t_n > 0$ and $n = \{1, 2, \dots, N\}$. Let

β_n be the number of vehicles residing within G 's communication range at t_n , where $0 \leq \beta_n \leq \beta_{\max}$ and β_{\max} is the maximum number of vehicles that can be present within the segment of length D_c at any time.

Let x_n be the state of the system at t_n . In the presented model herein, the system state is a vector whose elements represent the weights of the vehicles residing within G 's communication range at t_n . Let w_i^n be an integer representing the weight of vehicle i at time t_n , where $n = \{1, 2, \dots, \beta_n\}$ and $0 \leq w_i^n \leq w_{\max}$. w_i^n is a function of vehicle i 's remaining residence time, remaining request size, and location along D_C , which indicates the vehicle's numerical ranking at time t_n . Note that, selecting the value of w_{\max} is remarkably strenuous as a large value of w_{\max} poses a serious limitation to one's ability to solve the Markov Decision Process accurately. This is known as the Bellman's *curse of dimensionality*, which is the well-known exponential increase in time and space required to compute an optimal solution to the MDP as the number of possible states (state space size) increases [35].

Now, it is important to mention that, the number of vehicles present within the coverage range of G is not constant. In fact, since vehicles arrive to the RSU according to a Poisson process, then, according to [42], the number of vehicles present with the segment of length D_C follows the Poisson distribution. Hence, in order to avoid the complexity of having a variable vector size to represent the system state, we can justifiably assume that, at any arbitrary instant t_n at equilibrium, the number of vehicles present within the coverage range of the RSU is β_e . As such, the system state matrix at t_n is a vector of size β_e containing the corresponding weights of β_e vehicles. Note that, in the case where $\beta_e - \beta_n > 0$, the values of $\{w_{\beta_n+1}^n, w_{\beta_n+2}^n, \dots, w_{\beta_e}^n\}$ will be set to $-\infty$. On the other hand, whenever $\beta_n > \beta_e$, the RSU will ignore the most recently arriving vehicles until either a vehicle completes its download request or a vehicle leaves its communication range. As such, the system state at time t_n is a fixed

size vector denoted by $x_n = \{w_1^n, w_2^n, \dots, w_{\beta_e}^n\}$.

Feasible Actions and Rewards

Let E be the state space where, at any time t_n , the system state $x_n \in E$. This subsection describes the feasible actions at time t_n and their associated single step reward:

- A is the action space, where the action at time t_n is $a_n \in A$. Note that, $a_n = 1$ if the vehicle with associated weight w_1^n is selected, $a_n = 2$ if the vehicle with associated weight w_2^n is selected, and so forth. Therefore, $A = \{1, 2, \dots, \beta_e\}$. Note that, the feasible set of actions at time t_n is $A_n \subset A$. Since PEARL may only select a vehicle whose associated weight is not $-\infty$. Therefore, in the case where $\beta_e - \beta_n \geq 0$, $A_n = \{a_1, a_2, \dots, a_{\beta_n}\}$, whereas, when $\beta_e - \beta_n < 0$, $A_n = \{a_1, a_2, \dots, a_{\beta_e}\}$.
- $r_n(x_n, a_n)$ gives the single step reward of the system at time t_n if the current state is x_n and action a_n is taken. According to PEARL, the single-step reward r_n is the number of downloaded bits to the selected vehicle at time t_n . Whenever a vehicle departs from G 's coverage range with an incomplete download request, the single-step reward is penalized by the remaining number of bits which need to be downloaded in order to fulfill that vehicle's request. As such, PEARL strives for a larger number of transmitted bits per time slot, and at the same time, tries to avoid the undesired event where a vehicle departs from G 's range with an incomplete download request. Note that, and according to [59], the RSU consuming the same amount of energy per time slot may transmit at larger rates for closer vehicles. Therefore, it becomes clear now that PEARL prefers to select vehicles closer to the RSU over the far ones. However, PEARL may be forced to schedule service for farther vehicles in order to prevent the

penalty incurred by the above-described unfavourable event.

Input From the Environment

At the beginning of each time slot, G collects all the parameters associated with the set of in range vehicles, and then feeds the network information to PEARL, which, at time t_n , becomes aware of the following:

1. P_n being the remaining power in the RSU's battery, $0 \leq P_n \leq P_t$.
2. T_n being the time until the next recharge, $1 \leq T_n \leq T$.
3. β_n being the number of vehicles residing within G 's communication range, $0 \leq \beta_n \leq \beta_{\max}$.
4. $\overline{J}_n = \{J_1^n, J_2^n, \dots, J_{\beta_e}^n\}$ being a vector of size β_e containing the remaining sojourn times of each vehicle v_i , $i \in (1, 2, \dots, \beta_e)$ and $0 \leq J_i^n \leq r_{\max}$
5. $\overline{H}_n = \{H_1^n, H_2^n, \dots, H_{\beta_e}^n\}$ being a vector of size β_e containing the remaining request sizes for each vehicle v_i , $0 \leq H_i^n \leq H_{\max}$.
6. $\overline{d}_n = \{d_1^n, d_2^n, \dots, d_{\beta_e}^n\}$ being a vector of size β_e containing the distances between G and each of the in-range vehicles, $0 \leq d_i^n \leq G_R$, where $G_R = D_C/2$.

At this stage, and according to Figure 5.3, PEARL will calculate the weight associated with each vehicle within G 's communication range and hence realize the system state.

Vehicle Weights

As earlier mentioned, the power required for G to communicate with a vehicle residing within its coverage range increases remarkably as the separation distance between the transmitter and the receiver increases. Therefore, in a greedy power saving

mode, the RSU may always prefer to serve the closest vehicle in order to consume the least amount of energy, and thus, conserve its battery power for subsequent service requests. However, under such operational policy, vehicles may suffer from a deteriorated quality of service especially when leaving the communication range of the RSU without a completed service request. This event is referred to as an undesired in PEARL’s formulation in the next sections.

Now, in order to bias the RSU towards serving vehicles with the least amount of consumed energy while avoiding undesired events, vehicle i ’s weight at time t_n , previously defined as w_i^n , gives it a priority depending on its remaining sojourn time, remaining request size, and its separation distance from G . In fact, as a vehicle becomes closer to G (*i.e.*, d_i^n decreases), its weight increases since now, the RSU may transmit data at a high rate rather than transmitting data to farther vehicles using the same amount of energy. Let R_i^n be vehicle i ’s remaining residence time at the beginning of n^{th} time slot. Whenever R_i^n decreases, vehicle i ’s weight increases as well, which is a signal for the RSU to complete vehicle i ’s download request before it leaves its communication range (and avoid being penalized). Finally, vehicle i ’s weight increases as H_i^n increases, which will prioritize vehicles with larger remaining request size.

Recall that, the system state x_n at time t_n is a vector whose elements correspond to the weights of the set of in range vehicles, where, as previously defined, $0 \leq w_i^n \leq w_{max}$. Hence, the size of the state space E is $(w_{max} + 1)^{\beta_e}$. It becomes clear now that a large value of w_{max} results in an intractable MDP whose state space size is remarkably huge. On the other hand, a smaller value of w_{max} might not give PEARL enough information and differentiation between the different vehicles requesting service at a particular time slot. As such, the choice of the value of w_{max} has to account for the trade off between the time and space needed to achieve an optimal policy and the

level of differentiation and prioritization between vehicles.

System Dynamics

This subsection lays out the equations that govern the evolution of the system dynamics:

- Power remaining in the next time slot:

$$P_{n+1} = P_n - P_c^n \quad (5.3)$$

where P_c^n is the power consumed by G in the n^{th} time slot, and $1 \leq n \leq N$.

- A vehicle's remaining request size:

$$H_i^{n+1} = \begin{cases} H_i^n, & \text{if } a_n = i \\ H_i^n - K_i^n \times \tau, & \text{if } a_n \neq i \end{cases} \quad (5.4)$$

where $K_i^n = g(P_c^n, d_i^n)$ is the rate at which the RSU serves the selected vehicle. Note that, for a fixed amount of transmit power, the data rate decreases drastically as the separation distance between the transmitter and receiver increases.

- A vehicle's remaining sojourn time:

$$R_i^{n+1} = R_i^n - \tau, \text{ for } i = \{1, 2, \dots, \beta_n\} \quad (5.5)$$

In the case where $\beta_n > \beta_e$, whenever a vehicle i either departs from G 's communication range or completes its download request during the time slot starting at t_n , G will consider the weight of another vehicle $j \neq i$ at t_{n+1} where H_j^{n+1} and R_j^{n+1} are Random variables with known distributions.

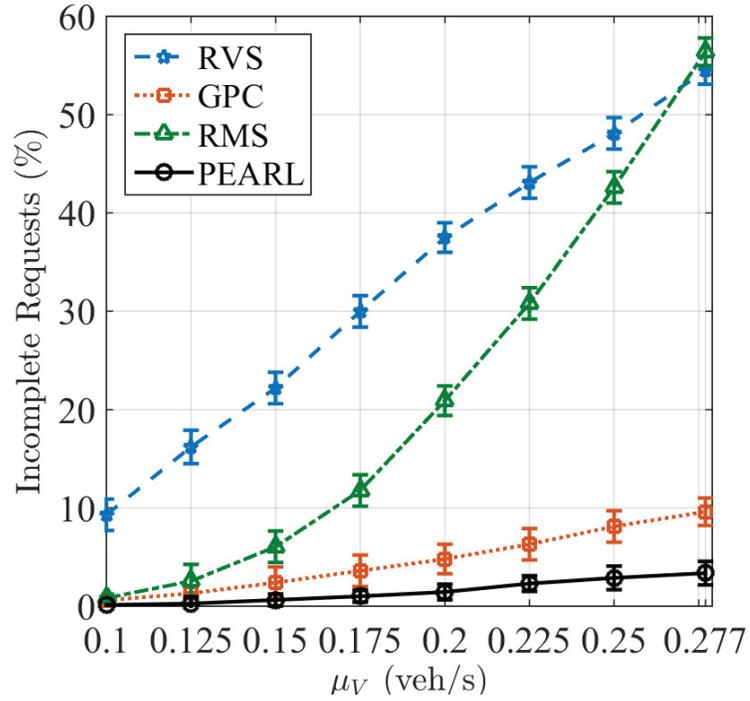
5.4.4 MDP Solution Approach:

Recall from Chapter 2 that, the optimal control of a MDP requires the determination of a stationary policy π defining which action a_n should be applied at time t_n in order to maximize an aggregate objective function of the immediate rewards. According to [34], the optimal policy associated with optimal value function given in Equation (2.4) achieves the maximum reward expression laid out in Equation (2.1). However, in order to solve Equation (2.5), the knowledge of the transition probability function, $P(x_{n+1}|x_n, \pi(x_n))$ is required. Note that, the formulated Markovian domain herein lacks the state transition mapping, *i.e.*, $P(x_{n+1}|x_n, a_n)$. Therefore, the reinforcement learning method, Q-learning, presents itself as a simple way for the RSU to learn the optimal policy by experiencing the consequences of actions without the requirement of an established transition function. Consequently, PEARL implements a stochastic iterative Q-learning algorithm (detailed in Algorithm 2.1 in Chapter 2.3) and uses observations from online samples in order to realize the optimal scheduling policy π^* .

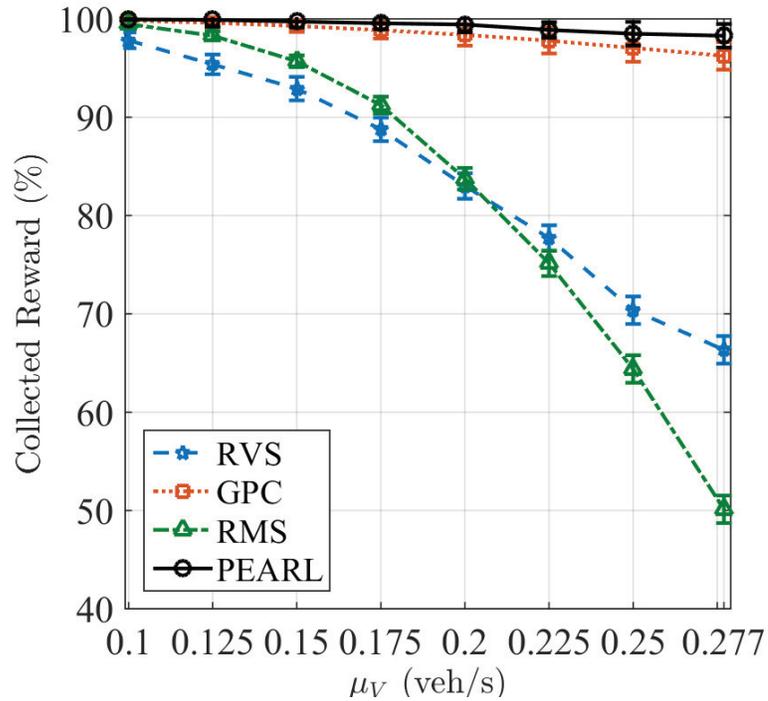
5.5 Simulation Results

5.5.1 Simulation Setup

In the simulation setup of this work, the Simple Free-flow Traffic Model (SFTM), which was laid out in [50], is adopted. Using the discrete-event simulator Veins ([60]), the vehicular traffic model presented in Section III is validated. Furthermore, the realistic mobility traces generated by SUMO were fed as a mobility input for PEARL exploration phase. The presented results herein were averaged over multiple runs of the simulations. In this section, the performance of PEARL is evaluated in terms of:



(a) Incomplete Request Percentage



(b) Collected Reward Percentage

Figure 5.4: PEARL Performance Evaluation

- Incomplete request percentage
- Collected rewards percentage
- Average per-vehicle fulfilled request percentage
- Network throughput

PEARL is compared with three other scheduling algorithms namely:

1. RVS: Random Vehicle Selection algorithm where, at time t_n , the RSU randomly chooses a vehicle $v_i \in S_n$ to be served [11].
2. GPC: Greedy Power Conservation algorithm where, at time t_n , the RSU chooses the vehicle $v_i \in S_n$ which resides in the lowest energy consumption zone compared to the remaining vehicles residing within G 's communication range.
3. RMS: Rate Monotonic Scheduling algorithm without pre-emption where, at time t_n , the RSU chooses the vehicle with the highest priority. A vehicle's priority, according to traditional RMS algorithms [79], is inversely proportional to its period, *i.e.*, the shorter the period, the higher the priority and vice-versa. Herein, a vehicle's period is the time it requires until it completes its download request.

Vehicular nodes arrive at a unidirectional highway segment of length D_C with multiple lanes according to a Poisson process and travel with a constant average speed drawn from a truncated Gaussian distribution. Each vehicle has an associated service request size to be downloaded from the RSU. A vehicle admitted to service may download at a rate of B_c Mbps depending on its corresponding separation distance with the RSU. Upon its departure from the RSU's communication segment, its associated remaining request size and average throughput are recorded for the network's performance analysis. Table 5.2 lists the simulator's input parameters.

Table 5.2: PEARL Simulation Input Parameters

Parameter	Value
Discharge period	$T = 12$ (hours)
Time slot length	$\tau = 0.1$ (ms)
Vehicular arrival rate	$\mu_V \in [0.1; 0.277]$ (veh/s)
Min and Max vehicle speed	$V_{\min} = 3, V_{\max} = 50$ (m/s)
Min and Max request size	$Q_{\min} = 1, Q_{\max} = 10$ (MB)
RSU covered segment	$D_C = 1000$ (m)
Channel data bit rate	$B_c \in [1; 27]$ (Mbps)
Learning rate	$\alpha(n) = 1/n$
Discount factor	$\gamma = 0.5$

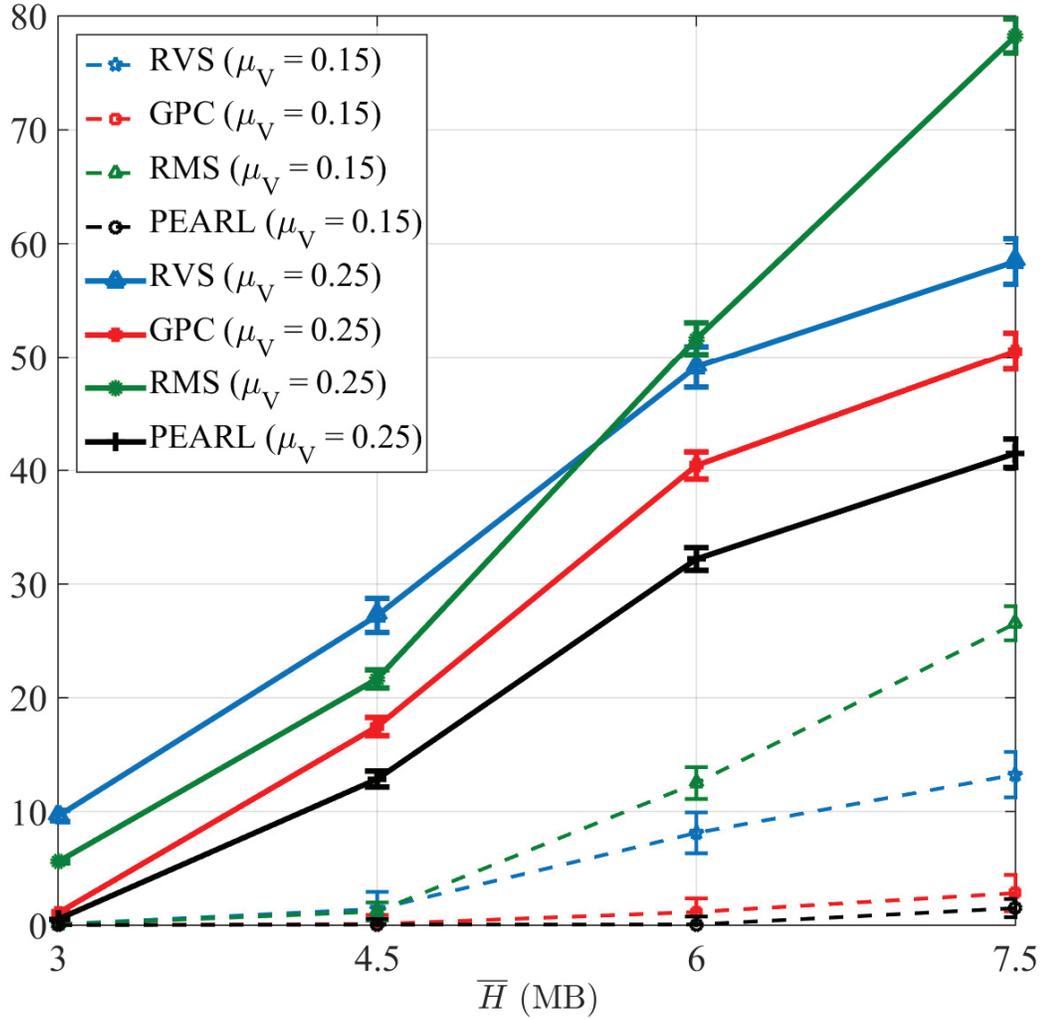


Figure 5.5: QoS under Variable Request Size

5.5.2 Simulation Results

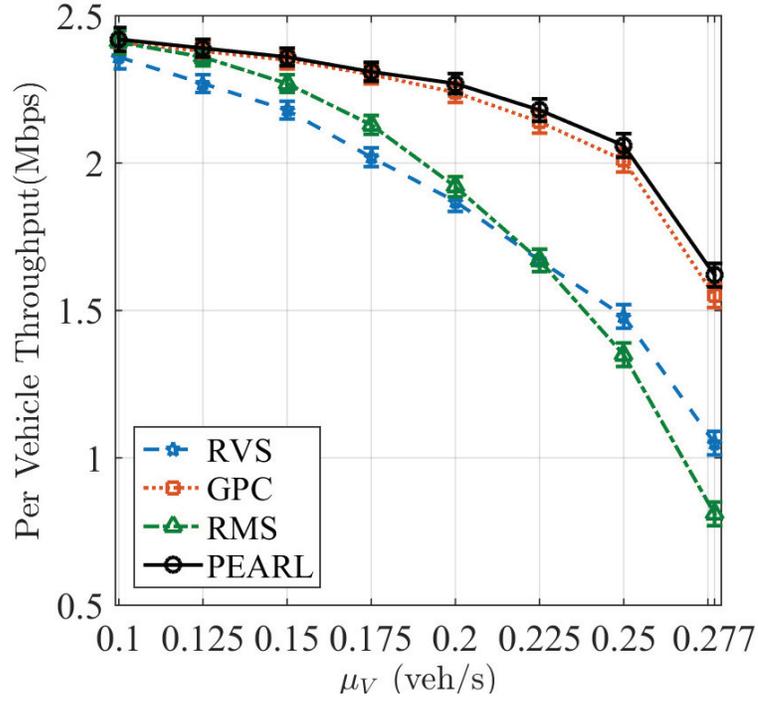
Figure 5.4 evaluates PEARL's performance when compared with the three previously described scheduling algorithms, namely, RVS, GPC and RMS. Figure 5.4(a) plots the percentage of vehicles leaving G 's communication range with an incomplete service request as a function of the vehicular arrival rate. It is clear that the number of incomplete requests increases as the vehicular arrival rate increases under the three scheduling algorithms. In fact, an increase in μ_V is accompanied by an increase in the number of vehicles present within the range of the RSU. As μ_V increases, the likelihood of selecting a certain vehicle will decrease, independent of the scheduling discipline. Consequently, a vehicle will spend less time receiving service and the total number of vehicles departing from G 's communication range with incomplete service requests will increase. Figure 5.4(a) also shows that PEARL outperforms RVS, GPC as well as RMS in terms of incomplete service requests. Under RVS, the selection method is random, and no service differentiation is applied, and therefore, the number of vehicles whose associated download request is not fulfilled increases remarkably as more vehicles are present within G 's communication range. Now, for GPC, G is admitting to service the vehicle which resides in the minimal energy consumption zone compared to the set of in range vehicles. Whenever μ_V is small and the vehicular density is low, a large portion of the vehicles have enough time to complete their download request whenever they are residing in low energy consumption zones, however, when μ_V increases, more vehicles will concurrently reside in low energy consumption zones and G will randomly choose between the multiple available vehicles, and therefore, the time during which a vehicle receives service is now not enough to complete the download request. Under RMS, the vehicle with the smallest remaining download request size is selected regardless of its location along D_C . Whenever the vehicular load is small, *i.e.*, $\mu_V < 0.15$, RMS performs relatively well. However, as

the network load increases, the number of incomplete requests increases remarkably. This is an expected result knowing that the RMS is only a good scheduling algorithm whenever the requests are schedulable within a specific time frame. Finally, recall that, for PEARL, a vehicle departing G 's range with an incomplete service request is a undesired event which the agent is trained to avoid. Therefore, the deployment of the well-trained PEARL agent guarantees that the majority (more than 95 %) of departing vehicles have completed their download service request. Figure 5.4(b) plots the percentage of collected rewards which is defined as the total number of downloaded bits over the total number of requested bits in a discharge period. In practice, a higher percentage of collected rewards results in a better QoS as well as increased RSU revenues. Figure 5.4(b) shows that the portion of total downloaded bits over the total requested bits decreases as more vehicles are present within G 's communication range. Furthermore, it is clear that the adoption of the reinforcement learning algorithm PEARL results in higher collected rewards than its counterparts RVS, GPC and RMS. It is important to note that, admitting the vehicle residing in the minimal energy consumption zone into service according to the GPC scheduling algorithm results in a high percentage of collected rewards since G is transmitting data to the selected vehicle at the highest achievable data rate at that particular instant. Consequently, and as previously stated, whenever the vehicular density is low, G is able to transmit a large portion of the requested file size. As μ_V increases, and more vehicles are present within the segment of length D_C , the smart scheduling algorithm PEARL outperforms GPC in terms of the percentage of collected rewards as PEARL learns to wait for approaching vehicles to enter the lowest energy consumption zone and transmit their data at a rate of 27 Mbps rather than admitting these vehicles to service when they are in higher energy consumption zones when no

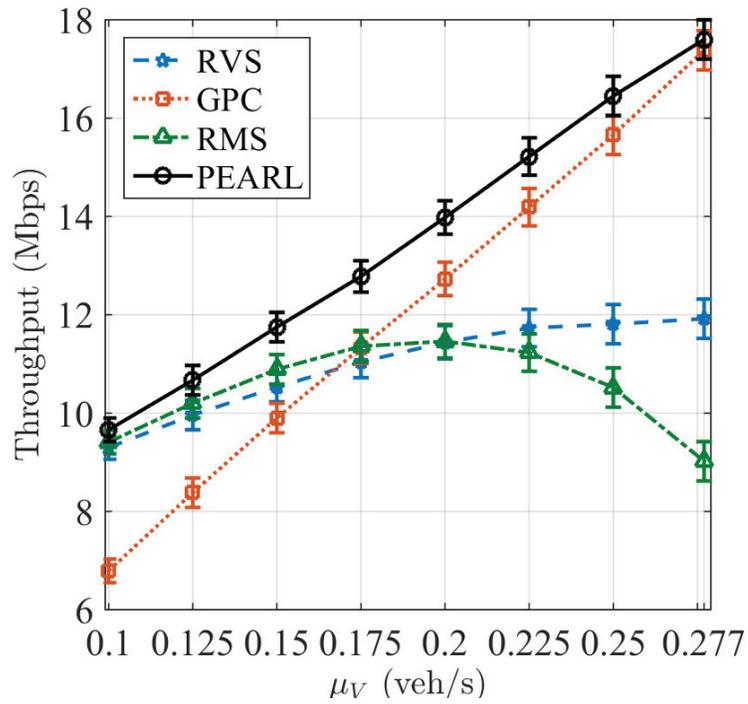
other vehicles are present in the lowest energy zones. Figure 5.5 shows that the percentage of vehicles departing from RSU’s coverage range with an incomplete service request increases as the average request size increases. The QoS also deteriorates as the vehicular arrival rate increases, which emphasizes the result in Figure 5.4(a). It is clear that PEARL outperforms all the other scheduling benchmarks irrespective of the size of the average service request.

Figure 5.6 plots the per-vehicle and the network throughputs when the RSU is operating under three different scheduling algorithms. Figure 5.6(a) shows that the per-vehicle throughput deteriorates remarkably as μ_V increases under the RVS and RMS scheduling disciplines. This is expected since, under RVS, the RSU may choose a vehicle residing in high energy consumption zones and the associated service transmission rate is therefore very small. Hence, the amount of time a vehicle spends receiving service, in this case, is extremely inefficient. Also, under RMS, the vehicle whose remaining request size is smallest may highly likely be present in a high energy consumption zone. This means that G is transmitting at a low rate. Hence, under RMS, the per-vehicle throughput decreases quickly as more vehicles are present within G ’s communication range. On the other hand, GPC and PEARL show significant enhancements on the level of per-vehicle throughput when compared to the RVS method. This is in fact due to the frequent selection of vehicles residing in low energy consumption zones allowing the RSU to transmit at a fast data rate. Figure 5.6(a) also shows that PEARL tops GPC in terms of the per-vehicle throughput under all considered vehicular arrival rates. Following the same reasoning presented for Figure 5.4(b), PEARL is trained to select the vehicle that most contributes to the total rewards thus rendering the exploitation of the service time slot highly efficient.

Figure 5.6(b) plots the overall network throughput under the four implemented scheduling algorithms for various free flow vehicular arrival rates. It is clear that



(a) Per-vehicle Throughput



(b) Network Throughput

Figure 5.6: Per-vehicle and Network Throughputs

PEARL outsmarts the three scheduling disciplines RVS, GPC and RMS in terms of the achievable network throughput. Under RVS, G randomly admits a vehicle residing within its communication range into service. As more vehicles are present within the considered roadway segment of length D_C , more vehicles are present in low energy consumption zones, and whose selection allows for faster data transmission. Hence, the network throughput increases as μ_V increases. Now, when the network is operating under GPC and the vehicular arrival rate is small, G is serving the vehicle residing in the lowest energy consumption zone compared to the set of all vehicles present within G 's communication range. In the very likely event that no vehicles reside in low energy consumption zones, G has no choice but to serve vehicles in high energy consumption zones resulting in slow data transmission and hence, decreased network throughput. However, as μ_V increases, GPC results in higher network throughput than RVS since now, more vehicles are present within the segment of length D_C and G is serving vehicles in lower energy consumption zones, and hence transmitting data at a faster rate. Now, under RMS, when the vehicle arrival process is slow, the vehicles with smaller remaining request sizes are residing in low-to-medium energy consumption zones which allows the RSU to transmit at an acceptable data rate. However, once more vehicles are present within the segment of length D_C , the vehicles having the smallest remaining file size fall at the edge of G 's communication range where the data download rate is smallest. Hence, the network throughput under RMS has this parabolic shape. When the RSU's operation is dictated by PEARL's optimal scheduling policy, the achieved network throughput is greater than that achieved under RVS and GPC for all considered vehicular arrival rates. In fact, PEARL is trained to efficiently schedule the vehicle's download service in such a way that maximizes the number of downloaded bits per unit time. The result illustrated in Figure 5.6(b) is an improved network throughput under PEARL compared to the

other three scheduling algorithms.

5.6 Conclusion and Future Research Direction

This chapter addresses the problem of energy-limited RSUs in a vehicular network. A Markov Decision Process is formulated and solved using a reinforcement learning technique, namely the Q-learning algorithm. The resolution is a Protocol for Energy-efficient Adaptive scheduling using Reinforcement Learning (PEARL), which is proposed for the purpose of increasing the number of downloaded bits per unit time as well as avoiding the undesired event of a vehicle departing from the RSU's communication range with an incomplete service request. After a sufficient training period, PEARL exploits the realized optimal scheduling policy, which outperforms three benchmark scheduling algorithms in terms of several QoS metrics. In particular, the deployment of PEARL complements the RSU with the required intelligent identity, which serves to maintain the RSU's operation throughout the whole discharge period as well as decrease the number of vehicles departing from the RSU's coverage range with an incomplete service request.

This work is the first step of introducing machine learning techniques, in particular, reinforcement learning, in order to optimize RSU scheduling in a vehicular network. Our future research is directed towards studying an energy-constrained vehicular network which is composed of a tandem of RSUs deployed on a long roadway segment. The feasibility of Q-learning as well as other machine learning techniques will be investigated in order to solve such a large-scale problem.

Chapter 6

Optimizing Downlink Traffic

Scheduling - The Multi RSU case

6.1 Introduction

The IoV is foreseen to support a full-fledged, smart and efficient ITS by providing real-time traffic information, context-aware advertising as well as drive-through Internet access, provisioned through the help of stationary IoT GateWays (IoT-GW) deployed along roadways. The previous chapter presented supporting evidence about the fact that the significant barrier to the widespread deployment of IoT-GWs is the cost of provisioning electrical grid power connections, [6], as well as their remarkable energy consumption. Following the emerging need for energy-efficient wireless communications as well as the fact that grid-power connection is sometimes unavailable for IoT-GWs, [69], it becomes clear and more desirable to deploy green energy-efficient IoT-GWs, which are equipped with large batteries rechargeable through renewable energy sources such as solar and wind power [70, 71]. Energy-efficient and QoS-oriented scheduling policies must be employed at the IoT-GW in order to guarantee a desired level of performance in an eco-friendly environment. The objective of this

chapter is to establish a smart visionary vehicular networking infrastructure similar to the one illustrated in Figure 6.1.

The major entangled challenge associated with the proper inauguration of a full-fledged connected vehicular network is the efficient control and management of the operation of multiple RSUs deployed in tandem along roadways. Indeed, the highly dynamic and stochastic nature of vehicular networks, the randomness in the vehicle arrival process as well as the diversity of the requested services give rise to a particularly challenging scheduling problem for the efficient operation of the IoT-GWs. Multiple studies in the literature have addressed the scheduling problem in the context of V2I communications. For instance, the work in [28, 72, 73, 11, 80] proposed novel V2I scheduling algorithms for a single RSU equipped with an infinite power source. Other studies addressed the energy consumption issue in the single-RSU (*e.g.*, [44] and [75]) and multi-RSU (*e.g.*, [76]) scenarios, where the RSU is privileged with a priori knowledge of vehicle arrival instances and requests in order to resolve a complex optimization problem. The objective of this present work is to establish a universal, green, intelligent and scalable scheduling policy which acclimates to the random characteristics of a vehicular environment, overcomes the limiting assumptions and deficiencies of previous studies and finally, establish a vigilant backbone ITS that supports the development of the IoV. Precisely, herein, we consider a long road segment where several IoT-GWs are deployed in tandem. The scenario is illustrated in Figure 6.1. Each IoT-GW is connected using fiber or cellular links to a backend ITS central server, which is the acting agent for all the communications that take place in the network. IoT-GWs collect high-dimensional inputs corresponding to the network characteristics, forwards the collected data to the central agent that devises appropriate actions. In this work, the central agent is trained to realize an optimal scheduling policy that meets the following objectives:

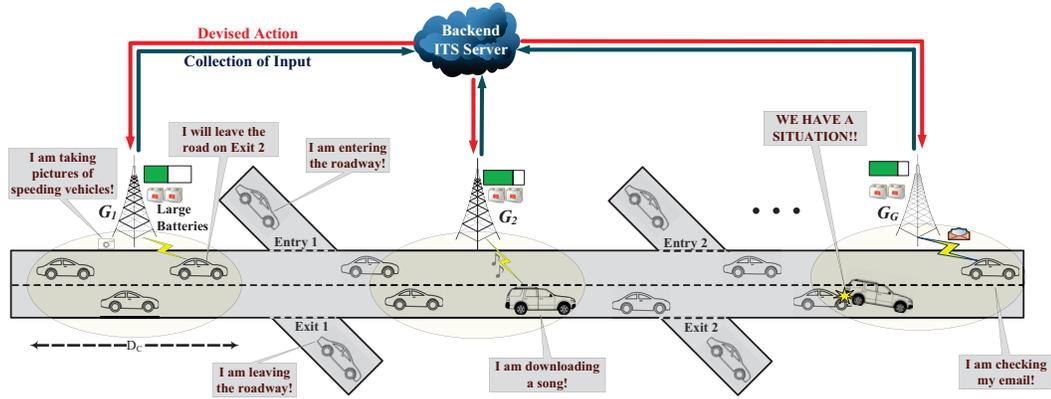


Figure 6.1: Energy-Limited Multi-RSU Vehicular Network

1. Communicate safety messages with minimum latency.
2. Minimize the mean response time as well as the mean total delay of non-safety-related download requests.
3. Satisfy the vehicles' download requirements before their departure from the road.
4. Maintain the entire vehicular network up and running by balancing the power consumption at each IoT-GW.

In the study presented in [12], the authors considered a vehicular network composed of a single battery-powered RSU and developed an MDP framework with discretized states in order to establish an optimal RSU-controlled scheduling policy. Therein, the resolution of the MDP was realized using reinforcement learning techniques [34]. The scalability of the solution proposed therein is poor especially when the number of state and action variables increase. Truly, the size of the table used to store Q-values grows exponentially and the system fails to distinguish between similar states and actions. Furthermore, the time required for the Q-table to converge increases intractably. In this present work, the state space dimension is enormously

large. Hence, the required computational time and effort to realize an optimal scheduling policy is prohibitively large, a phenomenon commonly referred to as the *curse of dimensionality* [35]. Consequently, recent advances in training deep neural networks (*i.e.* function approximation techniques) are exploited herein in order to overcome this complexity and, thus, promote the feasibility of instantiating a fictitious artificial agent that will: *a)* learn a scheduling policy from high-dimensional inputs using end-to-end deep reinforcement learning, *b)* derive efficient representations of the environment, and *c)* progress towards the development of a successful scheduling policy which meets the above-detailed objectives.

The remainder of this chapter is structured as follows. Section II lays out the problem statement and motivation as well as the novel contributions of this present study. Section III presents an overview of the most relevant related work in the context of IoV, V2I scheduling algorithms as well as deep reinforcement learning. A description of the V2I communication scenario is presented in Section IV. Section V presents the adopted vehicular mobility model. Section VI lays out a detailed presentation of the deep reinforcement learning model. An MDP model is formulated in Section VII. The performance of the proposed deep reinforcement learning algorithm is examined and compared to other existing scheduling heuristics in Section VIII. Finally, concluding remarks are presented in Section IX.

6.2 Problem Statement and Novel Contributions

The convergence of the IoV technology encompasses information communications, environmental protection, energy conservation as well as safe driving environment. The successful inauguration of the IoV technology requires the acquisition of core technologies and standards, which will be crucial to securing a strategic advantage. However, the integration of the IoV with other infrastructures should be as important

as the building of the IoV technology itself. Particularly, the ITS is envisioned to lay a solid basis for the IoV, which will eventually become an integral part of the larger IoT. This work aims to establish an efficient resource allocation scheduling policy that governs the operation of the IoT-GW, being the primary point of service in a vehicular environment. The objective of the proposed policy is to reduce the severity of (ultimately mitigate) a subset of the persisting challenges affronted in optimizing the eco-friendly operation of IoT-GWs in a highly dynamic ITS, subject to several random processes such as the emergence of safety-related messages, the vehicles' random service request sizes as well as the vehicles' arrival and departure points. Additionally, considering the scenario illustrated in Figure 6.1, it is clear that maintaining a balanced available energy at the IoT-GWs through proper communication and collaboration among the tandem of the connected IoT-GWs shall prolong the operation of the connected network. As such, safety-related information will continue to reach the maximum number of vehicles, thus, promoting a safer driving environment. Furthermore, the proper coordination of multiple IoT-GWs serves the purpose of fully satisfying a vehicle's service request by efficiently dividing the download requirements among the IoT-GWs which fall on that vehicle's path. Consequently, more vehicles will leave the roadway with a Complete Service Request (CSR), which is considered an important QoS metric for the network overall performance. The work presented herein eliminates the limiting assumptions and addresses the shortcomings of previous studies (*e.g.*, [75, 76, 12]). This work exploits a deep reinforcement learning technique, namely Deep Q-Network, to establish a safe, connected and energy-efficient vehicular network with multiple IoT-GWs. The following points highlight the identifying contributions of this study:

1. The adaptation of a vehicular network with a central ITS agent that governs the operation of multiple connected Iot-GWs deployed on a long road segment.

2. The realization of an energy-efficient and QoS-oriented scheduling policy, which dictates the operation of the energy-limited IoT-GWs. A deep reinforcement learning agent learns an adaptive dynamic scheduling policy that serves to balance the energy available at the IoT-GWs in order to maintain an operational connected ITS.
3. The exploitation of recent advances in training a deep reinforcement learning agent, which, first, constructs a state representation that implicitly contains all of relevant information about the current situation of the vehicular network, then observes the cost/reward of choosing a particular action, and finally, learns from current and past experience in order to realize an optimal scheduling policy and minimize the overall costs.
4. Artificial Intelligence provides the framework and tools to go beyond trivial real-time decision and automation in the IoV era. This work provides a solid ground to introduce machine learning techniques, in particular, deep reinforcement learning, in order to establish a smart, safe and green ITS, which complements the future IoV services and applications.

6.3 Related Work

6.3.1 IoV Enabling Technologies:

The complete realization of IoV could incur fundamental upgrades to the driving experience through the integration of intelligence within existing ITS applications. However, various vehicular communications problems are yet to be resolved for the proper realization and quick penetration of IoV enabling applications. Challenges in

the design and development of IoV are being investigated by researchers and practitioners interested in the future of vehicular communications. The authors of [81] provided a comprehensive review of routing protocols applicable in the context of IoV. The authors described the routing taxonomy from five different perspectives and then laid out preliminary guidelines for the efficient development of IoV routing protocols and technologies. The authors of [82] argued that the vehicular cloud constitutes an instance of the envisioned IoV comprising all the protocols and services required for the vehicle grid to operate efficiently and safely. They discussed the challenges associated with the exploitation of vehicular cloud computing to support autonomous and Internet-connected vehicles. The authors of [83] argued that the vehicular density as well as the high mobility of vehicles led to constant topological changes, and hence, content distribution became a challenging task. For this purpose, the authors proposed a Bayesian Coalition Game for content distribution using learning automata. The proposed algorithm showed an increased probability of content distribution with a lower end-to-end delay when compared to other content distribution counterparts.

6.3.2 V2I Scheduling-Based Access Methods:

The algorithms proposed in [28, 72, 73, 11, 80] overlooked the RSU energy consumption pertaining to the proposed scheduling discipline. Given the increasing concern over the energy consumption in wireless networks as well as the highly likely unavailability of permanent power sources in vehicular networks, the conventional design approaches may not be feasible to green communications and should be revisited. As such, the authors of [75] addressed the problem of scheduling for energy efficient RSU. Therein, the objective was to minimize the long term energy consumption subject to satisfying the communication requests associated with the passing vehicles. The authors first formulated lower bounds for total energy needed by a

RSU in order to serve a finite set of vehicular arrival demands. Then, the authors proposed three online scheduling algorithms which used vehicles' locations and speeds as inputs for a linear optimization problem which dynamically scheduled communication activity. In [84], the authors proposed an energy-efficient RSU deployment algorithm for the purpose of minimizing the transmit power of the RSUs subject to the constraints that all vehicles on the road should be covered. The authors then studied the network performance when the number of RSUs deployed on a roadway varied.

To the best of our knowledge, thus far, the literature has overlooked the possibility of establishing scheduling policies using machine learning techniques. Particularly, deep reinforcement learning augments the RSU with the ability to observe and analyse the environment, make decisions, learn from past experience, and eventually, perform optimal actions, which will serve the for the establishment of an efficient ITS in the IoV era. The next subsection lays out a brief overview of the state-of-the-art deep reinforcement learning models and their applicability in a vehicular environment.

6.3.3 Deep Reinforcement Learning:

In the Reinforcement Learning (RL) paradigm, an agent autonomously learns from past experience in order to maximize some reward signal. Learning to control an agent directly from high-dimensional inputs such as vision and speech is an extremely tedious task, known as the curse of dimensionality. The literature encloses numerous solutions which address this problem (*e.g.*, linear function approximation [36], hierarchical representations [37], state aggregation [38], etc.). These methods greatly rely on the system state representations, thus making the agent not fully autonomous and reducing its flexibility. The use of non-linear function approximation

techniques was relinquished as these methods turned out to be unstable and non-converging when used to represent the action-value function [36]. Only recently, the authors of [39] presented the Deep Q-Network (DQN) algorithm and tested it in a challenging framework composed of several Atari games. DQN achieved dramatically better results than earlier approaches and professional human players and showed a robust ability to learn representations from very high-dimensional input. In [40], the authors compared their DQN algorithm with the best performing methods from the reinforcement learning literature on 49 Atari games. Results showed that the DQN method outperformed the best existing reinforcement learning methods on 43 of the games without incorporating any of the additional prior knowledge about Atari games used by other approaches. Furthermore, the DQN agent performed at a level that was comparable to that of a professional human games tester across the set of 49 games, achieving more than 75% of the human score on more than half of the games. The authors of [39] and [40] exploited an experience replay mechanism ([85]) and a batch reinforcement learning technique ([86]), which made the convergence and stability of the proposed DQN model possible. DQN is foreseen to address the major long-standing challenge of RL by learning to control agents directly from high-dimensional inputs and state spaces. At this level, it becomes necessary to investigate the feasibility of similar techniques in reinforcement learning scenarios where an agent makes decisions that affect the state of the environment. This current work examines a variant of DQN in the context of a V2I communication scenario in a connected vehicular network with multiple RSUs, each of which serves as an IoT-GW.

List Of Symbols

Symbol	Description
--------	-------------

v_i	Speed of vehicle i (m/s)
V_{min}	Minimum vehicle speed (m/s)
V_{max}	Maximum vehicle speed (m/s)
\bar{V}	Average vehicle speed (m/s)
J_i	Vehicle i 's discrete sojourn time in the coverage range of an IoT-GW
H_i	Vehicle i 's download service request size
t_n	Beginning time of time slot n
G	Number of IoT-GWs
G_j	j^{th} IoT-GW
$v_{j,i}$	i^{th} vehicle residing in IoT-GW j 's communication range
$P_{G_j,n}$	Power remaining at j^{th} IoT-GW at t_n
$N_{j,n}$	Number of vehicles residing within j^{th} IoT-GW at t_n
$J_{j,i,n}$	Vector of size $N_{j,n}$ containing the remaining discrete residence times of each vehicle $v_{j,i}$ at t_n
$H_{j,i,n}$	Vector of size $N_{j,n}$ containing the remaining request sizes of each vehicle $v_{j,i}$ at t_n
S_j	Number of sensors within j^{th} communication range
$W_{j,i,n}$	Vector of size $N_{j,n} + S_j$ containing the waiting times of safety messages in the buffer of vehicles and sensors at t_n
$d_{j,i,n}$	Vector of size $N_{j,n}$ containing the separation distances between a vehicle $v_{j,i}$ and IoT-GW G_j at t_n
G_R	Transmission range of the IoT-GWs
x_n	System state at t_n
E	State space
A	Action space
a_n	set of IoT-GWs' actions at time t_n

A_n	Set of admissible actions at time t_n
τ	Length of a time slot (s)
N	Number of time slots in a discharge period
ρ	Vehicular density
B_c	IoT-GW transmission data rate

6.4 V2I Communication Scenario

As illustrated in Figure 6.1, consider a vehicular network consisting of a set of IoT-GWs equipped with large rechargeable batteries. The IoT-GWs are connected to a backend ITS central agent using fiber or cellular communication links. IoT-GWs are equipped with a single radio for downlink communication and operate independently from one another without any interference. The channel access time is assumed to be time-slotted, and IoT-GWs use transmit power control to maintain constant bit rate reception in downlink V2I communications regardless of the vehicle's location within the IoT-GW's coverage range. It is important to mention that, since there is a strong deterministic component of path loss versus distance, and the power consumption of an energy-efficient IoT-GW is dominated by downlink transmission power [87], then an IoT-GW will generally prefer to communicate with nearby vehicles rather than with more distant vehicles in order to minimize its energy consumption.

Vehicles enter the considered roadway segment from any several point according to a Poisson process. The vehicle arrival rate from entry i is equal to the vehicle departure rate from exit i . As such, the vehicular density in the considered road segment is contingent to the vehicular arrival process from the main entry point to that segment (A well-known technique for flow smoothing in traffic modelling[88]). It is assumed that, upon a vehicle's arrival to an IoT-GW's coverage range, it communicates its downlink service request as well as its expected exit point from the road. As

such, a vehicle may travel through one or more IoT-GWs communication zones. In the event where a vehicle or a sensor residing within the communication range of an IoT-GW senses hazardous road conditions (*e.g.*, a speeding car or a traffic collision), it raises a safety flag in order to notify the IoT-GW about the existence of a safety message, which should be communicated to the backend ITS server. It is assumed that there is perfect synchronization between the backend ITS server, the IoT-GWs as well as the vehicles residing within the considered roadway segment with the use of a Global Positioning System (GPS). This work borrows the communication rules from the WAVE protocol suite, where each node in the considered vehicular network (*i.e.*, sensor, vehicle or IoT-GW) periodically broadcasts announcement beacon messages containing information identifying the offered applications (in the case of the IoT-GW), information about the speed, location, direction of travel, download request size, and existence of a sensed safety message (in the case of a vehicle), and finally, sensed data and status information (in the case of a sensor). Vehicles that are requesting a communication link with the IoT-GW coordinate with the nearby IoT-GW in order to establish a connection.

A deep reinforcement learning agent is aware of the current network conditions of the IoT-GWs. At the beginning of each time slot, each IoT-GW forwards information about all the vehicles and sensors residing within its communication range to the central agent. The collected information is then fed to the deep reinforcement learning agent which devises a scheduling decision. As such, the IoT-GW grants access to a single vehicle or sensor with a permission to either upload the safety message it is carrying or continue downloading the requested data (in the case of a vehicle). In case the IoT-GW chooses to receive a safety message, it notifies the selected vehicle or sensor to transmit the carried safety message then the IoT-GW forwards the safety

message to the backend ITS server, which, in turn, broadcasts that message to all IoT-GWs. The safety message is now available at each IoT-GW and can be disseminated to the vehicles. Once a vehicle or a sensor receives the safety message they are carrying through a broadcast, they will drop their safety flag. During the time the IoT-GW is receiving a safety message, its energy consumption is minimal. However, when serving a vehicle's download service request, the IoT-GW's consumed energy increases as the distance to the receiving vehicle increases.

The deep reinforcement learning agent ensures proper communication and coordination among the IoT-GWs in order to maintain a balanced energy between the IoT-GWs. Furthermore, knowing a vehicle's exit point, the agent arranges for the completion of the vehicle's download request before its departure from the considered highway segment. Now, it is true that this work interprets download requests as delay tolerant, however, commuting passengers would appreciate a prompt response time during their residence time within an IoT-GW communication range. For instance, if passengers were requesting to download a video file, they would like to start buffering the data and watch the video as soon as possible. The file can be completely downloaded during their transit in forthcoming IoT-GWs. Finally, and most importantly, the IoT-GWs should manage to broadcast safety messages to concerned vehicles with minimal latency in order to conserve a safe driving environment.

6.5 Vehicular Traffic Model

Consider a vehicular network similar to the one illustrated in Figure 6.1. Several IoT-GWs are deployed in tandem on a long roadway segment with multiple lanes where vehicles enter the road from several entry points. For convenience, we describe the system with unidirectional vehicular traffic; however, the established scheduling

policy is applicable to the bi-directional case¹. Assume that the considered highway segment is experiencing steady free-flow traffic.

According to [43, 42, 50], vehicle arrivals from a particular entry point of the considered segment follow a Poisson process. Consequently, the overall vehicles' arrival process to the entire segment is also a Poisson process [54]. The per-vehicle speeds are i.i.d. random variables in the range $[V_{\min}; V_{\max}]$. These speeds are drawn from a truncated Normal distribution with average \bar{V} and standard deviation σ_V . It is assumed that vehicles maintain their respective speeds constant during their navigation period within the communication range of an IoT-GW [42],[50]. A vehicle's discrete sojourn time J_i has been laid out in Chapter 4.3. An arriving vehicle communicates its speed, download requirements as soon as it enters the coverage range of an IoT-GW. Also, with the use of the GPS, vehicles inform the IoT-GW about their exit points. Consequently, each IoT-GW keeps track of the characteristics of the vehicles residing within its coverage range. Note that, a vehicle i 's download service request size is a uniformly distributed Random Variable H_i between H_{\min} and H_{\max} . H_i is expressed in bits.

6.6 Markov Decision Process Model

In this work, a deep reinforcement learning agent, deployed at each IoT-GW, interacts with the vehicular environment in a sequence of actions, observations, and costs. At each time step, the agent selects an action from the set of feasible actions at that time. The IoT-GW will either suspend its services to listen to an announced safety message, or transmit data to a vehicle with a download request. The agent

¹The IoT-GW will learn the characteristics, dynamics and traffic conditions of the underlying environment, and realizes a scheduling policy accordingly.

then observes the changes in the environment and modifies the system state representation. The agent also receives a reward accordingly. In order to achieve the goals laid out in Section III, all the IoT-GWs should operate in a consistent, orderly and efficient way to balance the vehicular network’s available energy between them, report safety-related messages promptly and deliver a pleasing quality of experience for the travelling vehicles. After each selected action, an IoT-GW receives a step reward, which is a normalized indicator of how well is it contributing to accomplishing the previously-mentioned goals. The objective of learning is to construct an optimal action selection policy at each IoT-GW that serves to maximize the overall network’s performance. Now, it is worthwhile mentioning that, the received single-step reward depends on the whole previous sequence of actions and observations. As such, the impact of an action may only be seen after several hundreds/thousands of time-steps ahead.

6.6.1 Input From the Environment:

Let G be the total number of IoT-GWs. At the beginning of an arbitrary time slot (time t_n), each IoT-GW G_j ($1 \leq j \leq G$) observes the surrounding vehicular environment, collects all the parameters associated with the set of in-range vehicles and sensors, and chooses its action from the set of feasible actions at t_n . The input of IoT-GW G_j from the environment at time t_n is:

- $\overline{P_{G_j,n}}$: a vector of size G containing the remaining energy at each IoT-GW G_j where $j = 1, 2, \dots, G$ and $0 \leq P_{G_j,n} \leq P_t$, where P_t is the IoT-GW’s battery capacity.
- $N_{j,n}$: the number of vehicles residing within G_j ’s communication range, $0 \leq N_{j,n} \leq N_{\max}$.

- $\overline{J_{j,i,n}} = \{J_{j,i,1}, J_{j,i,2}, \dots, J_{j,i,N_{j,n}}\}$: G vectors of respective sizes $N_{j,n}$ containing the remaining discrete sojourn times of each vehicle $v_{j,i}$, $i \in (1, 2, \dots, N_{j,n})$ and $0 \leq J_{j,i,n} \leq J_{\max}$
- $\overline{H_{j,i,n}} = \{H_{j,i,1}, H_{j,i,2}, \dots, H_{j,i,N_{j,n}}\}$: G vectors of respective sizes $N_{j,n}$ containing the remaining request sizes for each vehicle $v_{j,i}$, $0 \leq H_{j,i,n} \leq H_{\max}$.
- $\overline{W_{j,i,n}} = \{W_{j,i,1}, W_{j,i,2}, \dots, W_{j,i,N_{j,n}+S_j}\}$: a vector of size $N_{j,n} + S_j$, where S_j is the number of sensors within G_j 's communication range. $\overline{W_{j,i,n}}$ contains the waiting times of the safety messages in the buffer of vehicles or sensors. In the case where vehicle $v_{j,i}$ or sensor s_j has no safety message to upload, its corresponding $W_{j,i,n}$ is set to a negative value (-1).
- $\overline{d_{j,i,n}} = \{d_{j,i,1}, d_{j,i,2}, \dots, d_{j,i,N_{j,n}}\}$: G vectors of respective sizes $N_{j,n}$ containing the separation distances between G_j and each of its in-range vehicles, $0 \leq d_{j,i,n} \leq G_R$, where G_R is the transmission range of the IoT-GWs.

The agent fully observes the current network state and is able to realize the system state representation at time t_n , denoted herein by x_n .

6.6.2 Immediate Single Step Costs

Let E be the state space where, at any time t_n , the system state $x_n \in E$. Let A_n be the action space at time t_n . The set of feasible actions for G_j at time t_n is $A_{j,n} \subset A_n$. $A_{j,n} = \{0, 1, 2, \dots, N_{j,n}\}$, where if $a_{j,n} = 0$, then G_j will receive a safety message and directly forward it to the backend ITS server. The IoT-GWs will broadcast the safety message in the following time slot. If $a_{j,n}$ is any value k between 1 and $N_{j,n}$, then G_j chooses to transmit packets to the k^{th} vehicle. Also, let $a_n = \{a_{1,n}, a_{2,n}, a_{3,n}\}$.

Let $a_{j,n}$ denote G_j 's action at time step t_n . Let $A_{j,n}$ be the set of admissible actions for G_j at t_n , therefore, $a_{j,n} \in A_{j,n}$. At time t_n , each IoT-GW chooses its

action, and accordingly, the network pays an immediate cost, a scalar value that reflects the righteousness of the IoT-GWs' actions. The immediate cost (negative reward) is the sum of the following normalized quantities:

1. Power consumed by each IoT-GW to transmit data to a selected vehicle. Note that the received signal strength decays exponentially as the separation distance between the IoT-GW and the selected vehicle increases [87]. Assuming that the IoT-GW controls its transmit power to maintain constant bit rate transmission, the power consumed to transmit to closer vehicles is significantly less than that consumed when transmitting to farther ones.
2. Normalized waiting time of the vehicles that have not received any service yet. This incurred cost trains the IoT-GW to minimize the average vehicles' response time.
3. Normalized total delay of completed service requests. Once a vehicle completes its download request, the system is charged a cost corresponding to the total delay of that completed service request. Since the deep reinforcement learning agent thrives to maximize its rewards (minimize negative rewards, *i.e.*, costs), it will learn to minimize the total end-to-end delay of download requests.
4. Penalty incurred on network due to the departure of a vehicle with an incomplete service request. The value of this penalty is a normalized quantity proportional to the remaining request size of the departing vehicle. As such, the IoT-GW is encouraged to fulfill the vehicles' download requests before their departure from the considered roadway. Recall that, a vehicle communicates its exit point upon its arrival to the highway, and the IoT-GWs should coordinate to completely serve that vehicle before its departure from the highway.
5. Penalty incurred on network due to the early cut-off of one of the IoT-GWs.

The network receives this penalty when any of the IoT-GWs shuts down once its battery is drained. The value of this penalty is proportional to the available energy at the other IoT-GWs. As a result, the network learns to balance the power consumption among the IoT-GWs such that when one of the IoT-GWs shuts down, the other IoT-GWs have very little amounts of energy remaining in their batteries.

Now, it is important to mention that, even if the impact of the occurrence of such event unveils in a single time-step (*i.e.*, when a vehicle departs or when an IoT-GW shuts down), the deep reinforcement learning agent realizes that the sequence of all its previous actions lead to this current system state. This is a clear example that the feedback about an action may sometimes be received after many thousands of time steps have elapsed. The goal of the agent is to interact with the IoT-GWs and select actions that maximize future rewards.

6.6.3 Proposed Solution Approach:

Recall that, the objective of this work is to realize an optimal scheduling policy which will govern the operation of the IoT-GWs in order to minimize the total expected negative rewards and, as a result, achieve the goals laid out in Section III. This problem has been formulated as an MDP whose states are modelled as a Markov chain, and a state-action dependent cost is incurred at each stage. A deep Q-learning algorithm is exploited to realize an optimal policy for the above-presented MDP. The deep Q-learning algorithm is presented in Algorithm 2.2 in Chapter 2.3.

Table 6.2: Simulation Input Parameters

Parameter	Value
IoT-GW Battery Capacity	$3 \times 50Ah$ batteries
Time slot length	$\tau = 0.1$ (s)
Vehicular densities	$\rho \in [2; 11]$ (veh/km)
Min and Max vehicle speed	$[60; 140]$ (km/h)
Min and Max request size	$[2; 12]$ (MB)
IoT-GW covered segment	$D_C = 1000$ (m)
Vehicles and IoT-GW radio range	500 (m)
Channel data bit rate	$B_c = 9$ (Mbps)
Learning rate	$\alpha(n) = 1/n$
Discount factor	$\gamma = 0.5$

6.7 Simulation and Results Discussion

6.7.1 The Learning Phase:

At the beginning of each time slot, each IoT-GW forwards the underlying network characteristics to the deep reinforcement learning agent. As such, the agent’s input, detailed in Section VII.A becomes ready to be fed to the neural network. The input to the neural network consists of $(G \times N_{max} \times 4)$ corresponding to the characteristics of all vehicles within all G IoT-GWs, and $\sum_j S_j$ that correspond to the safety messages’ waiting time in the sensors’ buffers. Recall that S_j is the number of sensors within G_j ’s communication range. We use a neural network consisting of two hidden layers as it was shown in [89] that two stages of feature extraction yields better accuracy than one. The first hidden layer of the neural network convolves $G (N_{max} \times 4)$ filters and then applies a rectifier nonlinearity [89]. The second hidden layer convolves $G (N_{max} \times 1)$ filters, again followed by a rectifier nonlinearity. The final hidden layer is fully-connected and consists of $(G \times N_{max})$ rectifier units. The output layer is a fully connected linear layer with a single output for each valid action. The literature encloses several deep reinforcement learning methods that use the system state and

action as input to the neural network, *e.g.* [90]. The disadvantage of such techniques is the need for a separate forward pass to compute the Q-value of each action, resulting in a cost that scales linearly with the number of actions. The work of [40] suggested an architecture in which there is a separate output for each possible action, and the system state is the only input to the neural network. The outputs correspond to the predicted Q-values of the individual actions for that input state. As such, a single forward pass through the neural network can compute the Q-values for all possible actions in a given state. This current work adopts this method to train the neural network. In the training phase of the simulations, an ϵ -greedy method was applied where ϵ decreased linearly from 1.0 to 0.1 over the first million training samples, and fixed at 0.1 thereafter. The agent was trained on a total of 50 million samples, which corresponds to two months of actual vehicular network operation. The replay memory used in training the neural network was of size 1 million most recent training samples, and the size of the minibatches was 100. Table 6.2 lists the values of all the hyper-parameters and optimization parameters as well as the simulator’s input parameters. After the training phase, the resulting neural network is exploited in order devise scheduling decisions for the IoT-GWs.

6.7.2 Simulation Setup:

In the simulations, the vehicular traffic model presented in Section V is adopted. Thorough simulations were conducted using the Simulation for Urban MObility (SUMO) simulator [51], which provides realistic vehicular mobility traces used as input for the simulation. The presented results herein were averaged over multiple runs to ensure a 95 % confidence interval. The proposed DQN algorithm is compared with four other scheduling algorithms namely:

1. Random: Random vehicle selection algorithm where, at time t_n , the IoT-GW

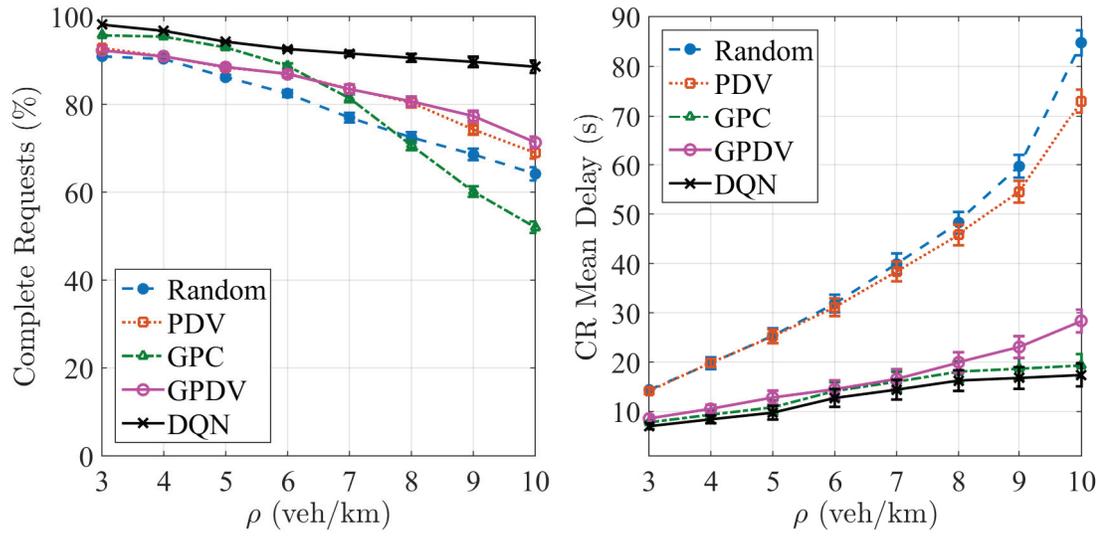
randomly chooses a vehicle within its communication range to be served [11].

2. PDV: Prioritizing Departing Vehicles, where at time t_n , the IoT-GW randomly chooses a vehicle from the set of vehicles, which are departing from the next exit.
3. GPC: Greedy Power Conservation algorithm where, at time t_n , the IoT-GW chooses the closest vehicle which contributes to the lowest energy consumption compared to the remaining vehicles residing within its communication range.
4. GPDV: Greedily Prioritize Departing Vehicles, where at time t_n , the IoT-GW chooses the closest vehicle from the set of vehicles, which are departing from the next exit.

Under all the above scheduling algorithms, if there exists a safety message flag within any of the IoT-GWs' communication ranges, the corresponding IoT-GW will halt its services and notify the vehicle or sensor carrying that message to transmit it. The IoT-GW will then forward the safety message to the backend ITS server, and in the forthcoming time slot, all the IoT-GWs will broadcast the received safety message. In this section, the performance of the proposed DQN algorithm is evaluated in terms of: *a)* Vehicles' Completed Request (CR) Percentage which is the percentage of completely fulfilled vehicle requests, *b)* CR Mean Delay and *c)* Network Lifetime being the time until one of the IoT-GWs cuts-off.

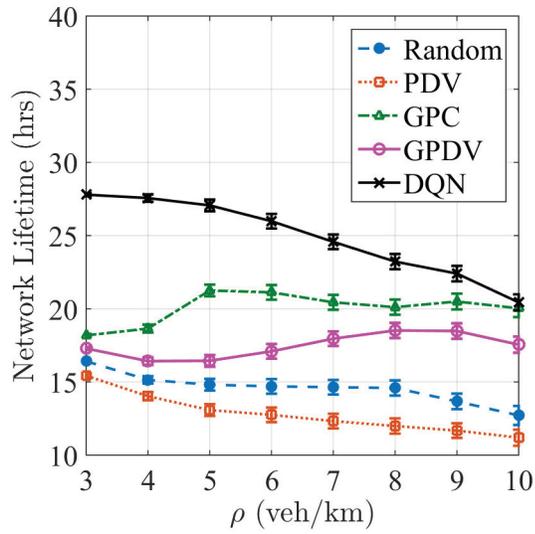
6.7.3 Simulation Results:

Figure 6.2 evaluates the performance of the deep reinforcement learning agent when compared with the four previously described scheduling algorithms. Figure 6.2(a) plots the percentage of vehicles leaving the vehicular network with a complete service request as a function of the vehicular density. It is clear that the number of CRs



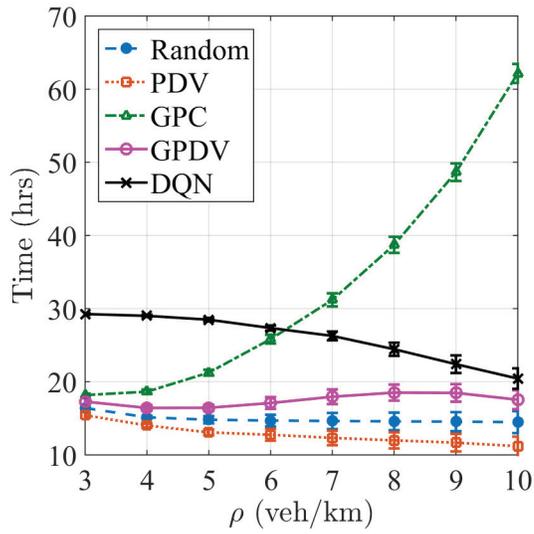
(a) CR Percentage

(b) CR Mean Delay

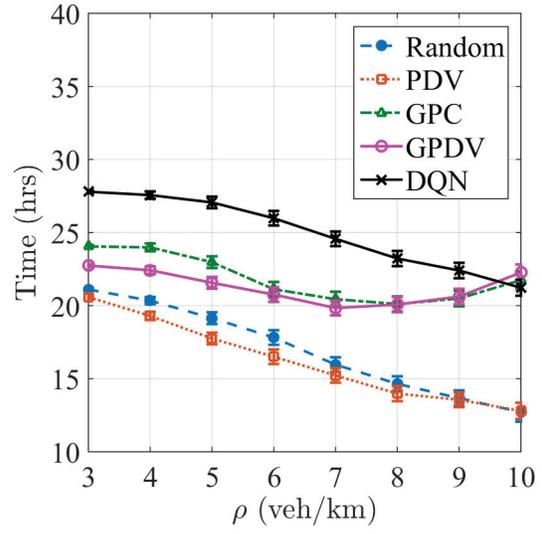


(c) Network Lifetime

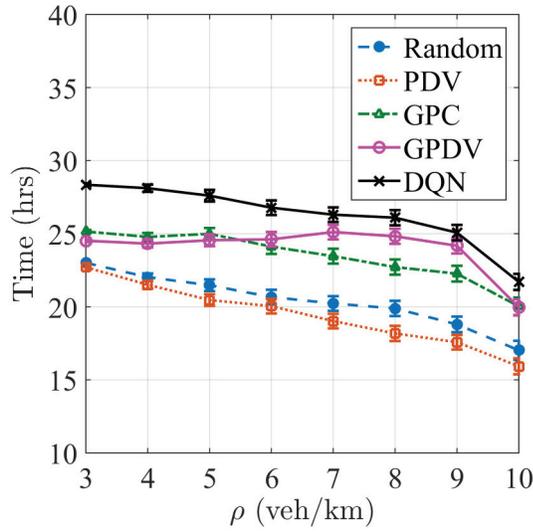
Figure 6.2: Performance Evaluation and Comparisons



(a) Lifetime of G_1



(b) Lifetime of G_2



(c) Lifetime of G_3

Figure 6.3: IoT-GWs' Lifetime

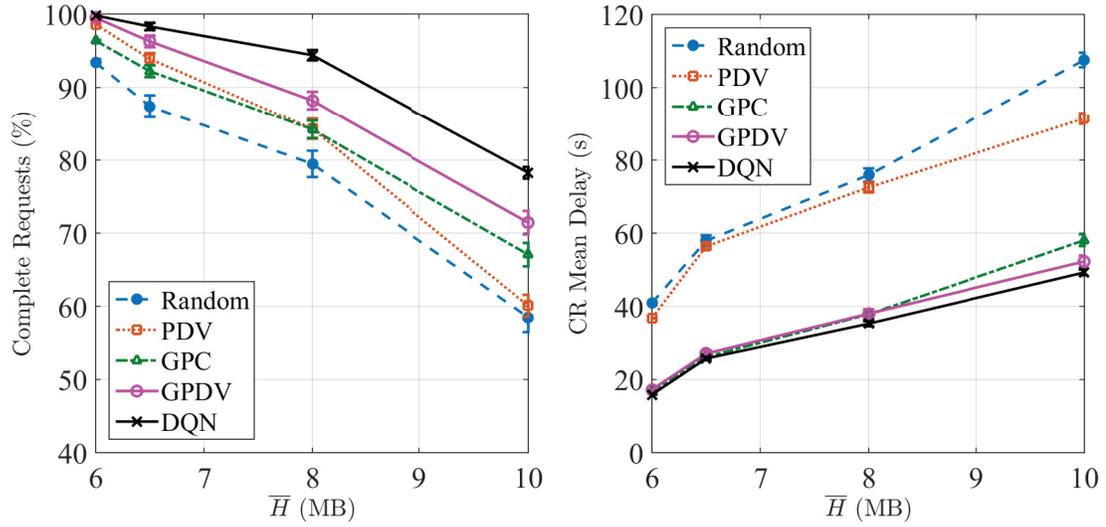
decreases as ρ increases under all the scheduling algorithms. In fact, as ρ increases, the likelihood of selecting a certain vehicle will decrease, independent of the scheduling discipline. Consequently, a vehicle will spend less time receiving service and the total number of vehicles departing from the vehicular network’s communication range with complete service requests will decrease. Figure 6.2(a) also shows that the DQN method outperforms all four other scheduling algorithms in terms of CR percentage. Under a random selection policy, there is no vehicle prioritization, and therefore, the number of vehicles whose associated download request is fulfilled decreases remarkably as the vehicular density increases. Now, for GPC, each IoT-GW is admitting to service the vehicle which resides in the minimal energy consumption zone compared to the set of remaining in-range vehicles. Whenever ρ is small, a large portion of the vehicles have enough time to complete their download request while being the closest to the IoT-GW, however, when ρ increases, the time during which a vehicle is closest to the IoT-GW is not enough to fully complete its download request. This clearly results in deteriorated QoS levels for larger values of ρ when the IoT-GWs are implementing a greedy power conservation scheduling policy. Under both PDV and GPDV, the IoT-GWs are prioritizing the vehicles that are departing from the next exit, and consequently, the two scheduling algorithms show similar results in terms of CR percentage. Furthermore, as ρ increases, and due to the prioritization notion, PDV and GPDV show better results than the random selection algorithm as well as the GPC method. Finally, it is clear from Figure 6.2(a) that the DQN agent outperforms all the scheduling algorithms it is compared to. Recall that, during the training phase, a vehicle departing from the network with an incomplete service request is an undesired event which incurs a high cost. Therefore, the DQN agent will learn to avoid such an event and as such, the deployment of the well-trained DQN agent guarantees that the majority of departing vehicles have completed their

download service request.

Figure 6.2(b) plots the CRs mean total delay as a function of vehicular density. Under the random and PDV algorithms, it is clear that most of the vehicles are receiving service from multiple IoT-GWs. On the other hand, under GPC, GPDV as well as DQN, it is evident that most of the CRs are fulfilled by a single IoT-GW as the mean CR delay is less than the average vehicle residence time within an IoT-GW's communication range (*i.e.*, at an average vehicles' speed of 100 km/h, a vehicle spends 36 seconds in an IoT-GW's communication range). Although GPC and GPDV show good results in terms of CR mean delay, however, DQN outperforms both these algorithms. This indicates that the DQN agent is sometimes choosing to serve a vehicle even if it is neither the closest nor departing soon since that decision contributes to minimizing the mean CR total delay. Figure 6.2(c) plots the network lifetime, defined as the time until an IoT-GW cuts-off. The improvement DQN achieves over the other four scheduling algorithms varies from 5 to 60 % in terms of the network lifetime. This is a clear implication of the penalty incurred on the DQN agent, in the training phase, due to the early cut-off of one of the IoT-GWs. Recall that, since this penalty is proportional to the remaining energy at the other IoT-GWs, the DQN agent learns to balance the power consumption among the tandem of IoT-GWs. This fact is illustrated in Figure 6.3 where it is clear that all three IoT-GWs have a similar lifetime. It is worthwhile mentioning that, according to [87], the power consumption increases exponentially as the distance between the transmitter and receiver increases. As a result, under GPC, the lifetime of G_1 increases as the vehicular density increases. The interpretation of this result is as follows: as more vehicles reside within the IoT-GW's coverage range, it becomes more likely that the closest vehicle which still requires service is very close to that IoT-GW. As such, the latter consumes minimal amounts of

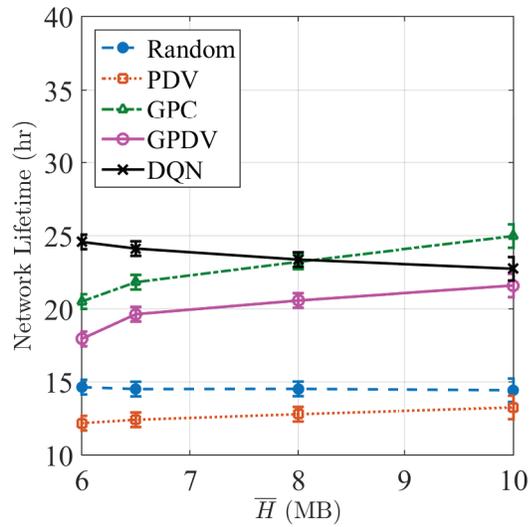
energy to serve the closest vehicle, and as such, its battery lifetime increases. However, this is only true for G_1 since when the vehicles arrive to G_2 and G_3 , a large portion of them have completed their service, and since an IoT-GW has to choose the closest vehicle which still requires service, it is less likely to find that vehicle in low energy consumption zones. As such, under GPC, the network lifetime is influenced by the lifetime of the downstream IoT-GWs, which is illustrated in Figure 6.3.

Figure 6.4 compares the performance of the DQN algorithm with the other benchmark scheduling algorithms as the vehicle mean service request is varied between 6 and 10 MB under a fixed vehicular density of $\rho = 6$ veh/km. Figures 6.4(a) and 6.4(b) show that the percentage of CRs as well as their mean delay decrease as the vehicles' mean service size increases under all scheduling algorithms. This is an expected and consistent result. Also, it is clear from Figures 6.4(a) and 6.4(b) that the DQN method achieves better results in terms of completed requests percentage and mean delay when compared to the other four scheduling algorithms. In terms of network lifetime, the DQN method outperforms all its counterparts for vehicles' mean request sizes less than 8 MB. When $\bar{Q} = 10$ MB, the GPC algorithm results in a longer network lifetime. This is due to the fact that, when arriving vehicles request to download large file sizes, it becomes less likely to serve these vehicles by a single IoT-GW, and chances are high that the closest vehicle to an IoT-GW still requires service. As earlier mentioned, an IoT-GW serving close-by vehicles consumes less amounts of energy, and as such, it extends its battery lifetime. It is true that GPC outperforms DQN in terms of network lifetime for larger values of vehicle requests, however, this is at the expense of deteriorated QoS levels revealed by a smaller percentage of complete service requests and longer CR delays.



(a) CR Percentage

(b) CR Mean Delay



(c) Network Lifetime

Figure 6.4: Performance Evaluation for $\rho = 6$ (veh/km)

6.8 Conclusion and Future Research Direction

The proper inauguration of a full-fledged, smart and efficient ITS is foreseen to support the legitimate realization of the next generation 5G network by providing several benefits including easier content sharing and efficient computation offloading. As such, the need to establish an intelligent and connected vehicular environment has become an emerging research priority. This work develops an artificial DQN agent to support the efficient operation of a vehicular network by learning a scheduling policy from high-dimensional inputs using end-to-end deep reinforcement learning. This agent derives efficient representations of the environment, learns from past experience, and progress towards the realization of a successful scheduling policy in order to extend the lifetime of a green, safe and connected vehicular network and achieve acceptable levels of QoS. The proposed DQN algorithm outperforms several existing scheduling benchmarks in terms of completed request percentage (improvement of 10-25 %), mean request delay (improvement of 10-15 %) and total network lifetime (improvement of 5-65 %) under variable vehicular densities and vehicle request sizes.

Chapter 7

Discussion and Future Directions

7.1 Conclusions

This thesis addressed multiple challenges associated with the inauguration of a safe, smart, and green intelligent transportation system. Mainly, it focused on the challenges of connectivity establishment, medium access control schemes and the energy-efficient operation of a vehicular network. At first, Chapter 2 of this thesis provided a comprehensive overview of the state-of-the-art medium access control protocol standards for wireless access in vehicular environments, namely: *a)* the IEEE 802.11p protocol and *b)* the IEEE 1609.4 protocol. This was followed by a survey of a selection of existing work addressing numerous problems and limitations of these protocols. Chapter 2 also provided a concise description of Markov Decision Processes and their resolution using machine learning methods, particularly, reinforcement learning and deep learning. These latter techniques were exploited in Chapters 5 and 6 respectively for the purpose of optimizing the operation of an efficient vehicular network.

Motivated by the recent interests in a fully connected vehicular network, Chapter 3 of this thesis examined the probability of establishing a connectivity path between

vehicles residing in dark areas of a roadway and a remote RSU. After carefully identifying the events whose occurrences incur changes in a vehicular network's topology, the average end-to-end delivery delay was derived and validated through simulations. The per-vehicle and network throughputs were evaluated using a Markov Chain framework. The reported results showed that a path is highly likely to be available between a source vehicle and a destination RSU under high vehicular densities; however, the collision probability under these high vehicular densities caused throughput deterioration.

Next, Chapter 4 of this thesis examined two novel MAC schemes to support V2I communications. A comprehensive study of the system from a vehicle's perspective was presented. Then, a mathematical framework was established with the objective of modelling a vehicle's on-board unit's queue and evaluate its performance under the newly proposed MAC algorithms, namely: Random Vehicle Selection (RVS) and Least Residual Time (LRT), in terms of several quality-of-service metrics. The reported results showed that LRT outperformed RVS in terms of vehicular average service time. This was not the case in terms of the vehicular response time since under LRT, a vehicle waited longer amounts of time before its remaining residence time became the least and hence, gained access to communicate with the RSU.

Next, and due to the emerging discussion and condemnation of the growing energy consumption of mobile networks, Chapter 6 of this thesis addresses the problem of RSU perseverance and productiveness in the context of a vehicular scenario where the RSU is energy-limited. The problem was formulated using a Markov Decision Process and resolved using reinforcement learning techniques. The trained agent engaged in exploration and exploitation phases and learned to govern the operation of a RSU to efficiently utilize its available energy and maintain an acceptable QoS during the battery discharge period. The reported results showed that the deployment of the

proposed reinforcement learning agent equips the RSU with artificial intelligence, which served to maintain the RSU's operation throughout the whole discharge period and limited the number of vehicles departing from the RSU's coverage range with an incomplete service request.

Finally, driven by the expeditious evolution of the Internet of Vehicles as well as the user's persisting demand for continuous connectivity, Chapter 7 of this thesis addressed both safety and Quality-of-Service concerns in a connected, smart, safe and green vehicular network. Bounded by the limitations of classical reinforcement learning methods, this work resorted to recent advances in training deep neural networks to evaluate the righteousness of the RSU's actions. As a result, a deep reinforcement learning agent was founded to learn a scheduling policy from high-dimensional inputs corresponding to the characteristics of the underlying network. The reported results showed that the proposed deep Q-network served to extend the lifetime of the battery-powered vehicular network while promoting a safe environment that provisioned acceptable QoS levels.

7.2 Future Work

For the past two decades, the research community has been actively crafting the foundation of a life-changing intelligent transportation system. Ever since, the world has witnessed the birth of an enormous number of ITS-enabling applications, whose associated challenges attracted significant research attention.

This final section of this thesis highlights prominent ITS applications and persisting challenges and recommends contemporary methods and techniques that promote the proper functionality of an efficient ITS.

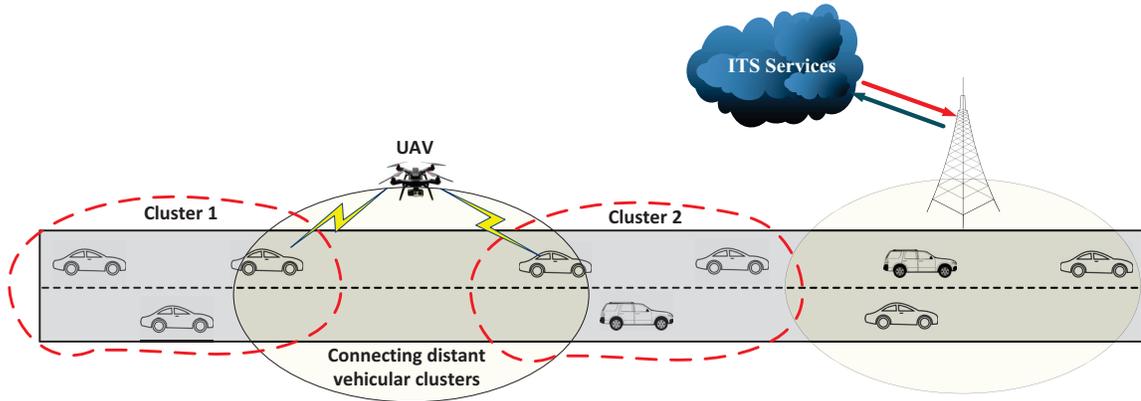


Figure 7.1: Enhanced Connectivity Using UAVs

7.2.1 Secure Communications in a Vehicular Environment

With the high expectations of the widespread penetration of communicating vehicles, the security deficiencies of vehicular networks have come under scrutiny. Owing to the nature of vehicular applications, security flaws can result not only in packet losses or network dysfunctions but also in fatal accidents. Security mechanisms that can account for privacy, authentication, integrity, and non-repudiation while having an acceptable communication overhead are necessary to the operation of ITS applications. A wide range of studies on vehicular networking security problems and solutions have been recently proposed (*e.g.*, [91] and [92]). However, many security challenges still remain to be solved to support and enable a highly secure ITS infrastructure and vehicular communications.

7.2.2 UAV-Aided Multi-Hop Connectivity

Unmanned Aerial Vehicles (UAVs) are an emerging technology that can be harnessed for military, public and civil applications. The U.S. military has been exploiting UAVs for more than 25 years for border surveillance, reconnaissance and striking purposes. UAVs may be further utilized by the public authorities for public safety

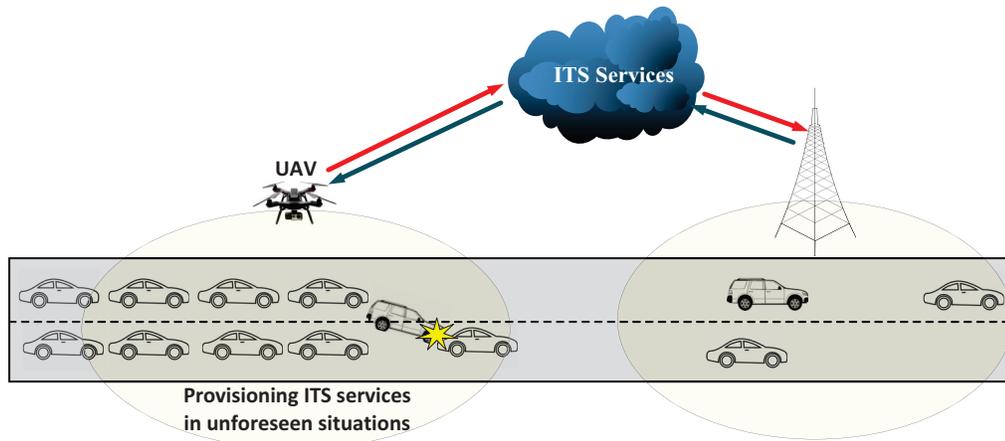


Figure 7.2: UAVs as Mobile RSUs

and transportation management to provide timely disaster warnings and assist in speeding up rescue and recovery operations when the public communication network gets crippled. A UAV-related application discussed in this very thesis is illustrated in Figure 7.1 where a broken connectivity path may be re-established using a UAV. The use of UAVs in a scenario similar to the one illustrated in Figure 7.1 is expected to *a)* increase the probability of an available path and *b)* mitigate the impact of uncooperative vehicles on the overall network connectivity. The characteristics of these UAVs such as their speed, number, capacity, as well as communication capabilities are to be determined after careful assessment of this application.

7.2.3 UAV-Aided Temporary V2I Services

In the scenario illustrated in Figure 7.2, UAVs behave like mobile RSUs whose flexible deployment allows vehicles to procure access to ITS services under unforeseen traffic conditions. Knowing that these UAVs are battery-powered, the efficient use of their available energy is a must. Chapter 5 of this thesis has established a solid ground for energy-efficient RSU operation in vehicular environments; however, the UAVs have their unique characteristics that require revised operational mechanisms.

Bibliography

- [1] Federal Communications Commission et al. Amendment of the commissions rules regarding dedicated short-range communication service in the 5.850-5.925 ghz band. Technical report, fcc 02-302. Technical report, FCC, 2002.
- [2] L. Davidson. How connected cars are driving the internet of things. Technical report, The Telegraph, 2015.
- [3] J. Frazer. Smart cities: Vehicle to infrastructure and adaptive roadway lighting communication standards. Technical report, GridAptive Technologies, 2012.
- [4] R. Atallah et al. Modelling of multi-hop inter-vehicular path formation for connecting far vehicles to rsus. In *2015 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2015.
- [5] P. Neelakantan et al. Computation of minimum transmit power for network connectivity in vehicular ad hoc networks formed by vehicles with random communication range. *International Journal of Communication Systems*, 27(6), 2014.
- [6] S. Pierce. Vehicle-infrastructure integration (vii) initiative: Benefit-cost analysis: Pre-testing estimates. March 2007.
- [7] R. Atallah et al. Vehicular networking: A survey on spectrum access technologies and persisting challenges. *Vehicular Communications*, 2(3), 2015.

- [8] R. Atallah et al. Multi-hop vehicle-to-infrastructure communications: A feasibility study, modelling and performance analysis. *IEEE Transactions on Vehicular Technology*, 2016.
- [9] R. Atallah et al. Throughput analysis of ieee 802.11 p-based multi-hop v2i communications. In *World of Wireless, Mobile and Multimedia Networks (WoW-MoM), 2015 IEEE 16th International Symposium on a. IEEE*, 2015.
- [10] R. Atallah et al. A novel drive-thru internet channel access scheme. *IEEE Communications Letters*, 18(6), 2014.
- [11] R. Atallah et al. Modeling and performance analysis of medium access control schemes for drive-thru internet access provisioning systems. *IEEE T-ITS*, 16(6), 2015.
- [12] R. Atallah et al. A reinforcement learning technique for optimizing downlink scheduling in an energy-limited vehicular network. *IEEE Transactions on Vehicular Technology*, 2016.
- [13] J. Kenney. Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, 99(7), 2011.
- [14] IEEE Standards Association et al. 802.11 p-2010ieee standard for information technologylocal and metropolitan area networksspecific requirementspart 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments. URL http://standards.ieee.org/findstds/standard/802.11_p-2010.html.
- [15] Wei Sun, Hesheng Zhang, Cheng Pan, and Jun Yang. Analytical study of the ieee 802.11 p edca mechanism. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 1428–1433. IEEE, 2013.

- [16] Jose R Gallardo, Dimitrios Makrakis, and Hussein T Mouftah. Performance analysis of the edca medium access mechanism over the control channel of an ieee 802.11 p wave vehicular network. In *Communications, 2009. ICC'09. IEEE International Conference on*, pages 1–6. IEEE, 2009.
- [17] K. Hafeez et al. Performance analysis and enhancement of the dsrc for vanet's safety applications. *IEEE Transactions on Vehicular Technology*, 62(7), 2013.
- [18] Stephan Eichler. Performance evaluation of the ieee 802.11 p wave communication standard. In *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pages 2199–2203. IEEE, 2007.
- [19] Chong Han, Mehrdad Dianati, Rahim Tafazolli, Ralf Kernchen, and Xuemin Shen. Analytical study of the ieee 802.11 p mac sublayer in vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 13(2):873–886, 2012.
- [20] Y. Wang et al. Ieee 802.11 p performance evaluation and protocol enhancement. In *Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference on*. IEEE, 2008.
- [21] C. Campolo et al. Multichannel communications in vehicular ad hoc networks: a survey. *IEEE Communications Magazine*, 51(5), 2013.
- [22] C. Campolo et al. Modeling broadcasting in ieee 802.11 p/wave vehicular networks. *IEEE Communications Letters*, 15(2), 2011.
- [23] J. Misic et al. Performance characterization for ieee 802.11 p network with single channel devices. *IEEE Transactions on Vehicular Technology*, 60(4), 2011.
- [24] Claudia Campolo, Antonella Molinaro, Alexey Vinel, and Yan Zhang. Modeling prioritized broadcasting in multichannel vehicular networks. *IEEE Transactions on Vehicular Technology*, 61(2):687–701, 2012.

- [25] Giuseppe Bianchi. Performance analysis of the ieee 802.11 distributed coordination function. *IEEE Journal on selected areas in communications*, 18(3):535–547, 2000.
- [26] Elmar Schoch and Frank Kargl. On the efficiency of secure beaconing in vanets. In *Proceedings of the third ACM conference on Wireless network security*, pages 111–116. ACM, 2010.
- [27] J. Alcaraz et al. Control-based scheduling with qos support for vehicle to infrastructure communications. *IEEE Wireless Communications*, 16(6), 2009.
- [28] MH. Cheung et al. Dora: Dynamic optimal random access for vehicle-to-roadside communications. *IEEE JSAC*, 30(4), 2012.
- [29] D. Niyato et al. A unified framework for optimal wireless access for data streaming over vehicle-to-roadside communications. *IEEE Transactions on Vehicular Technology*, 59(6), 2010.
- [30] WL. Tan et al. Analytical models and performance evaluation of drive-thru internet systems. *IEEE JSAC*, 29(1), 2011.
- [31] R. Zhang et al. A unified tdma-based scheduling protocol for vehicle-to-infrastructure communications. In *2013 WCSP*. IEEE, 2013.
- [32] R. Zhang et al. A novel centralized tdma-based scheduling protocol for vehicular networks. *Intelligent Transportation Systems, IEEE Transactions on*, 16(1):411–416, 2015.
- [33] R. Sutton et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

- [34] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.
- [35] W. B. Powell. What you should know about approximate dynamic programming. *Naval Research Logistics (NRL)*, 56(3), 2009.
- [36] J. Tsitsiklis et al. An analysis of temporal-difference learning with function approximation. *Automatic Control, IEEE Transactions on*, 42(5):674–690, 1997.
- [37] P. Dayan et al. Feudal reinforcement learning. In *Advances in neural information processing systems*, pages 271–271. Morgan Kaufmann Publishers, 1993.
- [38] S. Singh et al. Reinforcement learning with soft state aggregation. *Advances in neural information processing systems*, pages 361–368, 1995.
- [39] V. Mnih et al. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.
- [40] V. Mnih et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
- [41] A. Abdrabou et al. Probabilistic delay control and road side unit placement for vehicular ad hoc networks with disrupted connectivity. *IEEE Journal on Selected Areas in Communications*, 29(1), 2011.
- [42] M. Khabazian et al. A performance modeling of connectivity in vehicular ad hoc networks. *IEEE TVT*, 57(4), 2008.
- [43] S. Yousefi et al. Analytical model for connectivity in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 57(6), 2008.

- [44] Z. Yan et al. Fair downlink traffic scheduling for energy sustainable vehicular roadside infrastructure. In *2014 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, 2014.
- [45] K. Abboud et al. Stochastic analysis of a single-hop communication link in vehicular ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, 15(5), 2014.
- [46] M. Khabazian et al. Analysis of continuous communication availability in vehicular ad hoc networks. *IEEE Systems Journal*, 7(1), 2013.
- [47] J. He et al. Delay minimization for data dissemination in large-scale vanets with buses and taxis. *IEEE Transactions on Mobile Computing*, 15(8), 2016.
- [48] J. Fukuyama. A delay time analysis for multi-hop v2v communications over a linear vanet. In *2009 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2009.
- [49] H. Wu et al. Spatial propagation of information in vehicular networks. *IEEE Transactions on Vehicular Technology*, 58(1), 2009.
- [50] M. Khabbaz et al. A simple free-flow traffic model for vehicular intermittently connected networks. *IEEE T-ITS*, 13(3), 2012.
- [51] M. Behrisch et al. Sumo—simulation of urban mobility: an overview. In *Conference on Advances in System Simulation*. ThinkMind, 2011.
- [52] ITS Standards Fact Sheets. IEEE 1609 - family of standards for wireless access in vehicular environments (wave). available online - http://www.standards.its.dot.gov/fact_sheet.asp?f=80.

- [53] J. Huang et al. Vehicle density based forwarding protocol for safety message broadcast in vanet. *The scientific world journal*, 2014, 2014.
- [54] E. Cascetta. *Transportation systems engineering: theory and methods*, volume 49. Springer Science & Business Media, 2013.
- [55] D. Miorandi et al. Connectivity in one-dimensional ad hoc networks: a queueing theoretical approach. *Wireless Networks*, 12(5), 2006.
- [56] N. Wisitpongphan et al. Routing in sparse vehicular ad hoc wireless networks. *IEEE journal on Selected Areas in Communications*, 25(8), 2007.
- [57] L. Kleinrock. *Queueing systems, volume i: theory*. 1975.
- [58] W. Viriyasitavat et al. Dynamics of network connectivity in urban vehicular networks. *IEEE Journal on Selected Areas in Communications*, 29(3), 2011.
- [59] B. Alawieh et al. Investigating the performance of power-aware ieee 802.11 in multihop wireless networks. *IEEE Transactions on Vehicular Technology*, 58(1), 2009.
- [60] Christoph Sommer, Reinhard German, and Falko Dressler. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15, January 2011.
- [61] Francesco Malandrino, Claudio Casetti, Carla-Fabiana Chiasserini, and Marco Fiore. Offloading cellular networks through its content download. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2012 9th Annual IEEE Communications Society Conference on*. IEEE, 2012.
- [62] M. Khabbaz et al. Modeling and analysis of an infrastructure service request

- queue in multichannel v2i communications. *IEEE Transactions on Intelligent Transportation Systems*, 15(3), 2014.
- [63] S. Ou et al. A selective downlink scheduling algorithm to enhance quality of vod services for wave networks. *EURASIP Journal on Wireless Communications and Networking*, 2009(1), 2008.
- [64] M. Khabbaz et al. Delay-aware data delivery in vehicular intermittently connected networks. *IEEE Transactions on Communications*, 61(3), 2013.
- [65] Transport Canada. Canadian motor vehicle traffic collision statistics 2014 - <http://www.tc.gc.ca>.
- [66] E. Strinati et al. Holistic approach for future energy efficient cellular networks. *e & i Elektrotechnik und Informationstechnik*, 127(11), 2010.
- [67] US Department of Energy NREL. Transportation and the future of dynamic mobility systems, 2016.
- [68] D. Lister. An operators view on green radio. *Keynote Speech, GreenComm*, 2009.
- [69] K. Tweed. Why cellular towers in developing nations are making the move to solar power. *Scientific American*, 2013.
- [70] V. Chamola et al. Solar powered cellular base stations: current scenario, issues and proposed solutions. *IEEE Communications Magazine*, 54(5), 2016.
- [71] R. Atallah et al. Energy harvesting in vehicular networks: A contemporary survey. *IEEE WCM*, PP(99), 2015.
- [72] Y. Zhang et al. On scheduling vehicle-roadside data access. In *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, pages 9–18. ACM, 2007.

- [73] Y. Zhuang et al. On the uplink mac performance of a drive-thru internet. *Vehicular Technology, IEEE Transactions on*, 61(4):1925–1935, 2012.
- [74] Q. Ibrahim. Design, implementation and optimisation of an energy harvesting system for vehicular ad hoc networks’ road side units. *Intelligent Transport Systems, IET*, 8(3), 2014.
- [75] A. Hammad et al. Downlink traffic scheduling in green vehicular roadside infrastructure. *IEEE TVT*, 62(3), 2013.
- [76] A. Khezrian et al. Energy efficient scheduling in green vehicular infrastructure with multiple roadside units. 2015.
- [77] F. Zou et al. Energy-efficient roadside unit scheduling for maintaining connectivity in vehicle ad-hoc network. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication*, pages 1–8. ACM, 2011.
- [78] R. Roess et al. Traffic engineering. 2011.
- [79] J. Lehoczky et al. The rate monotonic scheduling algorithm: Exact characterization and average case behavior. In *Real Time Systems Symposium, 1989., Proceedings*. IEEE, 1989.
- [80] A. Reis et al. Deploying roadside units in sparse vehicular networks: what really works and what does not. *IEEE Transactions on Vehicular Technology*, 63(6), 2014.
- [81] J. Cheng et al. Routing in internet of vehicles: A review. *IEEE Transactions on Intelligent Transportation Systems*, 16(5), 2015.

- [82] M. Gerla et al. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In *Internet of Things (WF-IoT), 2014 IEEE World Forum on*. IEEE, 2014.
- [83] N. Kumar et al. Bayesian coalition game as-a-service for content distribution in internet of vehicles. *IEEE Internet of Things Journal*, 1(6), 2014.
- [84] B. Zhang et al. Energy-efficient roadside units deployment in vehicular ad hoc networks. In *6th International Conference on Wireless, Mobile and Multi-Media (ICWMMN 2015)*. IET, 2015.
- [85] J. Long Lin. Programming robots using reinforcement learning and teaching. In *AAAI*, pages 781–786, 1991.
- [86] S. Lange et al. Batch reinforcement learning. In *Reinforcement Learning*, pages 45–73. Springer, 2012.
- [87] T. Rappaport et al. *Wireless communications: principles and practice*, volume 2. Prentice Hall PTR New Jersey, 1996.
- [88] Transport for London. Traffic modelling guidelines: TfL traffic manager and network performance best practice, 2010.
- [89] K. Jarrett et al. What is the best multi-stage architecture for object recognition? In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009.
- [90] M. Riedmiller. Neural fitted q iteration—first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*. Springer, 2005.
- [91] J. Isaac et al. Security attacks and solutions for vehicular ad hoc networks. *IET communications*, 4(7), 2010.

- [92] J. Petit et al. Potential cyberattacks on automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 16(2), 2015.