

Efficient Traffic Control Protocols for Vehicular Ad-Hoc Networks

by

Maram Younis Saleh Bani Younes

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the Ph.D. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

© Maram Younis Saleh Bani Younes, Ottawa, Canada, 2015

Abstract

Traffic efficiency applications over road networks have been investigated recently using VANETs. This area of research is primarily concerned with increasing the traffic fluency over road networks. In this thesis, we first propose an efficient and accurate protocol to detect congested road segments in a downtown area using VANETs. We refer to this protocol as the Efficient COngestion DETection (ECODE) protocol. ECODE evaluates three different traffic characteristics of each road segment including traffic speed, traffic density, and the time required to travel the segment. Moreover, ECODE evaluates traffic characteristics and detects the congestion level in each direction of the road segment.

In addition, we propose an intelligent, dynamic, distributed, and real-time path recommendations protocol. We refer to this protocol as Intelligent path reCOmenDation (ICOD) protocol. ICOD is the first path recommendation protocol that does not rely on a central database of gathered traffic data for each area of interest. Eliminating centralized behavior resolves bottleneck as well as single point of failure problems, which in turn minimizes congestion and collision problems in VANETs. Furthermore, ICOD selects the path towards each destination in a hop-by-hop manner, which makes the turn decision at each road intersection more accurate and real-time. Different variants of ICOD are introduced that consider travel time, travel distance, fuel consumption, gas emissions, and context-awareness of each road segment parameters.

Moreover, two traffic balancing mechanisms are proposed in this thesis to distribute traffic over the road network evenly, namely Bal-Traf and Abs-Bal. These mechanisms eliminate the highly congested road segment scenarios that are caused by the path recommendation protocol. Bal-Traf detects and eliminates the highly congested output road segment at each road intersection. However, Abs-Bal aims to keep the traffic density balanced among all output road segments at each road intersection.

Finally, we propose an Intelligent Traffic Light Controlling (ITLC) algorithm to schedule the phases of each traffic light at isolated road intersections. This algorithm aims to decrease the queuing delay time of competing traffic flows and to increase the throughput of each signalized road intersection. ITLC has also been adapted in this thesis to the Arterial Traffic Lights (ATLs) algorithm for arterial road network scenarios. In ATLs the expected platoons on the arterial street are considered in the scheduling algorithm of each traffic light located on the arterial street coordinates. Transmitting packets among these traffic lights report the main characteristics of each predicted platoon.

Publications

The following publications by the author are relevant to this thesis:

Journals

1. M. Bani Younes, A. Boukerche, and G. Roman-Alonso, *An intelligent path recommendation protocol (ICOD) for VANETs*, Computer Networks, Elsevier, vol.64, Pages 225-242, 2014.
2. M. Bani Younes, A. Boukerche, *A Performance Evaluation of an Efficient Traffic Congestion Detection Protocol (ECODE) for Intelligent Transportation Systems Ad-Hoc Networks*, Elsevier, vol 24, Pages 317-336, 2015.
3. M. Bani Younes, A. Boukerche, G. Roman-Alonso, *Traffic Balancing-based Path Recommendations Mechanisms in Vehicular Networks*, Wireless Communications and Mobile Computing, Wiley. (Accepted)
4. M. Bani Younes, A. Boukerche, *Intelligent Traffic Light Controlling Algorithms using Vehicular Networks*, IEEE Transactions on Vehicular Technology. (Second Round)

Conferences

1. M. Bani Younes, G. Roman Alonso and A. Boukerche, *Time-distance path recommendation mechanisms for VANETs*, In Proceedings of the second ACM international symposium on Design and analysis of intelligent vehicular networks and applications (DIVANet '12), Pages 75-80, 2012.
2. M. Bani Younes, G. Roman Alonso and A. Boukerche, *Distributed Infrastructure-Based Road Congestion Avoidance Protocol for VANETs*, IEEE Globecom - Ad Hoc and Sensor Networking Symposium, 2012. (**BEST PAPER AWARD**)
3. M. Bani Younes and A. Boukerche, *Performance Evaluation of a Context-Aware Path Recommendation Protocol for VANETs*. In: The IEEE Global Communications Conference, Exhibition and Industry Forum (GLOBECOM 2013), 2013.
4. M. Bani Younes and A. Boukerche, *Efficient traffic congestion detection protocol for next generation VANETs*, IEEE International Conference on Communications (ICC), Pages 3764-3768, 2013.

5. M. Bani Younes and A. Boukerche, *A Traffic Balanced Mechanism for Path Recommendations in Vehicular ad-hoc Networks*, IEEE Global Communications Conference, Exhibition and Industry Forum GLOBECOM 2014.
6. M. Bani Younes and A. Boukerche, *An Intelligent Traffic Light Scheduling Algorithm Through VANETs*, IEEE 39th Conference on Local Computer Networks (LCN), Pages 637-642, 2014.
7. M. Bani Younes and A. Boukerche, *A Fault Tolerant Distributed Path Recommendation Protocol for Vehicular Networks*, IEEE International Conference on Communications (ICC), 2015. (Accepted)

Acknowledgements

First, I would like to deeply and sincerely thank Professor Azzedine Boukerche for his continuous professional and financial supports through these difficult years; without him this thesis would never have been finished. Secondly, words cannot begin to express my gratitude towards Professor Graciela Roman-Alonso who has shown great help in the completion of this thesis as well as her gentle and kind approach during her guidance. Many thanks are due to Philadelphia University in Jordan which made my dream of pursuing my Ph.D. degree in Canada come true. Their financial support has proved to be extremely beneficial to the successful completion of this degree and I cannot wait to join my old colleagues back there again. I also would like to thank my friends and colleagues who I met in the PARADISE research laboratory for being always helpful and optimistic. I wish them all the best in their studies and careers.

Moreover, I would like to thank my siblings: Rimah, Kaluida, Saleh, Jawhara, Anoud, Aissa, and Areen. They always made me laugh and updated me regularly with news about everything back home. I hope they all have the chance to achieve the best in their current and future endeavors. All thanks and gratitude to my parents: Dr. Younes Bani Younes and Lina Ksaimh, who always believed in me; they taught and encouraged me to always be myself.

Last but definitely not least, I wanted to thank my husband Dr. Mohammad Sababheh, who is the unlimited source of support and love in my life.

Glossary of Terms

A_{time} : Estimated arrival time

Abs-Bal : Absolute Traffic Balancing based Mechanism

ACC : The accuracy of ECODE

AF : Arterial Flow

ACK : Acknowledgement Message

ADV : Advertisement message

All_V : The count of all vehicles in each cluster

ATLs : Arterial Traffic Light Controlling Algorithm

ATSC : Adaptive Traffic Signal Control

Bal-Traf : Traffic Balancing based Mechanism

$BandCon$: The bandwidth consumption of ECODE

C_{area} : The area of the cluster

C_{ET} : The estimated traveling time required to pass through that cluster

C_{id} : The cluster identifier

C_{length} : The length of the cluster

C_n : The number of clusters

C_{V_j} : Cluster where V_j is located

C_{TD} : The traffic density of the cluster

C_{TS} : The traffic speed of the cluster

COC : Content Oriented Communications Protocol

$CoLe$: The assigned cost to each level of bad conditions

Count : The number of records in NT_j

D_k : The destination k

D_L : The density level

E_{time} : Ending cross time of the previous TL

ECODE : Efficient Congestion Detection Protocol

$ET(j)$: Estimated Traveling Time at V_j

ICOD : Intelligent Path Recommendations Protocol

IFTIS : Infrastructure-Free Traffic Information System

ITLC : Intelligent Traffic Light Controlling Algorithm

ITLs : Intelligent Traffic Lights

L_c : The length of each cluster

L_{RS_i} : The length of the road segment

L_{time} : Estimated leaving time

$Lane_{num}$: The number of lanes on the road segment

$LCFT$: Last cycle finishing time

MARL : Multi-Agent Reinforcement Learning

MAS_i : Maximum Allowable Speed for the road segment i

$NCFT$: Next cycle finishing time

NT : Neighboring Table

O_{per} : Overhead percentage

OIS : Optical Information System

Op_1 : Best turn option towards the destination

Op_2 : The second best option toward the destination

Op_3 : The worst option towards the destination

p_{D_k} : The path leading towards any destination D_k

PR2HB : Probabilistic Restricted 2-Hop Broadcast

R2HB : Restricted 2-Hop Broadcast

RA : ready area

RecomReport : Recommendation Report

RIS : Route Information Sharing system

RS_i : The road segment

RSU : Road Side Unit

RZ : Rebroadcast Zone

S_{sum} : The sum of all vehicle speed in each cluster

S_{time} : Starting cross time of the previous TL

Sd_i : Saturated density of the traffic flow i

SD_i : Saturated Density of the road segment i

SDF : saturated density factor

SerCo : The assigned cost to each located service

SRSs : Surrounding Road Segments

SSD : Safe Stopping Distance

STPN : Synchronized Timed Petri Networks

$T_COST(p_{D_k})$: The overall travel cost score of each possible path (p_{D_k})

T_{Erep} : Traffic Evaluation Report

TCC : Traffic Control Center

TD : The traffic density of the road segment

Td_{D_k} : The traffic density moving towards D_k
 $TD(j)$: Traffic Density at V_j
 Td_i : Traffic density of the output flow i
 TMR : Traffic monitoring evaluation report
 TL : Traffic Lights
 TR_{V_i} : The transmission range of each traveling vehicle
 $TS(j)$: Traffic Speed at V_j
TS : The traffic speed of the road segment
TT : The estimated traveling time of the road segment
 V_k : The closet vehicle to the center of the cluster
V2I : Vehicle to Infrastructure communication
V2V : Vehicle to Vehicle communication
VANET : Vehicular Ad Hoc Network
ZOR : Zone of Reference
 γ : The interval of each vehicle to broadcast the basic traffic data

Contents

1	Introduction	1
1.1	Vehicular Ad-Hoc Networks (VANETs)	1
1.2	Overview of VANETs Applications	2
1.3	Challenges and Issues of Traffic Efficiency Applications	3
1.4	Problem Statement and Research Objectives	5
1.5	Contributions	8
1.6	Thesis Outline	9
2	Literature Review	10
2.1	Introduction	10
2.2	Real-Time Traffic Evaluation	10
2.2.1	Disseminating Traffic Data in VANETs	13
2.2.2	Congestion Detection and Traffic Evaluation in VANETs	15
2.3	Intelligent Path Recommendation Protocols	19
2.3.1	Centralized Path Recommendations	21
2.3.2	Distributed Path Recommendations	23
2.4	Efficient Traffic Light Scheduling Algorithms	24
2.4.1	Cooperation Systems among Located Traffic Lights	26
2.4.2	Traffic Light Scheduling Mechanisms	28
2.5	Discussion	32
2.6	Summary	33
3	An Efficient Congestion Detection Protocol (ECODE)	34
3.1	Introduction	34
3.2	The Proposed Congestion Detection Protocol: ECODE	35
3.2.1	Dissemination and Gathering of Basic Traffic Data	35
3.2.2	Road Segment Clustering	37

3.2.3	Local Traffic Evaluation	38
3.2.4	Expanding Traffic Evaluation Area	40
3.3	ECODE and Its Execution Schemes	42
3.4	Performance Complexities of ECODE	43
3.4.1	Definitions	43
3.4.2	Performance Complexities of ECODE Phases:	43
3.4.3	Bandwidth Consumption of ECODE	45
3.4.4	Accuracy of Traffic Evaluation in ECODE	45
3.5	Performance Evaluation of ECODE and Its Execution Schemes Variants	46
3.5.1	Traffic Scenarios	46
3.5.2	The Performance Evaluation of ECODE	47
3.5.3	Comparison Study of ECODE to COC and StreetSmart	49
3.5.4	The Performance Evaluation of ECODE Execution Scheme Variants	54
3.6	Use Case: Traffic Distributed over Downtown Ottawa using ECODE	55
3.7	Summary	59
4	An Intelligent Path Recommendation Protocol (ICOD)	60
4.1	Introduction	60
4.2	The Architecture of ICOD	61
4.2.1	Assumptions	61
4.2.2	ICOD Architecture	62
4.3	Phases of ICOD	63
4.3.1	Road Segment Traffic Evaluation	64
4.3.2	Destination Advertisement Message Broadcasting	64
4.3.3	Path Construction	65
4.3.4	Path Recommendation	66
4.4	Traffic Parameters Considered in ICOD Variants	68
4.4.1	Traffic Parameters	68
4.4.2	Path Fitness Function	69
4.5	ICOD Variants	70
4.5.1	Congestion Avoidance Path Recommendation	70
4.5.2	Eco-Path Recommendation	70
4.5.3	Context-Aware Path Recommendation	72

4.6	Fault Tolerant Distributed Path Recommendation Protocol (TD-PR)	75
4.6.1	Fault Tolerant Mechanisms	75
4.6.2	Potential Failure Points in ICOD	75
4.6.3	The Fault Tolerant Distributed Path Recommendations (TD-PR)	76
4.7	Performance Evaluation of ICOD	79
4.7.1	Performance Evaluation of Congestion-Avoidance and Eco-path Variants	82
4.7.2	Performance Evaluation of the Context-Aware Variant	84
4.7.3	Impact of Road Segment Length and Size of Area of Interest	87
4.7.4	Comparative Performance of ICOD to RIS, TraffCon and ECO-Route	87
4.7.5	Performance Evaluation of TD-PR	90
4.8	Summary	92
5	Balancing Traffic Path Recommendation Mechanisms	94
5.1	Introduction	94
5.2	Overloaded Road Segments and Alternative Options	95
5.2.1	Road Intersections and Investigated Scenarios	95
5.2.2	Detecting and Predicting Overloaded Road Segments	96
5.2.3	Ranking the Output Road Segments Leading Towards each Destination	98
5.3	Balancing Traffic Path Recommendation Mechanisms	99
5.3.1	Balanced Traffic Path Recommendation Mechanism (Bal-Traf)	99
5.3.2	Absolutely Balanced Traffic Path Recommendation Mechanism (Abs-Bal)	102
5.4	Performance Evaluation of Bal-Traf and Abs-Bal Mechanisms	104
5.4.1	Comparative Performance of Bal-Traf and Abs-Bal to ICOD	105
5.4.2	The Impact of Congestion Level on the Road Network	107
5.4.3	Impact of the Road Network Size	110
5.5	Summary	110

6	Intelligent Traffic Light Controlling Algorithm	111
6.1	Introduction	111
6.2	Controlling Traffic Light Scenarios	112
6.2.1	Isolated Traffic Light	112
6.2.2	Signalized Road Networks	114
6.3	Intelligent Traffic Light Controlling Algorithm	116
6.3.1	Intelligent Scheduling Algorithm for Isolated Traffic Light	116
6.3.2	Arterial Traffic Lights Controlling Algorithm (ATLs)	120
6.4	Performance Evaluation of ITLC and ATLs	125
6.4.1	Performance Evaluation of ITLC Algorithm	126
6.4.2	Performance Evaluation of ATLs Algorithm	128
6.5	Summary	131
7	Conclusion and Future Work	133
7.1	Future Work	135

List of Tables

2.1	Traffic Data Gathering	11
2.2	Traffic Data Disseminating Mechanisms	14
2.3	Traffic Evaluation Protocols	16
2.4	Path Recommendation Protocols in VANETs	21
2.5	Intelligent Traffic Lights Controlling Systems	26
3.1	Simulation Parameters of the Road Segment	47
3.2	Simulation Parameters of the Ottawa Downtown	56
4.1	Simulation Parameters of The Path Recommendation Protocol	81
5.1	Simulation Parameters of Bal-Traf and Abs-Bal	105
6.1	Traffic Light Timing Variable Definitions [85]	113
6.2	Simulation Parameters of ITLC	126
6.3	Simulation Parameters of ATLS	128

List of Figures

1.1	User interfaces of applications in traveling vehicles.	3
2.1	Traffic evaluation and congestion detection in VANETs.	13
2.2	Path recommendation protocols in VANETs.	20
2.3	Isolated road intersection.	24
2.4	Mesh-alike signalized road network scenario.	25
2.5	Traffic light scheduling mechanisms.	27
2.6	Architecture of traffic efficiency applications in VANETs.	32
3.1	Road segment clustering.	37
3.2	Traffic evaluation report (T_{Erep}).	40
3.3	Communication delay and number of transmitted packets of ECODE. . .	48
3.4	End-to-End communication delay and number of transmitted packets in ECODE, COC, and StreetSmart, for different road lengths and different road densities.	50
3.5	Accuracy of ECODE, COC, and StreetSmart and accuracy in each direction.	52
3.6	Performance evaluation of the proactive execution scheme and a compar- ison performance of proactive, reactive and hybrid schemes.	53
3.7	The map of Ottawa downtown area and the SUMO representation of the map.	55
3.8	Traffic evaluation of downtown Ottawa.	57
4.1	Architecture of ICOD.	62
4.2	Weighting factor initialization algorithm.	71
4.3	Context-aware variant adaptation.	73
4.4	Grid-layout scenario.	80
4.5	Comparison analysis of <i>ICOD</i> variants.	83
4.6	Performance evaluation of the context-aware variant.	85

4.7	The impacts of road segment length and area size.	88
4.8	Comparison of the <i>Fastest</i> Variant to previous protocols.	89
4.9	The performance evaluation of TD-PR protocol.	91
5.1	Input and output traffic flows at road intersections.	96
5.2	Turn options towards each destination.	98
5.3	Grid-layout scenario with three destinations.	104
5.4	The comparative performance of the Bal-Traf and Abs-Bal mechanisms to ICOD.	106
5.5	The performance of the Bal-Traf and Abs-Bal mechanisms compared to ICOD for different percentages of congested road network scenarios. . . .	108
5.6	The impact of the road network sizes on the performances of the Bal-Traf, Abs-Bal mechanisms and ICOD protocol.	109
6.1	Primary phasing options for 4-leg intersection [85].	112
6.2	Signalized road networks [85].	114
6.3	Optimal scheduling case for arterial street [86].	115
6.4	Ready area at signalized road intersection.	117
6.5	The options of phases for each traffic light.	118
6.6	Parameters of traffic lights controlling algorithm on arterial streets. . . .	121
6.7	An arterial street in a downtown area.	122
6.8	Ready areas of adjacent road intersections in arterial street cases. . . .	123
6.9	The performance evaluation of ITLC compared to OAF [90].	127
6.10	The arterial factor.	129
6.11	The performance evaluation of ATLS compared to ART-SYS [109] and random system.	130

Chapter 1

Introduction

In this chapter, we overview the application categories that have been introduced regarding road networks using Vehicular Ad-Hoc Networks (VANETs) technology. We then explain the main challenges and issues specifically tied to traffic efficiency applications. We then present the objectives and contributions of this thesis. Finally, the outline of the thesis is summarized at the end of this chapter.

1.1 Vehicular Ad-Hoc Networks (VANETs)

Vehicular Ad-Hoc Networks (VANETs) have recently been incorporated into use over road networks. Several useful application protocols have been proposed to assist drivers and passengers during their trips [10], [55], [90]. VANETs is a special case of Mobile Ad-Hoc Networks (MANETs) in which the mobile nodes are configured as traveling vehicles. The mobility models of these vehicles are controlled by the boundaries of existing road networks and by driving rules. Moreover, the high speed of mobile nodes is the main differentiating characteristic VANETs and MANETs, particularly in highway scenarios [2].

Besides traveling vehicles, a set of Road Side Units (RSUs) are installed over road networks to assist and communicate with traveling vehicles in VANETs. These RSUs play the role of fixed infrastructure in the connecting networks. These fixed RSUs and traveling vehicles are all provided with a wireless communication transceiver (i.e., IEEE 802.11p). Transceiver equipments enable the communications between traveling vehicles (V2V), as well as the communication between traveling vehicles and located RSUs (V2I) [3], [4], [5], [6].

1.2 Overview of VANETs Applications

In recent decades, different applications have been introduced for road networks that use VANETs. These applications can be classified into three main categories: safety, traffic efficiency and infotainment applications [102], [103], [104]. The first category (i.e., safety applications) aims to increase the safety of passengers and drivers as well as the safety of the traveling vehicle itself. These types of applications save lives by avoiding or minimizing the effects of traffic accidents over the road network. These applications detect collisions and/or zones of crucial conditions; they then broadcast warning messages over the entire road network.

Applications for traffic efficiency aim to enhance the utilization of road networks and of available resources such as time and fuel. Generally speaking, by avoiding highly congested road segments one can achieve efficient travel time and decrease fuel consumption. Moreover, the schedule of located traffic lights plays a key role in controlling the traffic fluency over the road network. Efficient algorithms for traffic light control decrease the queuing delay time of traveling vehicles at each signalized road intersection; this enhances the utilization of the road network where a higher throughput can be obtained for each signalized road intersection. Real-time traffic should be evaluated regularly to report the traffic characteristics of each road segment. Reports of real-time traffic distribution are required mainly in order to design real-time traffic efficiency applications.

Finally, infotainment applications aim to make the driving experience more comfortable [7]. They provide access to different services such as: paying tolls easily without the need to stop, playing music, making phone calls, and listening to text messages, to mention a few applications. This category of applications is also intended to provide passengers with real-time entertainment such as video streaming and electronic game applications. Safety, traffic efficiency and infotainment applications are all introduced to drivers and passengers using interfaces such as wheel-mounted buttons, touch-screen interfaces or voice equipments (i.e., speakers and microphones). Figure 1.1 shows some of the interfaces that can be installed or attached to traveling vehicles.

In this thesis, we plan to investigate and to enhance traffic efficiency applications using VANETs. The challenges and issues of using VANETs to provide traffic efficiency applications are presented in the next section.

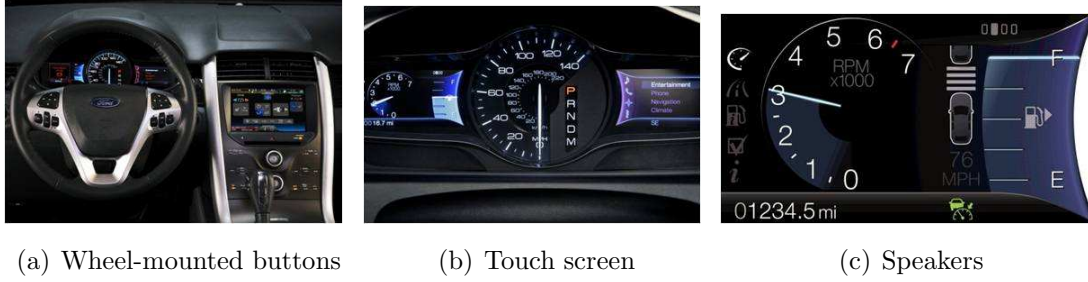


Figure 1.1: User interfaces of applications in traveling vehicles.

1.3 Challenges and Issues of Traffic Efficiency Applications

Traffic congestion is a road network condition that occurs when the existing traffic exceeds the available capacity of one or more road segments. It is usually characterized by slow traffic speed, long travel time and/or increased delay times in vehicular queuing. Traffic congestion, in general, leads to a significant increase in travel time, fuel consumption and environmental pollution. It also increases the potential for traffic accidents and creates hazardous conditions in the road network. Multiple factors affect the efficiency and usability of each road network, including road geometry, obstacles (e.g., potholes, construction sites, etc.), environmental factors (e.g., snow, icy roads, rain, fog, etc.), variations in traffic density, and the schedule of located traffic lights at signalized road intersections. These factors contribute generating the conditions of traffic congestion over highways and urban areas, thereby decreasing the efficient utilization of road networks.

The high speed of traveling vehicles presents the greatest challenge of all proposed applications in VANETs. For example, in order to evaluate the real-time traffic characteristics of any road segment, the basic traffic data of all traveling vehicles should be gathered and analyzed. This data is represented by the vehicle's speed, location and direction of motion at a certain period of time. However, the gathered data at each road segment should reflect the real-time traffic characteristics there; this means that an accurate and real-time protocol should be used to evaluate the traffic characteristics of each road segment.

In order to enhance traffic fluency and to guarantee smooth movement over the road network, intelligent path recommendation protocols are required. In these protocols, the path towards each targeted destination should be constructed to pass through the least congested road segments (i.e., the fastest alternative path). The context of each road

segment in terms of obstacles, environmental factors and locating services also affect the efficiency of choosing the best path over the road network. Possessing an outdated traffic report of surrounding road segments can lead to selection of the wrong path (i.e., a less efficient path). This is due to the fact that traffic characteristics change rapidly over road networks due to the high mobility and dynamic nature of VANETs. Moreover, the context of each road segment in terms of located services and road conditions change throughout the day, week, or time of year.

Finally, efficient schedules of located traffic lights decrease the queuing delay time of competing traffic flows at each road intersection; thus, they increase the traffic fluency over the road network. However, accurate and real-time evaluations of competing traffic flows are required in order to be considered by the scheduling mechanism. Old and inaccurate traffic reports lead to inefficient schedules for the timing cycles of each traffic light. Moreover, a significant scheduling algorithm is required to guarantee a fair and efficient schedule of competing traffic flows at each signalized road intersection.

In general, among the challenges and issues of traffic efficiency application protocols in VANETs, we will focus on the following:

- *Bandwidth flooding and redundant data*: the bandwidth of VANETs is utilized over time by repetitively broadcasting beacons. In highly congested traffic scenarios and/or large areas of interest, the bandwidth flooding is a more serious challenge. In these scenarios, using the typical broadcast forwarding mechanism to advertise the status of traveling vehicles consumes the bandwidth; this also floods the network with redundant data. The bandwidth flooding challenge is characterized by having high error rates and several interferences, which lead to packet loss.
- *Accuracy of traffic evaluations*: the traffic characteristics of each road segment are evaluated using the basic data of traveling vehicles over each area of interest (i.e., road segment). A certain node (i.e., Vehicle or RSU) is chosen at each area of interest for the purpose of gathering traffic data and evaluating its characteristics. In the case where the node in charge has obtained outdated or incorrect traffic data, wrong and useless traffic evaluations are produced. Moreover, if several status beacons were lost or dropped due to congestion or collision problems over the connecting network (i.e., VANETs), incorrect traffic evaluations could be reported.
- *Scalability*: it is not easy to design a highly scalable application protocol in VANETs; this is due to the broadcast mechanism of advertising the basic traffic data of traveling vehicles. Evaluating the traffic characteristics of large areas containing many

traveling vehicles is a challenging problem. The basic traffic data of each vehicle floods the network, thus producing inaccurate traffic evaluations, due to packet loss occurrence. Furthermore, the dynamic and rapidly changing nature of the location of vehicles in the road network decreases scalability. For example, selecting the best path towards each destination, in extended road networks, usually suffers from accuracy problems. Moreover, the traffic situation of any road segment, at the selected path towards any target destination, can be changed during the trip of each vehicle.

- *Reliability*: different points of failure could appear in the traffic efficiency application protocols, some of which are specific to the nature of the protocol in use (e.g., a cluster head in a cluster-based protocol is not working). However, other failures are general and can appear in any protocol (e.g., packet loss, a vehicle communication is down, etc).
- *Security Threats*: some malicious drivers may send compromised beacons that reflect a false status about the surrounding area, in order to deceive other drivers. The receiver drivers will believe that passing through these areas might, for example, consume more travel time; they therefore end up following an alternative route towards the target destination. Then, the malicious drivers could enjoy their trip through a light and comfortable traffic environment after having directed traffic elsewhere under false pretenses.

1.4 Problem Statement and Research Objectives

Traffic efficiency applications that use VANETs in road networks are considered challenging to implement. This is due to the highly dynamic nature of VANETs and to the scalable extended nature of the road networks. In order to provide a satisfactory quality to the user, the application must satisfy a stringent set of requirements. Therefore, by identifying the relevant requirements and satisfying them, the proposed applications can be considered successful. Several requirements that must be satisfied include network bandwidth, total delay and accuracy.

Several studies have investigated real-time traffic evaluation and congestion detection mechanisms over road networks [13], [15], [53]. These mechanisms gather the basic traffic data of traveling vehicles over the area of interest and evaluate the traffic characteristics in this area. These characteristics traffic speed, traffic density, traffic volume and/or

estimated travel time of each area of interest. The accuracy of traffic evaluation metrics is the main limitation of all previous traffic evaluation mechanisms. It is affected by high bandwidth consumption and packet loss occurrence over the VANETs network. Moreover, none of the previous traffic evaluation and congestion detection protocols have considered the direction of moving traffic over any road segment; this affects the accuracy of the traffic distribution estimation over the road network. For example, it is common seen scenarios, in which it is important to consider the direction of evaluated traffic, one direction of a road segment may be highly congested with traffic, while the opposite direction has a much lighter density of traffic.

Efficient real-time path recommendation protocols have been investigated in the literature [51], [56] in order to increase the traffic fluency over the road network. In general, all previously proposed path recommendation protocols have used a centralized architecture. First, the traffic characteristics of each located road segment are gathered in a central database. Then, a central processor recommends the optimal path towards each located destination according to the gathered data. The centralized architecture suffers from a bottleneck problem, which causes some packets to be dropped and causes an incorrect path to be generated. Moreover, a relatively high delay is required to gather the data and then to recommend the optimal path in the centralized architecture. Thus, the turn recommendation at each road intersection may be delivered late, becoming useless or incorrect.

Furthermore, the previously proposed path recommendation protocols have considered the optimal path of each traveling vehicle separately towards its targeted destination. None of these protocols have considered traffic volume generated at any output segment of road intersections. This means that the path recommendation protocol may generate highly congested road segment situation, or transfer the highly congested situation from one road segment to the next road segment at each road intersection.

In addition, intelligent traffic light controlling algorithms have been proposed in the literature [90], [109] to decrease the queuing delay time at each signalized road intersection. The sequence phases of the timing cycle at each traffic light are scheduled efficiently, so as to increase the traffic fluency over the road network. The accuracy of the characteristics of competing traffic flows at each signalized road intersection mainly affects the efficiency of the scheduling algorithm. Moreover, a significant scheduling algorithm is required at each located traffic light. An efficient schedule of located traffic lights decreases the queuing delay of traveling vehicles. It also increases the throughput of each road intersection.

In this thesis, which aims to address the previously discussed challenges, we propose an efficient and accurate protocol, named ECODE, that evaluates the traffic characteristics at each direction of located road segments separately. We also introduce several variants of an intelligent, distributed and dynamic path recommendation protocol (ICOD). For each traveling vehicle, these variants recommend the path towards its target destination in a hop-by-hop fashion. Moreover, the generated traffic volume at each output road segment, with path recommendation protocol in use, is investigated and the overloaded output road segment scenarios are eliminated by introducing two traffic balancing mechanisms (i.e., Bal-Traf and Abs-Bal). Finally, we introduce an intelligent traffic light controlling algorithm, adapting it for isolated road intersections (ITLC) and for signalized arterial road network scenarios (ATLs). We concentrate the research objectives for traffic efficiency applications over road networks as follows:

- *Efficient bandwidth consumption:* ECODE decreases bandwidth consumption, thereby decreasing the rate of collisions and congestion throughout the VANETs network. This protocol uses an efficient data disseminating mechanism that divides the area of interest into a set of non-overlapping clusters. It also defines a relay vehicle at each cluster to report the traffic characteristics of such a cluster, which minimizes the bandwidth consumption.
- *Distributed architecture:* instead of relying on a centralized database and a central processor, ICODE distributes the decision of selecting the best path towards each destination among a set of installed RSUs all over the area of interest. The distributed architecture of ICODE eliminates the bottleneck and single point-of-failure problems over VANETs.
- *Accuracy and correctness:* ECODE provides a highly accurate estimate about the congestion level and the traffic characteristics of each road segment. It uses three different metrics to evaluate the traffic of each road segment: traffic density, traffic speed and estimated travel time. Moreover, considering the direction of traffic while evaluating congestion levels enhances the accuracy of the traffic evaluations, particularly for a scenario in which there is a clear variation between the congestion level in opposite directions at the same road segment. ICODE recommends the path towards each targeted destination in a hop-by-hop fashion. The turn decision at each road intersection is made based on the traffic situations of the surrounding road segments at that time; this allows for decisions that are more dynamic, real-time and accurate. Finally, using an accurate traffic evaluation of competing traffic

flows at signalized road intersections will help produce a more efficient schedule of each intelligent traffic light.

- *Scalability*: ECODE aims to evaluate the congestion level and the traffic characteristics of any road segment. Multiple forwarding of the gathered data increases the boundaries of the overall area to be evaluated. Moreover, the distributed architecture of ICOD helps to extend the boundaries of each road network of interest.
- *Balancing traffic over the road network*: Bal-Traf and Abs-Bal are two traffic balancing-based mechanisms. They are proposed in order to estimate traffic volume at each output road segment by the path recommendation protocol in use. These mechanisms help to eliminate the generated highly congested road segment scenarios, while selecting the best path towards each destination.

1.5 Contributions

This thesis comprises the following four main contributions intended to enhance traffic efficiency over road networks:

- *Efficient COngestion Detection protocol (ECODE)*: this is an efficient and real-time traffic evaluation protocol. It is designed to evaluate the traffic characteristics of each road segment located in a downtown area. ECODE detects and reports the road segment direction of highly congested traffic to be considered by the real-time traffic efficiency applications.
- *Intelligent path reCommendation protocol (ICODE)*: this utilizes a set of RSUs scattered over the road network. These RSUs cooperatively recommend the optimal path towards each targeted destination, according to the real-time traffic characteristics of surrounding road segments. Several variants of ICOD have been introduced, including congestion avoidance, Eco-path and context-aware trajectory.
- *Traffic balancing-based path recommendations mechanisms (Bal-Traf and Abs-Bal)*: two traffic balancing based mechanisms are proposed for distributed path recommendation protocols. These mechanisms consider the generated volume of traffic at output road segments on each road intersection. Bal-Traf detects the estimated overloaded output road segments; it then eliminates these overloaded scenarios by directing some vehicles to other road segments as next hop towards the targeted

destination/destinations. However, Abs-Bal distributes the input traffic of each road intersection so as to balance it among the existing output road segments at each road intersection.

- *Intelligent traffic light controlling algorithm*: an intelligent traffic light controlling algorithm is designed. This algorithm has been adapted for isolated traffic light (ITLC) and for arterial signalized road network (ATLs). ITLC is first introduced to efficiently schedule the successive phases of each isolated traffic light. Moreover, ATLs cooperatively schedules the traffic over arterial signalized road networks.

1.6 Thesis Outline

The remainder of this thesis is organized as follows: in Chapter 2, we discuss the previously proposed traffic efficiency application protocols. Then, we introduce our congestion detection protocol (ECODE) to efficiently evaluate the real-time traffic characteristics of each road segment located over the investigated road network in Chapter 3. Several variants of an intelligent path recommendation protocol (ICOD) are presented in Chapter 4. In Chapter 5, two traffic balancing based mechanisms have been proposed. These mechanisms distribute the large volume of traffic among located road segments towards targeted destinations, at each road intersection. An intelligent traffic light controlling algorithm is presented in Chapter 6 to schedule the phases of timing cycle at each installed traffic light. Isolated road intersection and arterial signalized road network scenarios have been investigated using this algorithm. Finally, Chapter 7 concludes the thesis and highlights some future works.

Chapter 2

Literature Review

2.1 Introduction

The main focus of this thesis is to propose traffic efficiency application protocols for road networks. Therefore, in this chapter we investigate the main protocols in the literature that have been designed as traffic efficiency applications over road networks. First, in Section 2.2 we present the previously proposed real-time traffic evaluation and congestion detection protocols. In Section 2.3, the intelligent path recommendation protocols that have been proposed in the literature are investigated. Efficient traffic light controlling algorithms for isolated road intersections and for signalized road networks are discussed in Section 2.4. We also provide a classification of traffic efficiency application protocols, according to their functionality and to the targeted scenarios of these applications, in Section 2.5. Finally, Section 2.6 summarizes the chapter.

2.2 Real-Time Traffic Evaluation

Real-time traffic distribution over road networks is essential in order to find the optimal path towards each destination. Moreover, accurate distribution of traffic is required to efficiently schedule the sequence of phases of timing cycles in each traffic light. Accurate traffic characteristics and punctual delivery of traffic reports are important to guarantee, in order to produce useful and correct traffic efficiency applications. In order to evaluate the traffic characteristics over the road networks, several mechanisms have been introduced to gather real-time traffic characteristics. Four main technologies have been utilized to gather basic traffic data at each road area of interest: optical information

system, inductive loop detectors, mathematical modeling and VANETs technology. Table 2.1 summarizes the general advantages and the main limitations of each of these traffic data gathering mechanisms.

Table 2.1: Traffic Data Gathering

Mechanism	Advantages	Limitations
OIS: Optical Information System	On time delivery	Expensive, required processor and memory, not scalable and not always clear.
Inductive loop detectors	On time delivery, accuracy	Expensive and not scalable.
Mathematical modeling	Help to prove the correctness and the validity of the proposed algorithms	Do not reflect the real distribution over the road network.
VANETs	Free, fast and scalable	Limited accuracy, long delay, high communication overhead

Krajzewicz et al. [87] have used an optical information system (OIS) to gather the traffic characteristics of each area of interest. Moreover, Jain et al. [28] have installed fixed cameras at roadsides all over the area of interest. In OIS, video cameras have been used to gather the basic traffic data of each road segment. The gathered photos are analyzed regularly to detect high congestion scenarios in traffic flow. Indeed, this technology requires a significant number of camcorders, which are expensive to buy and to install. Significant algorithms and non-negligible time delay are required to analyze the gathered photos and to obtain the characteristics of each traffic flow. Moreover, to run the analyzing algorithms, powerful processors with adequate memory size are required. OIS clearly suffers from scalability limitations due to the zoom configurations of the cameras used. During night-time or rush weather (i.e., foggy or rainy), there is no guarantee of obtaining good clarity in the detected photos. This affects the accuracy of the evaluated traffic characteristics.

Mirchandani et al. [99] and Dunkel et al. [70] have used inductive loop detector equipment to count the number of vehicles traveling over investigated road segments during a given period of time. This equipment is installed at the beginning of each road segment; whenever a vehicle passes by the detector is controlled on that road segment. In general, the number of traveling vehicles, the boundaries of the investigated road segment, and the time factor are used to configure the traffic characteristics of such a road segment. However, inductive loop detectors are expensive to install and to maintain. They suffer from limited scalability issues and the exact location or real-time speed of each vehicle could not be accurately detected using this equipment.

On the other hand, many researchers [98], [94], [107], [108] have used several math-

ematical models such as Cellular Automata (CA) [8] and Balance Density (PED) [9] to estimate the traffic distribution over road networks. These models present estimates of traffic distribution and estimated mobility over the road network. The estimated traffic distributions help to mathematically prove the correctness and the validity of each proposed application. However, these mathematical estimates do not reflect the real distribution over the road network. Thus, there is no guarantee that in a real traffic scenario, the proposed applications (i.e., algorithms) would have the same estimated performance of the modeling approaches.

Recently, VANETs technology has been used to gather the traffic characteristics over the road networks [88], [90], [92], [101], [109]. The free, fast and scalable communications, among traveling vehicles and located RSUs in VANETs, attract many researchers wishing to utilize this technology in order to evaluate traffic characteristics over the road network. Several challenges have been faced in the proposed data gathering and traffic evaluation mechanisms to generate accurate and punctual traffic reports. Moreover, the communication overheads in terms of bandwidth consumption and end-to-end delay are the main challenges of these mechanisms.

In order to evaluate traffic characteristics and to predict the distribution of traffic over road networks, the basic traffic data of all traveling vehicles are required. Using VANETs, traveling vehicles periodically broadcast their basic traffic data (i.e., speed, location, direction, etc.). The traffic data of all traveling vehicles at the area of interest are gathered and analyzed, in order to generate a traffic characteristics report. The gathered data is classified based on the locations of traveling vehicles. The traffic characteristics of each area of interest (i.e., road segment) are investigated, including traffic density, traffic speed or traffic volume, etc. For each road segment, the obtained traffic characteristics are compared to its capacity. In the case that the traffic density of any road segment exceeds its capacity, the road segment is detected and reported as a highly congested road segment. Figure 2.1 illustrates the sequential steps of traffic evaluation and congestion detection mechanisms.

In this section, we first investigate the efficient data disseminating mechanisms that have been used accordingly to collect the basic traffic data of traveling vehicles. Then, the previously proposed traffic evaluation and congestion detection protocols are presented.

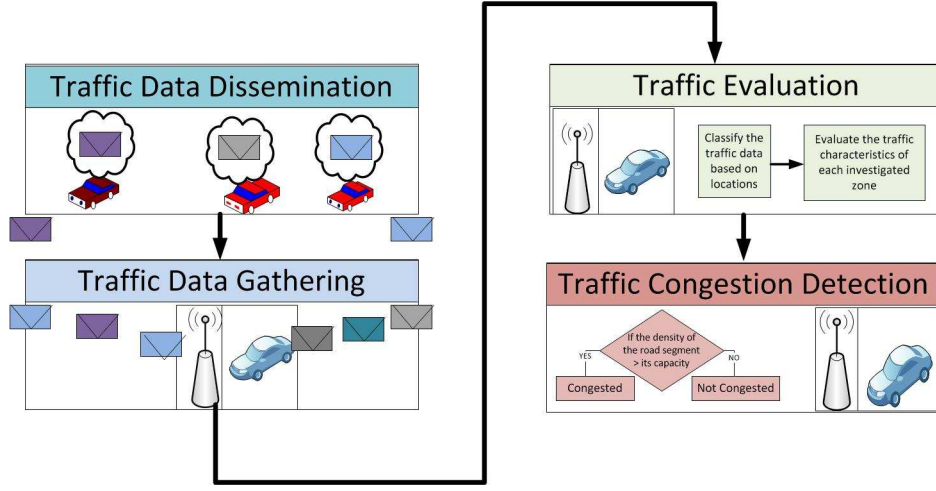


Figure 2.1: Traffic evaluation and congestion detection in VANETs.

2.2.1 Disseminating Traffic Data in VANETs

The limited transmission range of traveling vehicles allow each vehicle to gather the basic traffic data of its neighbors. Each vehicle is only aware of the traffic characteristics inside its transmission range. The scalable environment of road networks and the limited transmission range of traveling vehicles introduce serious challenges for efficient data dissemination. At first glance, using multi-hop broadcast communications of VANETs to disseminate data over the entire road network seems to be an attractive proposition. However, multi-hop broadcast mechanisms introduce many network challenges, including broadcast storm problems, memory restrictions, bandwidth restrictions and reliability problems. We investigate some of the previously proposed solutions that have attempted to tackle these challenges and to efficiently disseminate the traffic data. Table 2.2 summarizes and categories the previously proposed mechanisms of disseminating traffic data based on the efficient utilized technique; it also illustrates the network challenges that have been handled in each proposed mechanism.

By using some relay vehicles to forward the traffic data instead of a blind flooding mechanism, Xu et al. [16] and Chou et al. [46] resolved the problems with bandwidth consumption and restricted memory. Xu et al. [16] have proposed two broadcast methods: Restricted 2-Hop Broadcast (R2HB) and Probabilistic Restricted 2-Hop Broadcast (PR2HB). These methods are used for disseminating and gathering traffic data to expand the detection range of each RSU, and to improve the performance of a 2-hop broadcast mechanism. In the R2HB, each vehicle rebroadcasts its basic data only if it is located

Table 2.2: Traffic Data Disseminating Mechanisms

Paper	Approach	Technique	Challenges			
			Bandwidth Consumption	Memory	Reliability	Broadcast Storm
Chou et al., [46]	Broadcast to disseminate data.	Relay vehicle.	✓	X	X	✓
Xu et al., [16]	Geocast to disseminate data.	Relay vehicle.	✓	✓	X	✓
Inoue et al., [51]	Geocast to disseminate data.	Event driven.	✓	✓	X	✓
Dornbush et al., [53]	Need to Say principle.	Event driven.	✓	✓	X	✓
Zhong et al., [11]	Make a local decision when to disseminate a report, based on the vehicle status.	Data prioritization.	✓	X	X	✓
Shibata et al., [54]	Categorize data, and send important data prior to other data.	Data prioritization.	✓	✓	X	✓
Bauza et al., [1]	Send a statistical data related to each Area (Geocast).	Using statistical data.	✓	X	X	✓
Korkmaz and Ekici, [50]	The ACK message is sent to announce a correct received report.	ACK mechanism.	X	✓	✓	X

inside the rebroadcast zone (RZ) boundaries. On the other hand, in PR2HB, all broadcast messages from different vehicles can be rebroadcast; however, this opportunity is only made available to the most eligible vehicles inside RZs. The most eligible vehicle at any RZ, in this mechanism, is determined based on its current location and its real-time speed. However, although these methods have reduced bandwidth consumption and have decreased the number of transmitted messages, the low accuracy of gathered data and late traffic report deliveries are still the main limitations of these mechanisms.

Event-driven techniques have been utilized to disseminate data in a reactive manner instead of periodically broadcasting beacons [51], [53], [72], [73], [74], [75], [76], [77], [78], [79]. Inoue et al. [51] have proposed a data dissemination mechanism in which each vehicle broadcasts the driving experience of its traversed path only when it arrives at road intersections. On the other hand, Dornbush et al. [53] have designed a method of dissemination that seeks to reduce the communication overhead by only transmitting important data. For example, sudden changes in the traffic characteristics of the area of interest is a type of important data that should be disseminated. Low accuracy of

traffic evaluation is considered the main limitation of this system, especially in the case of slow-moving vehicles and light traffic density. Other research studies like Zhong et al. [54], Shibata et al. [11] prioritize the data to be broadcast, in order to reduce the effects of restricted bandwidth and limited memory. Zhong et al. [11] have introduced a mechanism in which each vehicle decides which traffic reports to disseminate, how many, and when. These reports are prioritized in terms of their value as reflected by supply and demand. Additionally, Zhong et al. [54] have proposed a method to categorize data based on the source (i.e., sender vehicle) location. The proposed method sends the most important data before other less important data.

In order to enhance the efficiency of the dissemination mechanisms, a statistical representation of gathered data can be computed and forwarded accordingly [1], instead of forwarding all gathered data [112], [113], [114], [115], [116]. Bauza et al. [1] introduced four different statistical functions to represent the traffic situations at each area of interest. These functions have been discussed through the following terms: mean, median-based, median-interval and median-interval neighbors. However, statistical representations can work efficiently only for small areas. Finally, Korkmaz et al. [50] used an acknowledgement (ACK) message to solve the hidden node problem. The ACK message is also used to solve the issue regarding unreliable communication media that appear in data dissemination protocols. In the case that any vehicle receives a traffic data report that is related to a certain area, it should send an ACK message; this will announce that the sender vehicle has received the traffic report correctly. If the sender did not receive the ACK message during a certain period of time, it should retransmit the traffic report message. This mechanism increases the reliability of the intended system.

2.2.2 Congestion Detection and Traffic Evaluation in VANETs

Once the traffic data of a certain area or road segment is gathered, vehicles or RSUs in charge can evaluate the traffic density [15], traffic speed [20] or estimated travel time [51], [71] of the relevant area. These traffic characteristics represent the main categories of traffic evaluation at each area of interest. Table 2.5 summarizes the main characteristics of select traffic evaluation and congestion detection protocols that have been introduced in the literature regarding VANETs.

Fukumoto et al. [15] proposed the Content Oriented Communications (COC) protocol. Each vehicle periodically broadcasts its basic data in COC. Based on the gathered information of surrounding vehicles, the traffic density of each area is then computed.

The higher the traffic density in each area is, the higher the reported congestion level. In other words, this protocol uses the traffic density metric to determine the level of traffic congestion at each area of interest. Although this protocol achieves good performance in terms of traffic evaluation accuracy, it faces some problems regarding the incurring of high bandwidth consumption for scalable areas, and it requires high end-to-end delay.

On the other hand, Padron et al. [20] have proposed a protocol by the name of “Voting”; in this protocol each vehicle estimates the traffic congestion level in its surrounding area based on its internal evaluated traveling characteristics. Each vehicle broadcasts the estimated internal congestion level to the neighboring vehicles according to its traveling speed, which is compared to the maximum speed limit in the road segment or the zone of interest. Receiver vehicles vote for or against the estimated traffic evaluation in order to reach a consensual decision regarding the level of traffic congestion in the respective area. For example, if a certain vehicle is moving slowly whereas the other neighboring vehicles are moving at a relatively higher speed, the general consensus that will be reached about this vehicle is that it is voluntarily driving slow and is not affected by any traffic congestion. On the other hand, if a vehicle is moving slowly on a certain road segment and the surrounding vehicles are driving at approximately the same speed, the conclusion drawn is that congestion in that area is the factor influencing all or most vehicles there, forcing them to drive slowly. Relatively high traffic density is required in each area in order to negotiate the congestion evaluation, otherwise the accuracy of the traffic evaluation will be low.

Table 2.3: Traffic Evaluation Protocols

Protocol	Communication	Traffic Evaluation	Considerations	
			Bandwidth Consumption	Accuracy
COC [15]	V2V	Traffic density	High	Good
Voting [20]	V2V	Traffic speed	High	Low
Virtual Sink [46]	V2V	Traffic volume	High	Good
Clustered Area [54]	V2V	Estimated traveling time	Medium	Low
SOTIS [34]	V2V	Traffic speed	High	Good
IFTIS [33]	V2V	Traffic density	Low	Good
StreetSmart [53]	V2I	Traffic speed	Low	Low
V2X [45]	V2I	Traffic speed	Medium	Medium
Lattice [51]	V2I	Estimated traveling time	Low	Low and Old

The aforementioned congestion detection protocols, COC [15] and Voting [20] are restricted to the vehicle transmission range area. In order to expand the area of evaluation,

after each vehicle evaluates the level of congestion in its local transmission range area, it forwards the traffic evaluation reports towards vehicles in the neighboring areas [54], [46], [34], [33]. Vehicles in the next-hop areas evaluate the extended congestion level. They combine the local congestion level and the evaluation reports received from neighboring areas.

Chou et al. [46] proposed a virtual data sink-based information dissemination mechanism using vehicular networks. The virtual sink node gathers the information about traffic volume at each investigated zone. This mechanism uses travel vehicles as mobile traffic detectors, where inter-vehicle communications deliver the basic traffic data among traveling vehicles. The level of traffic congestion is computed according to the estimated traveling time needed to cross each area. On the other hand, Shibata et al. [54] proposed a geocast protocol in which the targeted geographic region is divided into smaller manageable areas. Each vehicle measures the time it takes to pass through each area, and traffic information is collected by exchanging packets among moving vehicles. Each vehicle measures the time it takes to pass through an area upon entering/exiting a pair of roads, and it generates historical traffic information statistics from vehicles which have passed through the same pair of roads. In general, multi-hop communications expand the area over which each vehicle is aware of the traffic situation. However, it causes more bandwidth consumption and less accurate traffic evaluation due to the outdated gathered data.

An information dissemination approach to self-organizing the inter-vehicle network, named SOTIS, is introduced by Wischhof et al. [34]. In SOTIS, each vehicle is equipped with a local digital map, and an internal database is associated with geographical coordinates and is shown on indicators on the in-car display. Traffic conditions, particularly traffic speed, of each vehicle are evaluated locally, using the basic data from periodic hello messages. Traveling vehicles also periodically exchange traffic information messages that include the location of the vehicle, the intended segment (i.e., part) of road and the average traffic speed of this segment. If any car receives this traffic information message, it compares the content of the message with its internal database to decide if the information received is more accurate/updated than its best knowledge of the traffic situation. SOTIS has been mainly introduced into highway scenarios where the forehead vehicles can alert following vehicles of highly congested areas or bad traffic condition scenarios. However, on downtown areas, this system may deliver late or useless warning messages, especially in the case where vehicles in the opposite direction use a receive-carry-broadcast mechanism to deliver traffic condition information to distant vehicles

traveling in the same direction. Another infrastructure-free traffic information system (IFTIS) for vehicular networks is proposed by Jerbi et al. [33]. This protocol is one of the first research works that has considered the traffic characteristics at each road segment separately in an urban area. It aimed to evaluate the traffic density of each road segment in order to choose the best route through which to send packets over the communication network. The IFTIS protocol has adapted the location-based group concept by dividing the investigated road segment into a set of overlapping location-based groups (i.e., cells). The leader vehicle of each group is the closest vehicle to the center of each cell, traffic density of each cell is evaluated by the leader of each group. Leaders of cells deliver the traffic density reports to each road intersection at the end of the road segment. At the road intersection, the evaluation reports of all cells at the road segment are combined and the global traffic density of such a road segment is evaluated. A relatively high level of delay is reported in this system (i.e., 2-9 seconds). Moreover, the traffic density is evaluated for both directions of each road segment at the same time. In some scenarios, a high traffic density could be located in one direction while the other direction witnesses a very low traffic density. This causes inaccurate decisions to be made, especially for those applications that aim to identify the least congested paths for traveling vehicles or for efficient traffic light applications. Overlapping clusters decrease the accuracy of traffic evaluation for the entire road segment as well.

Lakas et al. [52], proposed a system that frequently disseminates and exchanges road traffic information in order to obtain more real-time paths towards each destination. The zone of reference (ZOR) has been proposed in this work to coordinate the area of interest. This coordinates the angles at the geometrical region covering a given area. The geometric region is typically a rectangle shape, and each angle is identified by its GPS coordinates. The ZOR area can be of any closed shape including a set of angles.

Using V2I communications of VANETs, each RSU computes the traffic congestion level of each area of interest. The basic traffic data of each area are delivered by traveling vehicles to the responsible RSUs, in order to evaluate the level of traffic congestion [16], [45], [51], [53]. Sandor et al. [53] proposed *StreetSmart* protocol; *StreetSmart* uses a combination of clustering and epidemic communication to find, disseminate and to report on dynamic traffic patterns of adjacent cluster areas. Only summary information from areas where there is unexpected traffic are exchanged in this system. Nodes exchange summary statistics using epidemic communications. Each node detects high congestion levels areas by analyzing the summary statistic messages gathered locally. The low level of accuracy is the main limitation of this system because several vehicles can participate in

more than one cluster; this is due to the overlapping cluster areas utilized in *StreetSmart*.

Schunemann et al. [45] introduce an algorithm that can be used by navigation systems to calculate routes circumnavigating highly congested roads. This algorithm is named V2X-Based traffic congestion recognition and avoidance. Traveling vehicles compute and send the average speed of each road segment to other vehicles and RSUs in their vicinity. However, the speed of traveling vehicles is not an accurate enough metric to detect the traffic congestion level in a certain area; this is because some vehicles may voluntarily move slowly. Another congestion detection mechanism was introduced by Inoue et al. [51], using V2I communications architecture. This proposed mechanism detects the congestion level per path in downtown scenarios based on the traveling time of moving vehicles. Whenever a vehicle finishes its trip (i.e., arrives at the targeted destination), it reports the driving experience to the nearest RSU. At this point, the actual travel time is compared to the optimal traveling time of the same trip path (i.e., the required travel time in the case where there is no traffic congestion in that trip). The path congestion index is determined at that RSU by comparing the actual traveling time to the optimal required time.

2.3 Intelligent Path Recommendation Protocols

Different navigation systems have been attached to and installed on the traveling vehicles in the last few years. These systems have been used to guide drivers towards their targeted destinations. The typical navigation systems provided in most cars these days find the shortest path between the driver's location and the targeted destinations. On the other hand, different real-time navigation applications such as Google [47], Waze [48] and I-GO [49], to mention a few, can be run from smart mobile devices. In this context, these applications consider travel time and traffic congestion as the dynamic parameters of each candidate's path towards any destination. However, these applications require Internet communication to gather the real-time data of the dynamic traffic parameters. Moreover, the performance of these applications varies based on the Internet's communication speed and strength.

Several protocols and mechanisms have been proposed to construct a dynamic path towards any destination in downtown and/or highway scenarios using VANETs technology. Most of these protocols aim to find the fastest path towards each destination over the road network [44], [45], [51], [56], [70], [121], [122], [123], [124], [125], [126].

In general, the previously proposed path recommendation protocols in VANETs can

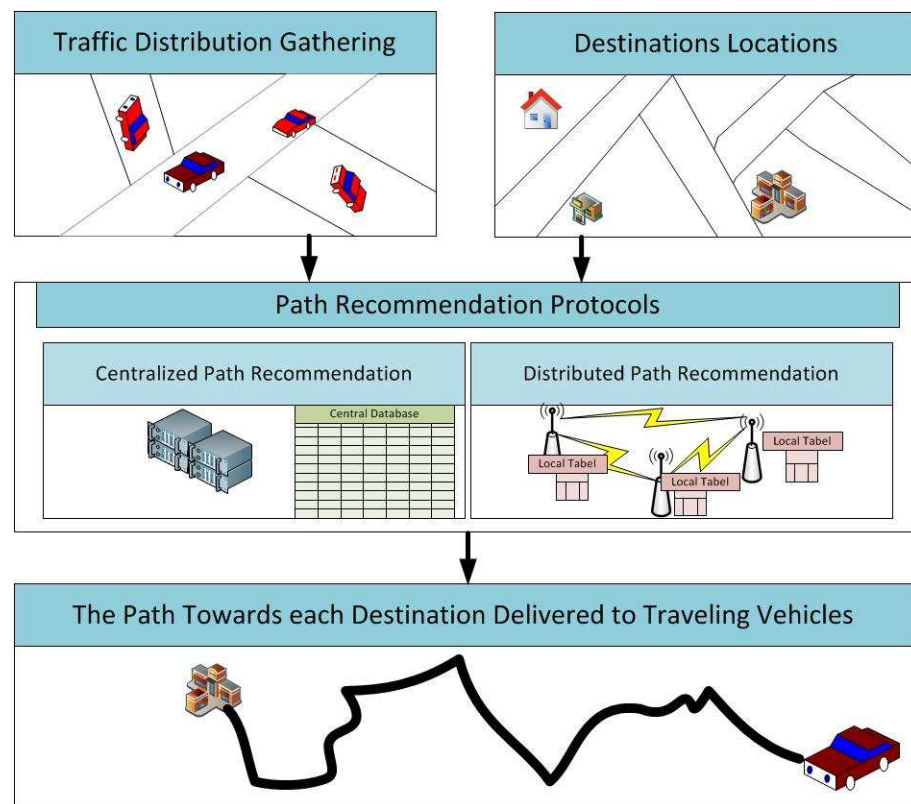


Figure 2.2: Path recommendation protocols in VANETs.

be classified into two main categories: centralized and distributed architectures. Centralized path recommendation protocols gather the traffic characteristics of the entire road network and the location information about the targeted destinations, in a central database. After that, a central processor selects the best path towards each destination, according to the gathered data, and reports the selected path back to each traveling vehicle. On the other hand, in the distributed path recommendation protocols, a set of RSUs are scattered all over the area of interest. Each RSU obtains the best directions towards all target destinations at each road intersection, according to the traffic characteristics of surrounding road segments. Communications among the located RSUs cooperatively update the local routing tables towards each destination. Figure 2.2 summarizes the sequence steps of path recommendation protocols in VANETs.

Table 2.4 summarizes the main characteristics and considerations of previously proposed path recommendations protocols. In the rest of this section we detail the solutions, considerations and characteristics of these protocols.

Table 2.4: Path Recommendation Protocols in VANETs

Protocol/ Author	Algorithm	Centralized	Real-Time	Scalability	Considerations			
					Time	Distance	Fuel and Gas	Context-Aware
Yamashita et al. [42] (RIS)	Dijkstra	✓	X	X	✓	✓	X	X
Inoue et al. [51] (Automobile)	K-paths	✓	X	X	✓	✓	X	X
Lakas et al. [52]	Dynamic Dijkstra	✓	✓	X	✓	X	X	X
Shibata et al. [54]	Dynamic Dijkstra	✓	X	✓	✓	X	X	X
Kitani et al. [55]	Ferrying Technique	✓	X	✓	✓	X	X	X
Collins et al. [56] (TraffCon)	K-shortest paths	✓	X	X	✓	X	✓	X
Kono et al. [81] (ECO-route)	Dijkstra	✓	X	X	X	X	✓	X
Wedde et al. [80] (BeeJamA)	Distributed	X	✓	✓	✓	X	X	X

2.3.1 Centralized Path Recommendations

Yamashita et al. [42] introduced a simple route guidance mechanism based on the Route Information Sharing (RIS) system. In this work, drivers would provide trip plan details to the route information server (i.e., central processor). This server then sends accumulated traffic information, based on all received planned routes, to drivers. The central processor computes the estimated future traffic flow at each road segment from the delivered plans towards each destination.

Inoue et al. [51] proposed an automobile control system that helps to alleviate traffic congestion in a lattice-like road environment, without drastically increasing the travel distance of each vehicle. The central processor produces k possible routes towards each vehicle destination. These k paths consist of the shortest path and $k - 1$ other alternative random paths that have been chosen based upon a set of random relay points over the area of interest. The traveling distance of these k paths should not exceed a pre-defined maximum distance threshold. Finally, the central processor uses the real-time traffic information to recommend the least congested route to traveling vehicles (i.e., fastest alternate path).

Furthermore, using geocast principles, Lakas et al. [52] proposed a vehicular communication system for traffic congestion detection and dissipation over road networks. This system used a modified dynamic Dijkstra algorithm, in order to find the least congested route towards any target destination. The congestion index used in this work reflects the ratio of actual travel time to ideal required time (i.e., where no traffic congestion is observed) of the respective path. The obvious overhead of this system occurs in the form of a large number of transmitted messages (i.e., high bandwidth consumption) intended to

guarantee real-time recommendations, especially for a large number of traveling vehicles. Moreover, a high delay is expected when determining each path in scalable areas.

Several proposed protocols [54], [55] have divided the area of interest into a set of clusters so as to facilitate traffic management and control, especially for large areas of interest. These protocols aimed to solve the congestion problem locally at each cluster and then to combine these cluster solutions together. Shibata et al. [54] divided the targeted road map into a set of predefined areas (i.e., clusters). The fastest path towards each destination can be determined based on its location and the alternate set of areas chosen. An extension of [54] was proposed by Kitani et al. [55], which aimed to improve the efficiency of information sharing by using a message ferrying technique. In this extended work, buses traveling on predetermined routine paths are used as carriers to deliver the traffic data among the nodes of the communication network; also, cars are defined as regular nodes. Although these protocols, which divided the area of interest into smaller manageable clusters, have handled the scalability issue, they all use a central processor at each cluster area to make the path recommendation decisions. However, late decisions, expired gathered data, central points of process, bottlenecks and single point of failure problems are the main limitations of these protocols.

Some of the intelligent path recommendation protocols in VANETs have considered other parameters besides reducing traffic congestion and minimizing the travel time objectives. Fuel consumption, gas emission and travel distance are some of these investigated parameters [56], [57], [58], [81]. As illustrated in Table 2.4, Yamashita et al. [42] and Inoue et al. [51] consider the travel distance of each selected path towards the targeted destination, where the fastest path should not exceed a pre-defined distance threshold. On the other hand, Collins et al. [57] and Kono et al. [81] have considered the fuel consumption and the gas emission parameters of each selected path. These research studies have produced eco-path recommendation protocols. For instance, Collins et al. [57] presented a vehicle routing algorithm (TraffCon). The latter algorithm optimizes the usage of road capacity and thus reduces vehicle trip time while decreasing fuel consumption and gas emission. A fitness function that weights the factors considered has been introduced in this work as well. In TraffCon, each vehicle first contacts the server processor by sending its location and targeted destination. The server obtains k shortest paths from the location of the vehicle towards the targeted destination, according to the entire gathered data. The server processor then evaluates the cost of each path by using the fitness function introduced in [56]. Finally, the path with the lowest overall cost is selected and recommended to the traveling vehicle.

Kono et al. [81] have introduced an ecological route search and driving protocol. This protocol uses the mathematical model introduced in [63] to estimate the fuel consumption of each road segment in urban areas. Then, the server processor uses the Dijkstra algorithm to find the path that consumes the least amount of fuel. Besides the bottleneck and single point of failure problems caused by the centralized architecture, the main limitation of this approach is the issue of accuracy. The traffic situation of each road segment can change more quickly than it is possible to gather the traffic data and to recommend the best path in terms of fuel consumption. Moreover, the process of estimating the fuel consumption should be sufficiently accurate, where several factors have to be investigated.

2.3.2 Distributed Path Recommendations

Due to the aforementioned challenges that have appeared in the centralized path recommendation protocols, recently distributed path recommendations are designed using VANETs technology. The distributed path recommendation protocols rely on real-time communications among a set of RSUs that are installed all over the area of interest, as illustrated in Figure 2.2. These RSUs aim to cooperatively find the path towards each destination, which can be adjusted and reconfigured at each road intersection (i.e., the path is constructed and updated in a hop-by-hop fashion).

Very few research studies have investigated the distributed architecture of path recommendation protocols using VANETs. BeeJamA [80] has been introduced as a distributed and self-evaluated vehicle routing guidance approach. The latter approach uses a multi-agent system which relies on a distributed V2I communications. The area of interest is divided into a set of small manageable clusters, and the traffic characteristics at each cluster are gathered and handled separately. Traveling vehicles request the next hop direction at each traffic intersection, and the located RSUs respond according to the foreseen traffic characteristics. BeeJamA is inspired by the honey bee's foraging, it has been intensively evaluated with respect to average travel time and congestion avoidance [80]. The path request and the recommendation response at each road intersection, among traveling vehicles and located RSUs, clearly cause a high bandwidth consumption and large delay time.

2.4 Efficient Traffic Light Scheduling Algorithms

Efficient traffic lights aim to intelligently schedule and to control competing traffic flows at each road intersection. The optimal schedule at each signalized road intersection is set based on the real-time traffic distribution over the road network. In general, the smaller the average queuing delay time at the signalized road intersection, the more efficient the scheduling algorithm. Moreover, an efficient scheduling algorithm produces higher throughput for each signalized road intersection and then enhances the utilization of the road network.

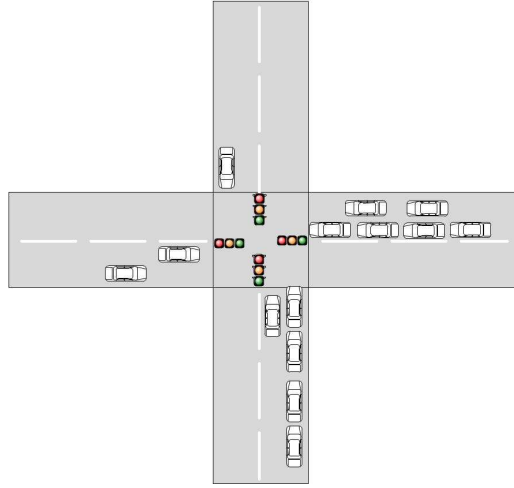


Figure 2.3: Isolated road intersection.

The schedule of phases sequence at each located traffic light has been investigated in the literature for three different signalized road network scenarios: isolated road intersection, arterial streets and mesh-alike signalized road networks. In an isolated signalized road intersection, the located traffic light only considers the traffic characteristics of the competing traffic flows at the respective road intersection. Figure 2.3 illustrates a typical isolated road intersection scenario with eight competing flows of traffic. For instance, traffic density, traffic speed, traffic volume and traffic distribution of each flow are the main parameters to assign priority and to set the sequences of the competing traffic flows to pass the intersection. VANETs has been used to gather the real-time traffic characteristics of all competing traffic flows at signalized road intersections. In the case that the gathered traffic characteristics are not accurate, or if any of the traffic reports are delivered late to the traffic schedule center, the schedule of such a traffic light is negatively affected.

On the other hand, in arterial street and mesh-alike signalized road network scenarios, the located traffic lights over the signalized road network cooperatively schedule the phases for each traffic light. Considering the schedule of neighboring traffic lights helps to enhance efficiency over these signalized road networks. Since vehicles usually travel in platoon fashion, communication between located traffic lights keeps each traffic light aware of the successive arriving platoons; the size and the estimated arrival time of each platoon essential considerations for intelligent traffic lights. Figure 2.4 illustrates a signalized mesh-alike road network scenario. Any single path over this road network, in Figure 2.4, can be seen as an arterial street; traffic flow over the arterial street has higher priority to pass each road intersection than other competing flows.

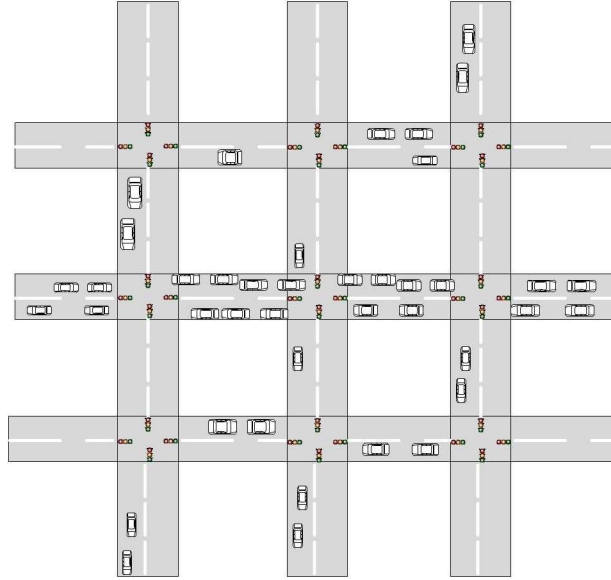


Figure 2.4: Mesh-alike signalized road network scenario.

Table 2.5 summarizes some intelligent traffic light, scheduling mechanisms that have been recently proposed for signalized road networks.

In general, for all signalized road network scenarios, we didactically divide the main problem into three main components: real-time traffic data gathering, cooperations among the located traffic lights and designing traffic light scheduling algorithms. Figure 2.5 illustrates the general architecture of intelligent traffic lights scheduling mechanisms.

First, regarding real-time traffic data gathering, the investigated mechanisms and protocols that have been presented in Section 2.2 can be adapted and used to evaluate

Table 2.5: Intelligent Traffic Lights Controlling Systems

Algorithm/ Mechanism	Consideration	Traffic Lights Controlling Mechanisms	
		Traffic Data Gathering	Schedule Technique
Adaptive TL [92]	Isolated Intersection	Vehicular Ad Hoc	Modified Webster's formula [82]
Phase-based Strategy [101]	Isolated Intersection	Vehicular Ad Hoc	Phase-based Strategy [101]
Simulated TLCS [87]	Isolated Intersection	Optical Information System (OIS)	Extending Green phase for long jam lane
OAF [90]	Isolated Intersection	Vehicular Ad Hoc	On-line Scheduling Algorithm
CATS [93]	Isolated Intersection	Clustering Vehicular Ad Hoc	Density-based Adaptive traffic signals
Markov Model [98]	Mesh-alike Road Network	Stochastic Estimation	Markovian Model
RHODES [99]	Arterial Street/ Mesh-alike Road Network	Inductive Loop Detectors	Hierarchical System
Graph Model [94]	Mesh-alike Road Network	Estimation based of GA algorithm	Graph Model
MARLIN-ATLIN [107]	Mesh-alike Road Network	Statistical and historical evaluation	Multi-agent reinforcement learning (Markov Game)
STPN [108]	Arterial Street/ Mesh-alike Road Network	Mathematical modeling according to the network input	Synchronized Timed Petri Nets scheduling phases
Smart City Framework [88]	Mesh-alike Road Network	Vehicular Ad Hoc	Smart City Framework
ART-SYS [109]	Arterial Street	Vehicular Ad Hoc	Fuzzy Logic

the real-time traffic characteristics of all traffic flows over the signalized road network investigated, as seen from the top level of Figure 2.5.

2.4.1 Cooperation Systems among Located Traffic Lights

In arterial street and mesh-alike signalized road network scenarios, characteristics of traffic along the entire investigated road network are required. These parameters should be taken into consideration in order to schedule the sequence of the competing traffic flows at each signalized road intersection. Using mathematical models, the estimated traffic distribution over the entire road network is formalized, and future distribution and movements are estimated based on the mobility model in use. On the other hand, for real-time traffic distributions over the road network, several mechanisms have been introduced to enable cooperation among the located traffic lights. These mechanisms can be classified into two main categories: centralized and distributed cooperation systems. In this section we investigate the advantages and drawbacks of each system in detail.

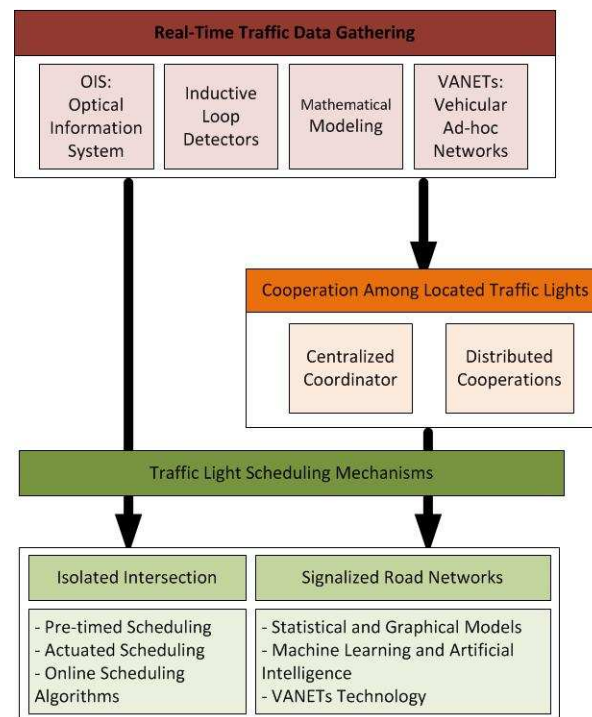


Figure 2.5: Traffic light scheduling mechanisms.

Centralized Coordinator

The characteristics of each traffic flow over the road network are reported to the traffic control center (TCC) in this system [109]. TCC serves as the focal point for the management of the transportation system in urban areas, it integrates data gathered from a variety of different sensor sources. Thus, it provides means for operators to manage traffic and to inform the located traffic lights. The central processor of TCC obtains the best schedule for each traffic light in the investigated area. This approach should decrease stops on roadways, resulting in an increase in traffic fluency for vehicles.

Using a centralized coordinator, the best schedule for each traffic light is found, with all road segments in the network fairly considered. However, a centralized coordinator can cause bottleneck problems over VANETs network; this is because all traffic flows report the traffic characteristics to this central processor. Moreover, high delay is expected in this system due to the time involved in reporting the traffic characteristics, processing them and delivering the best schedule back to each traffic light. In general, the centralized system suffers from scalability issues and timely reporting delivery. Moreover, the central coordinator processor does not consider any flow of traffic unless it receives the traffic

report of the respective traffic flow. Missing traffic reports cause problems with the accuracy of the best scheduling algorithm for each located traffic light.

Distributed Cooperations

In distributed cooperation systems, each located traffic light communicates with its neighboring traffic lights over the signalized road network using traveling vehicles. Communications among located traffic lights provide each one with estimated sizes of expected platoons and their arrival time at each input traffic flow. The scheduling algorithm of each traffic light is adapted according to the real-time data gathered and the schedule of neighboring traffic lights.

Barba et al. [88] have proposed a distributed framework of an urban grid-layout area, in which intelligent traffic lights have been installed over the road network. These traffic lights communicate with each other to gather weather information and traffic characteristics of located road segments within the investigated road network area. Moreover, they send warning messages to drivers about the real-time traffic characteristics within the road segment.

Furthermore, Mirchandani et al. [99] introduce a hierarchical traffic light controlling architecture (RHODES). In RHODES, each intelligent traffic light over the signalized road network gathers the traffic characteristics of its surrounding flows of traffic. It then reports these gathered traffic characteristics to neighboring traffic lights. Forwarding the traffic reports towards the neighboring traffic lights helps to inform each located traffic light within the signalized road network about the traffic distribution throughout the rest of the road network.

In general, the distributed system eliminates the bottleneck problem and decreases the delay time involved in reporting the traffic compared to the centralized cooperation system. However, the gathered data may not be consistent for all traffic lights, in the event that some packets have been dropped over the network. The schedule decision is made independently at each traffic light, which is affected by the accuracy of the gathered traffic data and the efficiency of the scheduling algorithm.

2.4.2 Traffic Light Scheduling Mechanisms

Several mechanisms have been proposed in the literature which aim to efficiently schedule the phases of the timing cycle at each traffic light. Here, we investigate the previously proposed traffic light scheduling mechanism for several signalized road network scenarios.

Isolated Road Intersection

The located traffic light at each isolated road intersection has been extensively investigated in the literature. All proposed mechanisms in this field have shown superior performance to the typical pre-timed scheduling mechanism. In the typical pre-timed mechanism, the schedule of all competing flows and the time period of each phase are permanently set. Historical studies of the traffic distribution over the investigated area help the engineers to set the parameters of each traffic light at the time of installation at the intersection.

On the other hand, intelligent traffic light scheduling algorithms require real-time traffic data gathering equipment. Utilizing the gathered traffic data, the sequence phases of each traffic light can be scheduled efficiently. Most of the previously proposed intelligent traffic light controlling algorithms have been designed following the actuated traffic light mechanisms or semi-actuated mechanisms. A traffic light control algorithm, which is embedded in SUMO simulator [83], is introduced by Krajzewicz et al. [87]. It aims to minimize the queuing delay time at each road intersection by considering the length of the traffic jam in front of the traffic light, in order to schedule the phases of the next timing cycle efficiently. In this research study, each intelligent traffic light tries to reduce the detected jams in front of it separately, by assigning a longer scheduled time to the traffic flow that contains a longer queue jam.

Priemer et al. [101] have considered an RSU installed at each isolated road intersection, which uses the phase-based strategy algorithm [101] to schedule the phases of the located traffic light and then to optimize the traffic fluency. The intelligent scheduling algorithm minimizes the delay time of traveling vehicles at each road intersection. On the other hand, Maslekar et al. [91] and Gradinescu et al. [92] have developed adaptive traffic signal control algorithms which utilize V2V communications in VANETs. The located traffic light, at each road intersection, considers the reported real-time traffic characteristics of all flows of traffic that are intending to pass the road intersection, in order to schedule the next timing cycle phases. This system [91] reduces wait time and the queue length at each road intersection. A modified Webster's formula [92] is also used to generate the schedule of each traffic light by Maslekar et al. [91].

Moreover, CATS [93] has been proposed as an adaptive traffic signal control algorithm that adjusts the timing pattern in accordance with vehicular demand. The timing cycle of each traffic light is set based on the estimated traffic density of competing traffic flows. Maslekar et al. [93] used the safe stopping distance (SSD) to set the length of the

adaptive timing cycle. On the other hand, on-line scheduling algorithms have been used to efficiently schedule the competing flows of traffic at each road intersection. Recently, Pandit et al. [90] have developed an on-line algorithm (OAF) to minimize the delay across the intersection by scheduling the optimal sequence of conflicted phases at each traffic light. First, a conflict graph is constructed indicating all competing traffic flows at each isolated intersection. Then, the rule of first come first serve schedules the competing platoons of traffic in each flow, using the estimated arrival time of each predictable platoon.

Signalized Road Networks

Several research studies have developed intelligent traffic light systems for road networks in urban and suburban areas, investigating arterial street and mesh-alike signalized networks. We classify the previously proposed intelligent traffic light systems, which have been introduced for the entire signalized road network, into three main categories. These categories are configured based on the utilized technology to generate the schedule of each traffic light in the signalized road network: statistical and graphical models, machine learning and artificial intelligence, and VANETs technology. Here, we investigate the previously proposed solutions in each of these categories:

- **Statistical and graphical models:** First, intelligent traffic light systems that aim to optimize traffic fluency all over the signalized road network have been investigated either statically [98], [99] or graphically [94]. Yu et al. [98] proposed a probabilistic model of controlling signalized road networks. The optimal timing plan of each traffic light has been proposed as a decision-making problem for a controlled Markov process [98]. The Markovian model is developed as a traffic light controlling system; it incorporates Robertson's platoon dispersion traffic model between intersections, and employs the value iteration algorithm to find the optimal decision for the controlled Markov process. Moreover, Mirchandani et al. [99] proposed a real-time traffic light controlling system. The input of this system is a real-time characteristic of traffic flow of the investigated road network. The system divides the traffic control problem, in a hierarchical and interconnected fashion, into several subproblems. The controlling algorithm is divided into two successive phases: the intersection control phase and the network flow control phase. A decision tree is used to develop the traffic flow controlling system over the road network. A graphical scheduling solution is presented by Chen et al. [94], in which

initially *genetic algorithm* [95], *particle swarm optimization* [96] and *the ant colony optimization* [97] algorithms are adopted to obtain an optimal solution for each traffic light schedule setting. A graphical representation of the targeted area is then used to find the optimal schedule for the traffic flows over the entire road network.

- Machine learning and artificial intelligence:** The second category of technology used to produce the optimal schedule for each traffic light in urban road networks is comprised of machine learning and artificial intelligence mechanisms. El-Tantawy et al. [107] applied the multi-agent reinforcement learning (MARL) approach to the adaptive traffic signal control (ATSC) problem for road network systems. El-Tantawy et al. [107] also proposed a multi-agent reinforcement learning system for an integrated network of adaptive traffic signal controllers (MARLIN-ATSC). Two possible modes are proposed by MARLIN-ATSC: independent mode and integrated mode. The integrated mode shows a reduction in the average intersection delay and travel time compared to the independent mode, especially for simulation scenarios of longer duration. Furthermore, synchronized timed petri networks (STPN) are utilized by Huang et al [108] to design and to analyze an urban traffic light controlling system. The applications of STPN are modularized into eight-phases, six-phases and two-phases of traffic light controlling systems. The traffic network acts as the input for the proposed procedure. This procedure also recommends the required number of traffic lights and the phase transaction of each traffic light based on the STPN model.
- The technology of VANETs:** Finally, VANETs technology has been used to deliver the traffic schedules of neighboring traffic lights over signalized road networks [127], [128], [129], [130], [131], [132], [133], [134]. Thus, these traffic lights cooperatively find the best schedule for each timing cycle, while considering the previous schedules of neighboring traffic lights. Barba et al. [88] developed a smart city framework using intelligent traffic lights (ITLs). In this framework, an ITL is installed at each road intersection, where four streets (i.e., eight flows of traffic) connect at the intersection. These ITLs send warning messages to the traveling vehicles to inform drivers about the traffic and the weather conditions of the different road segments in the city. This framework can be used to design an adaptive traffic light system for road networks. Tomescu et al. [109] investigated the arterial street scenario (i.e., open network). An adaptive traffic light controlling system, that

considers real-time traffic information including driver behavior and environmental factors, is introduced in an arterial street scenario. In the proposed system (ART-SYS), every road intersection is initially controlled by the traffic characteristics of its surrounding road segments. The cooperative connections between neighboring intersections are then considered in the controlling system of each adaptive traffic light.

2.5 Discussion

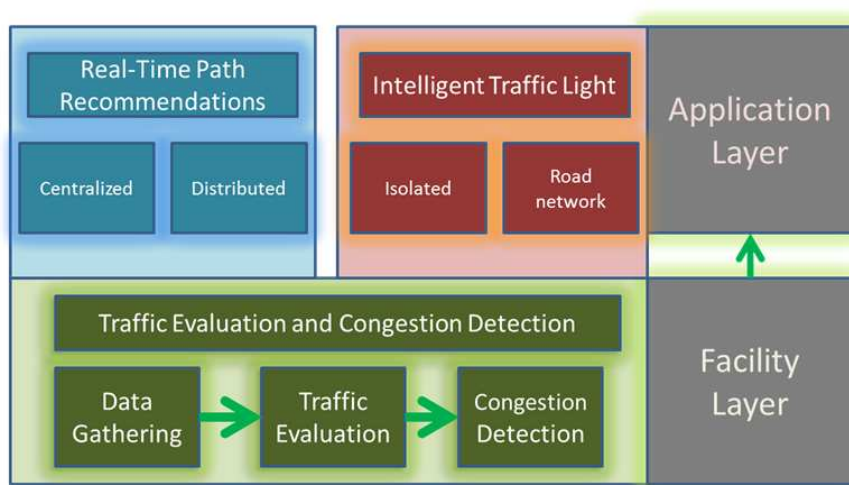


Figure 2.6: Architecture of traffic efficiency applications in VANETs.

We investigate the traffic efficiency applications that have been proposed for road networks using VANETs technology. Figure 2.6 summarizes the architecture of the traffic efficiency applications that have been proposed in the literature. First, the traffic evaluation and congestion detection mechanisms form a facility layer for optimal path recommendation protocols and efficient traffic light scheduling algorithms. Traffic evaluation mechanisms gather the traffic data of all traveling vehicles, which is used to evaluate the traffic characteristics of each investigated area or road segment. These characteristics include traffic speed, traffic density, traffic volume, and/or estimated traveling time. Then, areas with high traffic congestion are detected and reported. Each area or road segment is detected as having a congested or overloaded status in one of the following three scenarios: the traffic density exceeds the ideal capacity; traveling vehicles are forced

to move at a slower speed than the maximum speed limit; or, the estimated travel time is more than the optimal travel time of that area or road segment.

The optimal path recommendation protocols use the traffic evaluation of the investigated road network to select the best path towards each destination. Several parameters of these traffic characteristics have been considered, including travel distance, travel time, fuel consumption, gas emission and context of located road segments considerations. These path recommendation protocols can be classified into centralized and distributed approaches. Bottlenecks and single points of failure on the communication network are the main limitations of the centralized approach. However, the distributed approach requires extra communication overhead in terms of bandwidth consumption to keep all located processors synchronized about the gathered traffic data.

On the other hand, efficient traffic light controlling algorithms have been investigated for isolated road intersections and for signalized road network scenarios. In general, efficient traffic lights at isolated road intersections gather the real-time traffic characteristics of competing traffic flows. They then obtain an efficient schedule of the sequence phases for each timing cycle. On the signalized road network, cooperation between located traffic lights are required to generate a schedule that considers the general traffic distribution over the entire signalized road network. Centralized and distributed cooperation systems have been introduced to cooperate among located traffic lights. Moreover, several scheduling algorithms have been introduced to schedule the phases of traffic at each traffic light efficiently.

2.6 Summary

In this chapter, we have reviewed the traffic efficiency applications that have been proposed over the road network using VANETs. All traffic efficiency applications require real-time traffic distributions over the road network. The process of gathering real-time traffic characteristics of each road network using VANETs has faced several challenges, especially for scalable areas of interest and large numbers of traveling vehicles. Optimal path recommendations have been investigated to enhance traffic fluency over the road networks, taking several real-time traffic characteristics into consideration. Finally, we researched efficient scheduling algorithms of each traffic light for isolated road intersections and for signalized road networks. These scenarios have been handled using VANETs technology to enhance the utilization of the road network and the available resources.

Chapter 3

An Efficient Congestion Detection Protocol (ECODE)

3.1 Introduction

In this chapter, we present an Efficient COngestion DETection (ECODE) protocol that aims at evaluating the real-time traffic characteristics of each road segment (i.e., the road section connecting between any two adjacent intersections). Moreover, ECODE efficiently and reliably detects road segments with high traffic congestion in any urban grid-layout area. The traffic situation and the traffic congestion level of each road segment change from time to time. In order to obtain a real-time traffic evaluation of each road segment, ECODE was run repeatedly with the proactive, reactive and hybrid execution schemes presented in this chapter. These schemes aim also at investigating the effects of several data dissemination mechanisms that are usually used to control the congestion over the connecting network (i.e., VANETs). The complexities of ECODE are discussed; its performance is evaluated and compared to previous protocols in this field. An extensive set of scenarios and experiments have been used, which are in turn implemented by NS-2. Finally, in this chapter we have used ECODE to evaluate traffic congestion level and to investigate traffic distribution over some prominent streets in the downtown area of Ottawa; this study has been set up as a use case example of ECODE.

3.2 The Proposed Congestion Detection Protocol: ECODE

ECODE first evaluates traffic characteristics in terms of traffic speed, traffic density and estimated travel time of each road segment separately. The consideration of three different characteristics enhances the accuracy of the traffic evaluation at any road segment. In the case of inaccurate information derived from investigating any of these characteristics due to voluntarily slow-moving vehicles or inaccurate traffic density evaluations, other characteristics can help estimate the traffic situation accurately. Moreover, when ECODE evaluates these characteristics, it considers the direction of traffic flow in each road segment. It is common to see high traffic congestion in one side of any road segment, while the other side experiences very low traffic density; ECODE is able to detect these scenarios efficiently. The consideration of traffic direction is new in this field of research, and it is essential for accurate path recommendations and for efficient traffic light controlling applications. Finally, when considering location-based cluster mechanisms in our work, it is important to note that the level of accuracy and the efficiency level of the protocol are both enhanced when the configured clusters do not overlap with each other.

In order to evaluate traffic in a certain road segment, the basic traffic data of traveling vehicles in such a road segment are considered. Each vehicle gathers the basic traffic data from surrounding vehicles. After that, a location-based cluster mechanism is applied to divide the area of interest (i.e., the road segment) into a set of adjacent, non-overlapping and manageable clusters. Relay vehicles are selected to evaluate the traffic in each cluster locally, and to forward the traffic evaluation report towards the neighboring clusters. Finally, multi-hop communications among relay vehicles help to expand the traffic evaluation area to cover the entire road segment. More details about the successive phases of ECODE are given in the following sections.

3.2.1 Dissemination and Gathering of Basic Traffic Data

Each vehicle broadcasts an advertisement beacon message (*ADV*) periodically. The *ADV* message declares the location, speed and travel direction of the sender vehicle on each road segment. Whenever any vehicle receives an *ADV* message from the surrounding vehicles, it adds the basic traffic data of the sender vehicle to its Neighboring Table (*NT*); this is done only if the sender vehicle is located on the same road segment. The direction of each traveling vehicle can be obtained from the *ADV* message. The *ADV* message

contains a special field that determines whether the vehicle is moving east, west, north or south. At each road segment, two opposite directions of traveling vehicles should be detected; the traffic characteristics of each direction are evaluated separately. In the case that ECODE uses an *ADV* message that does not contain a specific field for the direction of moving vehicles (e.g., typical safety beacons), the travel direction of each vehicle can be obtained by comparing the location information of any two successive beacons sent by the same vehicle.

Algorithm 1 illustrates the data gathering phase systematically. Each vehicle gathers information pertaining to the location, speed and travel direction of the surrounding vehicles only if it receives the *ADV* message of these vehicles; and, if they are located at the same road segment.

Algorithm 1: Basic Traffic Data Gathering

Data: NT_i : Neighbor Table in V_i and RS_{V_i} : Road Segment where V_i is located.

```

1  $V_i$  broadcasts an  $ADV_i$  message;
2 When  $V_i$  receives  $ADV_j$ ;
3 if  $RS_{V_i} == RS_{V_j}$  then
4   |  $V_i$  adds the  $ADV_j$  to  $NT_i$  table ;    /* If  $V_i$  and  $V_j$  are located on the same road segment */
5 else
6   |  $V_i$  drops the message ;                /* Otherwise the contents of the message should be ignored */
7 end
```

If all vehicles broadcast the *ADV* message at the same time, most of these messages will collide with each other; this will occur despite the use of the Carrier Sense Multiple Access CSMA methods [110] to control the communication channels at VANETs. Traveling vehicles will then miss most of the basic data of their neighboring vehicles. Numerous research studies have investigated the adaptive beacon transmission in a vehicular environment to solve this problem [35], [36], [37]. In general, any of these solutions can be used to reliably disseminate the basic traffic data over VANETs. However, we present in this chapter a simple data dissemination solution with a single retransmission and a small dissemination time interval; this solution guarantees the minimal requirements of ECODE. First, in order to decrease the percentage of collisions between the message broadcasts and to increase protocol accuracy, each vehicle chooses a random time in a small interval period to broadcast the *ADV* message (γ). Here, γ is the empirically assigned interval of each vehicle, where vehicles choose a random time during γ to broadcast the basic traffic data.

Moreover, each vehicle broadcasts its *ADV* message twice (i.e., single message retransmission); in the case where any of these broadcast messages collide at any of the

two broadcast attempts, there is another opportunity to receive the basic traffic data message of the sender vehicle from the following broadcast attempt. There is a chance that some of these broadcast messages will never be received by some vehicles after both attempts, especially for very small broadcasting intervals; this will slightly affect the accuracy of the traffic evaluation. However, empirically this technique performs well by increasing the accuracy of evaluation for the traffic characteristics of each road segment.

3.2.2 Road Segment Clustering

In this phase, each road segment is clustered into a set of virtual, adjacent and non-overlapping areas. Traffic characteristics are evaluated locally in each cluster. The number and size of these clusters are determined based on the length of the road segment (L_{RS_i}) as well as the transmission range of each traveling vehicle (TR_{V_i}). The length of each cluster (L_c) should be less than the TR_{V_i} ($L_c < TR_{V_i}$), while the number of clusters (C_n) is computed by Equation 3.1.

$$C_n = \left\lceil \frac{L_{RS_i}}{L_c} \right\rceil \quad L_{RS_i} > TR_{V_i} \quad (3.1)$$

The length of each cluster area (L_c) is set to be less than the TR_{V_i} ; this is to enable $(TR_{V_i} - L_c)/TR_{V_i}\%$ of traveling vehicles in each cluster to directly communicate with traveling vehicles in adjacent clusters. Figure 3.1 illustrates an example of a clustered road segment.

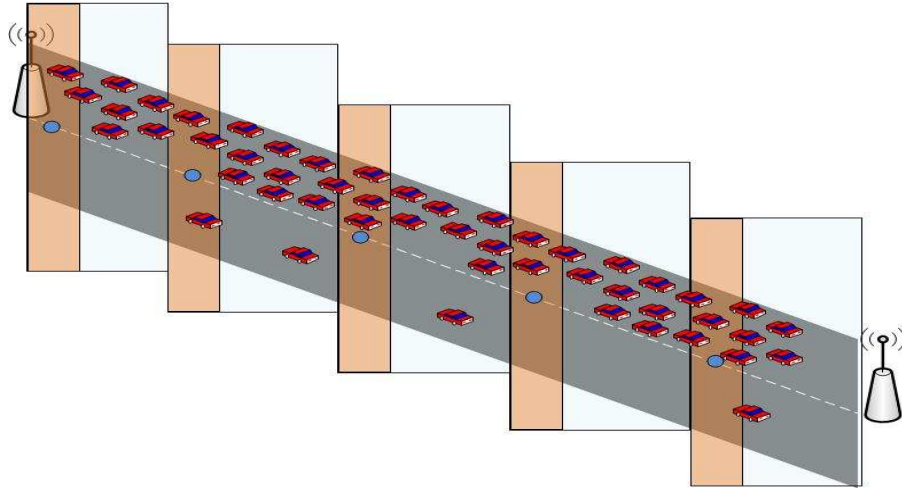


Figure 3.1: Road segment clustering.

We assume that each vehicle is equipped with a digital map that summarizes the physical characteristics (i.e, length, width, coordinates, maximum allowed speed, etc.) of the surrounding road segments. Each vehicle knows the length and coordinates of the road segment it is currently traveling through; therefore, each vehicle can configure the number of clusters and the boundaries of each cluster zone on that road segment. The first cluster starts from the beginning edge of the road segment and ends after L_c . The following clusters start from the outer edge of the last configured cluster and end after $2L_c$. The last cluster starts from the outer edge of the last configured cluster and ends at the final edge of the road segment. The length of the last cluster can be less than L_c , based on the length of the tested road segment.

3.2.3 Local Traffic Evaluation

Each vehicle determines the cluster in which it is located; it also evaluates the traffic characteristics of that cluster zone based on its knowledge of the surrounding vehicles located within the same cluster. The boundaries of any cluster are included inside the boundaries of each vehicle transmission range. Each vehicle computes traffic speed, density and travel time of each cluster, for each direction of traffic flow in the road segment. In our protocol, the traffic speed represents the average speed of all vehicles (See Algorithm 2, Line 9), while the traffic density represents the number of located vehicles per square meter (See Algorithm 2, Line 10). In this work, we define a saturated traffic density (SD_i) of each road segment (i). As long as the traffic density (TD_i) of any road segment or cluster is less than the SD_i , vehicles should be able to drive at the maximum allowable speed (MAS_i) of such a road segment. Otherwise, if TD_i is more than SD_i , vehicles will be forced to drive slower than MAS_i ; this is due to the high level of traffic congestion over the road segment.

In some scenarios, at a vehicle V_j , the evaluated TD_i is less than SD_i at the located cluster of the road segment i ; the traffic speed of such a cluster is, however, slower than MAS_i . In these scenarios one of the following two cases can be concluded: a) vehicles are voluntarily traveling slowly in this cluster; b) due to some kind of obstacles (e.g., large vehicles) or to collision occurrences over the communications network, V_j misses some beacon messages of traveling vehicles in the respective road segment cluster, and thus the TD and TS evaluation in V_j are not accurate. In order to determine which case causes slow travel speed over the road segment cluster (i.e., traffic congestion or voluntarily slow drivers), V_j sends a request message to one of its neighboring vehicles V_k (i.e., the closest

vehicle to the center of the cluster); V_j requests the traffic evaluation report of V_k which includes the traffic speed (TS_i) and traffic density (TD_i) of such a cluster. If the TD_i in V_k is less than SD_i as well and the TS_i is less than MAS_i , this means that most drivers have voluntarily chosen to drive slowly in this cluster. Otherwise, there is a problem with the gathered data in V_j ; in this case, the TS_i and TD_i of V_j should be updated by the obtained values of V_k . These details are illustrated systematically in *Algorithm 2 lines 14-20*.

Algorithm 2: Local Traffic Evaluation Algorithm

Data: NT_j : Neighbor Table in V_j ; $Count$: number of records in NT_j ; C_{V_j} : Cluster where V_j is located; C_{area} : the area of the cluster; C_{length} : the length of the cluster; S_{sum} : the sum of the speed of all vehicles in each cluster; All_V : the number of all vehicles in each cluster; $TS(j)$: Traffic Speed at V_j ; $TD(j)$: Traffic Density at V_j ; and $ET(j)$: Estimated Travel Time at V_j ; MAS_i : Maximum Allowable Speed for the road segment i ; SD_i : Saturated Density of the road segment i ; V_k : the closet vehicle to the center of the cluster; $TS(k)$: Traffic Speed at V_k ; $TD(k)$: Traffic Density at V_k .

```

1   $S_{sum} = 0$ ;
2   $All_V = 0$ ;           /* Add the speed of all vehicles and count the number of vehicles */
3  for  $Count$  do
4      if  $NT_j(k)$  is in  $C_{V_j}$  then
5           $S_{sum} = S_{sum} + Speed_{V_k}$ ;
6           $All_V = All_V + 1$ ;
7      end
8  end
9   $TS(j) = S_{sum}/All_V$ ;           /* Computing the traffic speed of the road segment */
10  $TD(j) = (All_V/C_{area})$ ;       /* Computing the traffic density of the road segment */
11 if  $TD(j) < SD_i$  &  $TS(j) < MAS_i$  then
12      $V_j$  requests the traffic report of  $V_k$ ; /* Only in low density and high traffic speed case */
13      $V_j$  receives the traffic report of  $V_k$ ;
14     if  $TD(k) < SD_i$  &  $TS(k) < MAS_i$  then
15          $ET(j) = C_{length}/MAS_i$ ;
16     else
17         update  $TD(j)$ ;
18         update  $TS(j)$ ;
19          $ET(j) = C_{length}/TS(j)$ ;
20     end
21 else
22      $ET(j) = C_{length}/TS(j)$ ; /* No need to confirm with other vehicles in this case */
23 end

```

The estimated travel time (ET_i) is computed based on the length of each cluster zone and the traffic speed (TS_i) of that zone. If the TD_i of the cluster in the road segment is less than SD_i , ET_i is computed based on the cluster length and the MAS_i of the road segment (i). The traffic speed (TS), traffic density (TD) and estimated traveling time (ET) are the main metrics that determine the local traffic situation in each cluster. Algorithm 2 illustrates systematically the steps that each vehicle (V_j) follows to evaluate the traffic condition in each cluster, where any vehicle (V_j) is located. The receiver vehicle stores the local traffic evaluation of the cluster where it is located, in a local traffic table

(T_{traf}). The local traffic table (T_{traf}) contains the main characteristics of each cluster on the targeted road segment. The main fields of the T_{traf} table include the cluster identifier (T_{id}), traffic speed (T_{TS}), traffic density (T_{TD}) and estimated travel time (T_{ET}) of each cluster.

3.2.4 Expanding Traffic Evaluation Area

In order to expand the area evaluated to cover the entire road segment, each vehicle aggregates the traffic evaluation of all cluster areas in such a road segment. Adjacent clusters can communicate directly by transmitting traffic evaluation reports. Transmitting the traffic evaluation reports between adjacent clusters helps to obtain a more scalable traffic evaluation. This report tabulates the traffic evaluation of all clusters of which vehicles are informed. Figure 3.2 illustrates the main fields for each record in the traffic evaluation report; as we can see, each record contains a cluster id (C_{id}), information regarding traffic density (C_{TD}), traffic speed (C_{TS}) and the estimated travel time (C_{ET}) required to pass through that cluster.

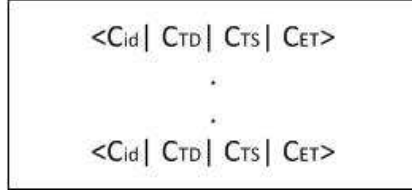


Figure 3.2: Traffic evaluation report (T_{Erep}).

Whenever any vehicle V_i receives the traffic evaluation report of any adjacent cluster, it inserts the traffic evaluation of this cluster as a new record into the internal T_{traf} table. Selecting suitable relay vehicles to transmit the traffic evaluation report in each cluster helps reduce bandwidth consumption and the delay time needed to gather the traffic evaluation data. Algorithm 3 illustrates the process undertaken by any vehicle, after receiving a traffic evaluation report (T_{Erep}).

The traffic characteristics are first evaluated at each cluster separately based on the basic broadcast traffic data of each vehicle. In the case where a certain vehicle broadcasts its basic data while it is in Cluster A and then moves to Cluster B immediately, it will be only considered in the traffic evaluation of Cluster A. It must be considered that this vehicle, which is either in Cluster A or Cluster B, depends on its location when it broadcasts the basic data beacon. Both considerations will have the same effect on the

Algorithm 3: Expanding the Evaluated Area

Data: T_{traf} : the local traffic evaluation table in each vehicle; T_{Erep} : the traffic evaluation report message.

```

1 while  $V_i$  receives a  $T_{Erep}$  do
2   Update the  $T_{traf}$  table :
3    $T_{traf} = T_{traf} \cup T_{Erep}$ ;
4    $V_i$  sets the  $T_{Erep}$  by the  $T_{traf}$  records;
5   if  $V_i$  is selected as a relay vehicle then
6      $V_i$  broadcasts the  $T_{Erep}$  ;
7   end
8 end

```

overall evaluation of the entire road segment; this is due to the method of combining the traffic evaluation of all located clusters on the road segment.

Selecting Relay Vehicles

In order to achieve a solution for more efficient bandwidth consumption, a relay vehicle is selected in each cluster based on its location; it is selected to transmit the traffic evaluation report for its respective cluster. In each configured cluster, we have defined a virtual reporting zone, as illustrated in Figure 3.1. Vehicles in the reporting zones are responsible for transmitting the traffic evaluation report towards the adjacent clusters. We set the reporting zone of the cluster by the length of $(TR_{V_i} - L_c)$; this enables any vehicle in this reporting zone to communicate with all vehicles in its adjacent clusters. All vehicles in the next cluster will receive the local traffic evaluation of the previous cluster. The traffic evaluation is broadcast by the relay vehicle that is selected over the reporting zone, as shown in *Algorithm 3 Lines 5-7*.

Relay vehicles are selected in each reporting area, where the relay vehicle in each cluster is the closest vehicle to the center of the reporting area. The relay vehicle for each cluster can be changed over time due to the vehicle mobility behavior. This works in ECODE because all vehicles in each cluster have the same data, so any vehicle can become a relay vehicle. Moreover, at each round we choose the relay vehicle that is close to the boundaries of the cluster, with the intention of delivering the traffic reports to all vehicles located in the next cluster.

Global Traffic Evaluation

The process of continuously updating the T_{traf} table with the surrounding clusters information, and the forwarding of the updated data T_{Erep} , is essential for evaluating the traffic characteristics of the road segment. It enables RSUs, installed at the end of such

a road segment, to gather the traffic evaluation of all clusters in this road segment from nearby traveling vehicles. Real-time traffic monitoring evaluation reports (*TMR*) are generated by these RSUs, summarizing the traffic situation of each road segment in the applicable direction. The traffic evaluation reports of these RSUs feature the aggregated traffic evaluation of all adjacently considered clusters. These *TMR* reports should be available for any real-time traffic efficiency application that would require the traffic distribution data pertaining to its characteristics over the road network, including path recommendation and traffic lights control.

3.3 ECODE and Its Execution Schemes

In this section, we propose three different execution schemes through which ECODE can be iteratively run: proactive, reactive and hybrid schemes. These schemes detect dynamic changes in traffic characteristics over the downtown area. The objective for the use of these schemes is to investigate the effects of the implemented technology at the congestion control function. This will be done in terms of accuracy and bandwidth consumption. The main difference between these mechanisms is observed by the frequency with which successive phases of the protocol are executed. These different schemes are explained in the following text.

- **Proactive:** In this scheme, each vehicle broadcasts its basic travel data periodically. The length of each time interval can be set externally as a variable, in order to guarantee a level of flexibility. However, different time interval lengths are expected to achieve different performance results in terms of bandwidth consumption and the accuracy of the congestion level evaluation during a certain period of time.
- **Reactive:** In this scheme, vehicles only broadcast the *ADV* message if they detect a certain changeable traffic situation (e.g., a sudden change in vehicle velocity, receive a request from the nearest RSU, etc). An inaccurate traffic evaluation is expected in this scheme. However, low bandwidth consumption is expected of this scheme due to the reduced number of transmitted packets.
- **Hybrid:** This scheme combines elements from the proactive and reactive schemes, in order to balance between their advantages and drawbacks. In this scheme, each vehicle broadcasts the *ADV* message to periodically declare its basic data in a similar way to the proactive scheme, with the exception that a longer time interval

is used. In addition, if any vehicle detects a certain changeable traffic situation, it also broadcasts the *ADV* message declaring its basic traffic data.

3.4 Performance Complexities of ECODE

In this section, we investigate the complexities of bandwidth consumption and the accuracy of traffic evaluation metrics in ECODE.

3.4.1 Definitions

- The bandwidth consumption of ECODE (*BandCon*) is the total number of transmitted messages during a given period of time. We assume that the size of these messages is standard.
- The accuracy of ECODE (*ACC*) is the ratio between the number of detected vehicles in a certain road segment and the real number of vehicles that exist during the period of time the information was gathered.

3.4.2 Performance Complexities of ECODE Phases:

More detail about the computation complexities of *BandCon* and *ACC* values are investigated through the discussion of the main phases of ECODE. We investigate each phase that affects the bandwidth consumption and/or the accuracy of ECODE:

Basic Data Dissemination

During this phase, each vehicle declares its basic traveling data to its neighbor vehicles. This is done by broadcasting an *ADV* message, as explained in Section 3.2.1. We refer to the number of *ADV* messages as ADV_{Mess} , and to the number of traveling vehicles in the investigated road segment as N . Each vehicle broadcasts its *ADV* message twice, as justified in Section 3.2.1. Thus, ADV_{Mess} is computed here by Equation 3.2:

$$ADV_{Mess} = 2 \times N \quad (3.2)$$

On the other hand, in the context of accuracy, there is a probability (p_0) that a collision will take place between *ADV* messages, which have been sent at the same time.

This means that the basic data of these vehicles will not be considered in the following phases.

p_0 : The probability that each vehicle (V_i) does not know about one or more vehicles in its transmission range after the data dissemination phase has occurred; this is due to collision occurrence between those *ADV* messages that have been sent simultaneously.

Mathematically, we can explain this probability in Equation 3.3:

$$p_0 = \sum_{i=2}^n \left(\frac{\theta}{\gamma}\right)^{i-1} \quad (3.3)$$

where θ is the interval of time; during this interval, if two or more vehicles send out the *ADV* messages, these messages could be collided together; and, γ is defined as the interval of time for data dissemination and the time interval of data gathering, from Section 3.2.1.

Equation 3.4 computes the accuracy of the traffic evaluation by ECODE, over every vehicle transmission range (TR_{V_i}):

$$Accuracy = \left(\frac{V_c - V_c \times (p_0)^2}{V_c} \right) \quad (3.4)$$

where V_c is the number of traveling vehicles over certain cluster (c) of the road segment.

In general, from Equations 3.3 and 3.4, we can see that when the interval of time for data dissemination is increased (γ), the accuracy of ECODE increases.

Expanding the Traffic Evaluation Area

In each cluster a relay vehicle is selected to compute and transmit the local traffic characteristics ($T_{E_{rep}}$) in that cluster; this is in order to expand the evaluation area. This step is only required in the case where the length of the investigated road segment is longer than the transmission range of each vehicle ($L_{RS_i} > TR_{V_i}$).

We denote the number of relay vehicles that should transmit the $T_{E_{rep}}$ messages as $E(RV_i)$. In the optimal case, each cluster has one relay vehicle; this vehicle is the closest vehicle to the center of the reporting area. The number of transmitting messages ($T_{E_{rep}}$) in any road segment scenario depends on the number of cluster areas in the road segment. Each relay vehicle retransmits an updated $T_{E_{rep}}$ message for $Cn - 1$ times. The total number of transmitted $T_{E_{rep}}$ messages ($ToT_{E_{rep}}$) is given by Equation 3.5:

$$ToT_{Erep} = C_n \times E(RV_i) \times (C_n - 1). \quad (3.5)$$

As we can see from Equation 3.5, the total number of T_{Erep} messages depends on the number of selected relay vehicles in each cluster, and on the number of clusters in each road segment.

3.4.3 Bandwidth Consumption of ECODE

In general, Equation 3.6 computes the total *BandCon* of ECODE:

$$BandCon = \begin{cases} ADV_{Mess}, & L_{RS_i} \leq TR_{Vi} \\ ADV_{Mess} + ToT_{Erep}, & L_{RS_i} > TR_{Vi} \end{cases} \quad (3.6)$$

From Equations 3.2 and 3.5, we can rewrite Equation 3.6:

$$BandCon = \begin{cases} 2 \times N, & L_{RS_i} \leq TR_{Vi} \\ 2 \times N + C_n \times E(RV_i) \times (C_n - 1), & L_{RS_i} > TR_{Vi} \end{cases} \quad (3.7)$$

From Equation 3.1, the *BandCon* can be re-written in Equation 3.8:

$$BandCon = \begin{cases} 2 \times N, & L_{RS_i} \leq TR_{Vi} \\ 2 \times N + \frac{L_{RS_i}}{L_c} \times E(RV_i) \times (\frac{L_{RS_i}}{L_c} - 1), & L_{RS_i} > TR_{Vi} \end{cases} \quad (3.8)$$

We can infer from Equation 3.8 that the total bandwidth consumption of ECODE (*BandCon*) depends only on the number of traveling vehicles (N) in the case that L_{RS_i} is less than or equal to TR_{Vi} . On the other hand, in the case where L_{RS_i} is more than TR_{Vi} , besides the number of traveling vehicles (N), the *BandCon* depends on the following parameters: the length of the road segment (L_{RS_i}), the transmission range of each traveling vehicle (TR_{Vi}) and the number of selected relay vehicles in each cluster ($E(RV_i)$).

3.4.4 Accuracy of Traffic Evaluation in ECODE

This section addresses the accuracy evaluation of the traffic characteristics in any road segment scenario. In a situation where traveling vehicles are distributed over several non-overlapped transmission ranges in the road segment, the computed probability (p_0)

in Equation 3.4 should be divided by the number of clusters existing on the road segment. Thus, Equation 3.9 computes the potential for accurately evaluating traffic in any road segment scenario.

$$ACC = \frac{V_n - \frac{V_n \times (p_0)^2}{(C_n)^2}}{V_n} \quad (3.9)$$

where V_n is the number of traveling vehicles over the investigated road segment.

Over all, we can see from Equation 3.9 that the accuracy of ECODE (ACC) augments when the length of the investigated road segment is increased; this is the case when the number of vehicles is the same, and when the transmission ranges are the same. ACC also increases by increasing the intervals of data dissemination and the interval time of data gathering (γ).

3.5 Performance Evaluation of ECODE and Its Execution Schemes Variants

In this section, we present the performance evaluation of ECODE and the evaluation of its different execution schemes, which have been proposed as an iterative execution of ECODE. The performance of ECODE is evaluated using an extensive set of simulation experiments that use NS-2 [64]. The simulation parameters that have been used in these experiments are summarized in Table 3.1. In the first set of experiments, we evaluate ECODE for different road lengths and traffic density scenarios; this is done to ensure that ECODE works successfully and that it does not suffer from high overhead. In the second set of experiments, we compare ECODE to two other previously proposed congestion detection and traffic evaluation protocols (i.e., COC [15] and StreetSmart [53]) in order to show the advantages of ECODE over these protocols. Finally, we test and compare the different execution schemes of ECODE on various scenarios with different road lengths, where the traffic density of each road changes over time. We have run each experiment for 30 different traffic scenarios to guarantee a high confidence interval of 95%. In most of the illustrated figures, we plot the average value of each 30 experimental results; and, in some figures we have plotted each obtained interval as well.

3.5.1 Traffic Scenarios

In the scenarios that have been used to obtain the traffic evaluation, the road length varies from 200 meters to 1000 meters; meanwhile, the traffic density at each road segment

Table 3.1: Simulation Parameters of the Road Segment

Parameter	Value
Wireless Medium	IEEE802.11
Transmission Range (m)	200
Vehicle Speed (m/s)	3 - 17
Simulation Time (s)	500
Simulation Area (m^2)	20 X 200 - 20 X 1200
Number of RSUs	2
Number of Vehicles	20 - 500
Simulation Map	Road segment in an urban area
Cluster Length (L_c)	75% of TR_{V_i}

varies from 0.025 to 0.125 vehicle per meter in each lane. The density level (D_L) for each scenario is the ratio between the number of vehicles (N) at each road segment and the road segment length (L_{RS_i}) per lane on the respective road segment ($Lane_{num}$): this is illustrated in Equation 3.10:

$$D_L = N / (L_{RS_i} \times Lane_{num}) \quad (3.10)$$

These traffic scenarios feature an occurrence of a traffic jam in only one of the investigated road segment directions. Thus, 80% of all vehicles were assigned to move in one direction while the remaining 20% were assigned to move in the other direction. On the other hand, for evaluating the execution schemes, we used a set of mobility scenarios for different one-direction road segment lengths; in this set of scenarios, the traffic density varies in time from 0.025 to 0.125 vehicles per meter per lane. The simulation time for these experiments is 500 seconds, and the traffic congestion level changes each 50 seconds. The congestion level increases by increasing time; it does so until it reaches a high congestion level of 0.125 vehicles per meter in each lane after 200 seconds. The congestion level decreases then until it reaches the low level of 0.025 vehicles per meter in each lane.

3.5.2 The Performance Evaluation of ECODE

In this section, we evaluate the performance of ECODE protocol in terms of end-to-end delay and bandwidth consumption (i.e., number of transmitted packets) for different road segment lengths and different road density scenarios.

In Figures 6.9(a) and 6.9(b), we illustrate the end-to-end delay of ECODE. We define the delay time, in our experiments, as the time when the vehicle starts broadcasting the

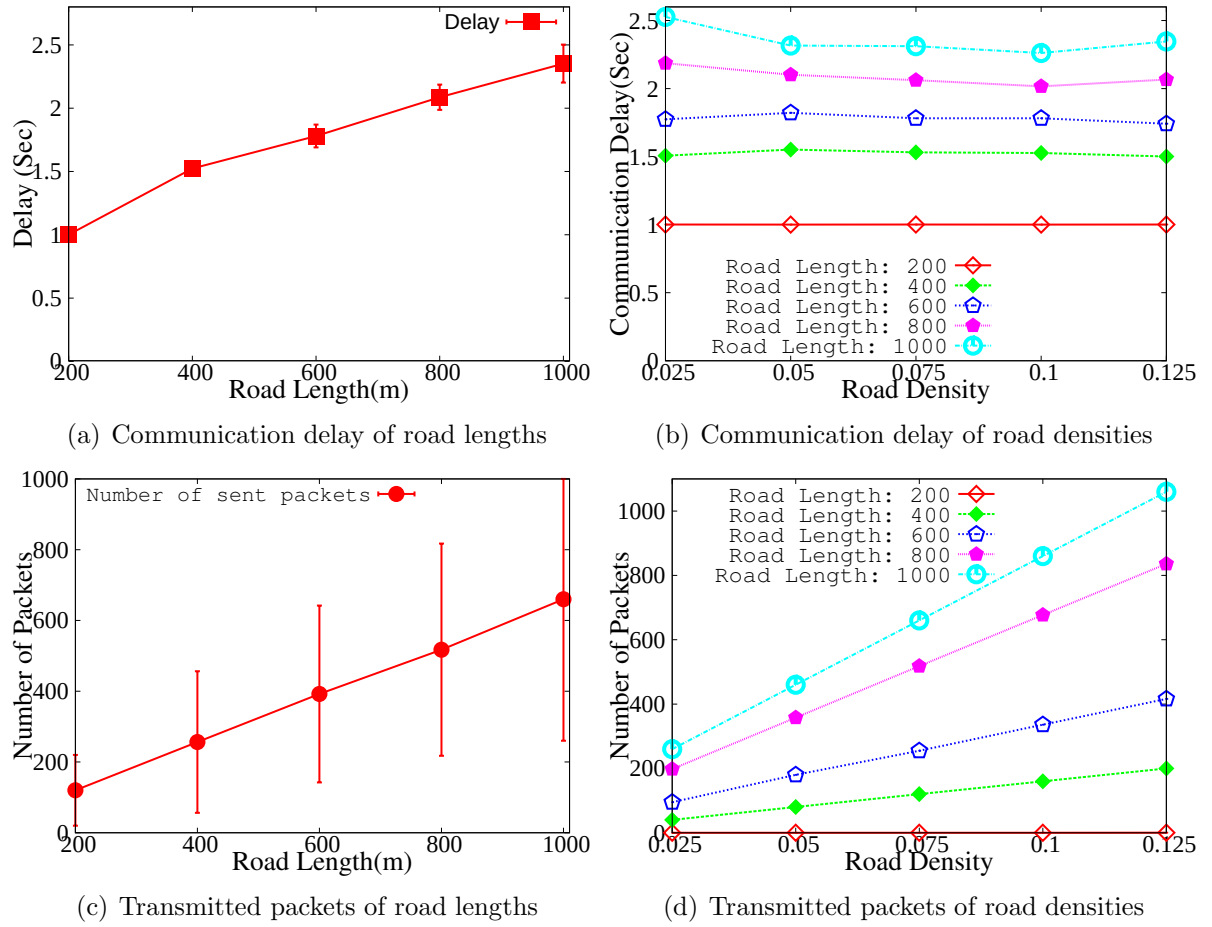


Figure 3.3: Communication delay and number of transmitted packets of ECODE.

ADV message to when the traffic characteristics and the congestion level are computed and reported by the responsible RSU. From Figures 6.9(a) and 6.9(b), we can infer that ECODE needs at most 2.5 seconds to compute the traffic characteristics over any investigated road segment length. From Figure 6.9(a), we see a longer road segment incurs a greater delay by ECODE. On the other hand, as we can see from Figure 6.9(b), the density of each road segment has a negligible effect on the delay compared to the road segment length parameters. For long road segments, high traffic density requires less time delay than low traffic density scenarios; we justify this result by the traffic distribution of each cluster in low traffic density scenarios, in which more time is required to report each cluster traffic characteristic.

In regards to the bandwidth consumption, we have investigated the number of transmitted packets of different scenarios. First, from Figure 3.3(c), we can see a longer road

segment leads to an increase in the number of sent packets. This is to be expected, because with longer road segments, more vehicles are expected to exist in a context where each vehicle transmits its basic data. Furthermore, a greater number of clusters occur on longer road segments, which means more T_{Rep} messages are transmitted to report on the local traffic evaluation of each cluster. From Figure 3.3(c), we can see bandwidth consumption at each road segment length, as well as for the greatly different road densities investigated. Figure 3.3(d) indicates that a greater number of messages sent with a higher density in each road segment; this is simply explained by the *ADV* broadcast messages of each traveling vehicle. Another interesting result can be observed in Figure 3.3(d); even if the number of vehicles remains the same, more messages should be transmitted if vehicles appear on longer road segments. This is due to the fact that a higher number of clusters configured on longer road segments; this means that more T_{Rep} messages should be sent.

3.5.3 Comparison Study of ECODE to COC and StreetSmart

In this section, we compare our proposed protocol (i.e., ECODE) to two other previously proposed traffic evaluation protocols, COC [15] and StreetSmart [53]; this is undertaken in order to show the advantages of ECODE over these protocols. We first investigate the overhead of these protocols in terms of end-to-end delay and the number of transmitted packets (i.e., bandwidth consumption) in each scenario. We also study the accuracy of these protocols for different scenarios and under different assumptions.

Starting with the communication delay overhead, in Figures 3.4(a) and 3.4(b), we compare the end-to-end delay of ECODE to that of COC and StreetSmart for scenarios in which there are different road lengths and different road densities respectively. From Figure 3.4(a), we see that the delay times of COC and ECODE protocols increase when the road length is increased; we also notice how the delay observed in StreetSmart is the same for all road lengths. However, for all road lengths, ECODE obtains 70% faster results than the COC protocol, while it produces a delay time of 10% longer than that of StreetSmart. On the other hand, Figure 3.4(b) illustrates the comparison between these three protocols in terms of end-to-end communication delay for different road densities. From Figure 3.4(b), we see that the delay of ECODE is the same for all road density scenarios, whereas the communication delays of COC and StreetSmart increase whenever the road density increases. In general, for all investigated road density scenarios, ECODE achieves 70% faster results when compared to COC and 10% slower results when com-

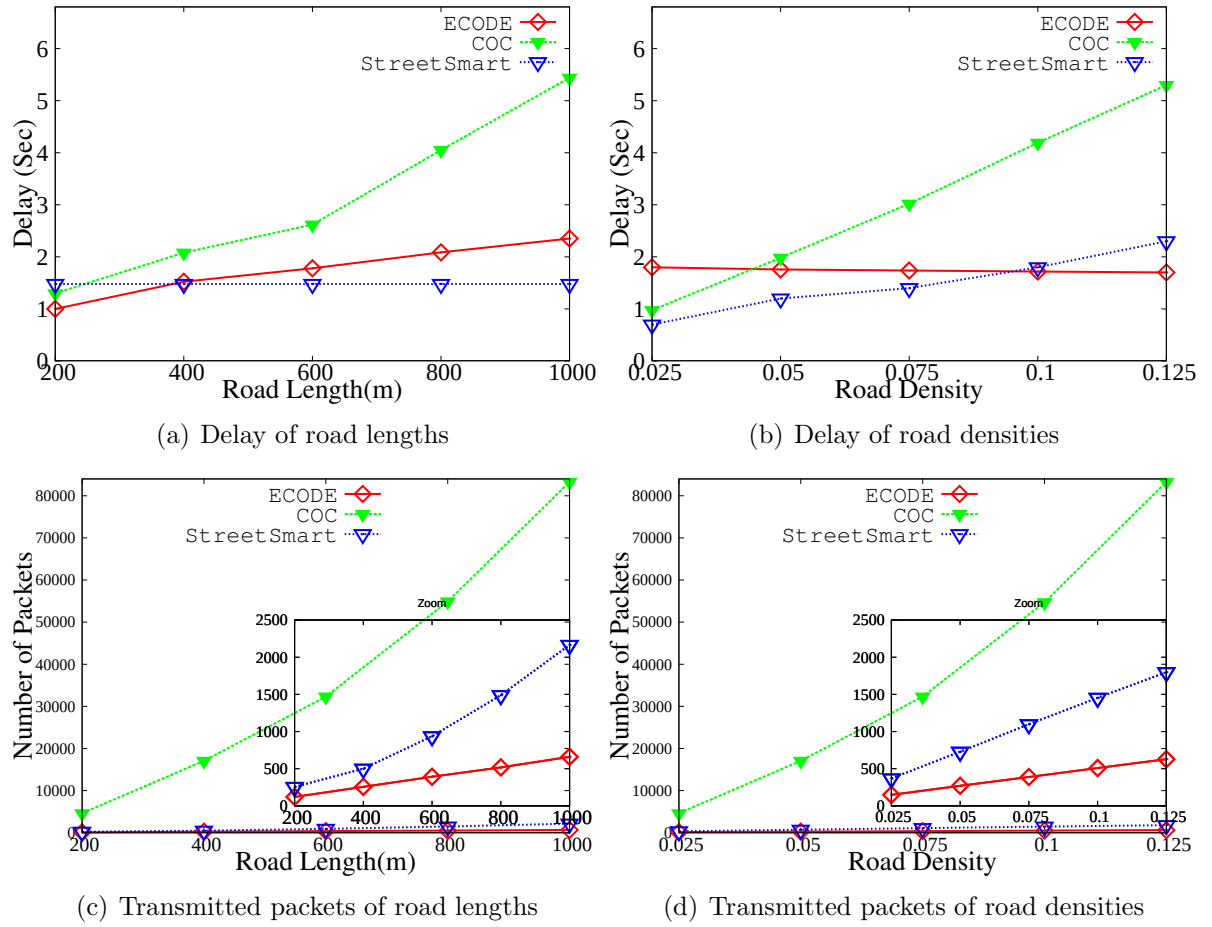


Figure 3.4: End-to-End communication delay and number of transmitted packets in ECODE, COC, and StreetSmart, for different road lengths and different road densities.

pared to StreetSmart. However, within higher traffic density scenarios, ECODE achieves a better performance than StreetSmart in terms of end-to-end communications delay.

The other communication overhead parameter we aim to investigate in this comparative study is the number of packets transmitted by each protocol (i.e., bandwidth consumption). Figures 3.4(c) and 3.4(d) respectively illustrate the comparison between ECODE, COC and StreetSmart in terms of the number of transmitted messages for different road lengths and road densities. As we see in both cases, COC incurs a huge number of messages compared to ECODE and StreetSmart. For different road lengths and traffic densities, ECODE sends out around 90% fewer messages than COC and 30% fewer messages than StreetSmart respectively. Figures 3.4(c) and 3.4(d) illustrate these results graphically.

For the rest of the comparison performance evaluation, we measure the accuracy of the congestion level obtained. We have defined two different techniques to evaluate the level of accuracy, which are explained in detail below. In the first technique, we compute the ratio between the number of vehicles that each protocol can detect on a certain road segment and the actual number of existing vehicles. The second technique computes the ratio between the number of vehicles that each protocol detects in each road segment direction and the actual number of existing vehicles on that direction.

Figures 3.5(a), 3.5(b) and 3.5(c) illustrate the ratio between the number of detected vehicles in each road segment and the actual number of existing vehicles. We have studied this parameter for different road lengths, road densities and numbers of vehicles. ECODE achieves a success ratio of approximately 100% in terms of its ability to detect the entire number of vehicles in the investigated road segment scenarios. On the other hand, the accuracy levels of COC and StreetSmart both decrease in conjunction with increases in road length, road density or in the number of traveling vehicles. However, COC protocol obtains a better level of accuracy when it is compared to StreetSmart in terms of its accuracy of detecting vehicles in each road segment.

In general, the ECODE protocol achieves an accuracy level that is 10%-25% higher than COC and StreetSmart protocols, respectively. The 10% failure in COC is caused mainly by lost messages; this is because COC uses the blind flooding mechanism in order to forward vehicle advertisement messages, which leads to different dissemination problems. On the other hand, StreetSmart handles the dissemination problem by using a cluster-based mechanism which forwards a statistical measurement for each cluster/hop. This statistical measurement considers that some vehicles exist in more than one cluster due to their highly mobile nature and to the overlapping between different adjacent clusters.

ECODE is the first congestion detection protocol proposed that considers the direction of traveling vehicles, as mentioned earlier in this chapter. All previously proposed protocols consider only half of the detected vehicles located in the first road direction, with the other half located in the opposite direction. We compared ECODE to COC and StreetSmart in terms of the congestion level occurring in each road segment direction. Figures 3.5(d) and 3.5(e) show the comparison results for different road lengths and different road densities respectively. From these figures, we can clearly see that ECODE achieves a traffic evaluation that is 50% more accurate compared to COC and StreetSmart.

Ultimately, we summarize the finding of this comparative study by noting that the

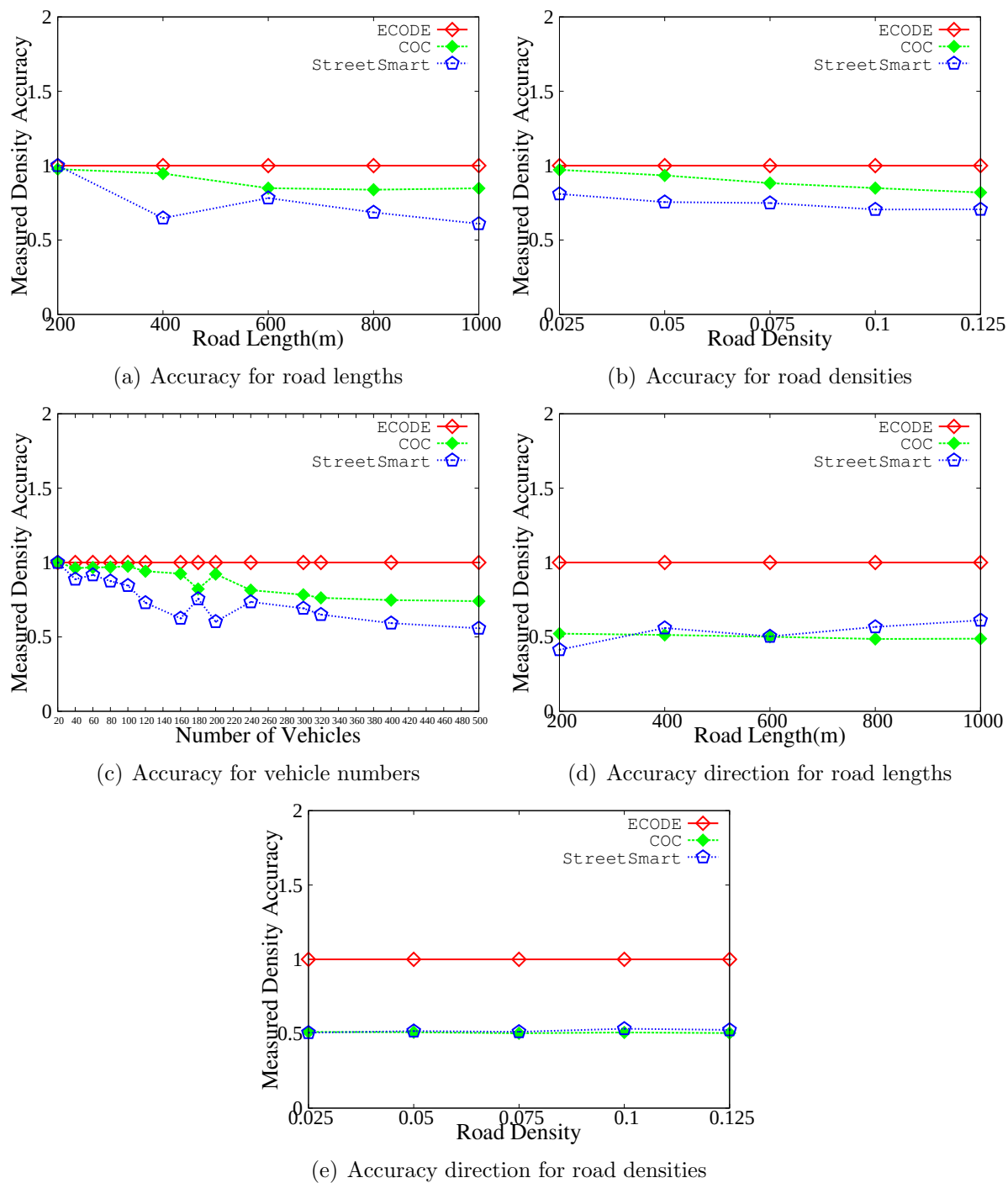


Figure 3.5: Accuracy of ECODE, COC, and StreetSmart and accuracy in each direction.

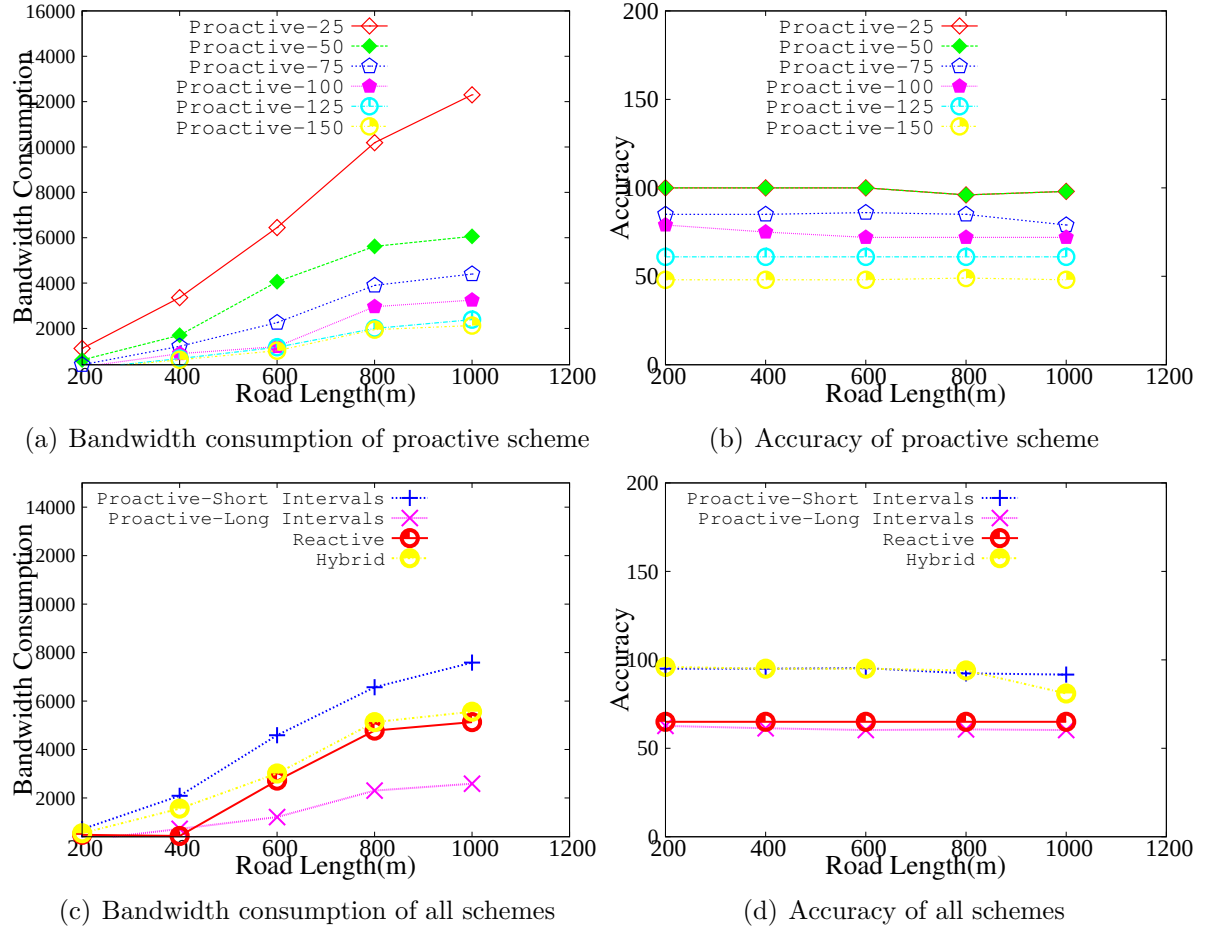


Figure 3.6: Performance evaluation of the proactive execution scheme and a comparison performance of proactive, reactive and hybrid schemes.

COC protocol suffers from a high communication overhead in terms of end-to-end delay and the number of transmitted packets; however, it achieves a higher level of accuracy in terms of its ability to detect the vehicles in the area of interest when compared to the StreetSmart protocol. ECODE achieves the best performance in terms of bandwidth consumption and evaluation accuracy compared to COC and StreetSmart protocols, but has a delay time of 10% longer when compared to StreetSmart protocol.

3.5.4 The Performance Evaluation of ECODE Execution Scheme Variants

In this section, we examine the performance of ECODE execution schemes (i.e., proactive, reactive and hybrid) in terms of bandwidth consumption and the accuracy of the evaluated congestion level. In this set of experiments, we set the interval of the proactive scheme to 25, 50, 75, 100, 125 and 150 seconds. As we can see from Figures 3.6(a) and 3.6(b), the bandwidth consumption of the proactive scheme decreases when the length of the periodic interval is increased; however, the accuracy of the detected level of congestion decreases by increasing the length of the periodic interval. The accuracy levels obtained when setting intervals to 25 and 50 seconds are almost equivalent; however, in the first case (i.e., 25 seconds), the bandwidth consumption is much greater than in the latter case (i.e., 50 seconds). We infer from Figures 3.6(a) and 3.6(b) that with greater intervals, more bandwidth consumption is saved; however, this negatively affects the accuracy for all proactive iterative execution schemes.

In order to compare the reactive and hybrid schemes to the proactive scheme, we classify the obtained results of the proactive scheme into short and long intervals. We compute the average evaluation results of 25, 50 and 75 second intervals as short interval evaluations and 100, 125 and 150 second intervals as long interval evaluations. Figures 3.6(c) and 3.6(d) show that, the reactive scheme consumes less bandwidth compared to the proactive scheme with short intervals. However, the reactive scheme does not achieve a high level of accuracy; if a vehicle is moving at the same speed while the congestion level and the number of neighbors have increased, the vehicle will not broadcast the *ADV* message, and the ECODE protocol will not measure the congestion level of that road segment. The hybrid scheme in our experiments is executed every 200 seconds, as is the proactive scheme. However, in the case where any vehicle detects a certain change in its moving speed, it broadcasts the *ADV* message with the basic data to start the traffic evaluation processes. As shown in Figures 3.6(c) and 3.6(d), the hybrid scheme consumes more bandwidth than the reactive scheme, while it achieves a level of evaluation accuracy that is close to the proactive scheme with short time intervals.



(a) Map of Ottawa downtown area

(b) SUMO representation of the map of Ottawa

Figure 3.7: The map of Ottawa downtown area and the SUMO representation of the map.

3.6 Use Case: Traffic Distributed over Downtown Ottawa using ECODE

In this section, we aim to evaluate a real traffic scenario of an area in downtown Ottawa as a use case implementation of ECODE. In this study, we first generated a set of real-time traffic mobility scenarios in this area using SUMO [83]. These mobility scenarios reflect the real traffic distributions in downtown Ottawa at several different times of day. Then, we compared the traffic on three main parallel streets (i.e., O'Connor, Bank and Kent) that connect the south and north ends of the downtown area. We intend to investigate the traffic distribution in these streets for different traffic density scenarios (e.g., peak hours, night hours, etc). This comparison study will help drivers traveling daily to choose the best street to follow according to the distributed traffic situation. Furthermore, it introduces a good background for researchers to choose the best street or road segment on which to run and test the VANET protocols that require a certain level of traffic density or connectivity.

Figure 3.7(a) illustrates the investigated area of downtown Ottawa; Figure 3.7(b) illustrates the representation of this area by SUMO [83]. As we can see from Figure 3.7(a), O'Connor, Bank and Kent streets are three parallel streets that connect the north and south parts of downtown Ottawa.

- O'Connor Street: Multi-lane one way street constructed as a north-south route.
- Bank Street: Runs through the heart of the downtown area. It is a prominent retail center and is constructed as a two-way route.
- Kent Street: Multi-lane one-way street constructed as a south-north route.

Our comparison of the traffic on these three streets, beginning at Catherine Street and ending at Albert Street, is illustrated in Figure 3.7(a). We have generated different scenarios with a different number of traveling vehicles, that examine different time periods throughout the day. We intend to investigate traffic distribution, traffic speed, and the estimated travel time for these streets during different scenarios.

Table 3.2: Simulation Parameters of the Ottawa Downtown

Parameters	Value
Simulation Area (m x m)	2000 x 2000
No. of RSUs	15
No. of Vehicles	200 - 1000
Transmission Range (m)	250
Map Layout	Downtown Ottawa
Simulation Time (Sec)	1000

Table 5.1 illustrates the parameters used in our simulation; we implemented these experiments using NS-2 [64]. In our simulation environment, we installed an RSU at some road intersections of the investigated streets in order to perform a detailed traffic investigation for the entire area of interest.

We have executed each experiment for 30 different traffic scenarios; the results plotted in Figure 3.8 illustrate the average for these 30 different executions. First, we measured the traffic density, the traffic speed and estimated the time required to travel on these streets. This is shown in Figures 3.8(a), 3.8(b) and 3.8(c), with the travel time lasting about 20 minutes during the peak hours (i.e., around 1000 vehicles in the area of interest). As we can see from Figure 3.8(a), the traffic density in both directions of Bank Street is 90% more than the traffic density of O'Connor and Kent Street. Each experimental scenario had different traffic characteristics, particularly the two directions of Bank Street. For example, the traffic density on the south-north side was higher than the density in the north-south side in some scenarios; on the other hand, in other scenarios, the density on the north-south side was higher than the density of the south-north side. However, when we evaluated the average of 30 different scenarios, the traffic density in both sides more closely resembled each other.

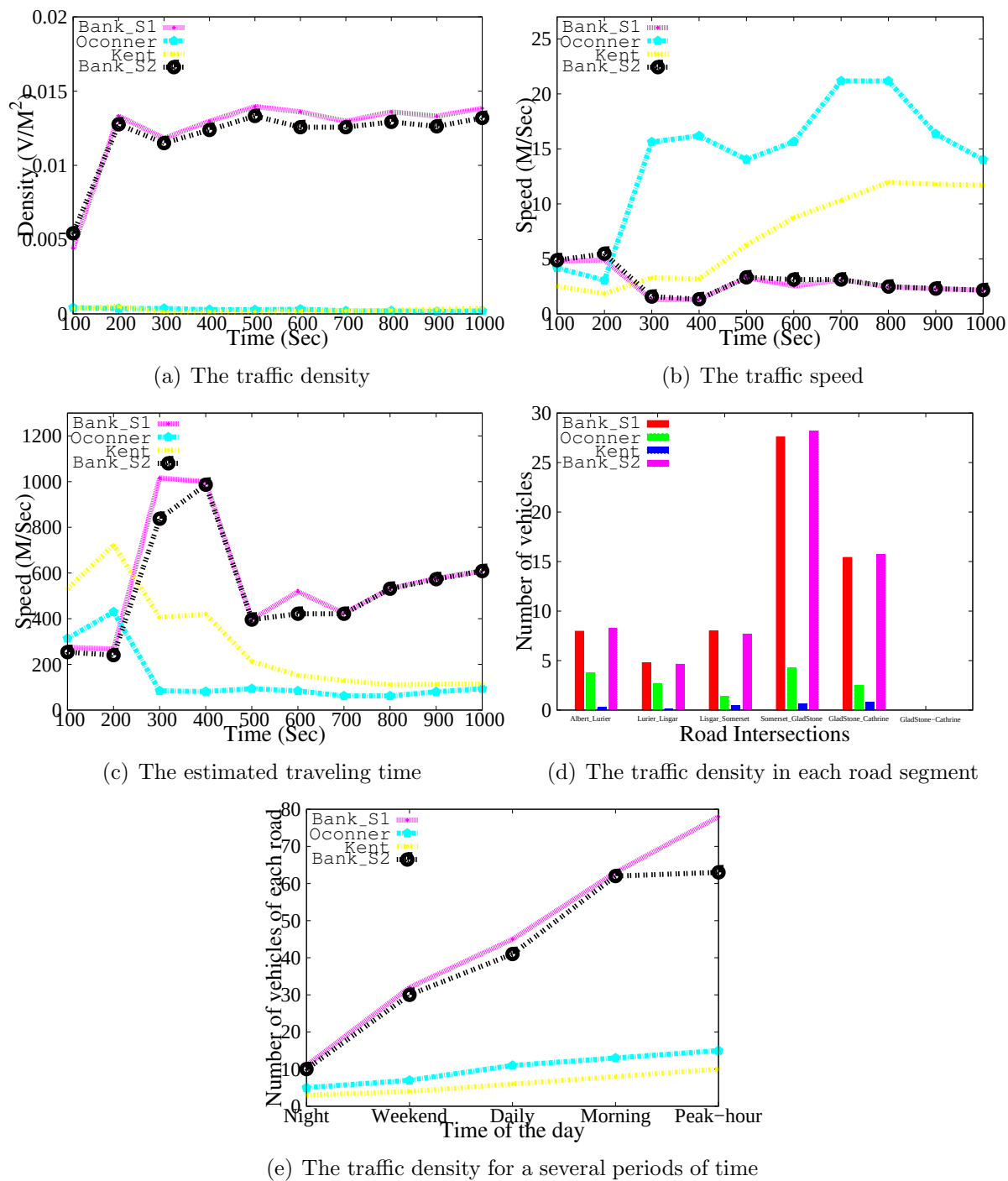


Figure 3.8: Traffic evaluation of downtown Ottawa.

From Figure 3.8(b), we can see that the traffic speed of Bank Street is 80% slower than the traffic speed of O'Connor Street and 60% slower than the traffic speed of Kent Street in peak hour scenarios. Moreover, Figure 3.8(c) illustrates that the required estimated time needed to travel Bank Street from north to south is 80% longer than the required estimated time needed to travel O'Connor Street in the same direction. In addition, traveling along Bank Street from the south to the north direction takes 60% more time than traveling along Kent Street. However, in the first two intervals of time, when a small number of vehicles passed over the area of interest, a small number of vehicles were simultaneously traveling along Bank Street; thus, the traffic speed on Bank Street was faster than the traffic speed on O'Connor Street and Kent Street. Furthermore, the estimated travel time on Bank Street was shorter than the estimated travel time on O'Connor Street and on Kent Street during these two intervals of time.

In Figure 3.8(d), we also investigated the traffic distribution among a set of popular road intersections (i.e., Albert-Laurier, Laurier-Lisgar, Lisgar-Somerset, Somerset-Gladstone, and Gladstone-Catherine) for the peak hour scenarios. From Figure 3.8(d), we can see that the number of vehicles traveling on Somerset-Gladstone and Gladstone-Catherine, out of all the investigated streets, is greater than the number of vehicles traveling over other sections of these streets. Moreover, the average number of vehicles on both sides of Bank Street at each road section are different than each other.

Finally, in Figure 3.8(e), we study the impact of the number of traveling vehicles on traffic distribution by investigating these streets for different traffic scenarios. We generated different scenarios with a different number of traveling vehicles: 200 (night hours), 400 (weekend hours), 600 (daily hours), 800 (morning hours) and 1000 (peak hours). As the Figure indicates, by increasing the number of vehicles, more vehicles travel through Bank Street, either from the south to the north section or from the north to the south section. The traffic density of Bank Street becomes relatively higher than the traffic density of O'Connor and Kent Streets, in conjunction with the increase in the number of traveling vehicles for all scenarios.

3.7 Summary

In this chapter, we proposed ECODE protocol, which aims to evaluate traffic characteristics and to detect highly congested road segments in downtown area scenarios. Three different parameters of traffic characteristics have been used to detect highly congested road segments and to enhance the reliability of ECODE. The performance of ECODE have been extensively evaluated for several scenarios.

Moreover, different execution schemes of ECODE have been introduced and discussed, to execute the protocol repeatedly: proactive, reactive and hybrid. These schemes aim to investigate the effects of the implemented techniques in the congestion control function of VANETs to study its impact on the performance of ECODE. The performance of the hybrid scheme achieves the best results when compared to the proactive and reactive schemes.

Finally, this chapter comments on the use case study scenario, which intends to investigate the traffic distribution in the downtown area of Ottawa. This study demonstrates that Bank Street is more congested by traffic than O’Conner Street and Kent Street. Moreover, more vehicles travel through Bank Street during any investigated time and/or level of traffic density.

Chapter 4

An Intelligent Path Recommendation Protocol (ICOD)

4.1 Introduction

This chapter introduces an intelligent, real-time and distributed path recommendation protocol. We refer to the introduced protocol as an Intelligent path reCOmmenDation protocol (ICOD). This protocol aims to find the best path towards each destination in any urban grid-based layout area. The path is reactively configured at each road intersection, based on the real-time traffic characteristics of the Surrounding Road Segments (SRSs) of that intersection; that is, any road segment that starts or ends at the road intersection. In order to handle the centralized behavior problems (i.e., bottleneck, single point of failure, etc.) which appeared in the previous path recommendation protocols, ICOD has constructed the path towards each destination in a distributed manner.

Three variants of ICOD are introduced to enable flexible selection of the best path towards each destination, according to the driver's concerns and priorities: congestion avoidance, Eco-path and context-aware path recommendation. The congestion avoidance variant aims to reduce traffic congestion by recommending each vehicle take the least congested path towards its destination (i.e., fastest path). The Eco-path variant recommends the best and most economical path in terms of fuel consumption and gas emission parameters of each candidate path. The context-aware variant considers the context of each traveled road segment while selecting the desired path. Thus, the located services and the road conditions are considered. We report on the performance of these variants of ICOD and compare them with other path recommendation protocols in this

field. Finally, a fault tolerant variant of ICOD has been designed and presented, and we report on its performance at the end of this chapter.

4.2 The Architecture of ICOD

In this section, we present the general structure of ICOD; before we introduce the details of the ICOD architecture, we present the assumptions that are considered in this protocol.

4.2.1 Assumptions

With the objective of designing a fully distributed and real-time path recommendation protocol, we consider the following assumptions:

- In order to focus on the study of distributed behavior of the ICOD protocol, we consider an RSU at each road intersection all over the area of interest (i.e., Grid-based layout). Each RSU manages and controls the traffic within its Surrounding Road Segments (SRSs). This assumption can be released if each RSU handles the information of a larger group of surrounding road intersections, or if there is a vehicle close to each road intersection that is capable of handling the RSU tasks. In addition, the required RSUs are simple where no memory or high processing capability are needed.
- We assume each vehicle is equipped with an IEEE 802.11p transceiver, GPS receiver and digital maps. Many vehicles are already equipped with such devices and equipment these days.
- We assume that each destination has an associated RSU (i.e., the nearest RSU to that destination) that broadcasts an advertisement message illustrating the position and characteristics of that destination.
- Finally, a minimum traffic density is assumed to exist on each road segment. The traveling vehicles help deliver messages between RSUs at road intersections. In the case where there is no connectivity over a certain road segment, the road segment witnesses a low traffic intake, and vehicles can then travel over such a road segment at the maximum allowed speed.

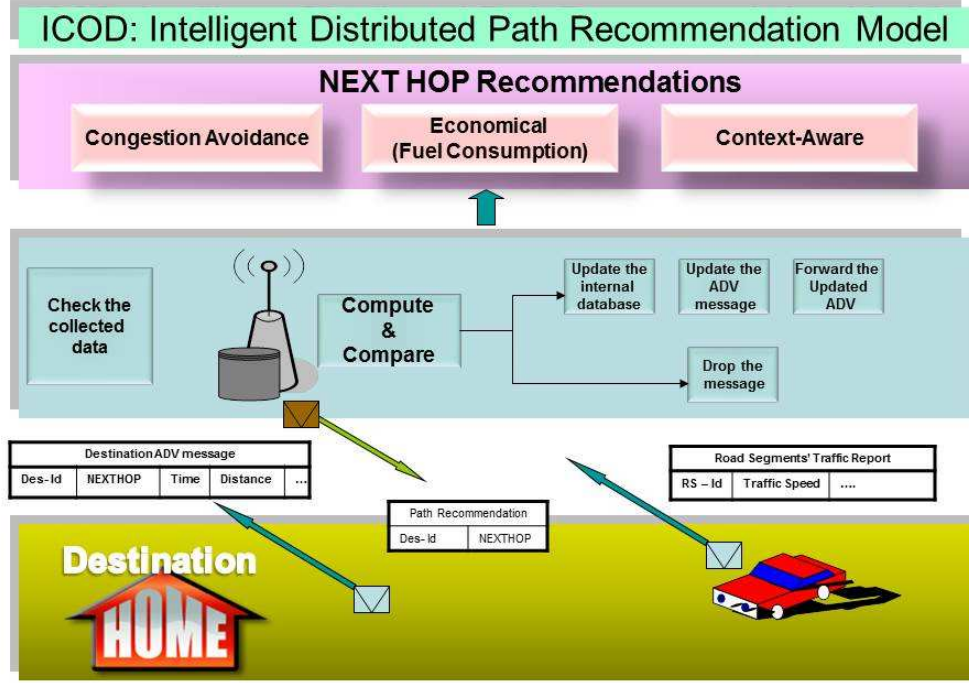


Figure 4.1: Architecture of ICOD.

4.2.2 ICOD Architecture

Figure 4.1 illustrates the main components of ICOD. At the bottom level of Figure 4.1, the mission of traveling vehicles is to gather and to deliver road segment traffic data to the responsible RSU at the end of each road segment. At the same time, destinations also broadcast advertisement messages that illustrate their locations and the initial travel expenses that are required for arrival.

At the middle level of Figure 4.1, RSUs located at road intersections communicate cooperatively using traveling vehicles, to construct and to recommend the best path available towards each targeted destination. These RSUs obtain and use basic information pertaining to the traffic situation of surrounding road segments and destination locations. The local database of each RSU includes two tables for surrounding road segment characteristics and the destination specifications. The *Surrounding Road Segment* table gathers the traffic situation of the directly surrounding road segments (i.e., the road segment that starts or ends at this road intersection), in a respective RSU. This table is filled and updated by current traveling vehicles at each road segment; this is done to

guarantee real-time and up-to-date traffic data. Six main fields in the Surrounding Road Segment table are presented with the characteristics of each road segment RS_i . These fields include: (1) the identifier of the road segment (RS_{id}), (2) travel speed of the road segment (TS), (3) travel distance of the road segment (TD), (4) travel time of the road segment (TT), (5) services over the road segment, and (6) the conditions of the road segment.

On the other hand, the *Destination* table stores the data with the information of the best path towards each destination, as evaluated according to the parameters considered (i.e., selected variant). Four main fields are presented in the destination table including: (1) the destination identity (D_i), (2) the best next hop towards the destination (NEXT-RSU), (3) best obtained travel distance (BTD), and (4) best obtained travel time (BTT). This table is filled and updated by broadcasting and forwarding the advertisement message (ADV_{D_i}), which is related to each destination (D_i). More details concerning the advertisement messages of the destinations and the communication process between the scattered RSUs are explained in Section 4.3.2 and Section 4.3.3 respectively.

In Figure 4.1, we can also see the different variants of ICOD, introduced at the top level of ICOD architecture. These variants consider different parameters in order to recommend the desired path according to user concerns and priorities (e.g., travel time, fuel consumption, road segment context, etc). In order to enhance flexibility and to guarantee a more complete protocol, a fitness function has been added that balances between the different considered parameters. The fitness function details followed by the different variants proposed of ICOD are presented in Sections 4.4 and 4.5 respectively.

4.3 Phases of ICOD

The proposed ICOD protocol has been designed, in a distributed manner, to recommend the best path leading towards each destination. In this section, we explain the details and the phases of this protocol. The ICOD protocol is composed of a set of sequential phases that are executed frequently and/or reactively upon respective traffic situations and in the context of road condition changes. This protocol produces real-time path recommendations that react to current traffic, as well as to traffic changes in road condition situations. The details of ICOD phases are presented in the rest of this section.

4.3.1 Road Segment Traffic Evaluation

During this phase, traveling vehicles on each road segment (RS_i) cooperatively evaluate the traffic and gather context situations of RS_i . Vehicles are also responsible for the reporting of real-time traffic evaluation to the responsible RSU at the end of the respective RS_i . The traffic situation is evaluated by considering all vehicles in the area of RS_i . Several traffic evaluation mechanisms and protocols [15], [53] can be used to gather the information of traveling vehicles on each road segment, to then evaluate the dynamic traffic parameters (i.e., traffic density (TD), traffic speed (TS), estimated traveling time (TT), traffic congestion level, etc). We have used our ECODE protocol for this step; the details of ECODE are presented in Chapter 3, where traveling vehicles cooperatively evaluate the traffic characteristics of each road segment direction. In ECODE, traveling vehicles also reliably deliver the traffic evaluation to RSUs located at the end of each road segment.

4.3.2 Destination Advertisement Message Broadcasting

Each RSU associated with a certain destination (D_i) broadcasts an advertisement message (ADV_{D_i}). This message declares the location of the respective destination, and it initiates the incurring of expenses needed to reach the destination in terms of travel time and distance. The advertisement message (ADV_{D_i}) is considered to be the vital component of ICOD, since it initiates the path towards each destination. Upon receiving the ADV message from each RSU, the receiver RSU updates the message using the information and experience it has obtained regarding the surrounding road segments, before forwarding it. In other words, the message delivers the best experience that each RSU has obtained about reaching the located destination to all of its direct neighbor RSUs (i.e., RSUs that are located at the end of its Surrounding Road Segments).

The destination ADV message contains four main fields: destination identifier, next-hop identifier, travel time and travel distance. The destination identifier field (DES_ID) is permanently set at each ADV_{D_i} message to specify which destination is represented by this message. On the other hand, the next hop identifier ($NEXTHOP_ID$) is changed respectively by the identifier of the most recent RSU that has forwarded this message. When the RSU associated with the destination D_i sends the initial ADV_{D_i} message, the DES_ID and $NEXTHOP_ID$ fields are set by the identifier of that RSU associated with the destination D_i . The fields of travel time (T_time) and travel distance ($T_distance$) are updated at each receiver RSU; this is based on the surrounding road

segment situations and the gathered data of that respective RSU.

4.3.3 Path Construction

The path towards each destination is constructed from the destination position to each road intersection, covering the entire area of interest. Algorithm 4 illustrates the internal procedure undergone at each RSU in order to construct the best path leading towards each destination.

Algorithm 4: Path Construction Algorithm

Data: T_i : Destination Table in RSU_i , $SRSs$: Surrounding Road Segments.

```

1  Whenever  $RSU_i$  receives an  $ADV_{D_k}$  from  $RSU_j$  ;
2  UPDATE  $ADV_{D_k}$  message:
3  {
4   $T\_time = T\_time + RS_{ij}.TT$ ;
5   $T\_distance = T\_distance + RS_{ij}.l$ ;
6  }
7  if  $T_i$  contains any record related to  $D_k$  then
8      COMPUTE:
9      {
10      $T\_COST(ADV) = w_1 \times T\_time + w_2 \times T\_distance$ ;
11      $T\_COST(T_i) = w_1 \times BTT + w_2 \times BTD$ ;
12     }
13     if  $T\_COST(ADV) < T\_COST(T_i)$  then
14         UPDATE  $T_i$ ; /* Update the table if the ADV message recommend a better path */
15         {
16          $T_i.NEXTRSU = j$ ;
17          $T_i.BTT = T\_time$ ;
18          $T_i.BTD = T\_distance$ ;
19         }
20     else
21         DROP the Message; /* No need to update the table or to forward the message */
22         exit;
23     end
24 else
25     ADD a record to  $T_i$  related to  $D_k$ ;
26 end
27 UPDATE  $ADV_{D_k}$ : NEXTHOP-ID = i ;
28 FORWARD updated message;
29 if  $RSU_i$  detects any change in the congestion level of any  $SRSs$  then
30      $RSU_i$  sends a set of  $ADV$  messages related to all  $D_k$  in its  $T_i$  to its neighbor RSUs ;
31 end

```

Whenever an RSU_i receives an ADV_{D_k} message from RSU_j , it first updates the $T_distance$ and T_time fields of the received ADV_{D_k} message; it does this by adding the length of RS_{ij} ($RS_{ij}.l$) and the following respective estimated travel time ($RS_{ij}.TT$): see Lines 2-6 in Algorithm 4. Notice that RS_{ij} is the road segment that links RSU_i and RSU_j . RSU_i then checks its *Destination* table (T_i) to see if it has a record related

to the destination D_k . In the case that T_i has no record related to the destination D_k , the RSU_i adds a new record to its T_i using the updated fields of ADV_{D_k} to fill the new record. Then, the receiver RSU_i updates the $NEXTHOP_ID$ field in the ADV_{D_k} with its own identifier, and forwards the final updated ADV_{D_k} message to all of its direct neighbor RSUs (see Lines 7-28). On the other hand, if T_i has a certain record related to D_k , this means that the arrival of the newly received message suggests the existence of a new path running towards the destination D_k . Here, RSU_i chooses between the two candidate paths, based on the benefits that each path has to offer in terms of travel distance and/or time. The overall cost of each path is computed in (Lines 10 and 11). More details concerning the overall cost of the different variants of ICOD are presented in Section 4.4, where the fitness function is also presented.

If the overall travel cost score of the path recommended by table T_i ($T_COST(T_i)$) is less than or equal to the overall travel cost score of the path recommended by the ADV_{D_k} message ($T_COST(ADV)$), the RSU_i should drop the received ADV_{D_k} message without updating its *Destination* table or forwarding that message because no better path is obtained. Otherwise, when the $T_COST(ADV)$ score is lower than $T_COST(T_i)$, the receiver RSU_i needs to update the D_k record, at its table, according to the received and updated ADV_{D_k} . After this step, the RSU_i updates the ADV_{D_k} message by assigning its ID to the $NEXTHOP_ID$ field and by forwarding the updated message to its direct neighbor RSUs. Successions of receiving, updating and forwarding ADV messages related to different destinations help all RSUs to obtain the best path towards all destinations.

Furthermore, in its aim to obtain intelligent and real-time performance, whenever an RSU_i detects a change in the congestion level of any of its surrounding road segments, it initiates a set of ADV messages for each record in its *Destination* table (see Lines 29-31). RSU_i sends these messages to its neighbor RSUs; as previously explained, these RSUs adjust their *Destination* tables and the constructed path according to received messages.

4.3.4 Path Recommendation

After obtaining the best path leading towards each destination, each RSU broadcasts a Recommendation Report (*RecomReport*). The *RecomReport* contains a list of all destinations detailing the next best road segment to follow, at the road intersection where the RSU is installed. Each vehicle, on the surrounding road segments, obtains the best direction to take towards its destination from this *RecomReport* message.

In order to deliver the *RecomReport* in a scalable and reliable manner, traveling

Algorithm 5: Greedy Forwarding Algorithm.

```

1 if  $V_i$  receives RecomReport message then
2   if  $V_i.recom_t < RecomReport_t$  then
3      $V_i$  adjusts its path towards its destination;
4      $V_i$  computes  $T$  (See Equations 4.1, 4.2);
5     if  $T == 0$  &&  $V_i$  did not receive RecomReport then
6        $V_i$  re-broadcast RecomReport;
7     end
8   else
9     drop the RecomReport message;
10  end
11 end

```

vehicles in each road segment must use a greedy routing protocol to forward the received report. Algorithm 5 illustrates the process undergone by each vehicle to forward the *RecomReport* throughout the traversed road segment. Instead of directly forwarding the *RecomReport* messages, vehicle (V_i) should verify whether it is the best vehicle to forward the message, or whether there is a more suitable vehicle (FV_i). First, each V_i checks the distance (X) to the sender. If V_i is the furthest vehicle away that is expected to receive this message, it should forward the message immediately. Otherwise, it should wait for a certain period of time (T) to receive the same forwarded message from another vehicle. This waiting time T depends on the position, speed (S) and direction of V_i .

Equations 4.1 and 4.2 respectively show how to set T_1 and T_2 at each V_i on both directions of any RS_i . For any V_i moving away from the RSU sender, the suitable time to wait before another attempt to send the *RecomReport* is given by T_1 in Equation 4.1. If the vehicle is moving towards the sender, the time T_2 it should wait is given by Equation 4.2. In the case that V_i does not receive any forwarding *RecomReport* message within T_1 time, V_i should forward the message itself (see Lines 5-7 in Algorithm 5).

$$T_1 = \frac{D - X}{S}, \quad (4.1)$$

$$T_2 = \frac{X - D/2}{S}, \quad (4.2)$$

where D is located at the furthest distance from the sender at which the message can be detected.

4.4 Traffic Parameters Considered in ICOD Variants

Travel distance, travel time and travel cost (i.e., fuel consumption and gas emissions) are the basic parameters considered in our protocol. In the following paragraphs, we investigate how to compute each of these parameters. We then introduce a fitness function that combines and prioritizes these parameters in order for them to be used in our protocol. Applying this fitness function, ICOD measures the cost of each elected path and then chooses the optimal path.

4.4.1 Traffic Parameters

Travel Distance: represents the distance length of each path, computed by adding the lengths of all road segments ($RS_i.l$) in the respective path. Equation 4.3 illustrates how to compute the travel distance (TD_{D_k}) of a certain path leading towards D_k at each RSU:

$$TD_{D_k} = \sum_{i=1}^n RS_i.l, \quad (4.3)$$

where n is the number of road segments in the respective path.

Travel Time: represents the estimated travel time of each path. This is based on the current road segment traffic speed ($RS_i.TS$) and the length of each road segment ($RS_i.l$). $RS_i.TT$, the estimated travel time of each road segment, is computed using Equation 4.4:

$$RS_i.TT = \frac{RS_i.l}{RS_i.TS}. \quad (4.4)$$

The travel time (TT) for each path towards the destination (D_k) is computed by using Equation 4.5:

$$TT_{D_k} = \sum_{i=0}^n RS_i.TT. \quad (4.5)$$

Travel Cost: represents the fuel consumption and the gas emissions of each path. From [62], the fuel consumption (ΔF) during a small interval of time (ΔT) is estimated for each traffic scenario based on: (1) traffic acceleration (a), (2) traffic speed (v), (3) tractive force (R_T) and (4) vehicle mass (M_v). Equation 4.6 is used to estimate the fuel consumption of each traveling vehicle [62],

$$\Delta F = [\alpha + \beta_1 R_T v + (\beta_2 M_v a^2 v / 1000)] \Delta T, \quad (4.6)$$

where α , β_1 , and β_2 are constant values associated with individual vehicles.

Based on the CO_2 rate in milligrams per milliliter of fuel (f_{CO_2}) the Carbon Dioxide (CO_2) percentage is estimated directly from the fuel consumption by Equation 4.7 [62]:

$$\Delta E_{CO_2} = f_{CO_2} \Delta F \quad (4.7)$$

For a better approximation of ΔF during any interval of time (ΔT), we add ΔF over small intervals to get:

$$\begin{aligned} \Delta F &= \int_0^{\Delta T} (\alpha + \beta_1 R_T v + (\beta_2 M_v a^2 v / 1000)) dt \\ &= \alpha \Delta T + \beta_1 R_T \Delta D + (\beta_2 M_v / 1000) \int_0^{\Delta T} a^2 v dt, \end{aligned}$$

where ΔD is the travel distance during ΔT .

For the last integral, the mean value theorem can be invoked to get

$$\Delta F = \alpha \Delta T + \beta_1 R_T \Delta D + (\beta_2 M_v a_0^2 v_0 / 1000) \Delta T, \quad (4.8)$$

where a_0 and v_0 are the acceleration and the speed of the traffic at some time (T_0) in $[0, \Delta T]$.

From Equation 4.8, we can see that the travel cost of any path can be estimated based primarily on the travel distance (ΔD) and the travel time (ΔT). For any path construction, any drastic increase in the travel time and/or distance will increase the fuel consumption and the release of gas emissions into the atmosphere.

4.4.2 Path Fitness Function

In order to combine the previously investigated parameters, we introduce a simple fitness function in Equation 4.9. The introduced fitness function weights and balances the travel time (TT_{D_k}) and the travel distance (TD_{D_k}) of each path leading towards any destination. It provides an alternative consideration of the travel time and distance, while computing the overall travel cost score ($T_COST(p_{D_k})$) of each possible path (p_{D_k}) leading towards any destination D_k :

$$T_COST(p_{D_k}) = w_1 TT_{D_k} + w_2 TD_{D_k}, \quad (4.9)$$

where w_1 , and w_2 are weighting factors that can obtain only the value of 0 or 1.

Using 0 or 1 values for w_1 and w_2 allows the RSU to consider only the time or only the distance of the selected path.

4.5 ICOD Variants

The different variants of ICOD are introduced in this section.

4.5.1 Congestion Avoidance Path Recommendation

This variant aims to investigate the travel time of each candidate path in order to indicate the traffic congestion level. The expected travel time of each road segment changes from time to time, based on the traffic speed and traffic density over the road segment. In a road segment with greater traffic congestion, the traffic speed will slow and the expected travel time on that road segment will increase. This variant of ICOD aims to find the fastest alternative path leading towards each destination, where the fastest path represents the least congested path. In Equation 4.9, while computing the overall cost of alternative candidate paths, $w1$ is set to 1 and $w2$ is set to 0. In this variant, only the travel time parameter is considered for each path. The traffic situation changes from time to time at each road segment. The real-time communications between the various RSUs installed serve to adjust the path towards each destination, which is done according to the real-time traffic situation of all road segments.

4.5.2 Eco-Path Recommendation

The large increase in the travel time and/or distance leads to greater fuel consumption and higher gas emissions. Here, we introduce a new variant that aims to balance between the travel time and the travel distance of each path. This variant is expected to produce a more efficient path in terms of fuel consumption and gas emissions.

Each RSU sets the weight factors, $w1$ and $w2$, based on the benefit ratio that the path will obtain in terms of travel distance and/or time needed to reach each destination (D_k). Whenever an RSU receives an ADV_{D_k} message, it first verifies whether that message has obtained any benefits in terms of the travel time and/or the travel distance; it does this after computing the benefit ratio of the travel distance (Dis_{ratio}) and the travel time ($Time_{ratio}$). The benefit ratios are computed by Equations 4.10 and 4.11 accordingly:

$$Time_{ratio} = ADV.TT_k / T(k).BTT, \quad (4.10)$$

where $ADV.TT_k$ is the travel time in the advertisement message (ADV_{D_k}) and $T(k).BTT$ is the best travel time that has been obtained at the destination table in the RSU.

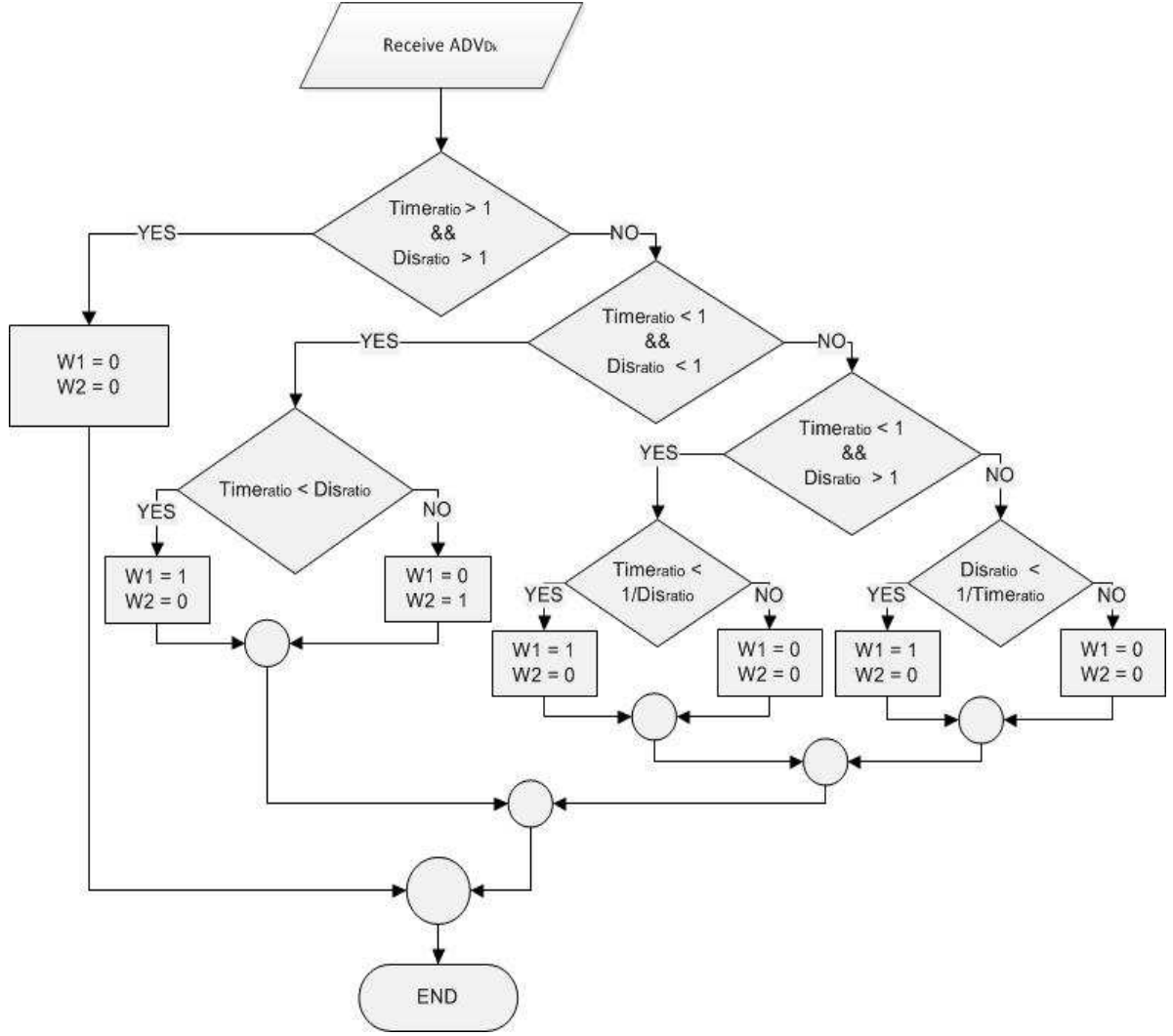


Figure 4.2: Weighting factor initialization algorithm.

$$Dis_{ratio} = ADV.TD_k / T(k).BTD, \quad (4.11)$$

where $ADV.TD_k$ is the travel distance in the ADV_{D_k} and $T(k).BTD$ is the best travel distance that has been obtained at the destination table in the RSU.

Figure 4.2 illustrates four different initialization cases for the weighting factors based on the benefit ratio comparison. In the first case, where both values of $Time_{ratio}$ and Dis_{ratio} are greater than 1, the ADV_{D_k} message does not achieve any benefits in terms of travel time or distance. In other words, the most recently obtained path at this RSU is better than the path recommended by the received ADV_{D_k} . Therefore, $w1$ and $w2$

are both set to 0 in this case, which will cause a drop down for the message without any updating or forwarding. Second, if $Time_{ratio}$ and Dis_{ratio} are both less than 1, then the path delivered by the ADV_{D_k} message achieves a better performance in terms of the travel time and distance. The updated ADV message should be forwarded, carrying a new candidate path to the surrounding RSUs. Third, in the case where $Time_{ratio}$ is less than 1 and Dis_{ratio} is more than 1, the new path will obtain benefits in terms of travel time; however, vehicles will need to travel a longer distance. The RSU compares the $Time_{ratio}$ and the reciprocal of the Dis_{ratio} ; the reciprocal value represents the loss ratio (i.e., overhead). For the updated ADV message to be forwarded, w_1 must be set to 1 and w_2 must be set to 0. This can only take place if the obtained benefit, in terms of travel time is greater than the overhead of the travel distance. Otherwise, both w_1 and w_2 are set to 0. Finally, if the Dis_{ratio} is less than 1 and the $Time_{ratio}$ is more than 1, the weighting factors are also set based on the comparison between Dis_{ratio} and the reciprocal of $Time_{ratio}$. As shown in Figure 4.2, for the last three cases, the weighting factors are set according to the larger obtained benefit of travel time and distance.

4.5.3 Context-Aware Path Recommendation

In real scenarios, some road segments are located at commonly targeted spots such as hospitals, schools or gas stations. These road segments usually suffer from higher traffic congestion and longer travel time, while needing to be easily maneuvered by drivers. For example, road segments around a hospital may need to be less congested in order to allow emergency cases to reach the hospital safely and on time. In addition, the conditions of road segments over a certain area vary; some road segments are in good conditions, while others have many potholes and bad conditions. Some drivers prefer to travel extra time and/or distance rather than traveling over road segments in bad conditions.

Here, we introduce a flexible variant that considers special services as well as the road conditions all over the area of interest. This variant can be included in any of the previously explained two variants (i.e., congestion avoidance and Eco-path). It mainly aims to recommend an alternative path leading towards each destination for the purpose of reducing the percentage of vehicles passing through road segments located at special services.

A certain cost ($SerCo$) is assigned to each service in order to rank the priority of special road segment avoidance. The more important the service is, the larger the ranked cost should be. At the same time, road segment conditions are reported with consider-

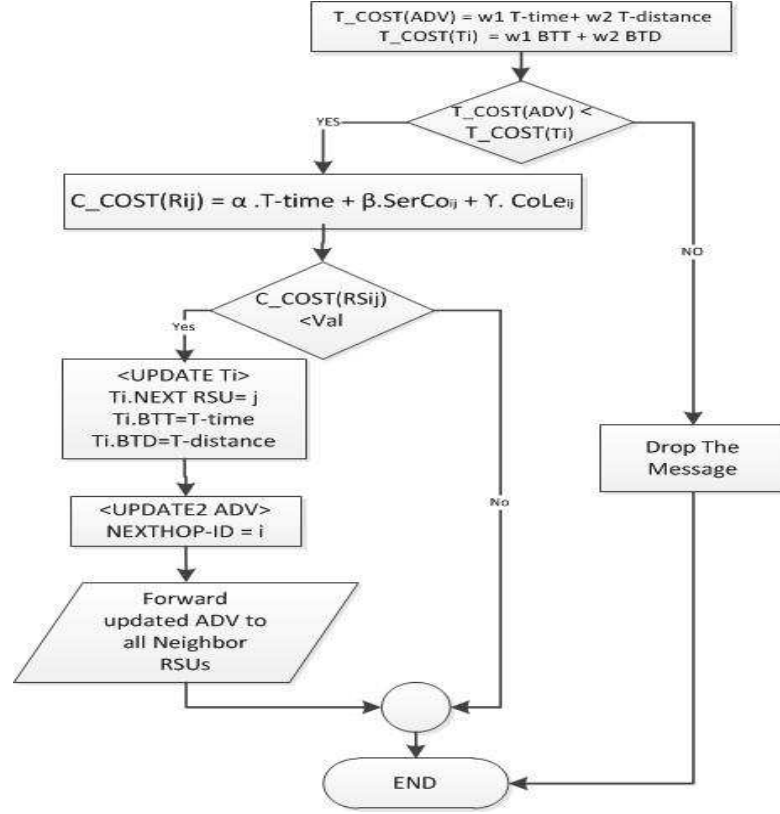


Figure 4.3: Context-aware variant adaptation.

ation for the existence of potholes, obstacles, and weather conditions, and other similar factors. Regarding road conditions, a certain cost (*CoLe*) is assigned to each level; with greater severity in the conditions of the road segment comes a larger assigned cost. The costs brought about by *SerCo* and *CoLe* are assigned according to the expected travel time of the different traffic congestion levels. For example, for very important special services or for very bad road segment conditions, a high cost is assigned that is equal to the average travel time of highly congested road segments. On the other hand, for less important services, such as gas stations, or in the case of moderate conditions experienced in road segments, the assigned rank of cost is the same as the average travel time for low-level congested road segments. The *SerCo* and *CoLe* values can be changed automatically based on the time of the day, the day of the week, etc. For instance, the *SerCo* value of a road segment where a school service is located should be high during the peak hours where all students are traveling to or from the school. However, this *SerCo* value should be low during night hours or weekends.

The context-aware variant adapts the path recommendation in Algorithm 4 to con-

struct paths that consider the context of each road segment. As shown in Figure 4.3, if the *Destination* table of RSU_i contains a record related to the destination D_k , RSU_i compares the traveling cost of ADV_{D_k} ($T_COST(ADV_{D_k})$), to the traveling cost of *Destination* table ($T_COST(T_{D_k})$) in the RSU. Only if $T_COST(ADV_{D_k})$ is less than $T_COST(T_{D_k})$, the RSU_i will use Equation 4.12 to compute the context cost ($C_COST(RS_{ij})$); this context covers the event in which road segment RS_{ij} (i.e., the road segment that connects between the receiver RSU_i and the sender RSU_j) is traversed.

In order to guarantee an adequate level of congestion-free experience for special road segments, we assigned the constant value 0.5, for the weight of the special road segment cost γ_2 .

$$C_COST(RS_{ij}) = \gamma_1 \cdot T - time + \gamma_2 \cdot SerCo_{ij} + \gamma_3 \cdot CoLe_{ij}, \quad (4.12)$$

where $SerCo_{ij}$ is the assigned service cost of RS_{ij} , $CoLe_{ij}$ is the condition level of RS_{ij} and $\gamma_1 + \gamma_2 + \gamma_3 = 1$.

In this variant, it is recommended that vehicles travel through each road segment (RS_i), if the context cost ($C_COST(RS_i)$) computed by Equation 4.12 does not exceed a certain threshold (Val). The Val value is also set based on the estimated travel time of each congestion level. If we want to guarantee a higher congestion-free level, we must assign the average estimated travel time of the low congestion level to the Val parameter. On the other hand, if we assign the average travel time of the moderate congestion level to the Val variable, a higher traffic congestion level is expected to occur over special road segments, and more vehicles will be expected to travel over road segments with bad conditions.

Finally, as shown in Figure 4.3, if $C_COST(RS_{ij})$ is less than the predefined value (Val), the RSU_i updates its *Destination* table, in accordance with the ADV_{D_k} message, and updates the $Node_{prei}$ field of ADV_{D_k} by using the RSU_i identifier, forwarding the updated ADV_{D_k} message towards its direct neighbor RSUs, with the exception of RSU_j (i.e., the sender RSU).

4.6 Fault Tolerant Distributed Path Recommendation Protocol (TD-PR)

In this section we briefly discuss the potential failures of each phase of ICOD. Then, we design a fault tolerant distributed path recommendation variant (TD-PR). The details of the TD-PR variant are also presented in this section.

4.6.1 Fault Tolerant Mechanisms

A faulty node and/or faulty link over the connecting network can affect the functionality, robustness and reliability of ICOD. These faults, over the VANETs network, can be either at the located nodes or at the available links between these nodes. Faults in VANETs can be classified as permanent, intermittent or transient [119]. The permanent faults will remain on the network unless they are repaired and/or removed by external administrator. Intermittent faults are unpredictable, and difficult to diagnose. Finally, transient faults will eventually disappear without any need for apparent intervention. In general, three aspects are associated to the fault tolerance including fault models, fault detection and diagnosis and fault resiliency [119].

Fault detection is considered the first step to fault correction, and it is important for resiliency. The comparison approach [111], the MM model [117] and the broadcast comparison model [118] are some of the most popular approaches that have been used to diagnose the fault scenarios. Considering that fault resiliency tolerates fault existence, detection aims to locate an alternate link or node to replace the faulty one [119]. Moreover, redundant data over the road network enhances the reliability of proposed protocols in case of the existence of faults [120].

4.6.2 Potential Failure Points in ICOD

The potential failure points at each phase of ICOD are investigated in this section.

- **Possible Failures During the Traffic Evaluation Phase**

1. Some vehicles fail to advertise basic traffic data.
2. Some receiver vehicles miss important advertisement messages due to failure links over the VANETs network.

3. Selected relay vehicles in the reported areas fail to report the traffic characteristics of the configured cluster.
4. RSUs located over the road network fail to receive the evaluation report of surrounding road segments.

- **Possible Failure During Destination Advertisement**

1. Targeted destinations fail to register at the closest RSU (i.e., associated RSU).
2. The associated RSU fails to broadcast the initial advertisement message.

- **Possible Failure During Path Construction**

The potential failures in this phase can be in the existing nodes or in the available link between RSUs and traveling vehicles:

1. RSUs fail to receive, process and/or forward the advertisement messages.
2. The communication link between two neighboring RSUs can be damaged while forwarding the message.
3. Traveling vehicles over long road segments fail to deliver the advertisement message between RSUs located over the road network.

- **Possible Failure During Path Recommendation**

1. Located RSUs fail to broadcast the recommendation report.
2. Some vehicles fail to receive the recommendation report due to link failure.
3. Traveling vehicles in between located RSUs fail to forward the recommendation report all over the road segment.

4.6.3 The Fault Tolerant Distributed Path Recommendations (TD-PR)

In this section, we introduce a fault tolerant distributed path recommendation (TD-PR) variant of ICODE. TD-PR detects the fault scenarios and aims to tolerate these faults efficiently at each phase of ICODE. Unlike previously proposed fault tolerant protocols, we do not assume that there is no faulty node or no faulty link during the setup phase.

In general, TD-PR handles the potential failures that are discussed and presented in Section 4.6.2. We investigate the fault tolerance solutions of each phase of ICOD in the following sections.

Fault Tolerant Traffic Evaluation

In the case that a certain vehicle has a problem with advertising its basic data over the road segment due to communication failure, camcorder or radar equipments can be used to physically detect the traffic characteristics of each vehicle. Using these equipments as an alternative solution to detect traveling vehicles over each road segment enhances the reliability and accuracy of the traffic evaluation phase.

In occasional scenarios, faulty links prevent some vehicles from gathering the basic traffic data of other surrounding vehicles. In such situations, these vehicles do not know about one or more of their neighboring vehicles. This problem can become more significant in a scenario where the relay vehicle that reported the traffic evaluation of the cluster does not know about some vehicles in such a cluster. In this scenario, inaccurate traffic evaluation can be generated for the road segment, due to inaccurate traffic evaluation reported of each cluster. In order to resolve this problem, the relay vehicle, which reports the traffic characteristics of each cluster, should validate the traffic data gathered about the cluster with some other vehicles inside such a cluster.

The relay vehicle sends a validation message to the vehicle located in the closest position to the center of the cluster. This validation message contains the gathered traffic data about the cluster in the local database of the reported vehicle. The receiver vehicle compares the contents of the validation message to local traffic data gathered about the respective cluster. In the case that the receiver vehicle has more data than the validation message, it sends an update message to the reported vehicle. The update message contains all traffic data missed by the reported vehicle. Otherwise, if the receiver vehicle does not have any extra data in its database, it sends an acknowledgement message to inform the relay vehicle that the traffic data it gathers is correct. When the relay vehicle receives the update/acknowledgement message, it updates the report message according to the received data, and broadcasts the report of such a cluster towards vehicles in the neighboring clusters.

Fault Tolerant Destination Advertisement

In order to make the destination advertisement phase more tolerate, each destination should register to more than one RSU over the road network. Each destination should choose the closest RSU as an associated RSU. The associated RSU of each destination should initiate the path construction process on behalf of the registered destinations, as illustrated in Section 4.3.2.

In the case where the associated RSU has failed to initiate the advertisement message on behalf of its registered destination/destinations, another RSU possessing a record of these destinations can initiate the message. This alternative RSU should wait for a predetermined period of time to receive the advertisement message from the associated RSU of the registered destination. In the case that this RSU did not receive any forwarding advertisement message related to the destination registered, it sends a request message to its neighboring RSUs. The request message aims to verify whether any of the neighboring RSUs have received an advertisement message related to the destination registered. If any neighboring RSUs have received a message related to this destination, that means the connecting link between the associated RSU and the alternative RSU has failed. These surrounding RSUs send back the details about the destination towards the requested RSU. Otherwise, if none of the neighboring RSUs have received the message, this indicates that the associated RSU to this destination has failed to broadcast the advertisement message. In this case, the alternative RSU should initiate an advertisement message on behalf of this destination instead of the associated RSU.

Fault Tolerant Path Construction

ICOD assumes that a minimum traffic density should be available at each road segment in any road network. This assumption is essential for long road segments, as traveling vehicles deliver messages between the RSUs located at the ends of the respective road segment, using a multi-hop communication technique. In order to release this assumption and to enhance the reliability of this phase, more infrastructure (i.e., RSUs) is required over the road network. This infrastructure is installed to deliver the advertisement messages between the located RSUs at the ends of long road segments, in the case that traveling vehicles failed to deliver these messages. Furthermore, a receive-carry-forward mechanism can enhance the reliability of this phase.

On the other hand, if any RSU located at a certain road intersection fails out, traveling vehicles that cross this intersection should cooperatively replace the damaged RSU while

constructing the optimal path towards each destination. In order to achieve this, each vehicle should have the ability of processing, recording and forwarding the traffic data, as do RSUs. Vehicles located close to the road intersection can receive the traffic evaluation reports of the existing road segments and the advertisement messages of adjacent road intersections. Thus, these vehicles should have the same database that can be recorded at the local database of the RSU over the road network. If the RSU located at any road intersection was detected in a failure situation, the closest vehicle to the center of that road intersection should broadcast the updated advertisement message towards adjacent road intersections.

Fault Tolerant Path Recommendation

In the case that a certain vehicle does not receive the recommendation report to select the best turn at the road intersection, it should send a request message asking for the recommendation report. If any neighboring vehicle/RSU received the request message, it should verify whether it is the best node to respond. The best node to respond is the one located in the closest position to the vehicle that requested the recommendation. If the best vehicle to respond also does not receive the recommendation report, it should wait a certain period of time to receive the recommendation message.

When other receiver nodes become aware that they are not the best candidates to respond, they set a waiting time to hear the best node forwarding the recommendation message. The waiting time at each of these nodes is set based on the distance between the respective node and the vehicle requested. The closer the node is to the requested vehicle, the less the waiting time sets to be. If the entire waiting time interval at any of these nodes passes without hearing any forwarding of the recommendation message, this node should forward the recommendation message itself.

4.7 Performance Evaluation of ICOD

In this section, we evaluate the performance of the introduced ICOD protocol. We begin by reporting on the performance of the congestion avoidance and the Eco-path recommendation variants compared to the path with the shortest distance (i.e., *Shortest* path). We illustrate the advantages and overheads of each variant in terms of communication overhead, travel time, travel distance, fuel consumption and gas emission metrics. We then investigate the advantages of the context-aware path recommendation variant in

terms of its ability to minimize the number of occasions in which it recommends special road segments and/or road segments in bad condition, compared to the *Fastest* path (i.e., congestion avoidance path recommendation variant). We also discuss the impact of different road segment lengths and different area of interest sizes on the performance of the different variants introduced. We also present the advantages of the different variants proposed, compared to previous path recommendation protocols: Route Information Sharing (RIS) [42], vehicle routing algorithm for Traffic Congestion (TraffCon) [56] and ECOlogical Route search and driving protocol (ECO-Route) [81]. The details of these protocols are presented in Chapter 2. Finally, we evaluate the performance of the fault tolerant variant (TD-PR) compared to the congestion avoidance variant of ICOD for different scenarios.

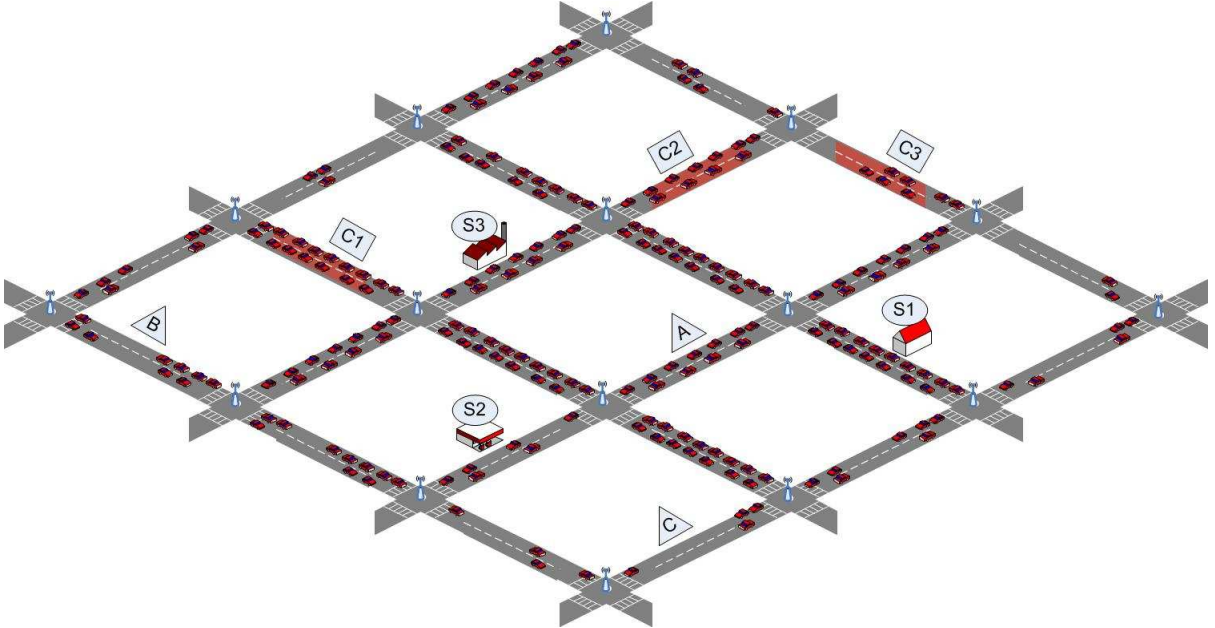


Figure 4.4: Grid-layout scenario.

The performance of ICOD is evaluated in different scenarios where all RSUs are meant to find an alternative path towards the following three different destinations: *A*, *B* and *C*. These destinations are located in a grid-layout that can be mapped to any real downtown grid-map, as shown in Figure 4.4. Special road segments are located at specifically marked services (*S1*, *S2*, *S3*); and, the road segments in bad condition have been highlighted with a different legend (*C1*, *C2*, *C3*) in Figure 4.4. This scenario is a simplified example that represents the extended case where a path is constructed from any road intersection towards any destination in the grid-layout scenario. The performance

of the proposed protocol has been evaluated through an extensive set of experiments using the Network Simulator *NS-2* [64]. Each experiment has been executed 30 times using different traffic scenarios, and the confidence interval obtained is more than 95% for each experiment. We have plotted most of the intervals; however, for some graphs the intervals were too small to plot. The parameters of the simulation used are illustrated in Table 4.1.

Table 4.1: Simulation Parameters of The Path Recommendation Protocol

Parameters	Value
Road Segment Length (m)	200, 400, 600, 800, 1000
Simulation Area (m X m)	1000 X 1000 - 10000 X 10000
Wireless Medium	IEEE802.11
No. of RSUs	16, 25, 36, 49, 64
No. of vehicles	200 - 2000
No. of Destination	3
Mobility Model	Map-Based Mobility
Transmission Range (m)	250
Map Layout	Grid-Layout
No. of Road Segments	40 - 160 bidirectional
Simulation Time (Sec)	5000
α	0.361
β_1	0.09
β_2	0.03
R_T	8
M_v	1250
Traffic speed	1-19 m/s
Traffic Acceleration	0-10 m/s ²
No. of Special RS.	3
No. of Bad Conditions RS	3
γ_2	0.5
γ_1	0.0, 0.25, 0.5
γ_3	0.0, 0.25, 0.5
No. of Faulty RSUs	1, 2, 3

Five congestion levels have been defined in our experiments in order to evaluate the proposed protocol. These congestion levels can be identified as follows: No_Congestion, Low, Medium, High, and Heterogeneous levels (i.e., a mixed scenario including Low, Medium, and High levels). In our experiments, the congestion level is classified based on the expected travel time of each road segment. Considering the No_Congestion scenario, all road segments in the downtown grid area have a short travel time. In the other congestion scenarios, only half of the road segments in the area of interest have No_Congestion experience, so the vehicle's travel time is relatively short. For the Low, Medium and High level scenarios, the other half of the road segments suffer from

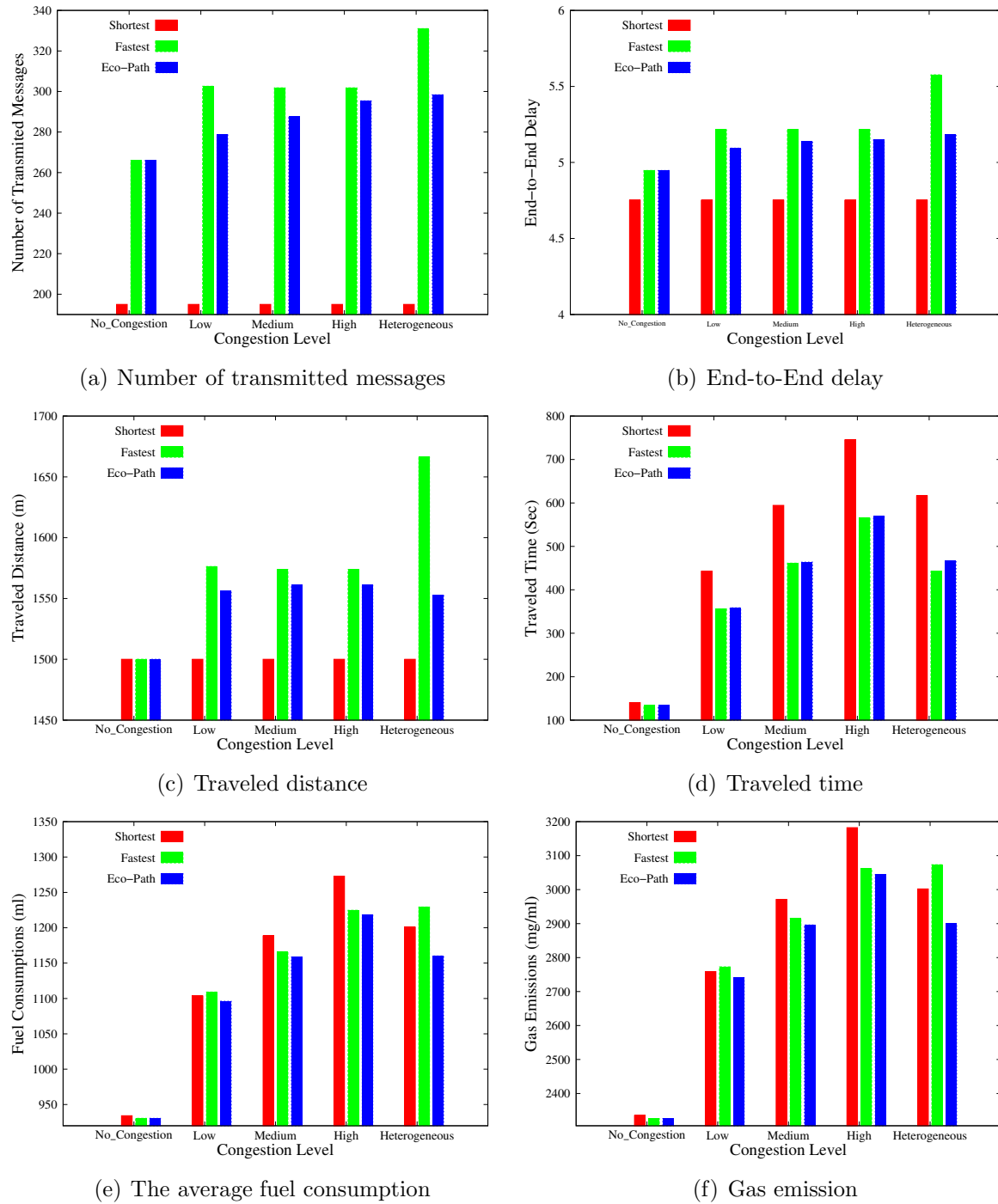
longer travel times at different levels. The higher the congestion level, the longer the vehicle travel time on those road segments. Finally, in the Heterogeneous congestion level scenario, half of the grid area has No-Congestion experience, while the other half suffers from a congestion experience ranging between Low, Medium and High levels. In order to focus on evaluating the performance of ICOD, we assume that each RSU has obtained and evaluated the real-time traffic situations of its surrounding road segments before it starts executing the experiments.

4.7.1 Performance Evaluation of Congestion-Avoidance and Eco-path Variants

Here, we compare the congestion avoidance path (i.e., *Fastest* path) and the Eco-path recommendation variants to the *Shortest* path. Figure 4.5 illustrates the performance evaluation of these variants for the presented congestion levels and scenarios. The elements listed below are illustrated in the aforementioned figure: number of transmitted messages (Figure 4.5(a)), End-to-End delay (Figure 4.5(b)), average travel distance (Figure 4.5(c)), average travel time (Figure 4.5(d)), fuel consumption (Figure 4.5(e)) and gas emissions (Figure 4.5(f)). In order to measure the fuel consumption and the gas emissions, we used Equations 4.6 and 4.7 respectively.

As shown in Figures 4.5(a) and 4.5(b), compared to the *Shortest* path, the congestion avoidance variant requires, on average, 30% extra communication messages and 10% greater delay time to construct the *Fastest* path. As previously explained, the communication overheads in ICOD depend on the topology of the investigated area and the dynamic change occurring in the travel time of each road segment. The travel distance is connected mainly to the topology, and can be predicted based on the location of each source and destination pair, while the travel time needs real-time and extra communications overhead to be estimated. From Figures 4.5(c) and 4.5(d), we can infer that the *Fastest* path increases the travel distance by 10% compared to the *Shortest* path, while it decreases the travel time by 30% for all congestion levels. The higher the congestion level is, the more efficient the congestion avoidance becomes in terms of decreasing the travel time.

Regarding fuel consumption and gas emission metrics, we can see from Figures 4.5(e) and 4.5(f), that the *Fastest* path decreases fuel consumption and gas emission metrics by 10% compared to the *Shortest* path. This is because the *Shortest* path increases the travel time drastically, especially in highly congested scenarios.

Figure 4.5: Comparison analysis of *ICOD* variants.

As illustrated in Figure 4.5, the performance of the Eco-path recommendation variant in terms of communication overhead, travel time and distance lies between the *Shortest* and *Fastest* path performances. However, Eco-path consumes, on average, 30% less fuel than the *Shortest* path fuel consumption. It also consumes 20% less fuel than the *Fastest* path variant. The Eco-path recommendation variant aims to recommend a path that is balanced between the travel time and the travel distance of the constructed path, in order to decrease fuel consumption.

4.7.2 Performance Evaluation of the Context-Aware Variant

In the evaluation study of the context-aware variant, we consider the following three cases:

Case(I): The traveling vehicles consider the travel time and the existence of the special road segments on the selected path, leading towards each destination. Drivers do not mind passing through road segments with bad conditions. In this case, $\gamma_1 = 0.5$, $\gamma_2 = 0.5$ and $\gamma_3 = 0.0$.

Case(II): The traveling vehicles consider the special road segments and the condition of the traveled road segments of the selected path. Drivers prefer to travel through road segments in good condition rather than reaching their destinations faster. In this case, $\gamma_1 = 0.0$, $\gamma_2 = 0.5$, and $\gamma_3 = 0.5$.

Case(III): The traveling vehicles consider travel time, the existence of special road segments and the conditions of the traveled road segments of the selected path. Drivers try to avoid traveling through road segments with adverse conditions but without drastically increasing the travel time of the selected path. In this case, $\gamma_1 = 0.25$, $\gamma_2 = 0.5$ and $\gamma_3 = 0.25$.

The comparative experiments aim to evaluate and compare the following: the travel time of vehicles; the travel distance of vehicles; the number of messages recommending special road segments; the number of occasions at which special road segments have been recommended; the number of messages recommending a road segment in bad condition, and the number of times road segments in bad condition have been recommended for congestion avoidance and the context-aware variant paths.

Starting with vehicle travel time for the different paths obtained, as illustrated in Figure 4.6(a), the context-aware variant path needs, on average, 20% more travel time compared to the *Fastest* path (i.e., congestion avoidance variant path). Case(II) needs the longest travel time among all compared cases, since the decision of updating the

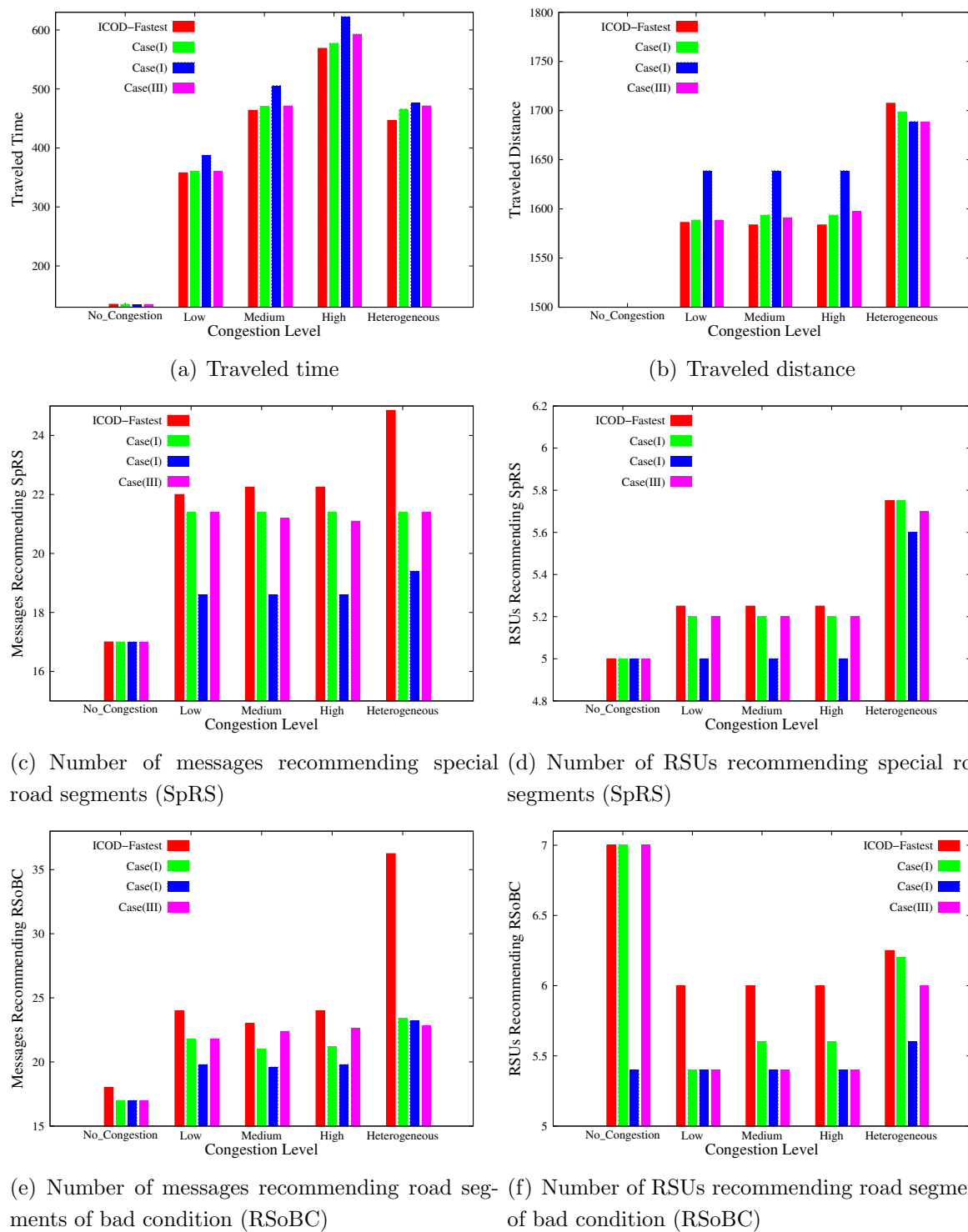


Figure 4.6: Performance evaluation of the context-aware variant.

internal database and forwarding the *ADV* message at each RSU is made based only on the locations of special and bad road segment conditions; this does not consider the traffic congestion level or the expected travel time. The travel time performances of Case(I) and Case(III) are located between the performance of Case(II) and the performance of *Fastest* path, because these two cases partially consider the travel time parameter.

As illustrated in Figure 4.6(b), the travel distances for the obtained paths of Case(I) and Case(III) are 10% longer than the travel distance of the *Fastest* path. On the other hand, the travel distance of the path obtained by Case(II) is 30% more than the travel distance of the *Fastest* path. The travel distance parameter varies based on the scenarios tested and on the locations of the congested road segments, special road segments or road segments in bad conditions.

Figure 4.6(c) shows the performance evaluation of the proposed protocol in terms of the number of messages that recommend each special road segment throughout the area of interest. From Figure 4.6(c) we infer that, compared to the *Fastest* path, the context-aware path decreases by 40% the number of messages that recommend each special road segment. Case(I) and Case(III) try to avoid passing through highly congested road segments, as well as through special road segments. However, Case(III) aims only to avoid passing through special road segments and road segments in bad conditions. Due to this behavior, the performance in terms of the number of messages that recommend special road segments in Case(I) and Case(III) are closer to the *Fastest* path performance than the number of messages in Case(II).

After obtaining the desired paths towards the located destinations, we check how many RSUs recommend a special road segment as a next road towards each destination. From Figure 4.6(d), the context-aware path decreases by an average of 25% the number of occasions that special road segments are recommended, compared to the *Fastest* path. By comparing Figure 4.6(c) to Figure 4.6(d), we see that there is a correspondence between the performance evaluation of number of messages recommending special road segments, and the number of RSUs that recommend a certain road segment as a route.

Finally, Figure 4.6(e) and 4.6(f) investigate, respectively, the number of messages that recommend a road segment that is in bad condition, and the number of RSUs recommending such road segments. The context-aware path reduces the number of messages that recommend each road segment in bad condition by 30% compared to the *Fastest* path. On the other hand, the number of times the obtained path passes through any special road segment is also decreased by 20% compared to the *Fastest* path.

4.7.3 Impact of Road Segment Length and Size of Area of Interest

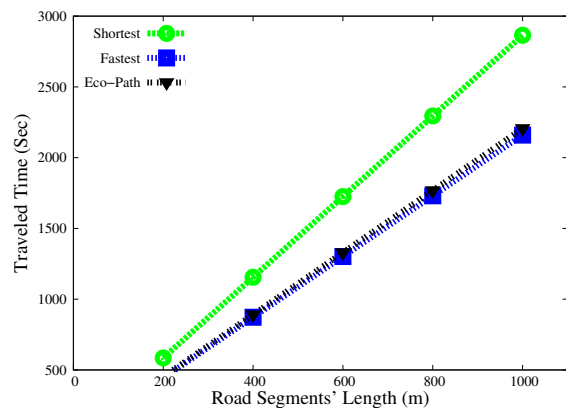
We have investigated the impact of the road segment length and the size of the area of interest (i.e., the different number of road intersections). As expected, for all variants, a longer segment of road is associated with greater travel expenses (i.e., time and distance) for the selected path, as well as needing a larger communication overhead to obtain that path. The traveling vehicles in each road segment reliably deliver messages between the existing RSUs at the end of each road segment, requiring more time and a greater number of transmitted messages within the forwarding process. Figures 4.7(a) and 4.7(b) show the travel time and number of transmitted messages, respectively, that are needed for different road segment lengths.

The estimated travel time, travel distance, communication overhead, and driving cost all increase when expanding the size of the area of interest. As we can see from Figures 4.7(d) and 4.7(e), the congestion avoidance variant is becoming more efficient in terms of its ability to decrease the estimated travel time compared to the *Shortest* path. This is same for the Eco-path variant, where the ECO-path saves more fuel compared to the *Fastest* and *Shortest* paths. For the context-aware variant, the number of recommendations of special road segments or of road segments in bad condition has been decreased by increasing the size of the area of interest; this is because more possible paths can be followed towards any destination by expanding the area of interest. Figure 4.7(f) illustrates the number of messages that recommend special road segments for different area sizes.

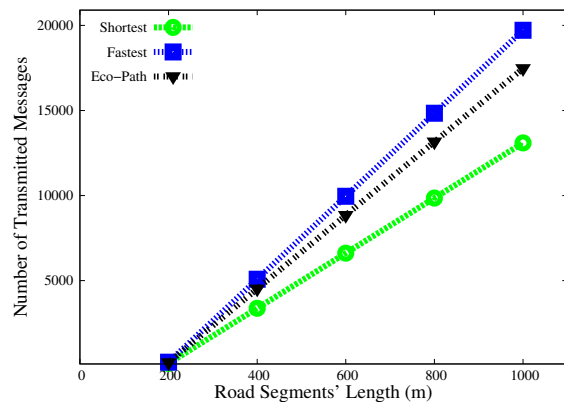
Over all, for the different area sizes evaluated, the travel expenses increase, as expected, and the increase in communication overhead is acceptable and reasonable in the context of vehicle traveling speed.

4.7.4 Comparative Performance of ICOD to RIS, TraffCon and ECO-Route

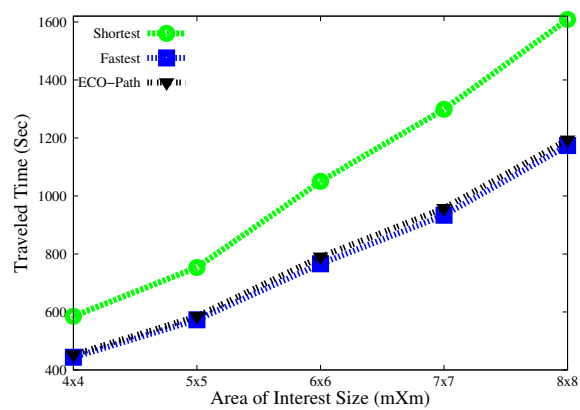
We compare the proposed protocol (ICOD) to three previous path recommendation protocols selected: Route Information Sharing (RIS) [42], vehicle routing algorithm for Traffic Congestion (TraffCon) [56], and ECOlogical Route search and driving protocol (ECO-Route) [81]; this comparative study has been performed using an extensive set of scenarios and experiments implemented by NS-2 [64].



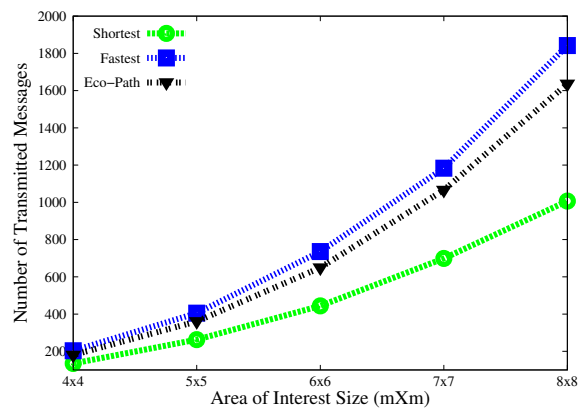
(a) Road length impact on travel time



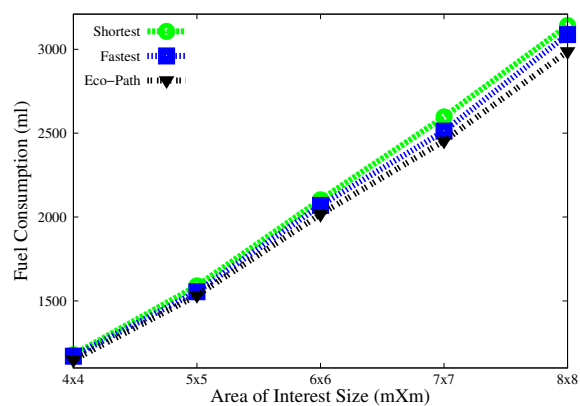
(b) Road length impact on number of messages



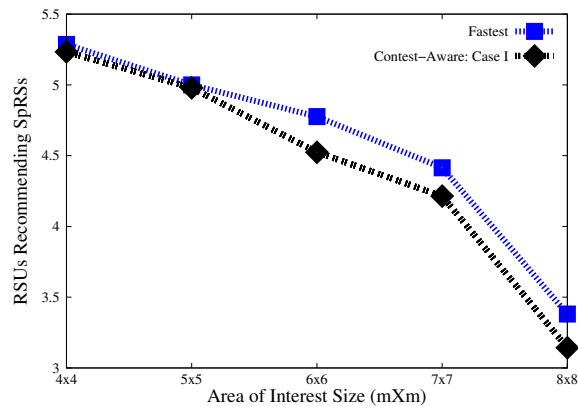
(c) Area size impact on travel time



(d) Area size impact on number of messages

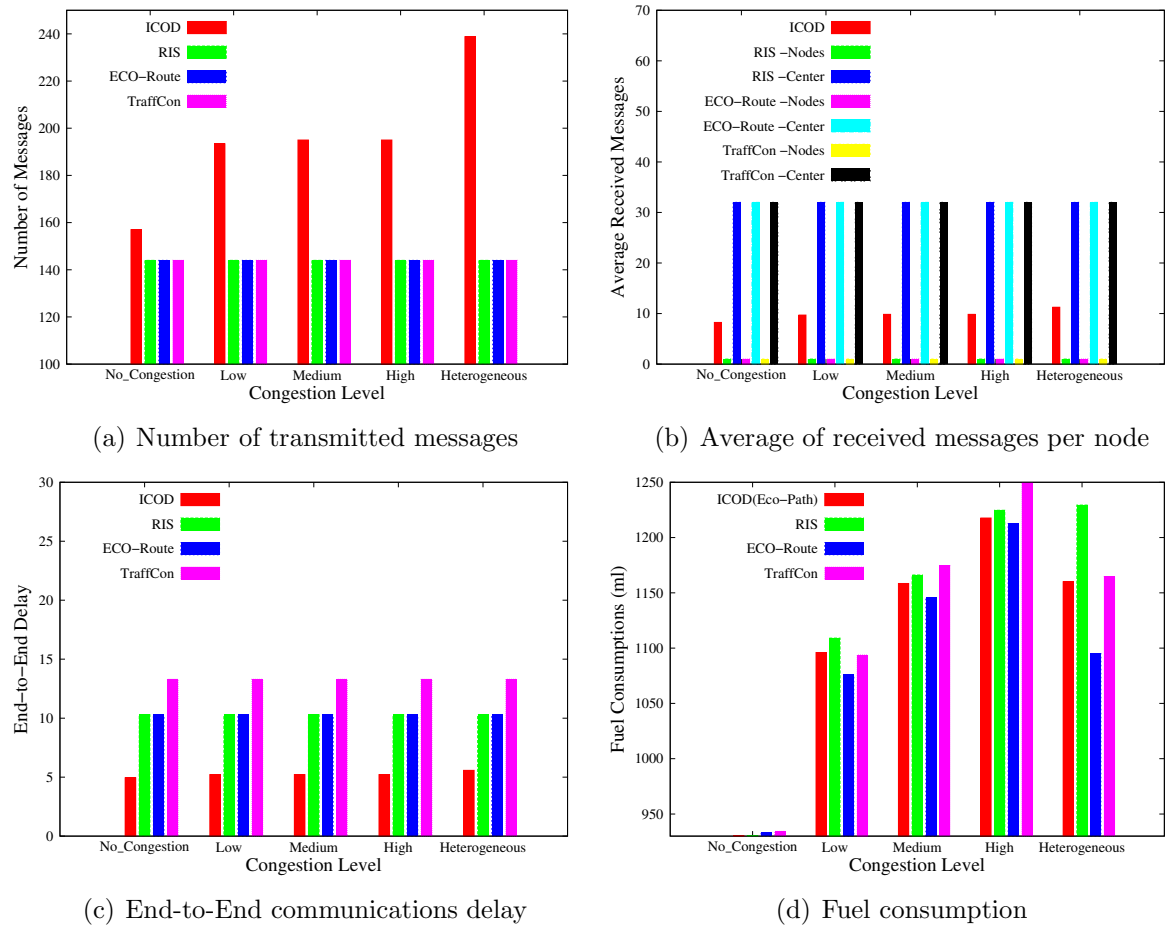


(e) Area size impact on fuel consumption



(f) Area size impact on SpRSs

Figure 4.7: The impacts of road segment length and area size.

Figure 4.8: Comparison of the *Fastest* Variant to previous protocols.

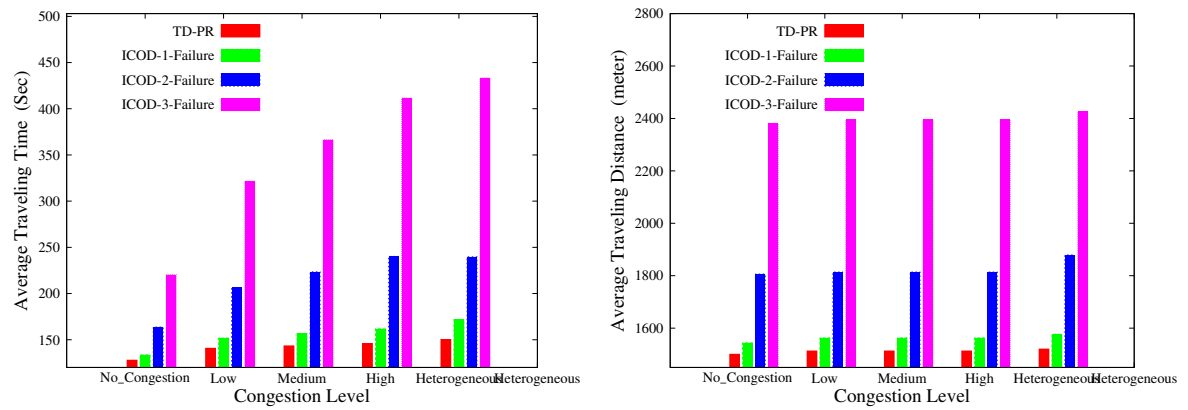
As illustrated in Figure 4.8(a), the number of transmitted messages by ICOD is on average 40% greater than the number of messages sent by the RIS [42], ECO-Route [81] and TraffCon [56] protocols. All of these protocols used a centralized communication approach, in which each RSU contacts the central server processor with the traffic situation of its surrounding road segments. The server responds to each RSU with the best path towards all destinations. ICOD needs to transmit different number of messages for different congestion levels. As mentioned previously, in ICOD, the path is constructed cooperatively between the existing RSUs. Due to the distributed behavior of ICOD, an extra and different quantities of transmitted messages are needed for each congestion level traffic scenario.

At the same time, Figure 4.8(b) illustrates the average number of messages received by each node in the RIS [42], ECO-Route [81], TraffCon [56] and ICOD protocols. A large number of messages is received by the central processor in the previous protocols because it collects the traffic data for the entire area of interest. These messages create a bottleneck problem; the central processor drops some packets when its buffer is full. In ICOD, the messages sent are distributed among all nodes in an approximately equal fashion without causing a bottleneck or a single point of failure problem, because there is no centrally targeted node.

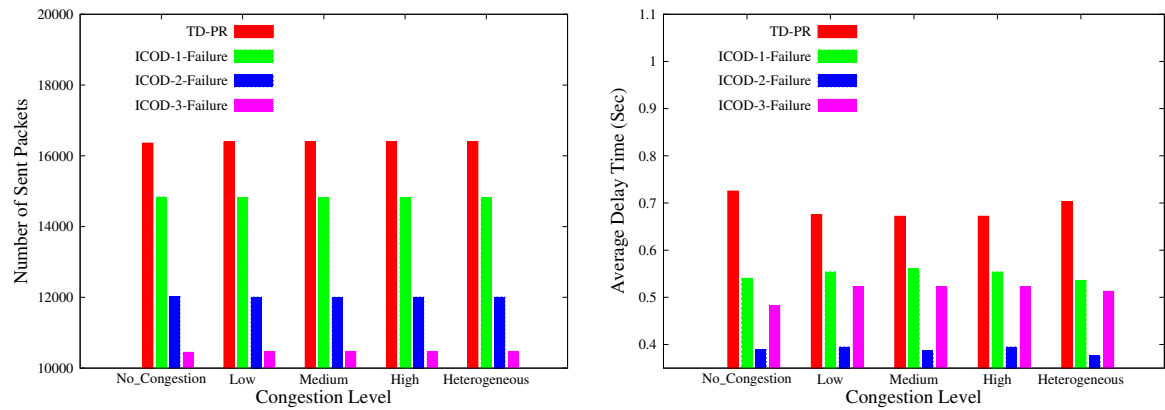
In the RIS [42], ECO-Route [81] and TraffCon [56] protocols, all RSUs must have direct contact with the central processor. The central processor finds the fastest path towards each destination, and then sends it to each traveling vehicle. Due to these successive processes, a high delay is expected, which causes late path recommendations. As shown in Figure 4.8(c), ICOD provides a more real-time path recommendation (i.e., 50% less delay time compared to RIS [42] and ECO-Route [81], and 60% less delay time compared to TraffCon [56]) that is more useful for traveling vehicles. The parallel processing and distributed behavior of ICOD illuminates the queuing and the central data gathering delay time.

4.7.5 Performance Evaluation of TD-PR

From Figure 4.8(d), we observe that the fuel consumption of ICOD is 20% less than that of the RIS protocol fuel consumption and 10% less than that of TraffCon fuel consumption. However, the fuel consumption of ECO-Route is 20% less than that the ICOD because ECO-Route concentrates solely on considering the fuel consumption of



(a) Average travel time towards each destination (b) Average travel distance towards each destination



(c) Number of transmitted packets (d) Average delay of obtaining the optimal path at each road intersection

Figure 4.9: The performance evaluation of TD-PR protocol.

each road segment and recommends the path that consumes the least amount of fuel, while ICOD considers different parameters in addition to fuel consumption.

In this section, we evaluate the performance of TD-PR compared to the congestion avoidance variant of ICOD when some failures occur among the existed nodes and/or links of the communication network (i.e., VANETs). We investigate several scenarios in which one or more RSUs fail to process and forward the advertisement messages during the path construction phase. An extensive set of simulation experiments using *NS-2* [64] has been utilized.

Figure 4.9 illustrates the performance evaluation of TD-PR compared to ICOD in cases that one, two or three RSUs failed to process and forward the advertisement message. As we can infer from Figure 4.9(a) and Figure 4.9(b), the average travel time and the average travel distance of vehicles is drastically increased in ICOD by increasing the number of faulty RSUs. TD-PR is able to tolerate the different faulty scenarios where less travel time and travel distance are obtained (i.e., a more reliable path is constructed).

On the other hand, TD-PR requires greater delay time and extra transmitted packets than ICOD, in these scenarios. As we can see from Figure 4.9(c) and Figure 4.9(d), the more the number of faulty RSUs in the congestion avoidance variant of ICOD, the smaller the delay time, and the smaller the number of transmitted packets.

4.8 Summary

This chapter proposed a fully distributed and intelligent path recommendation protocol, appearing in the form of ICOD. The proposed protocol is intended to recommend the path towards each destination throughout the area of interest. It aims to avoid and to reduce traffic congestion, fuel consumption and gas emissions released into the atmosphere; also, this protocol has taken into consideration special services located at each traversed road segment, as well as the conditions of these road segments. In order to achieve all of these goals flexibly, three different variants have been introduced to ICOD to recommend alternative paths: congestion avoidance, Eco-path and context-aware path selection. The first variant (i.e., congestion avoidance) is intended to find the least congested path towards any destination. The Eco-path variant balances between the travel time and the travel distance of the selected path; it performs well in terms of fuel consumption and gas emission metrics. Finally, the context-aware variant can be used with any other variant to guarantee a certain congestion-free level in special road segments (i.e., road segments that are located at commonly targeted services). Moreover, the context-aware variant

enables drivers to flexibly choose their desired path by either considering the travel time or the conditions of the traversed road segments. The introduced variants of ICODE use the same manner of distribution, which eliminates the centralized behavior problems. The fast communication and the hop-by-hop recommendation features introduce real-time path recommendations. This real-time and dynamic path reacts immediately to changes in traffic during each vehicle trip. Finally, a fault tolerant variant of ICODE has been designed and tested in this chapter (TD-PR); this variant aims to handle the potential failure problems in ICODE.

Chapter 5

Balancing Traffic Path Recommendation Mechanisms

5.1 Introduction

We have proposed an intelligent distributed path recommendation protocol, named ICOD, in Chapter 4. This protocol aims to select the optimal path leading towards each destination, while providing alternate routes so that vehicles can avoid highly congested road segments. However, in some scenarios, the input flow at any intersection can be highly congested, and most traveling vehicles are heading towards the same destination or different destinations located in close proximity. In these scenarios, the existing bottleneck problem is transferred from one road segment to the next road segment at each intersection. This exaggerates the traffic congestion problem all over the road network, instead of solving or reducing it. In other scenarios, the bottleneck problem is generated by the path recommendation protocol that is in use, over some output road segments. This occurs when most of the vehicles arriving from several input road segments, and moving towards the road intersection, receive recommendations to take the same output road segment towards their destinations. Due to these potential problems inherent in ICOD, in this chapter we introduce two balancing traffic path recommendation mechanisms: Bal-Traf and Abs-Bal. Bal-Traf is a reactive technique that is initiated when a certain output road segment of any road intersection is predicted to be in an overloaded traffic situation. In the case that the estimated traffic density of any output road exceeds its ideal capacity, Bal-Traf recommends that some vehicles, planning to pass by this road segment as next-hop, choose another less congested output road segment. On the other

hand, Abs-Bal is a proactive balancing traffic mechanism; it aims mainly to distribute the input traffic at each road intersection in an absolutely even manner among all output road segments. Furthermore, Abs-Bal considers the best travel time parameter of vehicles in addition to its purpose of balancing traffic in order to configure the fastest and least congested path.

5.2 Overloaded Road Segments and Alternative Options

The traffic volume and traffic density parameters compared to road segment capacity are the main factors used to detect highly congested (i.e., overloaded) road segments. In this section, we first define the input and output road segments of road intersections. We explain the technique used to detect and predict overloaded road segments. Finally, we specify how to rank the output road segments at each road intersection leading towards the targeted destinations; the main consideration of this ranking process is the estimated travel time of each next hop option towards the target destination.

5.2.1 Road Intersections and Investigated Scenarios

Grid-layout scenarios, commonly found in downtown areas, are investigated in this chapter. Figure 5.1 illustrates the input and output traffic flows of two adjacent road intersections, with an RSU at each road intersection (i.e., A and B). As we can see from Figure 5.1, each road intersection has four input flows and four output flows. In this phase, the RSU at each road intersection gathers destination reports from all input flows and traffic characteristic reports of output flows.

An RSU is expected at each road intersection over the downtown grid-layout area. In general, each road segment starts from a certain road intersection and ends at the next road intersection. This means that an RSU exists at the starting point of such a road segment, while another RSU is installed at the ending point. The starting and ending points are assigned based on the direction of traffic over the road segment. For example, in Figure 5.1, A is located at the starting point of direction (D1), and B is located at the ending point. However, B is located at the starting point of D2, and A is located at the ending point. The RSU at each road intersection is responsible for ranking the output road segments as optional next hop in the route towards each targeted destination (D_k). Ranking output road segments at each road intersection depends on the following

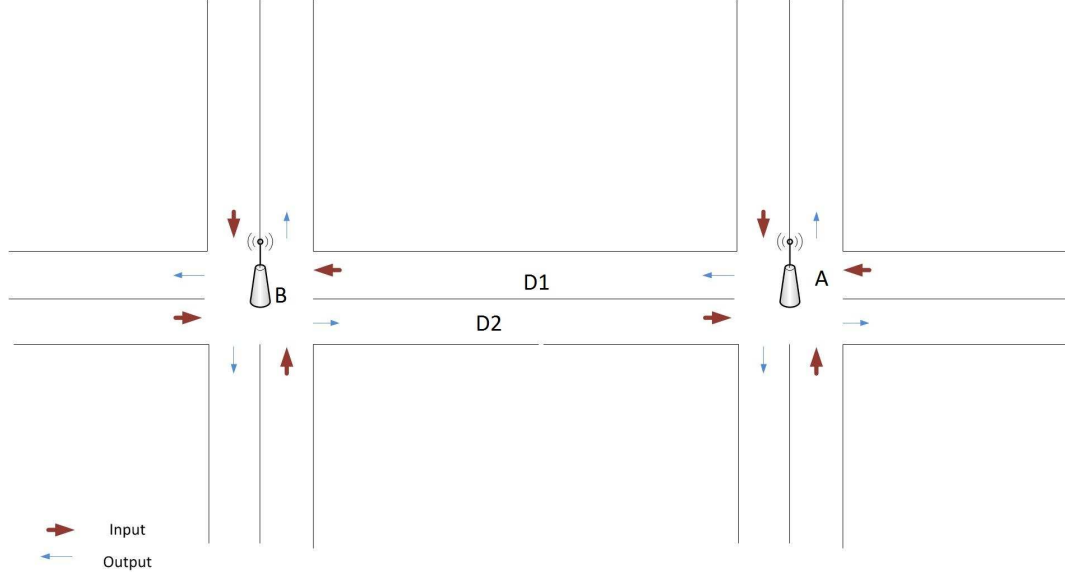


Figure 5.1: Input and output traffic flows at road intersections.

parameters: 1) the length of each road segment; 2) the average speed driven on such a road segment; and 3) the location of that road segment in respect to D_k . At the same time, these RSUs gather the destination reports of traveling vehicles at each input road segment. The destination reports categorize the traffic of the road segment based on the locations of targeted destinations for traveling vehicles. Thus, an appropriately located RSU can predict the best next output road segment to take towards each destination, based on the local gathered information. It then sends a recommendation message to traveling vehicles.

5.2.2 Detecting and Predicting Overloaded Road Segments

An overloaded road segment can be caused or exaggerated by the path recommendation protocol in use, as previously discussed. This occurs if the recommendation protocol suggests that the majority of vehicles take the same path towards each destination. In this section, we first describe how to detect overloaded road segments that already exist within the road network. Then, we investigate the scenarios in which the path recommendation protocol causes overloaded road segments, and determine how to predict these overloaded output road segments.

Detecting Overloaded Input Road Segments

Some road segments become overwhelmed by the presence of a large number of vehicles. These roads witness high traffic density which exceeds their capacity; this causes a reduction in traffic speed. Traveling vehicles cannot proceed at the maximum speed limit on such road segments, due to high levels of traffic congestion. In order to detect overloaded road segments, we introduce the saturation density (Sd_i) parameter of each road segment (i). The saturation density (Sd_i) is defined as the maximum traffic density that can travel smoothly over any road segment.

According to [66], the “optimum density” of freeways in the United States is described as ranging between [0.016 - 0.022] vehicles per meter per lane. When optimum density occurs, the maximum traffic flow can proceed on the road at the maximum allowed speed [66]. On the other hand, the “jam density” is described as ranging between [0.1 - 0.136] vehicles per meter per lane [66]; in the case of jam density, extreme traffic density is associated with a complete stop traffic flow. In our work, the interval of saturation density (Sd_i), is set between the optimum density and the traffic jam density intervals: saturation density is defined as ranging between [0.068 - 0.076] vehicles per meter per lane.

In general, in order to detect overloaded road segments, the real-time density (d_i) of each input road segment is compared to the saturation traffic density (Sd_i) of such a road segment. Whenever d_i is higher than Sd_i , the road segment i is configured as an overloaded road segment.

Predicting Overloaded Output Road Segments

As previously discussed, in some scenarios the congested road status is caused or generated by path recommendation protocols. For example, the RSU, located at any road intersection, recommends the most of the traveling vehicles moving towards the intersection (i.e., coming from differing road segment inputs), take the same output road segment as next hop. In this case, the overloaded road segment scenario is generated by this RSU, which runs a certain path recommendations protocol.

In order to predict the overloaded output road segments, each RSU counts the number of vehicles that are supposed to travel through each output road segment of the road intersection. The RSUs can perform this task according to the number of vehicles that are targeted for each existing destination; thus, the best output road segment is assigned

to each destination in that RSU's database. In the case where the estimated density (d_i) of any output road segment is greater than the saturation density (Sd_i) of that particular road segment, it is predicted that the output road segment will be overloaded.

5.2.3 Ranking the Output Road Segments Leading Towards each Destination

RSUs located on road intersections compute the cost of all next hop options leading towards each destination in terms of travel time. The cost of travel time is computed from the destination location to the road intersection location. At its internal database, the RSU in question records the cost of all candidate next hop options leading towards each destination. Three next hop options leading towards each destination are investigated for each input traffic flow at each RSU: right turn (RT), direct move (DT), and left turn (LT). Figure 5.2 illustrates these options for vehicles on the X flow at any road intersection.

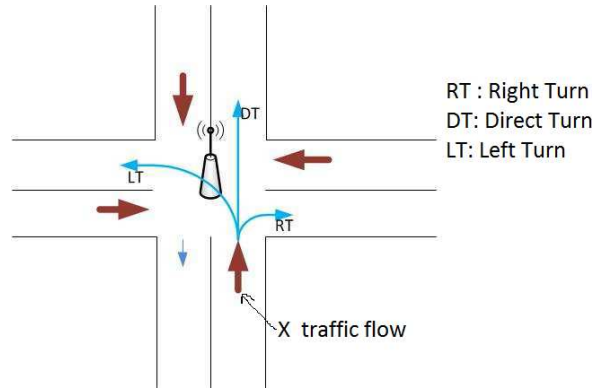


Figure 5.2: Turn options towards each destination.

RSUs rank the output options leading towards each destination; they prioritize the options requiring the least amount of travel time, while options with the longest travel time are ranked last (Op_1 , Op_2 and Op_3). A typical path recommendation protocol usually recommends that all vehicles travel through Op_1 , in order to reach their destinations in the fastest expected time. However, Op_2 and Op_3 are recommended for some traveling vehicles for the purpose of eliminating the predicted overloaded road segment scenarios that are generated over Op_1 . Ranking turn options towards each destination helps with the recommendation of the next best option; this is helpful when the best turn is not applicable or it has been overloaded.

5.3 Balancing Traffic Path Recommendation Mechanisms

In this section, we propose two balancing traffic-based path recommendation mechanisms (i.e., Bal-Traf and Abs-Bal). These mechanisms are designed to eliminate the bottleneck problem and highly congested road segment scenarios, while selecting the optimal path towards each destination. Bal-Traf and Abs-Bal both consider the generated traffic volume on output road segments at each road intersection. Bal-Traf first predicts the overloaded output road segment as explained in Section 5.2.2. Then, it distributes the estimated traffic of that overloaded road segment among other existing output road segments at each road intersection. Bal-Traf reactively helps to eliminate the bottleneck problem, by distributing the traffic only if it detects an overloaded output road segment scenario. Otherwise, vehicles travel along the best output road option (Op_1) towards their targeted destinations in Bal-Traf.

On the other hand, Abs-Bal recommends that vehicles leave the road intersection in an absolutely balanced manner (i.e., the same traffic density is generated at each output road segment of the relevant road intersection). Abs-Bal protectively eliminates overloaded road segment scenarios, unless all output road segments are in an overloaded situation. Incoming traffic at each road intersection is distributed evenly among outgoing traffic flows leading towards each targeted destination. This mechanism is intended to avoid generating drastically overloaded road segment scenarios. The details of these mechanisms are introduced in the rest of this section.

5.3.1 Balanced Traffic Path Recommendation Mechanism (Bal-Traf)

In this mechanism, each RSU responds by using the balancing traffic algorithm only if it predicts that any of the output road segments of the relevant road intersection will be potentially overloaded. The output traffic flow is considered to be overloaded if the estimated density of traveling vehicles (Td_i) at that flow is more than Sd_i (i.e., the saturated defined density), as explained in Section 5.2.2. Algorithm 6 illustrates the systematic procedure (Bal-Traf) used for eliminating the bottleneck and traffic congestion problems that are caused by the path recommendation protocols (e.g., ICOD).

Algorithm 6: Balancing Traffic Algorithm of Overloaded Road Segments

Data: TR: right turn; TL : left turn; TD : direct move; Op_1 : best turn option towards the destination; Op_2 : the second best option toward the destination; Op_3 : the worst option towards the destination; O_{per} : overhead percentage; Sd_i : saturation density of the traffic flow i ; Td_i : traffic density of the output flow i ; Td_{D_k} : the traffic density estimated to occur on the output flow moving towards D_k .

```

1 Each RSU computes the required travel time of all output options leading towards each  $D_k$ : TR, TL,
  TD;
2 Based on the required travel time, these options are sorted in ascending order:  $Op_1, Op_2, Op_3$ ;
3 while Possible Overloaded Roads do
4   if Output traffic flow  $i$  is detected as overloaded then
5     while More Destinations Required do
6       An RSU computes the  $O_{per}$  of the output flow ( $i$ ) using Equation 5.1;
7       An RSU computes the difference between  $Op_1$  and  $Op_2$  leading towards each  $D_k$  in terms
        of travel time:  $\text{cost}(Op_2) - \text{cost}(Op_1)$ ;
8        $D_k$ : the destination with the  $\text{Min}(\text{cost}(Op_2) - \text{cost}(Op_1))$ ;
9       if  $Op_2$  is an overloaded road segment then
10         $Op_2 = Op_3$ ;
11        Continue;
12      end
13      if  $Td_{D_k} > O_{per} \times Td_i$  then
14         $(Td_{D_k} - (O_{per} \times Td_i))$  of traveling vehicles are recommended to choose  $Op_2$  ;
15         $(O_{per} \times Td_i)$  of traveling vehicles are recommended to choose  $Op_1$  ;
16        No More Destinations Are Required;
17      else
18        All Vehicles traveling towards  $D_k$  should be recommended to choose  $Op_2$ ;
19         $O_{per} = (Td_i - Td_{D_k}) - Sd_i / Td_i$ ;
20        Remove  $D_k$  from the destination list;
21      end
22    end
23    Add all destinations  $D_{i,k,...}$ ;
24  else
25    Each vehicle proceeds along the best option ( $Op_1$ ) towards its targeted destination;
26    No Possible Overloaded Roads;
27  end
28 end
  
```

First, if the output flow i is detected to be an overloaded flow, the percentage of overload (O_{per}) of i is computed using Equation 5.1.

$$O_{per} = \frac{Td_i - Sd_i}{Td_i} \quad (5.1)$$

The RSU, located at the road intersection where the overloaded output road segment (Or_i) is detected, checks the list of destinations for which Or_i is the best next hop. For each destination on this list, the difference between the cost of the best next hop option (Op_1) and the second option (Op_2), understood as the $[\text{cost}(Op_2) - \text{cost}(Op_1)]$, is computed, *Algorithm 6 Line 7*. The cost of Op_1 (i.e., $\text{cost}(Op_2)$) is the required traveling time from the road intersection to the destination (D_k) if the output Op_1 is taken as next hop. However, $\text{cost}(Op_2)$ is the required travel time towards D_k ; this is when Op_2

is the next-hop option required to follow at the road intersection. The RSU recommends that vehicles travel towards this destination (D_i) select a path through Op_2 towards their targeted destinations only if Op_2 is not in an overloaded situation. In the case that Op_2 is an overloaded road segment, Op_2 is omitted from the options list and it is replaced by Op_3 (i.e., $Op_2 = Op_3$) and the travel expenses for D_k are incurred *Algorithm 6 lines 9-12*. The process of finding the destination that obtains the minimum value of $[\text{cost}(Op_2) - \text{cost}(Op_1)]$ is executed again in this case, using the new values of Op_2 and $\text{cost}(Op_2)$. The destination D_k is found, which has the lowest difference between the cost of Op_1 and Op_2 , understood as $\text{Min}(\text{cost}(Op_2) - \text{cost}(Op_1))$. The latter metric has been used to obtain the minimum average travel time for vehicles, while considering the balancing traffic metric and generated traffic volume characteristics over the road network.

The number of vehicles that are deported to travel through Op_2 or Op_3 should be configured to compute their density (Td_{D_k}) when taking the Op_2 or Op_3 road segment. In the case where Td_{D_i} is more than $(O_{per} \times Td_i)$, only $(O_{per} \times Td_i)$ of these vehicles will be recommended to choose the Op_2 ; $Td_{D_k} - (O_{per} \times Td_i)$ of the vehicles traveling towards D_k will, on the other hand, receive a recommendation to pass by the Op_1 , the estimated overloaded road segment and the best option, *Algorithm 6 lines 13-17*. This number of vehicles (i.e., $Td_{D_k} - (O_{per} \times Td_i)$) can travel through Op_1 without generating overloaded road segment scenarios. On the other hand, if Td_{D_i} is less than $(O_{per} \times Td_i)$, the RSU starts looking for another destination D_j ; the RSU will do this to recommend that vehicles travel towards D_j or suggest that some of them travel through Op_2 to eliminate the overloaded status over the road segment (Or_i), *Algorithm 6 lines 18-20*. However, sometimes eliminating the traffic congestion in one output road segment using Bal-Traf may cause a traffic congestion scenario on another output road segment. To handle this problem, the evaluating and balancing algorithm should be run repeatedly, until no outputting road segment is detected as an overloaded road segment.

In the scenario where the investigated road network is partially congested, Bal-Traf can eliminate the bottleneck problem. However, in the scenarios where the entire road network is congested or where most of the road segments over the road network are congested, Bal-Traf cannot be expected to completely eliminate the bottleneck problems in all output road segments. This is due to the fact that the entire road network is overloaded, which definitely leads to the generation of overloaded output road segments at some road intersections. However, in this scenario, Bal-Traf should decrease the total number of expected overloaded output road segments.

5.3.2 Absolutely Balanced Traffic Path Recommendation Mechanism (Abs-Bal)

This mechanism is intended to keep the traffic throughout the road network absolutely balanced among all existing output road segments. At each road intersection, the respective RSU suggests that vehicles leave the road intersection in a distributed fashion that is evenly balanced. The systematic explanation of the Abs-Bal mechanism is illustrated in *Algorithm 7*. The RSU computes the traffic density of each output road segment of the road intersection: Td_1 , Td_2 , Td_3 and Td_4 . The RSU then computes the average of densities for the output road segments (Ad), which is done using Equation 5.2.

$$Ad = \frac{Td_1 + Td_2 + Td_3 + Td_4}{O_n} \quad (5.2)$$

where O_n is the number of output road segments at each road intersection. The typical consideration of any road intersection scenario is four output road segments; however, it can be more or less than four output road segments in real scenarios.

The RSU recommends that some vehicles travel via the second (Op_2) or third (Op_3) option towards the target destination. Abs-Bal aims to keep the traffic density of all output road segments the same as the Ad value, regardless of the overloaded situation of these output road segments. First, the RSU computes the difference between travel time required through Op_1 and Op_2 , known as $dif1_{D_k}$, towards each destination D_k . It also computes the difference between the travel time required through Op_1 and Op_3 , known as $dif2_{D_k}$, *Algorithm 7 line 4*. Equation 5.3 and Equation 5.4 respectively illustrate how to compute $dif1_{D_k}$ and $dif2_{D_k}$ at each road intersection leading towards the destination D_k .

$$dif1_{D_k} = cost(Op_2) - cost(Op_1) \quad (5.3)$$

$$dif2_{D_k} = cost(Op_3) - cost(Op_1) \quad (5.4)$$

Second, the RSU sequentially checks the Td_i of each output road segment. Only in the case that Td_i of the output road segment i is more than Ad , the RSU checks the list of destinations for which the respective road segment is recommended as the best output road segment. The RSU selects the destination D_j which has the lowest value of $dif1_{D_j}$ among all these destinations. It checks the traffic density of the second potential (Op_2) road segment toward the D_j , (Td_{Op_2}). If Td_{Op_2} is more than Ad , the RSU should remove $dif1_{D_j}$ from the comparison list and it should add $dif2_{D_j}$. Then, it iteratively finds the minimum value among the new list and compares the Td of the alternative option to the

computed Ad value. The first detected alternative option (i.e., the option with minimum extra overhead travel time) that has a Td value less than the value of Ad is recommended as the preferred output road segment to take towards the destination (D_k).

Finally, in the case where the density of arriving vehicles traveling towards D_k (Td_{D_k}) is more than $(Td_i - Ad)$, only $(Td_i - Ad)$ of traveling vehicles towards D_k are recommended to take the second option Op_2 ; the other vehicles, however, proceed on Op_1 (i.e., road segment i) towards D_k , because the density of Op_1 should be equal to Ad as well, *Algorithm 7 lines 9-15*. On the other hand, if Td_{D_k} is less than $(Td_i - Ad)$, the RSU should select more destinations to deport traveling vehicles; so, they take the second or third options towards these destinations until the density of Op_1 becomes equal to Ad , *Algorithm 7 lines 15-17*.

Algorithm 7: Absolute Balancing Traffic Algorithm of Road Networks

Data: TR: right turn; TL : left turn; TD : direct move; Op_1 : best turn option towards the destination; Op_2 : the second best option toward the destination; Op_3 : the worst option towards the destination; Td_i : traffic density of the output flow i ; Td_{D_k} : the traffic density moving towards D_k ; Ad : the average density of all output road segments; $dif1$: the difference between the required travel time of Op_1 and Op_2 ; $dif2$: the difference between the required travel time of Op_1 and Op_3 .

```

1 Each RSU computes the required travel time of all output options towards each  $D_i$ : TR, TL, TD;
2 Based on the required travel time, these options are sorted in ascending order:  $Op_1, Op_2, Op_3$ ;
3 The RSU computes  $Ad$  of the existing road intersection by using Equation 5.2;
4 The RSU computes  $dif1$  and  $dif2$  towards each destination by using Equation 5.3 and 5.4 accordingly;
5 foreach output road segment  $i$  do
6   while  $Td_i > Ad$  do
7      $D_k$ : the destination with the Min( $dif1$ );
8     if  $Td_{Op_2} < Ad$  then
9       if  $Td_{D_k} > (Td_i - Ad)$  then
10         $(Td_i - Ad)$  of traveling vehicles towards  $D_k$  are recommended to take the second
11        option  $Op_2$ ;
12        other vehicles proceed on road segment  $i$  towards  $D_k$ ;
13         $Td_i = Ad$ ;
14         $Td_{Op_2} = Td_{Op_2} + Td_i - Ad$ ;
15      else
16        All vehicles traveling towards  $D_k$  are recommended to take the second option  $Op_2$ ;
17         $Td_i = Td_i - Td_{D_k}$ ;
18         $Td_{Op_2} = Td_{Op_2} + Td_{D_k}$ ;
19      end
20    else
21      Add  $dif2_{D_k}$  to the comparison list of  $dif1$ ;
22    end
23  end
24 end

```

5.4 Performance Evaluation of Bal-Traf and Abs-Bal Mechanisms

The performance of the proposed balancing traffic mechanisms are evaluated in this section. All experiments were executed as vehicles moved towards one of three destinations: *A*, *B* and *C*; these are located in a grid-layout scenario of a downtown area, as illustrated in Figure 5.3. Three different sets of experiments, designed to investigate the benefits and overheads of Bal-Traf and Abs-Bal mechanisms, have been introduced in this section. In the first set of experiments, different levels of traffic congestion were generated only over half of the located road networks. In the second set, a different number of road segments were considered congested in each experiment. Finally, the last set of experiments aimed to investigate the impact of the size of the downtown area on the performance of these two traffic balancing mechanisms. The parameters used in these experiments are illustrated in Table 5.1, all experiments have been implemented using NS-2 [64].

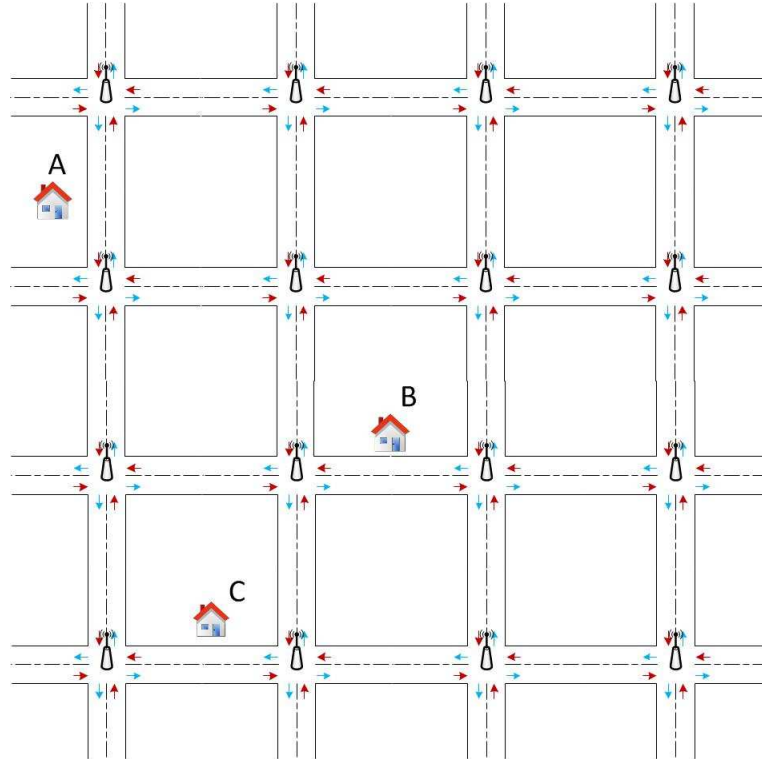


Figure 5.3: Grid-layout scenario with three destinations.

Table 5.1: Simulation Parameters of Bal-Traf and Abs-Bal

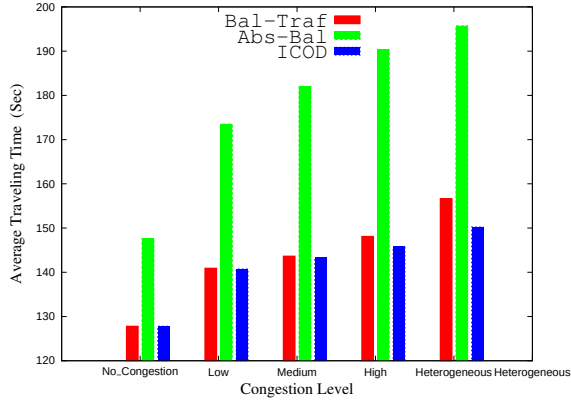
Parameters	Value
Simulation Area (m x m)	1000 x 1000 - 10000 X 10000
No. of RSUs	16, 25, 36, 49, 64
No. of vehicles	200 - 2000
No. of Destinations	3
Mobility Model	Map-Based Mobility
Transmission Range (m)	250
Map Layout	Grid-Layout: 4x4, 5x5, 6x6, 7x7, 8x8
No. of Road Segments	40 - 160 bidirectional
Simulation Time (Sec)	5000

5.4.1 Comparative Performance of Bal-Traf and Abs-Bal to ICOD

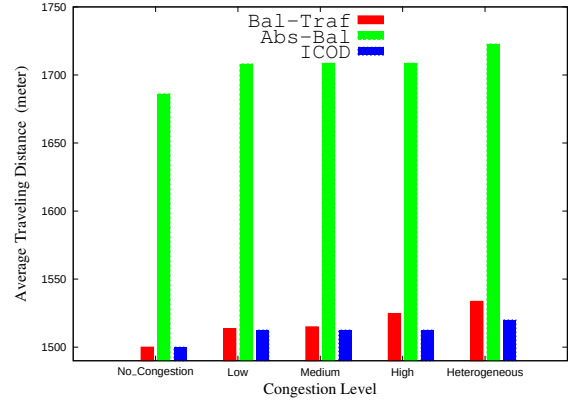
In this section, we simulate different traffic scenarios on grid-layout road networks. The generated scenarios can be in five main categories: No_{congestion}, Low, Medium, High and Heterogeneous congestion scenarios. The details specifications of these scenarios are presented in Chapter 4.

The performance of Bal-Traf and Abs-Bal are investigated for several defined scenarios. We compare these mechanisms to the ICOD protocol, evaluating vehicle travel time and travel distance towards each destination, the number of overloaded output flows generated, and the average density of each output road segment. Figure 5.4 illustrates the performance evaluation of Bal-Traf and Abs-Bal mechanisms compared to ICOD. Each experiment was executed for 30 different scenarios and the confidence interval of each experiment is 95%.

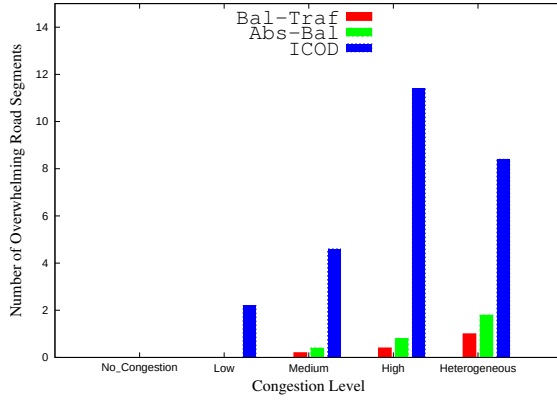
As we can see from Figures 5.4(a) and 5.4(b), Bal-Traf and Abs-Bal mechanisms create extra travel time and extra travel distance when compared to ICOD. In general, Bal-Traf requires 5% more travel time than ICOD, while Abs-Bal requires on average 40% more time to travel in the same scenarios. Bal-Traf requires 10% more travel distance than ICOD and Abs-Bal requires on average 50% more travel distance. As the network becomes more congested, the extra travel time and travel distance of these balancing traffic mechanisms will increase. This is due to the fact that a larger number of vehicles need to choose a longer trip distance with an increased travel time, in order to ensure that the road segments of the best path are less overloaded and the traffic is more balanced over the road network. In low congestion road network scenarios, the performance of the Bal-Traf mechanism is very close to the performance of ICOD. This fact is justified by the low number of output road segments that have been detected in an overloaded



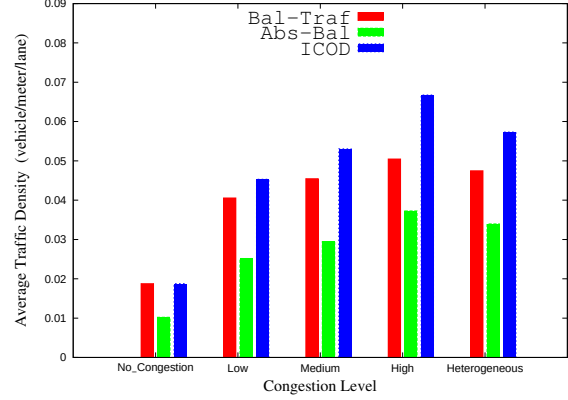
(a) The average travel time towards each destination



(b) The average travel distance towards each destination



(c) The number of overwhelmed road segments



(d) The average density of output road segments at each road intersection

Figure 5.4: The comparative performance of the Bal-Traf and Abs-Bal mechanisms to ICOD.

status; thus, no vehicles are required to change their path to reduce the generated traffic density at output road segments. On the other hand, Abs-Bal produces on average an extra 40% travel time and travel distance compared to ICOD, even in low congestion scenarios. This overhead is justified by the behavior of Abs-Bal, which distributes the traffic among output road segments regardless of the existence of overloaded road segment conditions.

As seen in Figure 5.4(c), when using ICOD, almost 40% of the road segments in the network experienced overloaded conditions. On the other hand, the Bal-Traf mechanism decreases the overloaded output road segments over the road network by an average of

70% compared to ICOD. The Abs-Bal mechanism also performs well in decreasing the number of overloaded road segments compared to ICOD. Bal-Traf achieves a performance that is, on average, 5% superior in decreasing the number of overloaded road segments when performance in terms of decreasing the number of overloaded road segments compared to Abs-Bal. However, in highly congested road network scenarios, Bal-Traf increases the traffic density of some output road segments drastically in order to decrease the number of road segments.

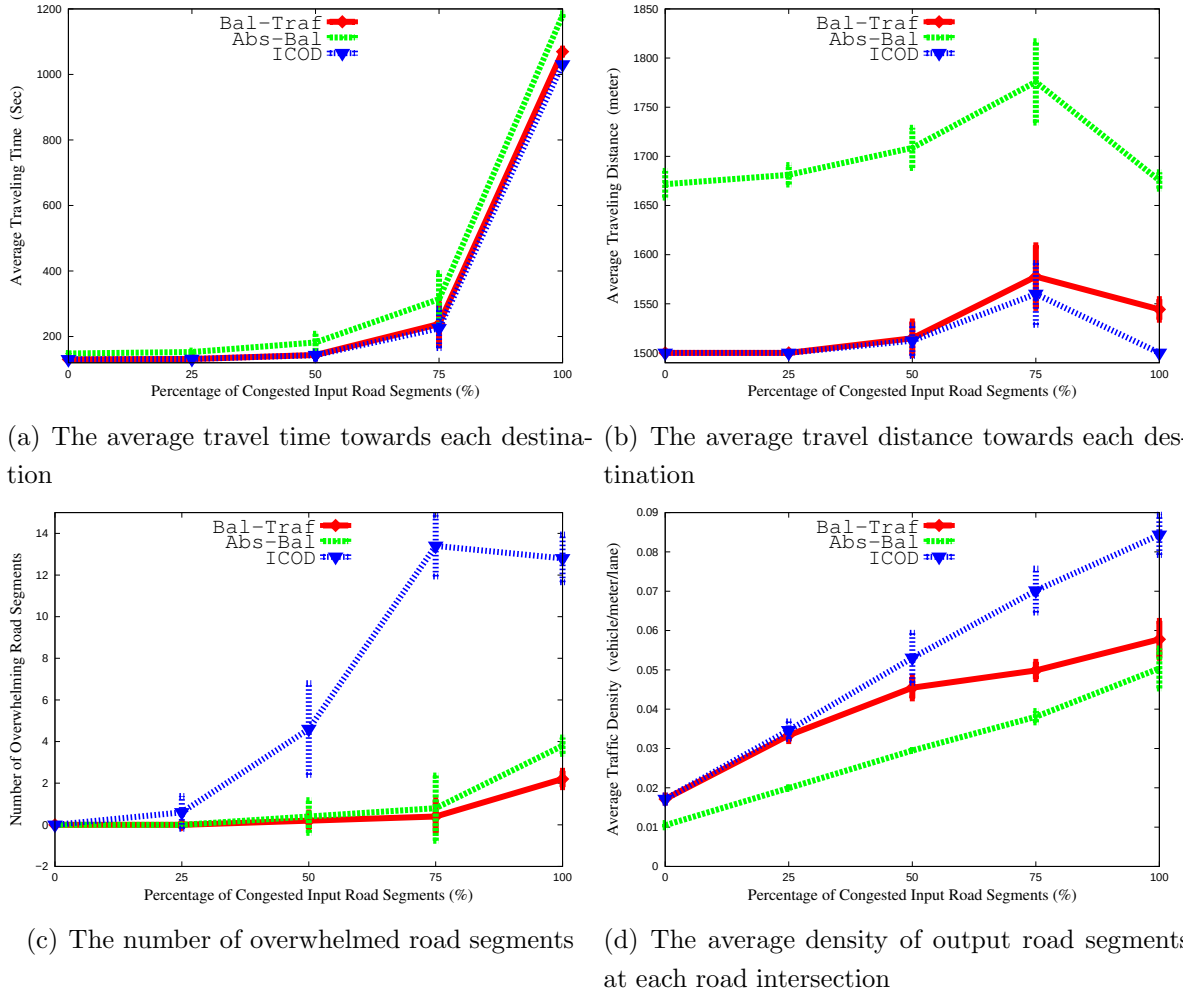
Finally, the traffic density of each output road segment decreases by an average of 10% when the Bal-Traf mechanism is used, compared to ICOD, as illustrated in Figure 5.4(d). Abs-Bal achieves the best performance in decreasing and balancing the traffic density of output road segments. On average, the Abs-Bal mechanism decreases the traffic density of the output road segments by 30% compared to ICOD.

5.4.2 The Impact of Congestion Level on the Road Network

In this section, we compare the performance of Bal-Traf and Abs-Bal mechanisms to ICOD for road network scenarios of different percentages. We have generated different sets of scenarios where 0%, 25%, 50%, 75% and 100% of the input road segments are congested. We set the traffic density and the estimated travel time of these congested road segments according to a medium level of traffic congestion scenario.

As we can see from Figure 5.5(a), the average travel time of vehicles towards each destination increases for all investigated mechanisms when the traffic congestion percentage is increased over the road network. In the case where the entire road network (i.e., 100% of the road network) is congested, the travel time increases drastically compared to other scenarios. This is justified by the fact that all vehicles need to travel through the highly congested road segments. In other scenarios where 25%, 50% or 75% of the road network is not congested, it was recommended that most vehicles travel through these non-congested road segments. The Abs-Bal mechanism requires the longest travel time while ICOD requires the shortest travel time in all investigated scenarios.

Regarding the travel distance of these mechanisms, Abs-Bal also requires the greatest travel distance while ICOD requires the least travel distance in all investigated scenarios, as illustrated in Figure 5.5(b). The same travel time is required by the Abs-Bal mechanism for scenarios in which no overloaded road segments are present in the network (i.e., 0% congested), and in which all road segments are overloaded (i.e., 100% congested).



(a) The average travel time towards each destination
(b) The average travel distance towards each destination

(c) The number of overwhelmed road segments
(d) The average density of output road segments at each road intersection

Figure 5.5: The performance of the Bal-Traf and Abs-Bal mechanisms compared to ICOD for different percentages of congested road network scenarios.

The same result is observed for the ICOD protocol in these scenarios. These results are justified by the fact that Abs-Bal and ICOD both recommend the exact same path for vehicles in these scenarios, regardless of whether the congestion level is 0% or 100%, because all road segments have almost the same relative traffic status. However, Bal-Traf performs differently in a situation where all road segments are overloaded rather than in a situation where all road segments have optimal traffic density. This is due to the large number of overloaded output road segments detected in the scenario where the entire road network is congested. Bal-Traf tries to decrease the number of overloaded road segments; this produces different paths than those produced when none of the road

segments are congested.

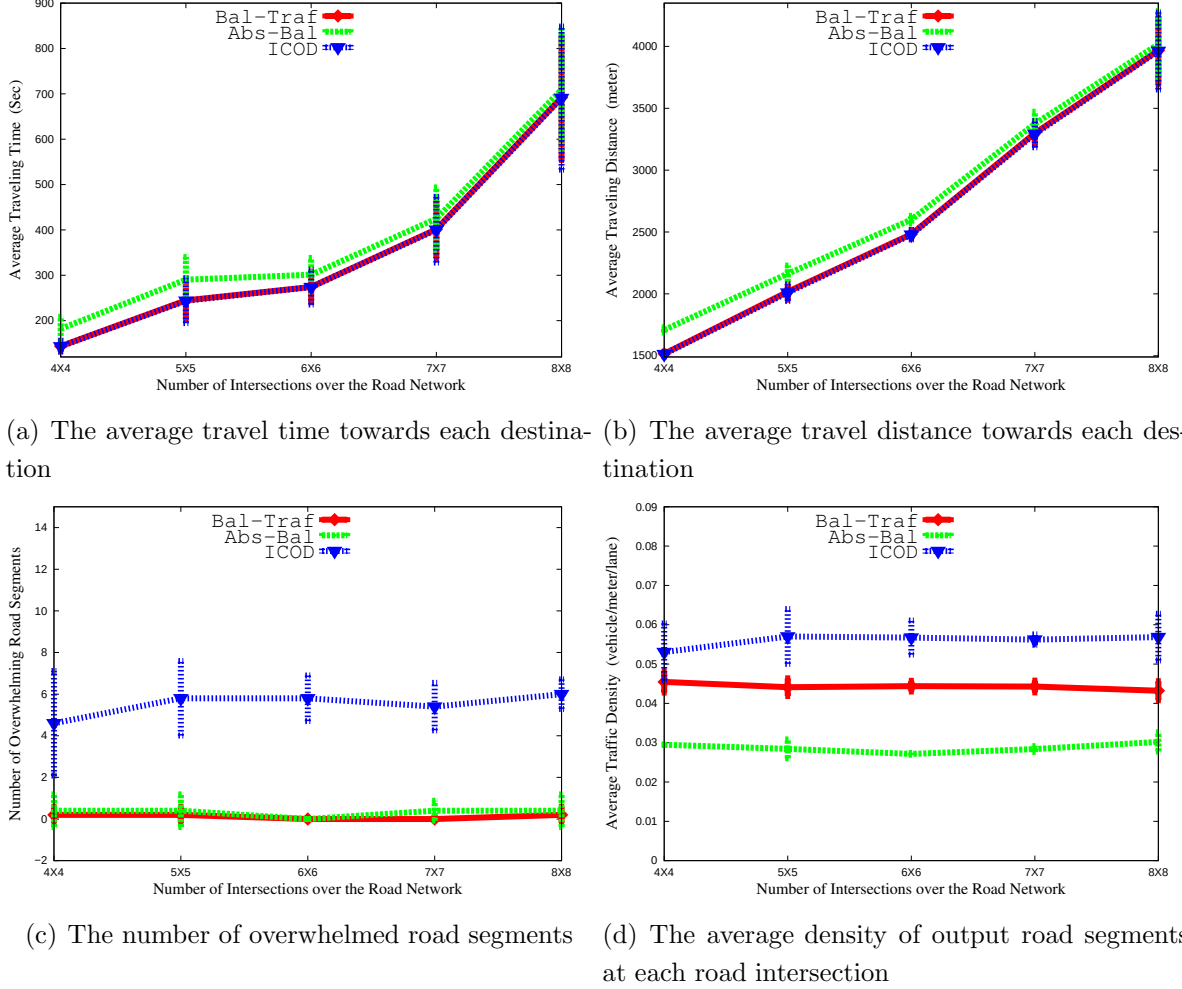


Figure 5.6: The impact of the road network sizes on the performances of the Bal-Traf, Abs-Bal mechanisms and ICOD protocol.

For all congested road network scenarios, Bal-Traf achieves the best performance in terms of decreasing the number of overloaded output road segments, as shown in Figure 5.5(c). Bal-Traf decreases the number of overloaded output road segments by 10% compared to Abs-Bal and by 50% compared to ICOD. Finally, Abs-Bal achieves the best performance of decreasing the traffic density of each output road segment for all congestion scenario percentages, as illustrated in Figure 5.5(d).

Overall, the percentage of congested road segments over any road network has an essential impact on the following parameters: travel time, travel distance, number of

overloaded output road segments and traffic density of each output road segment. By increasing the percentage of congested road segments over the road network, the travel time, number of overloaded road segments, and the traffic density of each output road segment are all increased.

5.4.3 Impact of the Road Network Size

In this section, we investigate the effects of road network sizes on the performance of the Bal-Traf and the Abs-Bal balancing traffic mechanisms. We have generated a set of scenarios for different road network sizes. Half of the road segments existing over these road networks have an optimal traffic density status; the other half of the located road segments have a medium traffic congestion level status.

The average travel time and average travel distance both increased in conjunction with an increased road network size for all investigated mechanisms, as shown in Figure 5.6(a) and Figure 5.6(b). From these figures, the performance of Abs-Bal in terms of travel time and travel distance is closer to the performance of ICOD in large road networks (i.e., 8×8) than it is in small road networks (i.e., 4×4).

However, the road network size does not have an impact on the performances of Bal-Traf or Abs-Bal in terms of the number of overloaded road segments or the traffic density of output road segments metrics. These results are illustrated graphically in Figure 5.6(c) and Figure 5.6(d) respectively.

5.5 Summary

In this chapter, we have proposed two balancing traffic path recommendation mechanisms (Bal-Traf and Abs-Bal) using VANETs. The Bal-Traf mechanism detects and eliminates the expected overloaded output road segments at each road intersection. ICOD recommends the best next hop towards each targeted destination with the least required travel time. However, Bal-Traf considers the traffic load of each output road segment, while recommending the next hop for vehicles traveling towards their destinations. Some vehicles need to travel additional time and/or distance in their trips to avoid causing traffic congestion over the road network. On the other hand, the Abs-Bal mechanism aims to completely distribute the input traffic of each road intersection in an absolute even manner among output road segments. The Abs-Bal mechanism achieves the best performance in terms of decreasing the traffic density of each output road segment.

Chapter 6

Intelligent Traffic Light Controlling Algorithm

6.1 Introduction

In this chapter, we propose the Intelligent Traffic Light Controlling (ITLC) algorithm, so as to efficiently schedule the phases¹ of traffic lights located at isolated signalized road intersections. Furthermore, an Arterial Traffic Lights Controlling (ATLs) algorithm has been adapted from ITLC for arterial street scenarios. At each road intersection, the intelligent traffic light gathers the real-time traffic characteristics of input traffic flows that intend to cross the respective signalized intersection. Then, the phases of the timing cycle² at the traffic light are scheduled so that the highest density traffic flow crosses the road intersection first. On the other hand, the ATLs algorithm is adapted in order to schedule the phases of each traffic light on the arterial street scenario. In ATLs, the intelligent traffic lights installed at each road intersection coordinate with each other to generate an efficient schedule for each traffic light over the signalized road network. The details of ITLC and ATLs are presented in this chapter. Moreover, we report on the performance of ITLC and ATLs compared to previous intelligent traffic light scheduling algorithm for several traffic scenarios using NS-2 [64].

¹different colors signals

²cycle of successive phases and time setting of each phase

6.2 Controlling Traffic Light Scenarios

Traffic lights are used to control and schedule competing traffic flows at each road intersection. In order to design an efficient traffic light controlling algorithm, the applicable operational concepts of the signalized road intersection are first investigated. In this section, we present the definition, properties and characteristics of isolated traffic lights and signalized road networks.

6.2.1 Isolated Traffic Light

Isolated traffic lights separately control the competing traffic flows at each road intersection. These traffic lights never consider the schedule of neighboring signalized intersections over the road network. For example, Figure 6.1 illustrates a typical 4-leg road intersection and the primary phasing options on a 4-leg road intersection. The 4-leg road intersection is shared among eight flows of traffic, two of which can proceed simultaneously through the road intersection (i.e., without conflicting with each other).

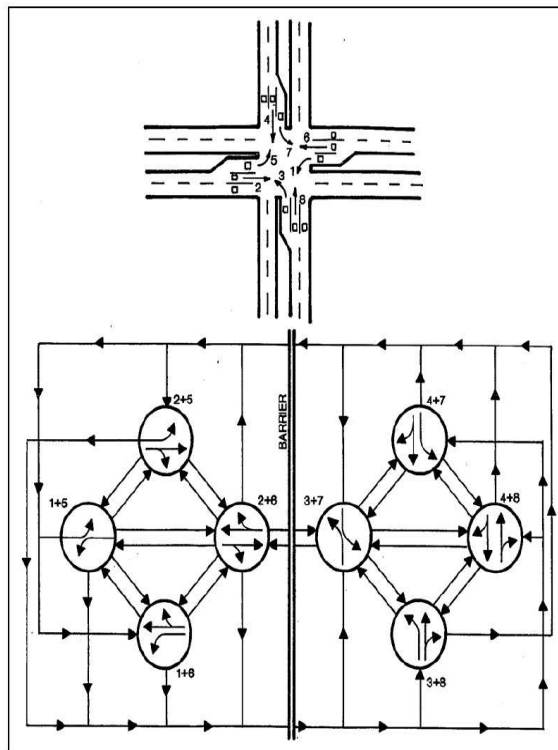


Figure 6.1: Primary phasing options for 4-leg intersection [85].

Table 6.1: Traffic Light Timing Variable Definitions [85]

Variable	Definitions
Cycle Length	The time required for one complete sequence of signal intervals (phases).
Phase	The portion of a signal cycle allocated to any single combination of one or more traffic movements simultaneously receiving the right-of-way during one or more intervals.
Interval	A discrete portion of the signal cycle during which the signal indications (pedestrian or vehicle) remain unchanged.
Split	The percentage of a cycle length allocated to each of the various phases in a signal cycle.
Offset	The time relationship, expressed in seconds or as a percent of cycle length, determined by the difference between a defined point in the coordinated green light and a system reference point.

The traffic light at such an intersection controls and schedules the sequence of successive phases of each timing cycle; meanwhile, it illuminates the conflict between the competing traffic flows. At each traffic light, the timing cycle variables, which include cycle length, phases, split intervals and offset parameters, should be set efficiently in order to enhance the traffic fluency over the road network [85]. Table 6.1 summarizes the definitions of these variables citeBOOK.

In general, Figure 6.1 illustrates all the options of the primary phasing for any timing cycle in a 4-leg intersection scenario. Variable-sequence phase and/or skip-phase techniques can be applied to enhance the efficiency of the traffic as well. In the typical controlling mechanisms (i.e., Pre-timed mechanisms), the cycle length, the length of each phase, the number of phases, and the sequence order of the phases are set permanently for each traffic light. Engineers usually use historically gathered traffic characteristics for the area of interest to select the optimal values for these parameters [85]; they set these values as they install the traffic lights.

On the other hand, traffic-actuated and volume-density systems have been used to control the timing values of each phase according to the real-time traffic characteristics of the competing traffic flows [85]. Real-time traffic detectors are required to develop these traffic light control systems efficiently for each isolated traffic light. The semi-actuated control system is used at intersections where a minor street has traffic volumes there are significantly lower than the major (i.e., arterial) street. This controlling system aims to minimize the interruption of traffic on the major street, while still providing adequate service to the minor street [86]. On the other hand, the fully-actuated system is employed when both streets at the intersection have relatively equal volumes and equal priorities. Based on the characteristics of the competing flows of traffic at the road intersection, the

initial green time and interval extensions are set for each phase in the timing cycle of the traffic light. The actuated system assigns a minimum green time to each phase, without a vehicle actuation [86]. Finally, the volume-density control system introduces the ability to reduce or to increase the vehicle extension intervals depending on the volume of the opposing traffic flows [86].

The interval time of the green and red phases in traffic-actuated and volume-density controlling systems are set mathematically. These parameters are set according to traffic speed, the capacity of the road intersection, the orderly traffic movement and the traffic volume of the competing flows [85], [86]. Accurate and real-time traffic characteristics of competing traffic flows are mainly required to obtain the best schedule for each traffic light.

6.2.2 Signalized Road Networks

In any signalized road network, the set of traffic lights installed over the road network coordinate with each other to achieve network-wide traffic operation objectives. In this section, we investigate the coordination between located traffic lights, in order to achieve the operational objectives of arterial street and mesh-alike road networks. The main objective of the arterial street scenario is to provide preferences for the progressive traffic flow along arterial flows, as illustrated in Figure 6.2(a). Cooperative scheduling among intelligent traffic lights, all over the arterial street, helps achieve these preferences. First, the traffic over the arterial street is divided into a set of successive platoons (i.e., a group of vehicles moving together). The green time of the arterial flow at each intersection is set to maintain the influence on the flow of these platoons over the arterial street.

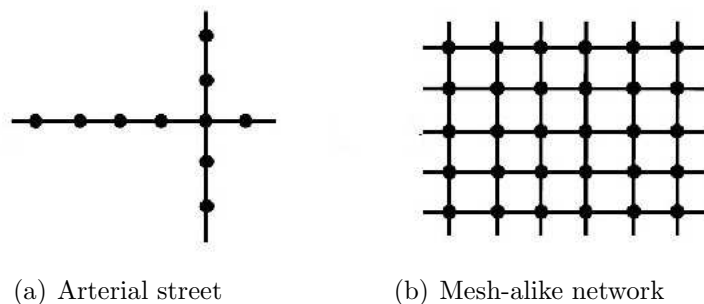


Figure 6.2: Signalized road networks [85].

The time relationship is established between the beginning of the green time at one

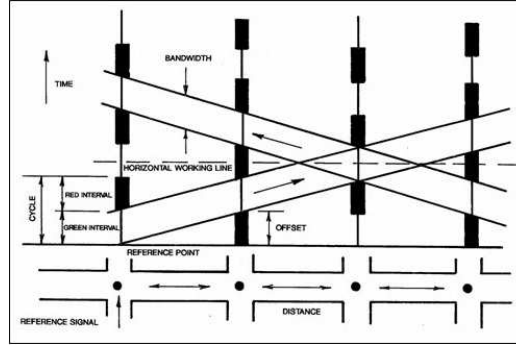


Figure 6.3: Optimal scheduling case for arterial street [86].

intersection and the beginning of the green time on the next intersection for the same flow of traffic (i.e., arterial flow). This time relationship can be set based on the distance between the adjacent road intersections and the average speed of vehicles on the connected road segment [85], [86]. Each traffic light on the arterial street should receive a green indication, before the platoons (i.e., a set of vehicles moving together) have arrived, in order to schedule the sequence of phases cooperatively with these previous traffic lights. This permits continuous and smooth traffic flow and reduces the queuing delay all over the arterial street. Figure 6.3 illustrates an example of the optimal scheduling case for a typical two-sided arterial street scenario that includes four road intersections [86]. In Figure 6.3, the timing cycle of each traffic light consists of two alternative phases: red and green. The throughput of the system represents the number of vehicles that can pass through the successive intersections smoothly (i.e., without stopping at any road intersection on the arterial street).

Accurate and real-time green indications are required between successive intersections. These indications concern traffic volume and the estimated arrival times of each platoon at the next road intersection. Cooperation, between located traffic lights on the arterial street, helps produce an efficient traffic schedule for the entire arterial street. It decreases the queuing delay time of traveling vehicles and enhances the throughput of road intersections, especially for over-saturated road network scenarios.

In the mesh-alike road network scenario, all flows of traffic are treated as arterial flows, as illustrated in Figure 6.2(b). In general, when two arterial streets cross each other at a certain road intersection, a signal timing interlock must take place in order for progression to occur through both flows of traffic [86]. The same cycle length and the timing plan must be used as a reference point for the timing at all signalized intersections in this scenario.

6.3 Intelligent Traffic Light Controlling Algorithm

In this section, we design an intelligent traffic light controlling (ITLC) algorithm for isolated road intersections. This algorithm specifies the sequence phases of each traffic light cycle and the time assigned to each phase. These parameters are set according to the real-time traffic characteristics of all competing traffic flows that intend to cross the signalized intersection. We then adapt the ITLC algorithm for the arterial street scenario by introducing an Arterial Traffic Lights (ATLs) controlling algorithm. ATLs sets the sequence phases of each traffic light cycle and the time of each phase according to the real-time traffic characteristics of competing traffic flows; it also considers the scheduling reports of the neighboring traffic lights located on the arterial street. Traveling vehicles over the arterial flows help deliver the scheduling reports among the traffic lights using the VANETs technology.

6.3.1 Intelligent Scheduling Algorithm for Isolated Traffic Light

The ITLC algorithm considers the real-time traffic characteristics of all competing traffic flows at isolated road intersections. ITLC simulates the signalized road intersection as a shared processor among eight flows of traffic, where two flows can be processed together at any time (i.e., non-conflicting flows can pass the road intersection simultaneously). Vehicles arrive at the road intersection at different estimated times; thus, each flow of traffic can be considered as a set of successive processes (i.e., platoons). Each platoon contains one or more vehicles that intend to cross the road intersection during the same green phase of the traffic light.

Ready Area

A virtual square area around each signalized road intersection is configured as the ready area in ITLC. Vehicles located inside this area are ready to cross the intersection. The distance from the traffic light to the boundary of the ready area (L_{ra}) is set based on the average traffic speed of all competing flows of traffic, and on the maximum allowed green time of the traffic light. Figure 6.4 illustrates an example of a ready area at a signalized road intersection. The ready area divides the successive platoons in each flow of traffic. Thus, it guarantees a fair usage of the road intersection among the competing flows of traffic. In the case where the traffic light assigns a red light signal to a certain flow of traffic, vehicles at such a traffic flow decrease their speed in preparation to stop inside

the ready area, at the closest available spot to the traffic light. For each flow of traffic, all vehicles located inside the ready area during the data gathering phase are considered to be in the same platoon of traffic that should pass through the road intersection at the same green phase of this flow.

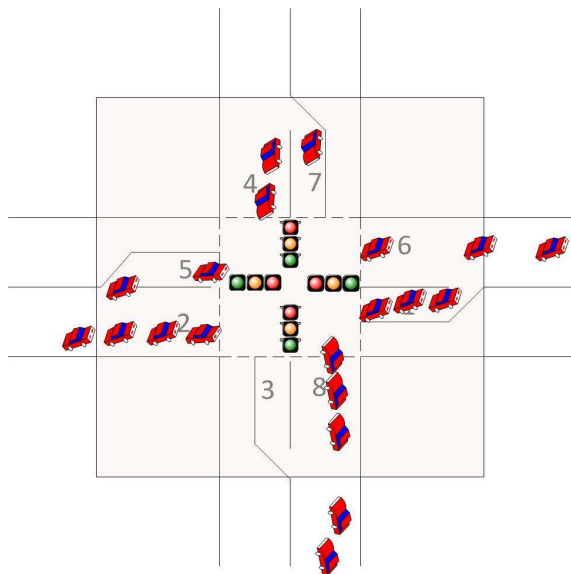


Figure 6.4: Ready area at signalized road intersection.

The period of time (T) for the green phase of each process should not exceed the maximum green time limit of the traffic light. In other words, all vehicles located inside the ready area during the data gathering phase should be able to pass the intersection during the maximum green time of the traffic light. T is computed by using Equation 6.1:

$$T = \theta + \frac{F_d}{S_{tf}} \quad (6.1)$$

where θ is a constant value that accounts for the startup delay of the very first vehicle in each platoon (i.e., process). F_d is the distance between the furthest vehicle in the configured platoon and the road intersection. Finally, S_{tf} is the speed of the traffic flow of this platoon.

Traffic Characteristics of the Competing Flows of Traffic

The traffic density (d_i), traffic speed (s_i) and estimated travel time (t_i) are computed for each traffic flow inside the ready area, according to our traffic evaluation and congestion detection proposed protocol (i.e., ECODE), which is presented in Chapter 3. Each

vehicle periodically broadcasts its basic traffic data (i.e., speed, location, direction, etc). Vehicles receive basic traffic data messages from neighboring vehicles; they record these messages combined with the received time. Moreover, each vehicle uses the digital map to determine which traffic flow it is located in, according to the coordinates of its location.

On the other hand, the located traffic lights periodically announce the boundaries of the ready areas according to the real-time traffic speed evaluated for the competing flows of traffic. Vehicles can verify whether they are located inside the ready area or not, after receiving this boundary information message. Then, only vehicles located inside the ready area use the gathered traffic data of the neighboring vehicles to compute the traffic characteristics of the platoon of vehicles located inside this area for each flow of traffic. Traffic characteristics include the number of vehicles in the platoon, traffic density (d_i), traffic speed and the required time to cross the intersection (T). Vehicles at each platoon reliably deliver the traffic characteristics of such a platoon to the respective traffic light. The traffic characteristics of each platoon are mainly used in the scheduling algorithm (i.e., ITLC) to set the sequence of phases and to assign the time of each phase efficiently.

The Scheduling Algorithm

As illustrated in Figure 6.1, eight options for the double traffic flow passing are available for each phase: P_{15} , P_{25} , P_{16} , P_{26} , P_{37} , P_{47} , P_{38} and P_{48} . Figure 6.5 illustrates these phases graphically, following the numbering sequence that was used in Figure 6.1 [85]. Four phases out of eight are selected in each traffic light timing cycle; however, all competing flows are scheduled through these phases. For example, if phases P_{15} and P_{26} are scheduled to pass, we should not schedule P_{25} and P_{16} during the same timing cycle of the traffic light because the vehicles in these phases have already been scheduled through the chosen phases (i.e., P_{15} and P_{26}).

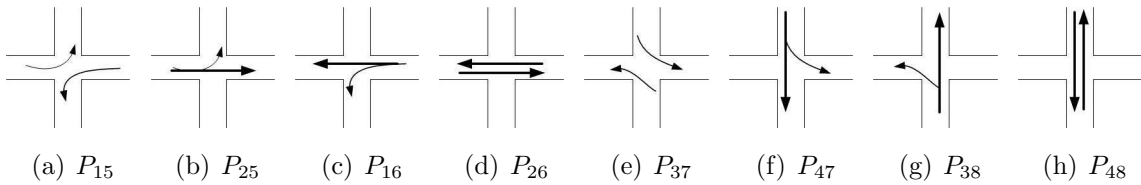


Figure 6.5: The options of phases for each traffic light.

In general, the schedule of all traffic flows start with the phase that has the largest platoon density ($Max(d_i)$). The ITLC algorithm schedules the eight competing flows of various platoons, with the largest density assigned first. Algorithm 8 systematically

Algorithm 8: Intelligent Traffic Light Scheduling Algorithm

Data: TL : traffic light; RA : ready area; d_i : the traffic density of the traffic flow i inside RA ; t_i : the required time for all vehicles inside RA , at the traffic flow i , to cross the traffic intersection.

```

1 compute  $d_i$  and  $t_i$  of all traffic flows inside  $RA$ ;
2 while  $d_i$  of any of the traffic flows at  $TL > 0$  do
3   let  $j$  the traffic flow with the maximum traffic density ( $d_j$ );
4   let  $i1$  and  $i2$  the traffic flows that can cross the traffic intersection simultaneously with the traffic
   flow ( $j$ ) ;
5   if  $d_{i1} > d_{i2}$  then
6      $P_{ji1} = \text{schedule}(j, i1)$ ;
7      $d_{i1} = 0.0$ ;  $t_{i1} = 0.0$ ;
8   else
9      $P_{ji2} = \text{schedule}(j, i2)$ ;
10     $d_{i2} = 0.0$ ;  $t_{i2} = 0.0$ ;
11  end
12   $d_j = 0.0$ ;  $t_j = 0.0$ ;
13  Adjust the  $t_k$  of all other traffic flows inside the ready area;
14 end

```

illustrates the schedule procedure. Two traffic flows are candidates that proceed simultaneously with the largest platoon density flow, *Algorithm 8 lines 3-4*. For example, if the traffic flow 1 has the largest platoon density, one of the traffic flows, either 5 or 6, is eligible to proceed with 1 (i.e., P_{15} and P_{16}). The traffic flow that has the largest platoon density among these two eligible flows is selected to proceed in the same phase with the largest platoon density (i.e., flow 1). However, the time of each phase is set based on the farthest vehicle inside the ready area of each traffic flow in the selected phase. As illustrated in Algorithm 9, since two flows of traffic are included in each phase, the longest required time (T) is selected for the respective phase schedule.

After a certain scheduling phase passes through the road intersection, the traffic density (d_i) and the required travel time (T_i) of each flow in the phase are reset to zero. This is done in the next iterative execution of the scheduling algorithm, until all competing flows of traffic are scheduled. The required time for each flow (T_i) is adjusted to subsequent traffic flows according to the new estimated location and speed of the farthest vehicle in each flow, *Algorithm 8 line 13*. Empty flows are assigned the time of zero for scheduling which eliminates the phase. When all vehicles, detected inside the ready area, pass the traffic intersection successfully, the process of evaluating the traffic distribution in the ready area and the scheduling algorithm phases are executed repeatedly for recently arrived vehicles.

ITLC is expected to increase the throughput of the road intersection because it assigns highest priority to the flow with the largest number of vehicles or with the highest traffic density. In each timing cycle, a fewer number of vehicles have to wait in the conflicted

traffic flows during each scheduled phase which decreases the average delay time per vehicle at the traffic light. The scheduled time for each phase is adjusted based on the location of the furthest vehicle in the platoon, which decreases the waiting delay time. Moreover, no vehicle should wait for empty flows because the assigned schedule times of these flows are set to zero.

Algorithm 9: Schedule Function

```

1 INPUT: traffic flows  $i$  and  $j$ 
2 if  $t_i > t_j$  then
3   |   return  $t_i$ 
4 else
5   |   return  $t_j$ 
6 end

```

6.3.2 Arterial Traffic Lights Controlling Algorithm (ATLs)

In general, a shorter distance between neighboring traffic lights on the signalized road network increases the importance of coordination between them. Along a corridor, traffic signals should be coordinated within 800 meters of each other unless they operate on different cycle lengths [86]. In arterial street scenarios, traveling vehicles usually move in a platoon fashion (i.e., each set of vehicles stay close to each other during their trips). During the green phase of any traffic light, a number of vehicles cross the road intersection towards the next road intersection over the arterial street. Several studies have been introduced aiming to produce an efficient scheduling algorithm for signalized road networks. Most of the previously proposed algorithms that have targeted the arterial street scenario aimed to provide a schedule for traffic lights located on arterial streets, without considering other traffic lights in the downtown areas.

Here, we adapted the ITLC algorithm to be used in the arterial street scenario and named it ATLs. In ATLs, intelligent traffic lights, installed at each road intersection, consider the scheduling reports of neighboring traffic lights. In addition, they consider the characteristics of the traffic flows that intend to cross the respective road intersection. Each traffic light weighs the received schedule reports of neighboring traffic lights, and prioritizes them besides the real-time traffic characteristics of competing flows at the respective road intersection (i.e., traffic density). Figure 6.6 illustrates the considered parameters of each intelligent traffic light, in ATLs. As illustrated in Figure 6.6, ATLs considers the reported data concerning the expected platoons from previous traffic lights over the arterial street. At the same time, it considers the real-time traffic reports of

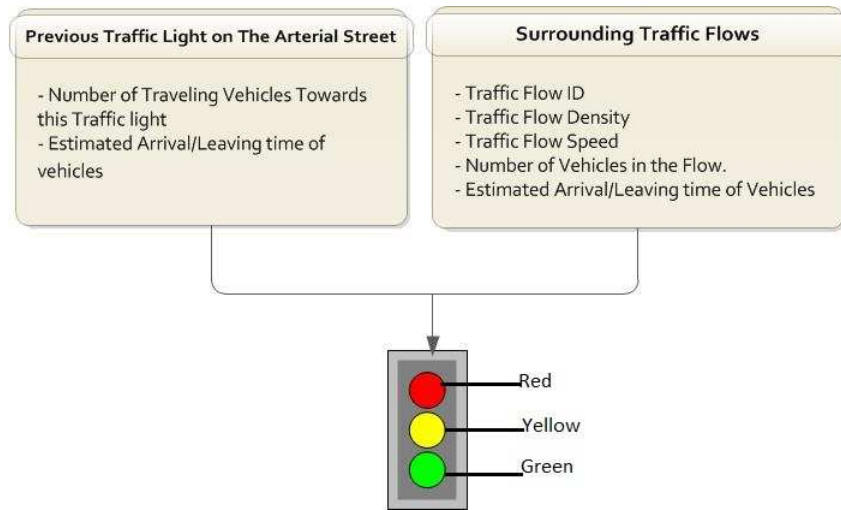


Figure 6.6: Parameters of traffic lights controlling algorithm on arterial streets.

competing flows of traffic at the road intersection. This algorithm intends to guarantee a green-wave over the arterial street, without disturbing other traffic flows on the entire road network.

Figure 6.7 shows a typical grid-layout example of the targeted scenario for ATLs. In this scenario, the arterial street is identified as one of the prominent streets in a grid layout of the downtown area. Each intelligent traffic light located on the arterial street reports the traffic schedule phases of surrounding flows to its neighboring traffic lights using the traveling vehicles over the connecting road segment. Considering the scheduling traffic report of the neighboring traffic lights introduces cooperative behavior between the located traffic lights.

Each intelligent traffic light, located on the arterial street, sends a reporting message (RM) to its neighbor traffic lights, located at the same arterial street. The reporting message (RM) contains: a) the number of traveling vehicles (N) in the incoming platoon moving towards the receiver traffic light; b) the starting cross time (S_{time}) of the respective platoon; c) the ending cross time (E_{time}) of the respective platoon. Traffic lights, located on the arterial street, initially use Algorithm 8 (i.e., ITLC) to generate the schedule phases at each traffic light. For the following timing cycles, an adaptive systematic procedure is introduced in Algorithm 10 (ATLs).

The ready area concept needs to be further adapted for use according to the distance between neighboring traffic lights. Figure 6.8 illustrates four different cases of the ready area settings around two successive road intersections on an arterial street scenario. As

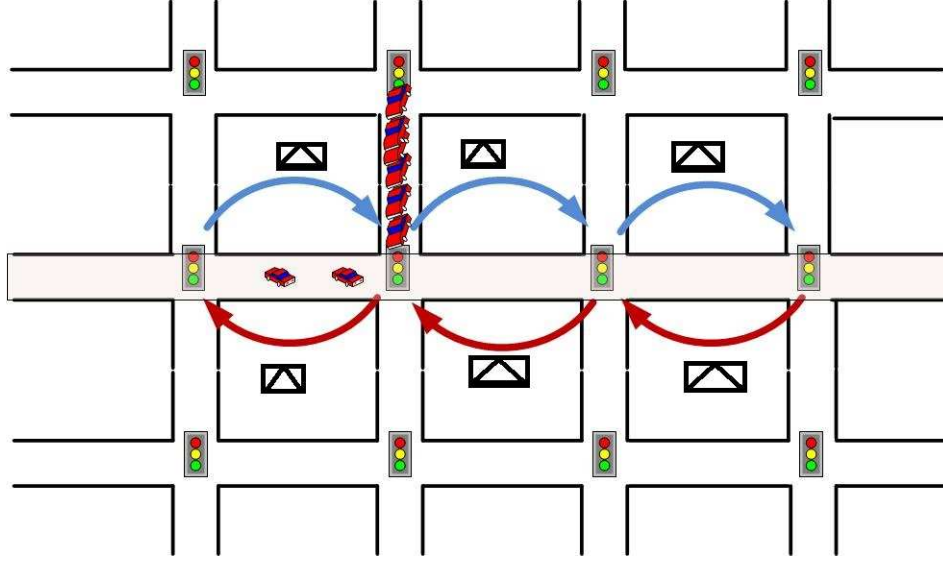


Figure 6.7: An arterial street in a downtown area.

we can see in Figure 6.8(a) and Figure 6.8(b), the ready areas overlap, however, in Figure 6.8(c) and Figure 6.8(d), the ready areas are completely separated. These cases can affect the saturation factor of the arterial road segment in between adjacent road intersections. In ATLs, we consider these cases based on the density of each traffic flow, and based on the saturation density of the flow by presenting the saturation density factor (SDF). Equation 6.2 computes the saturation density factor (SDF_i) of each traffic flow (i). SDF_i is used to schedule the phases of each cycle in any traffic light located on an arterial street.

$$SDF_i = \frac{d_i}{Sd_i} \quad (6.2)$$

where d_i is the density of the traffic flow i and Sd_i is the density required to completely saturate the traffic flow i . The value of SDF_i varies in the interval $[0.0, 1.0]$, where it can be zero in the case that no cars are located on the traffic flow i . On the other hand, SDF_i can be one in the case that the current traffic density (d_i) is equal to Sd_i .

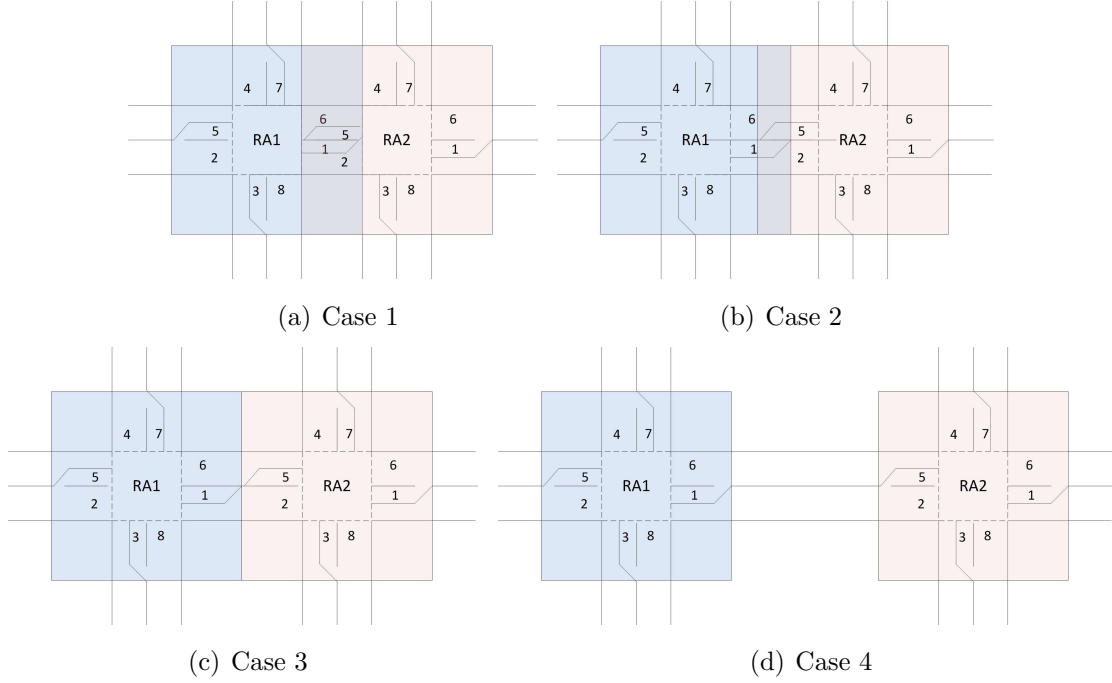


Figure 6.8: Ready areas of adjacent road intersections in arterial street cases.

The other factor to consider in Algorithm 10, where there is a need to find efficient scheduling phases, is the arterial factor (AF_i). AF_i is set to zero for any traffic flow that is not located on arterial street coordinations. However, for traffic flows located on the arterial street, AF_i is set to one. Equation 6.3 combines the saturation density factor (SDF) and the arterial street factor (AF) of each traffic flow i . In this equation, α is used to weigh the importance of the arterial factor compared to the saturation density.

$$Cf_i = \alpha \cdot AF_i + (1 - \alpha) \cdot SDF_i \quad (6.3)$$

ATLs uses the combined factor (Cf_i), computed by Equation 6.3, to schedule the competing flows at each traffic light. AF_i assigns a high priority for any traffic flow that has more than α saturation traffic density (SDF) and enables it to pass first over the arterial traffic flow, while the arterial flow waits, *Algorithm 10 line 10*.

As previously stated, a shorter timing schedule facilitates an efficient scheduling algorithm, as well as a smaller number of vehicles waiting for a green light signal. A smaller number of phases in each cycle produces a shorter time cycle, and thus a more efficient schedule. Because of this, we aim to schedule the arterial flow at the traffic light only once in each timing cycle. As we can see from *Algorithm 10 line 8*, we exclude the arterial flows from the scheduling processes; and, schedule the rest of the phases

Algorithm 10: Arterial Traffic Lights Controlling Algorithm (ATLs)

Data: TL : traffic light; RM : reporting message; N : number of vehicles crossing the previous TL ; S_{time} : starting cross time of the previous TL ; E_{time} : ending cross time of the previous TL ; AF : arterial flow; A_{time} : estimated arrival time; L_{time} : estimated leaving time; $LCFT$: last cycle finishing time; $NCFT$: next cycle finishing time.

```

1 use Algorithm 8 to schedule the initial cycle phases of  $TL$ ;
2 send  $RM$  of  $AF$  to the next  $TL$  on the arterial street;
3 when  $TL_1$  receives  $RM$  from  $TL_2$ , it computes:
4  $A_{time} = S_{time} + distance(TL_1, TL_2) / S_{tf}$ ;
5  $L_{time} = E_{time} + distance(TL_1, TL_2) / S_{tf}$ ;
6 if  $A_{time} < NCFT$  then
7   while  $CurrentTime < A_{time}$  do
8     use Algorithm 8 to schedule the next cycle phases of  $TL$ , without considering the  $AF$ ;
9   end
10  Use Equation 6.3 to compute the  $Cf_i$  of each competing flow  $i$ ;
11  while  $Cf_i$  of any of the traffic flows at  $TL > 0$  do
12    let  $j$  the traffic flow with the maximum combined factor ( $Cf_i$ );
13    let  $i1$  and  $i2$  the traffic flows that can cross the traffic intersection simultaneously with the
    traffic flow  $j$  ;
14    if  $Cf_{i1} > Cf_{i2}$  then
15       $P_{ji1} = \text{ART-schedule}(j, i1)$ ;
16       $Cf_{i1} = 0.0$ ;  $t_{i1} = 0.0$ ;
17    else
18       $P_{ji2} = \text{ART-schedule}(j, i2)$ ;
19       $Cf_{i2} = 0.0$ ;  $t_{i2} = 0.0$ ;
20    end
21  end
22 else
23    $Cf_j = 0.0$ ;  $t_j = 0.0$ ;
24   use Algorithm 8 to schedule the next cycle of phases of  $TL$ ;
25 end

```

using Algorithm 8; this is possible as long as the platoon of vehicles traveling from the previous traffic light has not yet arrived the signalized road intersection. The late arrival of arterial flows can be noticed in low saturated network scenarios or in the case where long distances exist between the adjacent intersections, such as Case 4 in Figure 6.8(d).

Algorithm 11: Arterial Schedule (ART-schedule) Function

Data: AF : arterial flow; A_{time} : estimated arrival time; L_{time} : estimated leaving time.

```

1 if  $i$  is an  $AF$  then
2   | if  $t_i < (L_{time} - A_{time})$  then
3     |    $t_i = L_{time} - A_{time}$ ;
4   | end
5 end
6 if  $t_i > t_j$  then
7   |   return  $t_i$ ;
8 else
9   |   return  $t_j$ ;
10 end

```

In the case where the platoon traveling from the previous traffic light is not expected during the next timing cycle, the phases of the cycle are scheduled according to the largest density first, using Algorithm 8 (ITLC). However, the traveling platoons are considered upon their expected arrival. As illustrated in Algorithm 10, the arterial traffic flows are considered after computing the combined factor (CF_i) of all competing traffic flows using Equation 6.3. The scheduling time assigned to each phase is the same as in Algorithm 9, and each phase is assigned the greatest required time between selected flows. This is done based on the location of the farthest vehicle in each flow of traffic inside the ready area. However, in some cases, the green time limit for the arterial flow is extended, to allow all vehicles in the configured platoon to pass through the road intersection together. The scheduling times of the arterial flows are illustrated in a systematic manner in Algorithm 11. In this case, the assigned time considers the reported arrival and departure time of neighboring traffic lights on the arterial street.

6.4 Performance Evaluation of ITLC and ATLS

In this section, we shall evaluate the performance of ITLC and ATLS. In order to investigate the advantages of ITLC and ATLS, we compared them respectively to previously proposed techniques, including OAF [90] for isolated intersection and ART-SYS [109] for arterial street scenario.

6.4.1 Performance Evaluation of ITLC Algorithm

Here, we evaluate the performance of ITLC, with isolated traffic lights. First, we used SUMO [83] to generate several mobility scenarios. In these scenarios, traveling vehicles aim to pass a road intersection that is controlled by an intelligent traffic light. A different number of vehicles have been generated to study the effect of traffic density over the road network on the performance of the proposed algorithms.

Table 6.2: Simulation Parameters of ITLC

Parameter	Value
Simulator	NS-2
Transmission range (m)	250
Simulation time (s)	2000
Simulation area (m^2)	1000 X 1000
Number of traffic lights	1
Number of vehicles	200 - 1000
Simulation map	4-leg traffic intersection

We have compared the performance of ITLC to a previously developed adaptive traffic signal control mechanism (OAF) [90]. This mechanism (i.e., OAF) claimed a better performance over the VANET-enabled vehicle-actuated control, the VANET-enabled Webster's and VANET-based fixed-time signal control, in terms of its ability to decrease the average waiting delay time for each vehicle at signalized isolated road intersections.

In our experiments, we first compare the queuing delay time of each vehicle located inside the ready area at the investigated signalized road intersection. We then compare the total delay, required for all vehicles in the road network, in both scheduling algorithms. Finally, we aim to evaluate the throughput of each compared mechanism, by counting the number of vehicles that cross the intersection per second. Table 6.2 illustrates the main parameters that have been used in these experiments.

Figure 6.9 illustrates the comparative performance of ITLC and OAF [90]. Each experiment has been executed for 30 different scenarios. The confidence interval for each experiment is 95%. Figure 6.9(a) illustrates the average delay per vehicle inside the ready area; ITLC decreases the average delay for each traveling vehicle by 30% compared to OAF [90]. The delay time required for detected vehicles to cross an intersection at a given time in ITLC was 25% less than the total required delay in OAF; this is as illustrated in Figure 6.9(b). From Figure 6.9(a) and Figure 6.9(b), it can be seen that ITLC decreases the queuing delay time of traveling vehicles by an average of 25% less than OAF for any isolated road intersection.

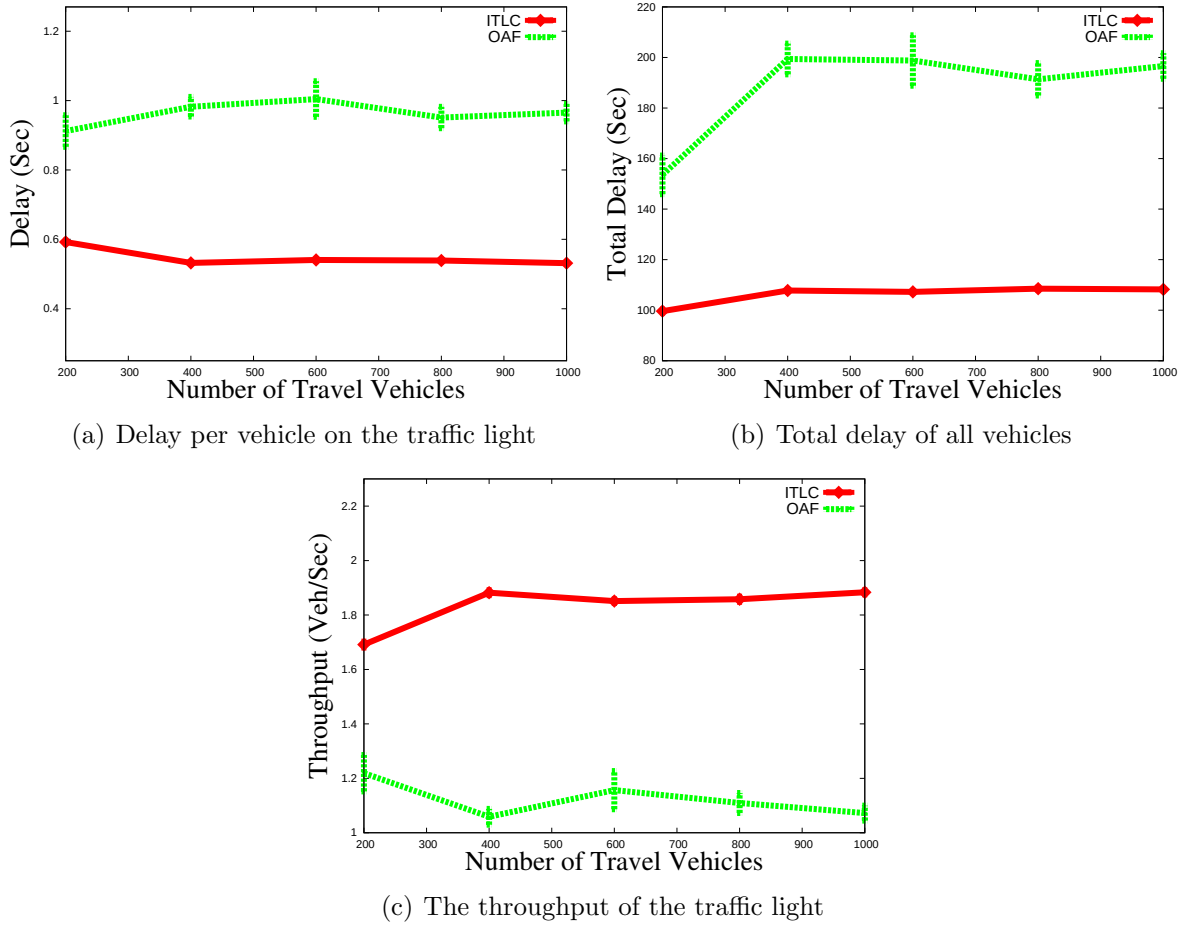


Figure 6.9: The performance evaluation of ITLC compared to OAF [90].

On the other hand, Figure 6.9(c) illustrates the throughput of an isolated road intersection that utilizes ITLC and OAF to schedule the phases of the respective traffic light. As Figure 6.9(c) indicates, the number of vehicles passing the road intersection per second while using the ITLC algorithm is 30% higher than the number of vehicles passing the intersection per second while using the OAF algorithm. This is due to the scheduling principle of ITLC, in which the flow of traffic that obtains the highest density inside the ready area is scheduled to pass the road intersection first. This mechanism assigns highest priority to the traffic flow with the highest density and the largest number of vehicles over other flows of traffic. A smaller number of vehicles also need to wait in conflicted flow at each scheduled phase of traffic. Moreover, the assigned time for these phases is adjusted regularly, according to the real-time location of the farthest vehicle inside the ready area at each flow of traffic.

6.4.2 Performance Evaluation of ATLS Algorithm

In this section, we evaluate the performance of the adapted ATLS algorithm compared to previously proposed intelligent traffic light controlling algorithms that have been introduced for arterial street scenarios (i.e., ART-SYS). We have used SUMO [83] to generate different mobility scenarios with different traffic densities. Table 6.3 illustrates the main parameters that have been used in these experiments. We have executed each experiment for 30 different scenarios, and we obtained a confidence interval of 95%.

Table 6.3: Simulation Parameters of ATLS

Parameter	Value
Simulator	NS-2
Transmission Range (m)	250
Simulation Time (s)	2000
Simulation Area (m^2)	1000 X 1000
Number of Traffic Lights	16
Number of Vehicles	200 - 1200
Simulation Map	4 X 4 grid-layout
Road Segment Length (m)	200
Ready Area Side Length (m)	100

In order to find the best weight of α in the combined formula (i.e., Equation 6.3), an extensive set of experiments over the investigated environment were executed. The average queuing delay time at each traffic light in the network is investigated in Figure 6.10(a). As seen in the Figure, the average delay at each traffic light in the system increases when the weight of the arterial factor (α) assigns a higher value. This is justified by the larger number of vehicles that have to wait on the conflicted traffic flows, while most of the time the highest priority for proceeding through the road intersection is given to the arterial flow. On the other hand, as we can see from Figure 6.10(b), a shorter delay time is experienced in the arterial flow when it uses a higher arterial factor. However, for the large values of the weight of AF_i (α), the performance of ATLS remains the same as the performance of previously proposed systems for arterial street scenario. In those systems, the arterial flow always has the highest priority over all other competing flows; all flows of traffic have to wait for the arterial flow to pass first, regardless of their saturation level [109].

From Figure 6.10(b), we can see that for high network densities, given that most roads in the network are saturated or almost saturated, the AF_i has a rational effect only if its weight (α) is less than 50%. No traffic flow was detected that had a 60%, 70%

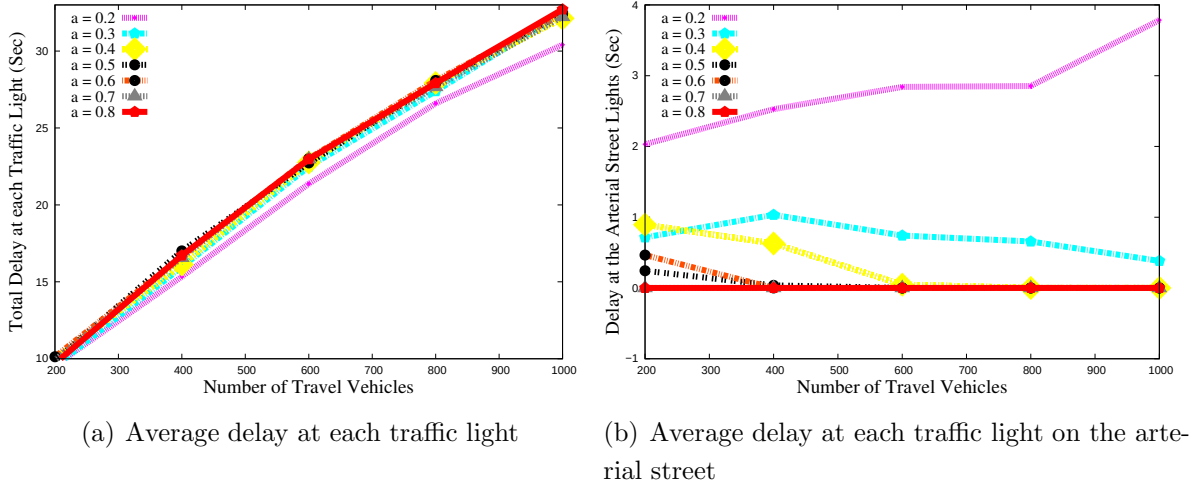


Figure 6.10: The arterial factor.

or 80% higher saturation factor than any traffic light in the investigated scenario. We can see from Figure 6.10 that ATLs with 20% AF_i weight performs the best in terms of decreasing the waiting delay on the arterial street scenario, without drastically increasing the average delay time of each traffic light in the road network system. This is the case for all network density scenarios.

In order to investigate the advantages of ATLs algorithm, we have compared the performance of ATLs to two algorithms, Random and ART-SYS [109]. Both were developed to control the schedule of each traffic light in the open road network. In these comparative experiments, we set the weight of AF_i in ATLs (α) to 20% for each traffic flow on each traffic light. In the Random system, each traffic light generates its schedule without considering the schedules of neighboring traffic light reports. It performs as a set of isolated traffic lights located on the road network. In ART-SYS [109], the highest priority to proceed is always assigned to the traveling vehicles on the arterial flow. Each traffic light located on that street schedules the traffic flow on the arterial street so that they may proceed at the moment when a platoon of vehicles arrives from the previous road intersection. This algorithm guarantees a high fluency for vehicles on the arterial street; meanwhile, it causes more delay for vehicles traveling on other traffic flows all over the road network, when compared to the Random algorithm.

Figure 6.11 illustrates the comparative performance evaluation of ATLs to previously proposed systems. The total waiting delay at each traffic light in the investigated road network is the highest for ART-SYS [109]. ATLs decreases the total delay at each traffic

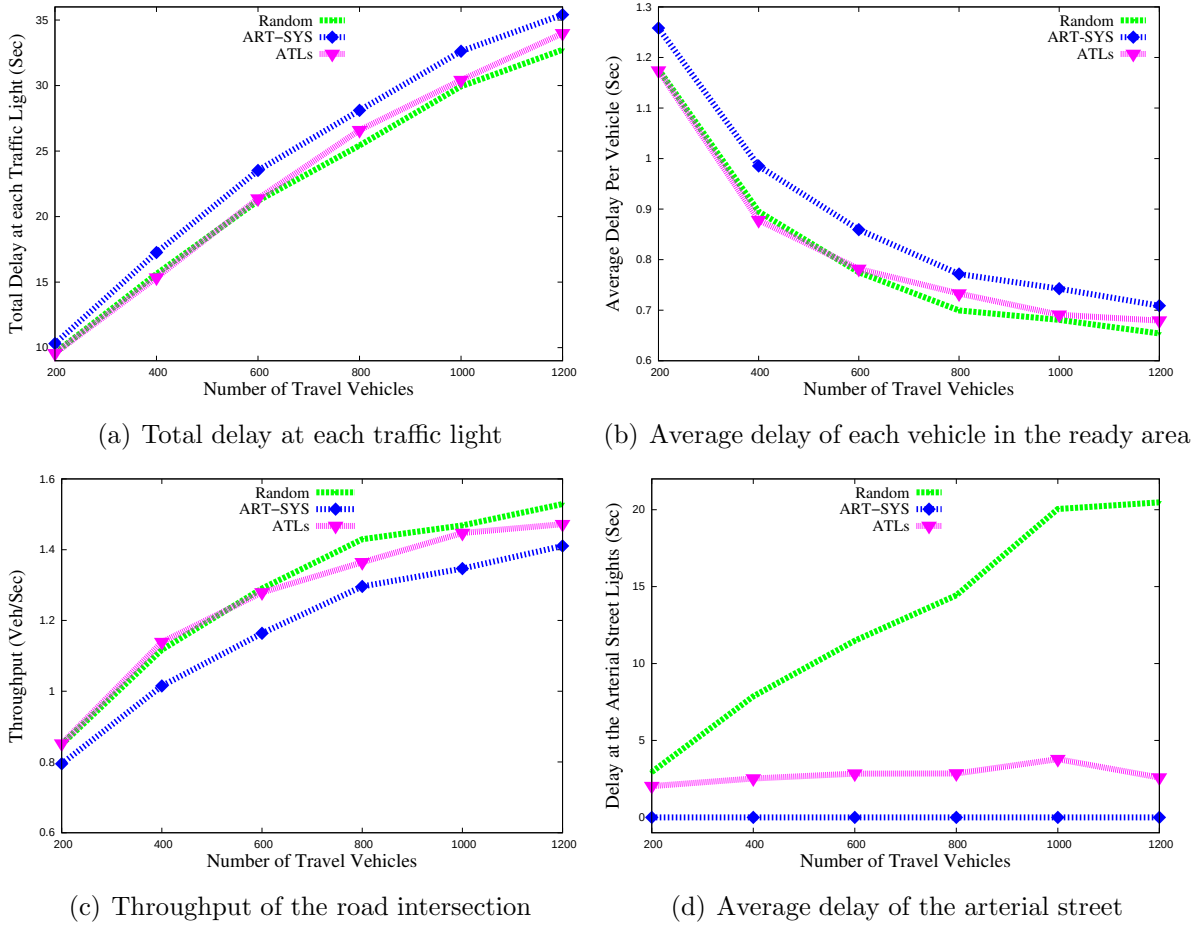


Figure 6.11: The performance evaluation of ATLs compared to ART-SYS [109] and random system.

light by 10% compared to ART-SYS [109]. However, ATLs increases the total delay by 1% compared to the Random mechanism. Figure 6.11(a) graphically illustrates these results: we see that ATLs allows the arterial flows to pass ahead of other traffic flow with higher density. Figure 6.11(b) illustrates the average delay for each traveling vehicle inside the ready area of each traffic light. As we can infer from Figure 6.11(b), ATLs decreases the average delay for each vehicle by 7% compared to ART-SYS [109]. At the same time, ATLs also increases the delay of each vehicle by 1% compared to the Random mechanism. Another interesting result can be seen in Figure 6.11(b), where vehicles in the ready area experience shorter delay times when the network is more saturated. This is due to the distribution of vehicles over road networks where larger platoons are constructed to cross the signalized road intersection at each green phase. In this case, the timing cycle of

each traffic light lengthens, which increases the average delay for traveling vehicles in the following schedule cycles. Moreover, since the ready area of any road intersection has a limited size and thus a maximum capacity, the graph in Figure 6.11(b) demonstrates a movement towards stable behavior by greatly increasing the traffic density on the road network. This positively affects the timing cycle of each traffic light; a maximum length can be predicted for each traffic light timing cycle.

The throughput of ART-SYS [109] is less than the throughput of the Random system. In ART-SYS the priority is always given to vehicles on the arterial flow, even when the traveling platoons are small. Vehicles on other traffic flows must wait, even if the number of these vehicles is larger or if they arrive earlier. The ATLS aims mainly to increase the throughput of each traffic light while partially considering arterial priorities. As we can see from Figure 6.11(c), ATLS increases the throughput of each signalized road intersection by 7% compared to ART-SYS. However, the throughput in the ATLS algorithm is decreased by 2% compared to the Random mechanism. The capacity of any road intersection throughput is limited; therefore, Figure 6.11(c) shows a closer throughput performance for road networks with higher traffic density.

Finally, Figure 6.11(d) illustrates the average delay of arterial flows at each signalized road intersection. As we can see from this figure, ATLS decreases the delay of the arterial flows by 70% compared to the Random mechanism. At the same time, ATLS increases the delay of the arterial street by 10% compared to ART-SYS. This result is to be expected, because ATLS conditionally considers the priority of arterial flows; ART-SYS, on the other hand, primarily considers this priority, and the Random system ignores it completely. All in all, ATLS achieves a good performance in terms of increasing the throughput of each traffic light and decreasing the total delay of each traffic light for the arterial street scenario. At the same time, ATLS enhances the traffic fluency for the arterial flows over the road network.

6.5 Summary

In this chapter, we have introduced an intelligent traffic light scheduling algorithm (ITLC) for isolated traffic light scenarios. Furthermore, we adapted this algorithm, subsequently known as ATLS, for arterial street scenarios. The ITLC algorithm utilizes VANETs technology to gather the real-time traffic characteristics of all competing flows of traffic, at each signalized road intersection. The flow with the maximum traffic density is scheduled to pass the road intersection first, while the traffic light phases are being

set. The ready area is defined at each signalized road intersection, in order to determine the size of each platoon that should pass through the intersection during the green phase that is assigned to this flow. The time assigned to each phase depends on the location of the farthestmost vehicle of each platoon in the traffic flow. ATLs, on the other hand, is intended to guarantee a high traffic fluency for the arterial flows. ATLs also considers the total traffic delay and the throughput of each traffic light in the arterial street scenario. In ATLs, each traffic light schedule incorporates the ratio between the traffic density of the competing traffic flows and the saturation density; this ratio is defined as a saturation factor. The arterial factor (AF_i) is also considered by ATLs to set the sequence of phases of each timing cycle at the located traffic light.

Chapter 7

Conclusion and Future Work

In this thesis, we investigated traffic efficiency applications over the road network with the use of VANETs. First, we presented literature describing state-of-the-art traffic efficiency applications proposed previously. We also classified the previously proposed traffic efficiency applications according to their functionality and targeted road traffic scenarios. We then presented our contributions in this field and reported on the performance of each proposed application, compared to the previously proposed mechanisms.

This thesis introduces four main contributions, which were all aimed at enhancing traffic fluency over road networks in urban areas.

- Designing an efficient traffic evaluation and congestion detection protocol (ECODE). It aims to separately evaluate the real-time traffic characteristics of each road segment in downtown areas. The traffic speed, traffic density, and estimated travel time of each direction of any road segment are evaluated and reported cooperatively by traveling vehicles. ECODE enhances the efficiency utilization of the communication network (i.e., VANETs), the total delay time involved in gathering the traffic characteristics, and the accuracy of the evaluated traffic characteristics. In general, all traffic efficiency applications require real-time traffic distributions over the area of interest. ECODE can be considered a facility function for the traffic efficiency applications of VANETs.
- Designing an intelligent path recommendation protocol (ICOD) intended to construct the path towards each targeted destination in a hop-by-hop fashion. A set of RSUs are installed throughout the area of interest to gather the real-time traffic distributions over a road network. These RSUs communicate with each other to

cooperatively find the optimal path towards each destination. At each road intersection, the traveling vehicles are given recommendations by the best turn option. This option is selected based on the targeted destination location, as well as the traffic situation of the surrounding road segments. Several variants of this protocol have been proposed, which consider congestion avoidance, economical (Eco-path), and context-awareness characteristics. The congestion avoidance variant gives recommendations to each vehicle to choose the fastest alternative path towards its intended destination, which prevent vehicles from traveling through highly congested road segments. The Eco-path variant suggests how each vehicle can choose the path that requires the least amount of fuel and produces the least amount of harmful gases. Finally, the context-aware variant considers the context of each road segment over the investigated area in terms of located common targeted services and conditions of the road.

- Proposing two traffic balancing mechanisms (i.e., Bal-Traf and Abs-Bal) ICOD considers the optimal path of each traveling vehicle towards its targeted destination separately. This may produce traffic congestion conditions at some output road segments, in the case that several vehicles are traveling towards the same destination or where destinations are close each other. In order to address this issue with ICOD, we have proposed the aforementioned mechanisms. Bal-Traf is a traffic balancing mechanism that predicts overloaded output road segments at each road intersection. It recommends some vehicles to take longer or slower paths towards their destinations in order to eliminate the highly congested road segment scenarios over the road network. On the other hand, Abs-Bal aims to keep the traffic absolutely balanced among all output road segments at each road intersection, while directing each vehicle towards its destination.
- Finally, proposing the ITLC algorithm to intelligently schedule the phases of each timing cycle at isolated traffic lights. ITLC controls the traffic light phases of each isolated road intersection. The time period of each phase and sequence of phases are set at each traffic light, based on the real-time traffic characteristics of competing traffic flows. Moreover, we have adapted this algorithm, named ATLCs, to arterial street scenario. The traffic lights located at the arterial street communicate with each other, and cooperatively schedule the phases of each traffic light according to the estimated characteristics of expected arrival platoons.

7.1 Future Work

We can identify several directions for further future research:

- We aim to investigate the estimation of traffic characteristics over road segments in the downtown area, without the need for basic data broadcasting of all vehicles. We plan to produce an accurate traffic evaluation mechanism that requires the basic data of small percentage of traffic characteristics.
- We aim to introduce an intelligent traffic light controlling algorithm that schedules the phases of traffic lights in any signalized road network. The real-time traffic distribution, the emergency traveling vehicles, physical characteristics of the road network, and weather conditions mainly affect the schedule of each traffic light.
- We aim to investigate the fault tolerance issues of our proposed protocols. Different points of failure could appear in our proposed protocols. Some of these failures depend on the nature of the protocol in use (e.g., a cluster head has failed in a cluster-based protocol), whereas others are general and can be appeared in any protocol (e.g., packet lost, communication failures, etc).
- We will focus mainly on the following aspects of security: authentication and key management, privacy, and secure positioning. Security concerns are very important for VANETs. There are several directions that could be investigated in order to make our protocols secure. We plan to investigate security aspects for our proposed protocols that will protect drivers and passengers from security attacks.
- We plan to implement our proposed service discovery protocols in real testbed in order to assess their performance in a real life scenario. Such testbed experiments will allow us to enhance our protocols in more practical manner. In this thesis, we have studied the performance of our protocols only through simulations.
- We plan to investigate the correctness and the validity of the proposed protocols mathematically. The parameters that have been set empirically in our protocols need to be investigated analytically in order to guarantee that the best performance can be obtained.

Bibliography

- [1] R. Bauza, J. Gozalvez, and J. Sanchez-Soriano, *Road Traffic Congestion Detection through Cooperative Vehicle-to-Vehicle Communications*, IEEE 35th Conference on Local Computer Networks (LCN), Pages 606-612, 2010.
- [2] S. Yousefi, M.-S. Mousavi, and M. Fathy, *Vehicular ad hoc networks (VANETs): challenges and perspectives*, 6th International Conference on ITS Telecommunications Proceedings, Pages 761-766, 2006.
- [3] L. Fan, and Yu. Wang, *Routing in vehicular ad hoc networks: A survey*, Vehicular Technology Magazine, vol.2, no.2, Pages 12-22, 2007.
- [4] H.A.B.F. De Oliveira, A. Boukerche, E. Freire Nakamura, A.A.F. Loureiro, *An Efficient Directed Localization Recursion Protocol for Wireless Sensor Networks*, IEEE Transactions on Computers, vol.58, no.5, Pages 677-691, 2009.
- [5] A. Boukerche, H.A.B. Oliveira, E.F. Nakamura, A.A.F. Loureiro, *Secure Localization Algorithms for Wireless Sensor Networks*, IEEE Wireless Communications Magazine, vol.46, no.4, Pages 96-101, 2008.
- [6] A. Boukerche, H. Oliveira, E. Nakamura, and A. Loureiro, *Localization systems for wireless sensor networks*, IEEE Wireless Communication Magazine, vol.14, no.6, Pages 6-12, 2007.
- [7] H.T. Cheng, H. Shan, W. Zhuang, *Infotainment and road safety service support in vehicular networking: From a communication perspective*, Mechanical Systems and Signal Processing, vol.25, no.6, Pages 2020-2038, 2011.
- [8] B. Luna-Benoso, J. C. Martinez-Perales, and R. Flores-Carapia, *Modeling Traffic Flow for Two and Three Lanes through Cellular Automata*, International Mathematical Forum, vol.8, no.22, Pages 1091-1101, 2013.

- [9] A. Bressan, and H. Ke, *Nash equilibria for a model of traffic flow with several groups of drivers*, ESAIM: Control, Optimisation and Calculus of Variations, vol.18, no.4, Pages 969-986, 2010.
- [10] P. Pongpaibool, P. Tangamchit, and K. Noodwong, *Evaluation of Road Traffic Congestion Using Fuzzy Techniques*, International Technical Conference (TECCON) of IEEE Region 10, Pages 1-4, 2007.
- [11] T. Zhong, B . Xu, and O. Wolfson, *Disseminating Real-Time Traffic Information in Vehicular Ad-Hoc Networks*, Intelligent Vehicles Symposium, IEEE, Pages 1056-1061, 2008.
- [12] S. Roy, R. Sen, S. Kulkarni, P. Kulkarni, B. Raman, and L.K. Singh, *Wireless across road: RF based road traffic congestion detection*, 2011 Third International Conference on Communication Systems and Networks (COMSNETS), Pages 1-6, 2011.
- [13] D. Huang, S. Shere, and S. Ahn, *Dynamic Highway Congestion Detection And Prediction Based On Shock Waves*, Dynamic highway congestion detection and prediction based on shock waves. In Proceedings of the seventh ACM international workshop on Vehicular InterNetworking (VANET'10), Pages 11-20, 2010.
- [14] H. Fler, J. Widmer, M. Ksemann, M. Mauve, and H. Hartenstein, *Contention-based forwarding for mobile ad-hoc networks*, Elsevier's Ad Hoc Networks, vol.1 , no.4, Pages 351-369, 2003.
- [15] J. Fukumoto, N. Sirokane, Y. Ishikawa, T. Wada, K. Ohtsuki, and H. Okada, *Analytic method for real-time traffic problems by using Contents Oriented Communications in VANET*, 7th International Conference on Information Technology Syatem (ITST'07) Telecommunications, Pages 1-6, 2007.
- [16] Y. Xu, Y. Wu, G. Wu, J. Xu, B. Liu, and L. Sun, *Data Collection for the Detection of Urban Traffic Congestion by VANETs*, Services Computing Conference (APSCC), IEEE Asia-Pacific, Pages 405-410, 2010.
- [17] X. Zhang, J. Hong, S. Fan, Z. Wei, J. Cao, and Y. Ren, *A Novel Real-time Traffic Information System Based on Wireless Mesh Networks*, IEEE Intelligent Transportation Systems Conference (ITSC), Pages 618-623, 2007.

- [18] B. Schunemann, J. Wedel, and I. Radusch, *V2X-Based Traffic Congestion Recognition and Avoidance*, Tamkang Journal of Science and Engineering, vol.13, no.1, Pages 63-709, 2010.
- [19] L.D. Chou, D. Li, and H. Chao, *Mitigation Traffic Congestion with Virtual Data Sink based Information Dissemination in Intelligent Transportation System*, Third International Conference on Ubiquitous and Future Networks (ICUFN), Pages 37-42, 2011.
- [20] F. M. Padron, *Traffic Congestion Detection Using VANET*, ProQuest, 2009.
- [21] G. Korkmaz and E. Ekici, *Urban Multi-Hop Broadcast Protocol for Inter-Vehicle Communication Systems*, In Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks (VANET '04), Pages 76-85, 2004.
- [22] S. Inoue, K. Shozaki, Y. Kakuda, *An Automobile Control Method for Alleviation of Traffic Congestions Using Inter-Vehicle Ad Hoc Communication in Lattice-Like Roads*, IEEE Global Communications Conference (Globecom) Workshops, Pages 1-6, 2007.
- [23] A. Lakas and M. Cheqfah, *Detection and dissipation of road traffic congestion using vehicular communication*, Microwave Symposium (MMS) Mediterranean, Pages 1-6, 2009.
- [24] S. Dornbush, A. Joshi, *StreetSmart Traffic: Discovering and Disseminating Automobile Congestion Using VANET's*, IEEE 65th Vehicular Technology Conference (VTC), Pages 11-15, 2007.
- [25] N. Shibata, T. Terauchi, T. Kitani, K. Yasumoto, M. Ito, and T. Higashino, *A Method for Sharing Traffic Jam Information using Inter-Vehicle Communication*, Third Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, Pages 1-7, 2006.
- [26] T. Kitani, T. Shinkawa, N. Shibata, K. Yasumoto, M. Ito, and T. Higashino, *Efficient VANET-Based Traffic Information Sharing using Buses on Regular Routes*, IEEE Vehicular Technology Conference (VTC), Pages 3031-3036, 2008.
- [27] K. Collins, G.-M. Muntean, *A vehicle route management solution enabled by Wireless Vehicular Networks*, IEEE Information Communication (INFOCOM) Workshops, Pages 1-6, 2008.

- [28] V. Jain, A. Sharma, and L. Subramanian, *Road traffic congestion in the developing world*, In Proceedings of the 2nd ACM Symposium on Computing for Development (ACM DEV'12), Pages 11, 2012.
- [29] V.-A. Dragoi, C. Dobre, *A model for traffic control in urban environments*, 7th International Wireless Communications and Mobile Computing Conference (IWCMC), Pages 2139-2144, 2011.
- [30] C.-J. Huang, I-F. Chen, K.-W. Hu, and H.-Y. Shen, *A Load Balancing and Congestion-Avoidance Routing Mechanism for Real-Time Traffic over Vehicular Networks*, Journal of Universal Computer Science, vol.15, no.13, Pages 2506-2527, 2009.
- [31] J. Dunkel, A. Fernandez, R. Ortiz, and S. Ossowski, *Event-driven architecture for decision support in traffic management systems*, Expert Systems with Applications, vol.38(6), Pages 6530-6539, 2011.
- [32] R. Bakhshi, F. Bonnet, W. Fokkink and B. Haverkort, *Formal Analysis Techniques for Gossiping Protocols*, ACM SIGOPS Oper. Syst. Rev., vol.41(5), Pages 28-36, 2007.
- [33] M. Jerbi, S.-M. Senouci, R. Meraihi, and Y. Ghamri-Doudane, *An Improved Vehicular Ad Hoc Routing Protocol for City Environments*, IEEE International Conference on Communications (ICC'07), Pages 3972-3979, June 2007.
- [34] L. Wischhof, A. Ebner, and H. Rohling, *Information dissemination in self-organizing intervehicle networks*, IEEE Transactions on Intelligent Transportation Systems, vol.6, no.1, Pages 90-101, 2005.
- [35] C. Thaina, K.N. Nakorn, and K. Rojviboonchai, *A study of adaptive beacon transmission on Vehicular Ad-Hoc Networks*, IEEE 13th International Conference on Communication Technology (ICCT), Pages 597-602, 2011.
- [36] B. Saeed, L. Bjorn, and L. Lavy, *On the Reliability of Safety Message Broadcast in Urban Vehicular Ad Hoc Networks*, Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '11), Pages 307-316, 2011.
- [37] R. Schmidt, T. Leinmuller, E. Schoch, F. Kargl, and G. Schafer, *Exploration of adaptive beaconing for efficient inter vehicle safety communication*, IEEE Network, vol.24, no.1, Pages 14-19, 2010.

- [38] ETSI TS 102 894-1 (V1.1.1), *Intelligent Transport Systems(ITS); Users and applications requirements; Part 1: Facility layer structure, functional requirements and specifications*, European Standard (Telecommunications series), vol.8, 2013.
- [39] ETSI EN 302 665 (V1.1.1), *Intelligent Transport Systems(ITS); Communications Architecture*, European Standard (Telecommunications series), vol.9, 2010.
- [40] Y. Zhu, Z. Li, H. Zhu, M. Li, and Q. Zhang, *A Compressive Sensing Approach to Urban Traffic Estimation with Probe Vehicles*, IEEE Transactions on Mobile Computing, vol.12, no.11, Pages 2289-2302, 2013.
- [41] A. Boukerche, *Algorithms and Protocols for Wireless and Mobile Ad Hoc Networks*, Wiley & Sons, 2008.
- [42] T. Yamashita, K. Izumi, and K. Kurumatani, *Effect of Car navigation with route information sharing for improvement of traffic efficiency*, Intelligent Transportation Systems, Pages 465-470, 2004.
- [43] M. Sniedovich, *Dijkstra's algorithm revisited: the dynamic programming connexion*, Control and cybernetics, vol.35, no.3, 2006.
- [44] X. Zhang, J. Hong, S. Fan, Z. Wei, J. Cao, and Y. Ren , *A Novel Real-time Traffic Information System Based on Wireless Mesh Networks*, IEEE Intelligent Transportation Systems Conference (ITSC), Pages 618-623, 2007.
- [45] B. Schunemann, J. Wedel, and I. Radusch, *V2X-Based Traffic Congestion Recognition and Avoidance*, Tamkang Journal of Science and Engineering, vol.13, no.1, Pages 63-709, 2010.
- [46] L.D. Chou, D. Li, and H. Chao, *Mitigation Traffic Congestion with Virtual Data Sink based Information Dissemination in Intelligent Transportation System*, Third International Conference in Ubiquitous and Future Networks (ICUFN), Pages 37-42, 2011.
- [47] "Google Maps Traffic", <http://maps.google.ca/>, Last accessed in September 2014.
- [48] "Waze GPS system", <http://www.waze.com/>, Last accessed in September 2014.
- [49] "I-GO Navigation System", <http://www.igonavigation.com/>, Last accessed in September 2014.

- [50] G. Korkmaz and E. Ekici, *Urban Multi-Hop Broadcast Protocol for Inter-Vehicle Communication Systems*, in Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks (VANET'04), 2004.
- [51] S. Inoue, K. Shozaki, and Y. Kakuda, *An Automobile Control Method for Alleviation of Traffic Congestions Using Inter-Vehicle Ad Hoc Communication in Lattice-Like Roads*, IEEE Globecom Workshops, Pages 1-6, 2007.
- [52] A. Lakas, M. Cheqfah, *Detection and dissipation of road traffic congestion using vehicular communication*, Microwave Symposium (MMS) Mediterranean, Pages 1-6, 2009.
- [53] S. Dornbush, and A. Joshi, *StreetSmart Traffic: Discovering and Disseminating Automobile Congestion Using VANETs*, IEEE 65th Vehicular Technology Conference (VTC), Pages 11-15, 2007.
- [54] N. Shibata, T. Terauchi, T. Kitani, K. Yasumoto, M. Ito, and T. Higashino, *A Method for Sharing Traffic Jam Information using Inter-Vehicle Communication*, Third Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, Pages 1-7, 2006.
- [55] T. Kitani, T. Shinkawa, N. Shibata, K. Yasumoto, M. Ito, and T. Higashino, *Efficient VANET-Based Traffic Information Sharing using Buses on Regular Routes*, IEEE Vehicular Technology Conference (VTC), Pages 3031-3036, 2008.
- [56] K. Collins and G.-M. Muntean, *A vehicle route management solution enabled by Wireless Vehicular Networks*, IEEE Information Communication (INFOCOM) Workshops, Pages 1-6, 2008.
- [57] K. Collins and G.-M. Muntean, *An Adaptive Vehicle Route Management Solution Enabled by Wireless Vehicular Networks*, 68th IEEE Vehicular Technology Conference (VTC), Pages 1-5, 2008.
- [58] K. Collins and G.-M. Muntean, *Route-Based Vehicular Traffic Management for Wireless Access in Vehicular Environments*, IEEE 68th Vehicular Technology Conference (VTC), Pages 1-5, 2008.
- [59] I. Ben Dhaou, *Fuel estimation model for ECO-driving and ECO-routing*, IEEE Intelligent Vehicles Symposium (IV), Pages 37-42, 2011.

- [60] E. Hart, J. Timmis, P. Mitchell, T. Nakamo, and F. Dabiri, *Some Initial Experiments in Calibrating Emissions Models Using an Evolutionary Algorithm*, Springer Berlin Heidelberg, Pages 125-132, 2012.
- [61] R. Akcelik, R. Smit, and M. BESLEY, *Calibrating fuel consumption and emission models for modern vehicles*, Paper presented at the Transportation Group Conference (IPENZ), 2012.
- [62] R. Akcilek and M. Besley, *Operating cost, fuel consumption and emission model in aaSIDRA and aaMOTION*, In 25th Conference of Australian Institutes of Transport Research, 2003.
- [63] T. Oguchi, M. Katakura and M. Taniguchi, *Carbon dioxide emission model in actual urban road vehicular traffic conditions*, Journal of Infrastructure Planning and Management, no.695(54), Pages 125-136, 2002.
- [64] "Network Simulator nss-2," <http://www.isi.edu/nsnam/ns/>, Last accessed in September 2014.
- [65] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, *SUMO - Simulation of Urban MObility: An Overview*, The Third International Conference on Advances in System Simulation(SIMUL), Pages 63-68, 2011.
- [66] H. Lieu, *Traffic-Flow Theory*, Public Roads (US Dept of Transportation), vol.62, no.4, 1999.
- [67] L. Wischhof and H. Rohling, *Congestion control in vehicular ad hoc networks*, IEEE International Conference on Vehicular Electronics and Safety, Pages 58-63, 2005.
- [68] V.-A. Dragoi, and C. Dobre, *A model for traffic control in urban environments*, 7th International Wireless Communications and Mobile Computing Conference (IWCMC), Pages 2139-2144, 2011.
- [69] C.-J. Huang, I-F. Chen, K.-W. Hu, and H.-Y Shen, *A Load Balancing and Congestion-Avoidance Routing Mechanism for Real-Time Traffic over Vehicular Networks*, Journal of Universal Computer Science, vol.15, no.13, Pages 2506-2527, 2009.
- [70] J. Dunkel, A. Fernandez, R. Ortiz, and S. Ossowski, *Event-driven architecture for decision support in traffic management systems*, Expert Systems with Variants, vol.38(6), Pages 6530-6539, 2011.

- [71] X. Hu, V. Leung, K. G. Li, E. Kong, H. Zhang, N. S. Surendrakumar, and P. TalebiFard, *Social drive: a crowdsourcing-based vehicular social networking system for green transportation*, In Proceedings of the third ACM international symposium on Design and analysis of intelligent vehicular networks and applications (DIVANet '13), Pages 85-92, 2013.
- [72] S. HosseiniNezhad and V, *Data dissemination for delay tolerant vehicular networks: using historical mobility patterns*, In Proceedings of the third ACM international symposium on Design and analysis of intelligent vehicular networks and applications (DIVANet '13), Pages 115-122, 2013.
- [73] A. Boukerche, S.K. Das, A. Fabbri, *Analysis of a Randomized Congestion Control Scheme with DSDV Routing in ad Hoc Wireless Networks*, Journal of Parallel and Distributed Computing, vol.61, no.7, Pages 967-995, 2001.
- [74] A. Boukerche, X. Cheng, J. Linus, *A performance evaluation of a novel energy-aware data-centric routing algorithm in wireless sensor networks*, Springer Wireless Network (WINE), vol.11(5), Pages 619-635, 2005.
- [75] A. Boukerche, C. Tropper, *A distributed graph algorithm for the detection of local cycles and knots*, IEEE Transactions on Parallel and Distributed Systems, vol.9, no.8, Pages 748-757, 1998.
- [76] A. Boukerche, M.S.M.A. Notare, *Behavior-Based Intrusion Detection in Mobile Phone Systems*, Journal of Parallel and Distributed Computing, vol.62(9), Pages 1476-1490, 2002.
- [77] A. Boukerche, S.K. Das, *Dynamic load balancing strategies for conservative parallel simulations*, 11th Workshop on Parallel and Distributed Simulation, Pages 20-28, 1997.
- [78] A. Boukerche, S. Hong, T. Jacob, *An efficient synchronization scheme of multimedia streams in wireless and mobile systems*, IEEE Transactions on Parallel and Distributed Systems, vol.13, no.9, Pages 911-923, 2002.
- [79] A. Boukerche, I. Chatzigiannakis, S. Nikolettseas, *A new energy efficient and fault-tolerant protocol for data propagation in smart dust networks using varying transmission range*, Computer Communications, vol.29(4), Pages 477-489, 2006.

- [80] H.F. Wedde, S. Senge, *BeeJamA: A Distributed, Self-Adaptive Vehicle Routing Guidance Approach*, IEEE Transactions on Intelligent Transportation Systems, vol.14, no.4, Pages 1882-1895, 2013.
- [81] T. Kono, T. Fushiki, K. Asada, and K. Nakano, *Fuel consumption analysis and prediction model for Eco route search*, In 15th World Congress on Intelligent Transport Systems and ITS America's 2008 Annual Meeting.
- [82] F. Webster, *Traffic Signal Settings*, Road Research Laboratory Technical Report, HMSO, London, UK.
- [83] M. Behrisch, L. Bieker, J. Erdmann and D. Krajzewicz, *SUMO Simulation of Urban Mobility: An Overview*, The Third International Conference on Advances in System Simulation (SIMUL), Pages 63-68, 2011.
- [84] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, *Recent Development and Applications of SUMO Simulation of Urban Mobility*. International Journal On Advances in Systems and Measurements, vol.5, no.3, Pages 128-138, 2012.
- [85] Dunn Engineering Associates In association with Siemens Intelligent Transportation Systems, *Traffic Control Systems Handbook*, Federal Highway Administration Office of Transportation Management, 2005.
- [86] F. Al-Nasser and M. Mahmoud, *Wireless Sensors Network Application: A Decentralized Approach for Traffic Control and Management*, *Wireless Sensor Networks - Technology and Applications*, Matin (Ed.).
- [87] D. Krajzewicz, E. Brockfeld, J. Mikat, J. Ringel, C. Ressel, W. Tuchscheerer, P. Wagner, and R. Wsler, *Simulation of modern Traffic Lights Control Systems using the open source Traffic Simulation SUMO*, 3rd Industrial Simulation Conference, Pages 299-302, 2005.
- [88] C.T. Barba, M.A. Mateos, P.R. Soto, A.M. Mezher, and M.A. Igartua, *Smart city for VANETs using warning messages, traffic statistics and intelligent traffic lights*, IEEE Intelligent Vehicles Symposium (IV), Pages 902-907, 2012.
- [89] D. Greenwood, B. Burdiliak, I. Trencansky, H. Armbruster, and C. Danneegger, *GreenWave Distributed Traffic Intersection Control*, In Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems- vol.2, Pages 1413-1414, 2009.

- [90] K. Pandit, D. Ghosal, H.M. Zhang, and C. Chen-Nee, *Adaptive Traffic Signal Control With Vehicular Ad hoc Networks*, IEEE Transactions on Vehicular Technology, vol.62, no.4, Pages 1459-1471, 2013.
- [91] N. Maslekar, M. Boussedjra, J. Mouzna, and H. Labiod, *VANET Based Adaptive Traffic Signal Control*, IEEE 73rd Vehicular Technology Conference (VTC), Pages 1-5, 2011.
- [92] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Iftode, *Adaptive Traffic Lights Using Car-to-Car Communication*, IEEE 65th Vehicular Technology Conference (VTC), Pages 21-25, 2007.
- [93] N. Maslekar, J. Mouzna, M. Boussedjra, and H. Labiod, *CATS: An adaptive traffic signal system based on car-to-car communication*, Journal of Network and Computer Applications, vol.36(5), Pages 1308-1315, 2013.
- [94] S.-W. Chen, C.-B. Yang, and Y.-H. Peng, *Algorithms for the traffic light setting problem on the graph model*, The 12th Conference on Artificial Intelligence and Applications, 2007.
- [95] Z. Michalewicz, *Genetic algorithms + data structures = evolution programs*, Springer, 1996.
- [96] J. Kennedy and R. Eberhart, *Particle swarm optimization*, Proceedings of the IEEE international conference on neural networks, vol.4, Pages 1942-1948, 1995.
- [97] E. Elbeltagi, T. Hegazy, and D. Grierson, *Comparison among five evolutionary-based optimization algorithms*, Advanced Engineering Informatics, vol.19, Pages 435-453, 2005.
- [98] X.-H. Yu, and W.W. Recker, *Stochastic adaptive control model for traffic signal systems*, Transportation Research Part C: Emerging Technologies, vol.14(4), Pages 263-282, 2006.
- [99] P. Mirchandani, and L. Head, *A real-time traffic signal control system: architecture, algorithms, and analysis*, Transportation Research Part C: Emerging Technologies, vol.9(6), Pages 415-432, 2001.
- [100] T. Bellemans, B. De Schutter, and B. De Moor, *Model predictive control for ramp metering of motorway traffic: A case study*, Control Engineering Practice, vol.14, no.7, Pages 757-767, 2006.

- [101] C. Priemer, and B. Friedrich, *A decentralized adaptive traffic signal control using V2I communication data*, 12th International IEEE Conference on Intelligent Transportation Systems (ITSC '09), Pages 1-6, 2009.
- [102] L.D. Baskar, B. De Schutter, J. Hellendoorn, and Z. Papp, *Traffic control and intelligent vehicle highway systems: a survey*, Intelligent Transport Systems(IET), Pages 38-52, 2011.
- [103] A. Boukerche, *Algorithms and Protocols for Wireless Sensor Networks*, John Wiley and Sons, vol.62, 2008.
- [104] A. Boukerche, H.A.B.F. Oliveira, E.F. Nakamura, A.A.F. Loureiro, *Vehicular Ad Hoc Networks: A New Challenge for Localization-Based Systems*, Computer Communications, vol.31(12), Pages 2838-2849, 2008.
- [105] A. Chen, B. Khorashadi, C.N. Chuah, D. Ghosal, and M. Zhang, *Smoothing Vehicular Traffic Flow Using Vehicular-Based Ad Hoc Networking Computing Grid (VGrid)*, IEEE Intelligent Transportation Systems Conference (ITSC '06), Pages 349-354, 2006.
- [106] B. Khorashadi , B. Liu , D. Ghosal , C. N. Chuah and M. Zhang *Smoothing vehicular traffic flow with VGrid*, In Transportation Research Board Annual Meeting, 2010.
- [107] S. El-Tantawy, B. Abdulhai, and H. Abdelgawad, *Multiagent Reinforcement Learning for Integrated Network of Adaptive Traffic Signal Controllers (MARLIN-ATSC): Methodology and Large-Scale Application on Downtown Toronto*, IEEE Transactions on Intelligent Transportation Systems, vol.14, no.3, Pages 1140-1150, 2013.
- [108] Y.-S. Huang, Y.-S. Weng, and M. Zhou, *Modular Design of Urban Traffic-Light Control Systems Based on Synchronized Timed Petri Nets*, IEEE Transactions on Intelligent Transportation Systems, vol.16, no.99, Pages 1-10, 2013.
- [109] O. Tomescu, I. M. Moise, A. E. Stanciu, and I. Batros, *Adaptive Traffic Light Control System Using Ad Hoc Vehicular Communications Network*, U.P.B. Sci. Bull., Series D, vol.74(2), 2012.
- [110] D. Cunha, F. Domingos, L.A. Vilas, A.C. Viana, and A.A.F. Loureiro, *Data Communication in VANETs: A Survey, Challenges and Applications*, 2014.

- [111] M. Malek. *A Comparison Connection Assignment for Diagnosis of Multiprocessor Systems*. Proceedings of the 7th annual symposium on Computer Architecture (ACM), Pages 31-35 1980.
- [112] O. Abumansoor, A. Boukerche, *A Secure Cooperative Approach for Nonline-of-Sight Location Verification in VANET*, IEEE Transactions on Vehicular Technology (TVT), vol.61, no.1, Pages 275-285, 2012.
- [113] A. Boukerche, S. Hong, T. Jacob, *A Distributed Algorithm for Dynamic Channel Allocation*, Mobile Networks and Applications, vol.7(2), Pages 115-126, 2002.
- [114] A. Boukerche, K. El-Khatib, L. Xu, L. Korba, *SDAR: a secure distributed anonymous routing protocol for wireless and mobile ad hoc networks*, 29th Annual IEEE International Conference on Local Computer Networks (LCN), Pages 618-624, 2004.
- [115] R.B. Batista, A. Boukerche, A.C.M.A.D. Melo, *A parallel strategy for biological sequence alignment in restricted memory space*, Journal of Parallel and Distributed Computing, vol.68(4), Pages 548-561, 2008.
- [116] A. Boukerche, R.W.N. Pazzi, R.B. Araujo, *HPEQ A Hierarchical Periodic, Event-driven and Query-based Wireless Sensor Network Protocol*, The IEEE Conference on Local Computer Networks (LCN), Pages 560-567, 2005.
- [117] J. Maeng and M. Malek, *A Comparison Connection Assignment for Self-Diagnosis of Multiprocessor Systems*, 11th International Symposium on Fault-Tolerant Computing, Pages 173-175, 1981.
- [118] D.M. Blough and H.W. Brown, *The Broadcast Comparison Model for On-Line Fault Diagnosis in Multiprocessor Systems: Theory and Implementation*, IEEE Transaction on Computer networks, vol.48, no.5, Pages 470-493, 1999.
- [119] M. Sharmin, S. Ahmed, and S. I. Ahamed, *SAFE-RD (Secure, Adaptive, Fault Tolerant, and Efficient Resource Discovery in Pervasive Computing Environments*, IEEE international Conference on Information Technology (ITCC), Pages 271-276, 2005.
- [120] W. Winoto, E. Schwartz, H. Balakrishnan, and J. Lilley, *The design and implementation of an intentional naming system*, 17th ACM Symposium on Operating Systems Principles (SOSP 99), 1999.

- [121] A. Boukerche, F. Xin, *A Voronoi Approach for Coverage Protocols in Wireless Sensor Networks*, IEEE Global Telecommunications Conference, GLOBECOM '07, Pages 5190-5194, 2007.
- [122] A. Boukerche, Y. Ren, *A secure mobile healthcare system using trust-based multicast scheme*, IEEE Journal on Selected Areas in Communications, vol.27, no.4, Pages 387-399, 2009.
- [123] A. Boukerche, X. Fei, *A coverage-preserving scheme for wireless sensor network with irregular sensing range*, Ad Hoc Networks, vol.5(8), Pages 1303-1316, 2007.
- [124] A. Boukerche, A. Roy, *Dynamic Grid-Based Approach to Data Distribution Management*, Journal of Parallel and Distributed Computing, vol.62(3), Pages 366-392, 2002.
- [125] A. Boukerche, K.R.L. Juc, J.B. Sobral, M.S.M.A. Notare, *An artificial immune based intrusion detection model for computer and telecommunication systems*, Parallel Computing, vol.30(5-6), Pages 629-646, 2004.
- [126] A. Boukerche, S. Das, A. Fabbri, *SWiMNet: A Scalable Parallel Simulation Testbed for Wireless and Mobile Networks*, Springer Wireless Networks, vol.7(5), Pages 467-486, 2001.
- [127] H.A.B.F. Oliveira, A. Boukerche, E.F. Nakamura, A.A.F. Loureiro, *Localization in time and space for wireless sensor networks: An efficient and lightweight algorithm*, Performance Evaluation, vol.66(35), Pages 209-222, 2009.
- [128] A. Boukerche, L. Bononi, *Simulation and modeling of wireless, mobile, and ad hoc networks*, Mobile ad hoc networking, Pages 373-409, 2004.
- [129] A. Bamis, A. Boukerche, I. Chatzigiannakis, S. Nikolettseas, *A mobility aware protocol synthesis for efficient routing in ad hoc mobile networks*, Computer Networks, vol.52,(1), Pages 130-154, 2008.
- [130] M. Elhadef, A. Boukerche, H. Elkadiki, *A distributed fault identification protocol for wireless and mobile ad hoc networks*, Journal of Parallel and Distributed Computing, vol.68(3), Pages 321-335, 2008.

- [131] S. Samarah, M. Al-Hajri, A. Boukerche, *A Predictive Energy-Efficient Technique to Support Object-Tracking Sensor Networks*, IEEE Transactions on Vehicular Technology, vol.60, no.2, Pages 656-663, 2011.
- [132] H.A.B.F. Oliveira, A. Boukerche, E.F. Nakamura, A.A.F. Loureiro, *Localization in time and space for wireless sensor networks: An efficient and lightweight algorithm*, Performance Evaluation, vol.6(3-5), Pages 209-222, 2009.
- [133] Y. Ren, A. Boukerche, *Modeling and Managing the Trust for Wireless and Mobile Ad Hoc Networks*, IEEE International Conference on Communications (ICC '08), Pages 2129-2133, 2008.
- [134] A. Boukerche, *Security and fraud detection in mobile and wireless networks*, in Handbook of Wireless Networks and Mobile Computing, Pages 309-323, 2002.