Improving User Experience and Energy Efficiency for Different Classes of Users in Vehicular Networks

Irina Tal

A Dissertation submitted in fulfillment of the requirements for the award of Doctor of Philosophy (Ph.D.)

Dublin City University

Faculty of Engineering and Computing,
School of Electronic Engineering

Supervisor: Dr. Gabriel-Miro Muntean

January, 2016

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of *Doctor of Philosophy* is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Student ID: 11212426

Signed:________________

Date:_________________

ACKNOWLEDGMENTS

First and foremost I would like to give thanks to God for giving me the strength to complete my PhD. Then, I would like to thank my dear and loving parents for providing me with the opportunity to be where I am today. They have sacrificied their own lives for me and my brother in order to give us everything, including a high education which they have always emphasized on. Mom, thank you for being always there for me and for living every moment of mine at a maximum intensity. Dad, thank you for believing in me. Thank you for all your prayers and for all your support. Extra special thanks go to my loving husband who was my biggest supporter in these last years. Thank you, darling for all your constant and invaluable support, for being so understanding and for your unconditional love and care. I couldn't have done it without you! Special thanks to my baby Sophie for being a great kiddo and allowing her mummy write her thesis ☺. You made mummy extra motivated. Another special thank you goes to my brother for all his encouragement, for believing in me and for just being a great big brother.

An extra special acknowledgement goes to my amazing supervisor, Gabi, for his guidance, extra-patience, especially in this last year, never-ending support, understanding and encouragement. Gabi thank you for believing in me, for pushing me forward when necessary, for being such a wonderful, enthusiastic and energetic supervisor. I have greatly benefited from your experience and wisdom, learning so many things from you. Thank you for leading me on this path toward a successful PhD and for being not only an extraordinary supervisor, but also an extraordinary person and friend.

Special thanks to my very good friend and former lab colleague Ramona for all the invaluable advices during these years, for receiving me so warmly in a cold Dublin, for always being there, believing in me and encouraging me, for listening to all my complaints and mostly for adopting me as her little sister.

I would also like to thank to Jenny, the lab co-director, and to all my lab collegues: Martin, Shengyang, Quang, Ronan, Ting, Longhao, Lejla, Yi Han, for their constant support, for all the lively chats and laughs we shared during these years.

I am blessed to be surrounded by great, wonderful and supportive people such as my good friends Adriana, Ioana, Arghir, Gabi's wonderful family, Aida & family, Bogdan & family, Irina, my godparents, and many others that I cannot mention here. Adriana you are a wonderful, amazing, trustful friend; you are like a sister to me. Thank you for your constant support, I am very happy that you are a constant part of my life. Words are too poor then to express the invaluable moral support and guidance offered by my godparents. A big thank you! Ioana & Arghir it was lovely sharing with you my first part of a PhD life.

I would like to extend a special thank you to the technical staff of the School of Electronic Engineering, Dublin City University, for their valuable support.

I would like to thank Dublin City University for funding my work under the Daniel O'Hare Scholarship Scheme and I would also like to acknowledge the support of LERO the Irish Software Research Centre.

Last but not the least, I gratefully acknowledge Dr. Pavlos Lazaridis, Dr. Xiaojun Wang, and Dr. Marissa Condon, for making this happen.

Dublin, January 2016

Irina Tal

LIST OF PUBLICATIONS

- **I.Tal** and G-M. Muntean, “Vehicular Communications-based Speed Advisory System for Electric Bicycles”, *IEEE Transactions on Vehicular Technology*, June 2015
- **I.Tal** and G-M. Muntean, “Towards Smarter Cities and Roads: A Survey of Clustering Algorithms in VANETs*”, Convergence of Broadband, Broadcast, and Cellular Network Technologies*, IGI Global, USA, April 2014
- **I.Tal**, Tianhua Zhu, G-M. Muntean, “On the Potential of V2X Communications in Helping Electric Bicycles Saving Energy”, *IEEE Vehicular Networking Conference (VNC),* Boston, USA, December 2013
- **I. Tal**, G-M. Muntean, “V2X Communication-based Power Saving Strategy for Electric Bicycles”, *IEEE Global Communications Conference (GLOBECOM),* Workshop on Vehicular Networks Evolution, Atlanta, USA, December 2013
- **I. Tal**, G-M. Muntean, “eWARPE – Energy-efficient Weather-aware Route Planner for Electric Bicycles”, *IEEE International Conference on Network Protocols (ICNP*), Göttingen, Germany, October 2013
- **I. Tal**, G-M. Muntean, “User-oriented Fuzzy Logic-based Clustering Scheme for Vehicular Ad-hoc Networks”, *IEEE Vehicular Technology Conference (VTC Spring),* Dresden, Germany, June 2013
- S. Djahel, M. Salehie, **I. Tal**, P. Jamshidi, “Adaptive Traffic Management for Secure and Efficient Emergency Services in Smart Cities”, *IEEE Pervasive Computing and Communication (PerCom) conference (WiP track),* San Diego, California, USA, March 2013
- **I. Tal**, G-M. Muntean, “Using Fuzzy Logic for Data Aggregation in Vehicular Networks”, *IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications*, Dublin, Ireland, October 2012

TABLE OF CONTENT

Acknowledgments ... iii
List of Publications ... v
Table of Content ... vi
List of Tables ... x
List of Figures ... xi
List of Abbreviations ... xvi
Abstract ... xviii
Chapter 1 Introduction ... 1
1.1. Motivation ... 1
1.2. Problem Statement and Proposed Solutions ... 3
1.3. Contributions to the State of the Art ... 5
1.4. Structure of the thesis ... 6
Chapter 2 Technical Background ... 7
2.1. Vehicular Networks ... 7
2.1.1. Vehicular Networks Overview ... 7
2.1.2. Supporting Standards ... 10
2.1.3. VANET Applications ... 14
2.1.4. VANET Characteristics and Challenges ... 15
2.2. Fuzzy Logic and Vehicular Networks ... 17
2.2.1. Introduction to FL ... 17
2.2.2. Fuzzy Logic Concepts ... 18
2.2.2.1. Linguistic Variables ... 18
2.2.2.2. Fuzzy Sets and Membership Functions ... 18
2.2.3. Designing a Fuzzy Logic System Step-by-Step ... 21
2.2.4. Fuzzy Logic Tools ... 25
2.3. Clustering and Vehicular Networking ... 27
2.3.1. Introduction to Clustering ... 27
2.3.2. Clustering in VANETs ... 29
2.3.3. Performance Assessment of Clustering in VANETs ... 31
2.3.4. Analysis of Topology-based Metrics for Clustering Algorithms Assessment in VANETs ... 32
2.3.4.1. Topology-based Metrics or General Metrics for the Evaluation of Clustering Algorithms in VANETs ... 34
2.3.4.2. Solution-dependent Topology-based Metrics ... 38
2.4. Chapter Summary ... 39
Chapter 3 Related Works ... 41

3.1. VANET-based Green Transportation Solutions 41
3.1.1. VANET-based Eco-routing Solutions 41
3.1.2. VANET-based Eco-driving Solutions 43
3.1.3 Discussion – Future Directions 44
3.2. Fuzzy Logic Solutions in VANETs 46
3.2.1. FL-based and Routing Protocols in VANETs 46
3.2.1.1. Routing Protocols in VANETs 46
3.2.1.2. FL-based Routing Protocols in VANETs 48
3.2.1.3. Discussion 52
3.2.2. FL-based MAC Protocols in VANETs 54
3.2.2.1. MAC Protocols in VANETs 54
3.2.2.2. FL-based MAC Protocols in VANETs 54
3.2.2.3. Discussion 58
3.2.3. FL-based Data Aggregation Solutions in VANETs 59
3.2.3.1. Data Aggregation in VANETs 59
3.2.3.2. FL-based data aggregation solutions in VANETs 60
3.2.3.3. Discussion 62
3.2.4. FL-based and Handover in VANETs 63
3.2.4.1. Handover in VANETs 63
3.2.4.2. FL-based Handover Solutions in VANETs 65
3.2.4.3. Discussion 66
3.2.5. FL Solutions in VANETs – Classification and Lessons Learnt 66
3.2.5.1. Classification 67
3.2.5.2. Lessons Learnt 68
3.3. Clustering Algorithms in VANETs 72
3.3.1. Static Clustering Algorithms 74
3.3.2. Mobile Clustering Algorithms 76
3.3.3 Discussion – Future Directions 81
3.4. Chapter Summary 84
Chapter 4 Proposed Architecture and Algorithms 86
4.1. Overall Architecture and Network Model 86
4.2. Speed Advisory System for Electric Bicycles - SAECy 89
4.2.1. SAECy Overview 89
4.2.2. SAECy Architecture 90
4.2.2.1. Data Collector and Processor 91
4.2.2.2. Computation and Recommendation Module 92
A) Bicycle Power Consumption Model 92
B) Bicycle Dynamics Model 93
C) FLS and Advised Speed Computation components 93
4.2.3. SAECy Algorithms 94

4.2.3.1. Green Optimal Speed Advisory Function 94
4.2.3.2. FL-based Wind-aware Speed Adaptation Policy 96
4.3. Energy Efficient Weather-aware Route Planner solution for Electric Bicycles – eWARPE 99
4.3.1. eWARPE Architecture and Functional Principle 99
4.3.2. eWARPE Algorithm 102
4.4. FuzzC-VANET - a Fuzzy Logic-based Clustering Scheme 105
4.4.1. FuzzC-VANET Principle and Network Model 105
4.4.2. FuzzC-VANET Algorithms 106
4.4.2.1. CHE Computation 109
4.4.3. User-oriented Fuzzy Logic-based Clustering Scheme 114
4.4.3.1. Architecture 114
4.4.3.2. Additions to the Clustering Scheme 116
A) Additions to the Clustering Beacon Message Format 116
B) Alterations to CHE Computation 116
4.5. Chapter Summary 117
Chapter 5 Performance Evaluation 119
5.1. SAECy's Performance Evaluation 119
5.1.1. Experimental Testing and Simulation Model Validation 119
5.1.1.1. Experimental Test-bed Description 119
5.1.1.2. Scenarios Description 120
5.1.1.3. Results Analysis – Experimental vs Simulation Results 123
5.1.2. Simulation-based Testing 124
5.1.2.1. Simulation Model 124
A) Simulation Platform 124
B) Simulation Settings 127
5.1.2.2. Scenarios Description 128
5.1.2.3. Comparison-based Assessment 130
5.1.2.4. Results and Analysis 130
A) Impact of the Traffic Lights Number and Topology 131
B) Impact of the Cyclist Weight on the Energy Consumption 138
C) Power Consumption vs Time 142
5.2. eWARPE Performance Evaluation 145
5.2.1. Survey - Interview 145
5.2.1.1. The problem and its dimensions 146
5.2.1.2. The interest in eWARPE and its benefits 147
5.2.2. Survey - Questionnaire 148
5.2.2.1. The problem and its dimensions 148
5.2.2.2. The interest in eWARPE and its benefits 150
5.2.3. Case Study 151

5.3. FuzzC-VANET Evaluation 156
5.3.1. Simulation Scenarios and Settings 157
5.3.2. Results and Analysis 159
5.3.2.1. Impact of traffic density: sparse vs dense traffic 159
5.3.2.2. Impact of penetration rate 163
5.3.2.3. Impact of time interval controlling CH re-election process 170
5.3.3. FuzzC-VANET Instantiation Testing 173
5.4. Chapter Summary 177
Chapter 6 Conclusions and Future Work 179
6.1. Conclusions 179
6.1.1. Overview 179
6.1.2. Contributions – Summary 181
6.2. Future Work 182
References 185
Appendix A .... A
Interview set of questions .... A
Appendix B I
Questionnaire set of questions I

LIST OF TABLES

2.1 FL Tool Comparison………………………………………………………………… 28

2.2 Notations used in topology-based metrics definition …………………….………… 34

2.3 General Metrics for the Evaluation of Clustering Algorithms in Vehicular Networks – Summary ………..………..………..………..………..………..………..………..………..….. 41

3.1 FL-based VANET Solutions - Classification …………………………….................. 71

4.1 Power Consumption Model Notations ……………………………………………... 93

4.2 Bicycle Dynamics Model Notations ……………………………………....…....….. 94

4.3 eWARPE FLS Rule Base………………………………………………………......... 105

4.4 Input/Output Linguistic Variables……………………………………………………. 114

4.5 Fuzz-C VANET Rule Base………………………………………………………....... 114

4.6 Alterations to the Rule Base………………………………………………………...... 118

5.1 Results Summary Experimental vs Simulation Testing ………………………………… 125

5.2 Simulation Parameters – List of Possible Values……………………....…....…........….. 128

5.3 Results Summary for Scenario 1 and 2 when varying Rolling Coefficient……………………………………....…....…... 129

5.4 Total Travel Time – Route 1…………………………………….....…....…........….. 139

5.5 Total Travel Time – Route 2……………………………………………………….... 139

5.6 Total Travel Time – Route 3……………………………………………………….... 140

5.7 Simulation Case-Study Results……………………………………….................... 154

5.8 Traffic Simulation Settings……………………………………………………......... 159

5.9 Network Simulation Settings……………………………………………………..... 159

5.10 Results – Clustered vs Non-Clustered Architecture…………………………………….. 178

LIST OF FIGURES

2.1 (a) Ad-hoc Architecture………………………………………………………………. 10

2.1 (b) WLAN/cellular Architecture……………………………………………………….. 10

2.1 (c) Hybrid Architecture………………………………………………………………... 11

2.2 Reference Protocol Stack of an ITS Station ……………………..………………….. 12

2.3 DSRC Channel Assignment in North America……………………..……………….. 13

2.4 DSRC Spectrum Allocation Worldwide……………………..………………………. 13

2.5 SPAT message example ……………………………………..…………………. 14

2.6 The Fuzzy sets of low and medium terms of speed ([48]) ………..…………………. 20

2.7 Classic FLS architecture ……………………………………..…………………. 21

2.8 Reinforcement learning-based real-time adaptive FLS architecture…………………. 25

2.9 Simple real-time adaptive FLS architecture……………………..…………………. 25

2.10 Illustration of node states in MANET clustering………………..…………………… 29

2.11 State machine representing CH-based clustering in VANETs..……………………… 30

2.12 Classification of the performance assessment metrics used in VANETs clustering solutions.. 33

3.1 Routing Protocol Description……………………………………………...………….. 48

3.2 VANET-based Routing Classification..……………………………….....………….. 48

3.3 VANETs MAC Protocols Classification..…………………………….....……….. 55

3.4 Data Aggregation – Functional Components..……………………………....……. 60

3.5 HO Phases…………………………...…………………………………...………….. 65

4.1 Overall Solution Architecture and Network Model…………………………………… 88

4.2 SAECy – Overview……………………………………………………….....….....… 90

4.3 SAECy – Architecture…………………………………………………….............. 92

4.4 eWARPE Architecture………….……………………………………………………… 101

4.5 Interaction between Functional Blocks………………………………………………… 102

4.6 *Precipitation chances* Membership…………………………………………………… 104

4.7 *Wind speed* Membership……………………………………………………………… 104

4.8 Clustered Network Model……………………………………………………………. 106

4.9 Clustering Scheme Finite State Machine…………………………………………… 107

4.10 Clustering beacon message…………………………………………………………. 107

4.11 FuzzC-VANET FLS……..…………………………………………………………. 111

4.12 *AD* Membership Functions…………………………………………………………. 113

4.13 *CHE* Membership Functions………………………………………………………. 113

4.14 System Architecture……………………………………………………………….. 116

4.15 Modified Beacon Structure…………………………………………………………. 117

5.1 Experimental Test-bed ……………………………………………………………… 122

5.2 Power consumption for bicycle equipped with speed advisory system – test-bed results 123

5.3 Power consumption for non-equipped bicycle – test-bed results……………………. 123

5.4 Simulation Platform…………………………………………………………....……. 127

5.5 Route 1 and Route 3……………………………………………………………....……. 130

5.6 Route 2…………………………………………………………....……....…………… 131

5.7 EC reduction – Route 1 (bicycle without assistance at start)………………………… 133

5.8 EC reduction – Route 2 (bicycle without assistance at start) ………………………… 133

5.9 EC reduction – Route 3 (bicycle without assistance at start)………………………… 134

5.10 EC reduction – Route 1 (bicycle with assistance at start)…………………………… 134

5.11 EC reduction – Route 2 (bicycle with assistance at start)…………………………… 135

5.12 EC reduction – Route 3 (bicycle with assistance at start)…………………………… 135

5.13 Number of stops for the non-equipped bicycle for the three routes………………… 136

5.14 Number of stops for the equipped bicycle for the three routes……………………… 136

5.15 Waiting times – Route 1………………………………………………………………. 137

5.16 Waiting times – Route 2………………………………………………………………. 138

5.17 Waiting times – Route 3………………………………………………………………. 138

5.18 EC reduction for SA2 – Route 1…………………………………………………………. 140

5.19 EC reduction for SA2 – Route 2…………………………………………………………. 141

5.20 EC reduction for SA2 – Route 3…………………………………………………………. 141

5.21 EC reduction for SA1 – Route 1…………………………………………………………. 142

5.22 EC reduction for SA1 – Route 2…………………………………………………………. 142

5.23 EC reduction for SA1 – Route 3…………………………………………………………. 143

5.24 Power vs time – Route 1……………………………………………………………….. 144

5.25 Power vs time – Route 2……………………………………………………………….. 145

5.26 Power vs time – Route 3……………………………………………………………….. 145

5.27 Proposed solution expected improvement on subjects' cycling experience…………. ... 148

5.28 Factors negatively influencing cycling – answers ……………………….……………… 150

5.29 Bad weather conditions influence on cyclists ………………………..………………... 150

5.30 Interest in eWARPE …………………….……………………………………………... 151

5.31 eWARPE estimated usefulness on improving cycling experience………………….. 152

5.32 Case-study route……………………………..……..……..……..……..……..…….. 153

5.33 Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (vg =4.17m/s)…………………………...………………… …………………………… 155

5.34 Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (vg =4.5m/s)…………………………...………………… …………………………… 155

5.35 Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (vg =5m/s)…………………………...………………… …………………………… 156

5.36 Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (vg =5.56m/s)…………………………...………………… …………………………… 156

5.37 Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (vg =6m/s)…………………………...………………… …………………………… 157

5.38 Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (vg =6.5m/s)…………………………..…………………………………………….. 157

5.39 Avg. Overhead……………………..…………………………………………….. 162

5.40 Avg. CML……………………..………………………………………………. 162

5.41 Avg. CHL……………………..……………………………………………….. 162

5.42 Avg. Percentage of vehicles clustered ……………………………………………….. 163

5.43 Avg. NoC……………………..………………………………………………... 163

5.44 Avg. CS...……………………..……………………………………………….. 163

5.45 Avg. Throughput……………………...…………………………………………… 164

5.46 Avg. Throughput/CM………………...…………………………………………… 164

5.47 Avg. Overhead vs penetration rate – Scenario 1……………………………………. 166

5.48 Avg. CML vs penetration rate – Scenario 1………………………………………… 166

5.49 Avg. CHL vs penetration rate – Scenario 1…………………………………………. 166

5.50 Avg. Percentage of vehicles clustered vs penetration rate – Scenario 1……………. 167

5.51 Avg. NoC vs penetration rate – Scenario 1…………………………………………. 167

5.52 Avg. CS vs penetration rate – Scenario 1…………………………………………… 167

5.53 Avg. throughput of the network vs penetration rate – Scenario 1…………………… 168

5.54 Avg. throughput/CM vs penetration rate – Scenario 1……………………………… 168

5.55 Avg. Overhead vs penetration rate – Scenario 2……………………………………. 168

5.56 Avg. CML vs penetration rate – Scenario 2………………………………………… 169

5.57 Avg. CHL vs penetration rate – Scenario 2…………………………………………. 169

5.58 Avg. Percentage of vehicles clustered vs penetration rate – Scenario 2……………. 169

5.59 Avg. NoC vs penetration rate – Scenario 2…………………………………………. 170

5.60 Avg. CS vs penetration rate – Scenario 2…………………………………………… 170

5.61 Avg. throughput of the network vs penetration rate – Scenario 2…………………… 170

5.62 Avg. throughput/CM vs penetration rate – Scenario 2……………………………… 171

5.63 Avg. CML vs TI…………………………………………………………………….. 171

5.64 Avg. CHL vs TI………………………………………………………………………… 172

5.65 Avg. Percentage of vehicles clustered vs TI………………………..…………….. 172

5.66 Avg. NoC vs TI…………………………………………………..…………….. 172

5.67 Avg. CS vs TI……………………………………………………….…………….. 173

5.68 Avg. throughput of the network vs TI……………………………………………….. 173

5.69 Avg. throughput/CM vs TI………………………………………………………… 173

5.70 Avg. Overhead (including FuzzC-VANET*)……………………….…………….. 174

5.71 Avg. CML (including FuzzC-VANET*)…………………………………………….. 174

5.72 Avg. CHL(including FuzzC-VANET*)……………………………………………… 175

5.73 Avg. Percentage of vehicles clustered(including FuzzC-VANET*)…..…………….. 175

5.74 Avg. NoC(including FuzzC-VANET*)……………………………..…………….. 175

5.75 Avg. CS(including FuzzC-VANET*)………………………………….…………….. 176

5.76 Avg. throughput of the network(including FuzzC-VANET*)……………………….. 176

5.77 Avg. throughput/CM(including FuzzC-VANET*)………………………………….. 176

5.78 Scenario 3 – hybrid VANET scenario……………………………………………….. 177

5.79 Zoom of Scenario 3 intersections………………………………………………….. 178

LIST OF ABBREVIATIONS

AODV – Ad hoc On-Demand Distance Vector Routing

AU – Application Unit

CAM – Cooperative Awareness Messages

CC – Cluster Changes

CCH – Control Channel

CCR – Cluster Reconfiguration Rate

CH – Cluster Head

CHCR – Cluster head Changing Rate

CHL – Cluster Head Lifetime

CL – Cluster Lifetime

CM – Cluster Member

CML – Cluster Member Lifetime

COA – Center of Area

CS – Cluster Size

DSRC – Dedicated Short Range Communications

EC – Energy Consumption

ETSI – European Telecommunication Standards Institute

EV – Electric Vehicle

eWARPE – energy-efficient Weather-aware Route Planner for Electric Bicycles

FL – Fuzzy Logic

FLC – Fuzzy Logic Controller

FLS – Fuzzy Logic System

FuzzC-VANET – Fuzzy Logic-based Clustering Scheme over VANETs

GID – Geometric Intersection Description

GLOSA – Green Light Optimal Speed Advisory

GPS – Global Positioning System

HO – Handover

HTTP – Hypertext Transfer Protocol

I2V – Infrastructure-to-Vehicle

ITS – Intelligent Transportation System

LTE – Long Term Evolution

MAC – Medium Access Control

MANET – Mobile Ad-hoc Networks

MAP – map data

NoC – Number of Clusters

OBU – On-Board Unit

OECD – Organisation for Economic Co-operation and Development

RSCH – Relative Speed Among Cluster Heads

RSU – Road Side Unit

RSWC – Relative Speed Compared to the Cluster Head Within a Cluster

SAECy – Speed Advisory System for Electric Bicycles

SCH – Service Channel

SPaT – Signal Phase and Timing Messages

TI – Time Interval

UMTS – Universal Mobile Telecommunications System

V2I – Vehicle-to-Infrastructure

V2V – Vehicle-to-Vehicle

V2X – encompasses V2V, V2I and I2V

VANET – Vehicular Ad-hoc Networks

WAVE – Wireless Access for Vehicular Environment

WLAN – Wireless Local Area Networks

ABSTRACT

Lately, significant research efforts are put in the area of designing smart cities by both academia and industry focusing on making use of city facilities (buildings, infrastructure, transportation, energy, etc.) in order to improve people's quality of life and create a sustainable environment. Vehicular ad hoc networks (VANET) play a main role in supporting the creation of smarter cities. VANET are based on "smart" inter-vehicle communications and with the infrastructure via so called V2X communications (i.e. V2V – vehicle-to-vehicle and V2I/I2V – vehicle-to-infrastructure/infrastructure-to-vehicle). V2X communications demonstrated their huge potential when designing not only intelligent transportation solutions, but also green transportation solutions. Most of the research in this area targets a single class of VANET users represented by the car drivers. This thesis addresses different types of users, mainly the cyclists, as cycling is one of the most sustainable and green forms of transportation.

The thesis proposes three main solutions: **Speed Advisory System for Electric Bicycles (SAECy)**, an **Energy Efficient Weather-aware Route Planner solution for Electric Bicycles (eWARPE)** and a **Fuzzy Logic-based Clustering Scheme (FuzzC-VANET)** over vehicular networks. SAECy provides on-route assistance to the cyclists in order to improve their cycling experience and reduce the energy consumption in the particular case of electric bicycles. SAECy uses mainly I2V communication for obtaining traffic light related information, but also weather information. eWARPE provides off-route assistance to the cyclists in order to support them in avoiding the adverse weather conditions as much as possible, but also to save the battery in the particular case of electric bicycles. FuzzC-VANET is a generic clustering scheme dedicated for VANET that can be employed for information dissemination in SAECy's context in order to enhance its performances and to increase the accuracy of weather information. FuzzC-VANET is also a response to two of VANET main issues, stability and scalability, and extends its benefits on all the other classes of users (i.e. drivers of different types of vehicles) in VANET.

The performance of the proposed solutions was evaluated through multiple assessment techniques: experimental testing based on a real test-bed, interviews and online questionnaire to measure the need for the solution proposed and the interest of users and simulations based on highly realistic scenarios.The results clearly show improved performance of the proposed solutions in comparison with other similar state-of-the-art solutions.

Chapter 1

INTRODUCTION

1.1. Motivation

Nowadays, smart cities represent a hot research topic for both academia and industry. The goal of this research is to make use of city facilities (buildings, infrastructure, transportation, energy, etc.) in order to improve people's quality of life and create a sustainable environment [1]. There is a considerable number of initiatives/projects that relate to smart cities. For instance, in 2010, IBM has launched IBM Smarter Cities[1] challenge, aiming to support 100 cities around the world in addressing some of their critical challenges. In the meanwhile the program has been extended to include 116 cities to date and the program has already been scheduled to run in 2016, too. In September 2015, US Government launched a smart cities initiative[2] aiming to improve cities services.$ 160 million were allocated for this initiative. In 2010, in the context of "Strategic Energy Technologies Information System", European Commission has launched the European Initiative on Smart Cities[3] with the main goal of reducing the greenhouse gas emissions by 40% by 2020, as compared to 1990. This initiative addresses four dimensions of the city: buildings, heating and cooling systems, electricity and transport. Related to transport, the declared aim is to build intelligent public transportation systems based on real-time information, traffic management

---

[1] IBM Smarter Cities Challenge. http://smartercitieschallenge.org/

[2] White House official website, https://www.whitehouse.gov/the-press-office/2015/09/14/fact-sheet-administration-announces-new-smart-cities-initiative-help

[3] European Commission, SETIS project website, http://setis.ec.europa.eu/implementation/technology-roadmap/european-initiative-on-smart-cities

systems for congestion avoidance, promote cycling and walking. In 2011, European Commission has launched the Smart Cities and Communities initiative, that covered only energy, but in 2012 this initiative emerged into the European Innovation Partnership for Smart Cities and Communities[4] that covered also the ICT (Information and Communication Technology) and transport sectors.

According to [2], the air pollution kills more than 3 million people across the world every year, about 3.5 million to be more exact, and causes health problems from asthma to heart disease for many other people. In the countries adhering to the *Organisation for Economic Co-operation and Development* (OECD), the cost of the health impact of air pollution was about 1.7 trillion dollars in 2010, data that included deaths and illness [2]. Road transport accounts for about 50% of this cost. In US for instance, road transport accounts for 27% of CO2 emissions [3]. Therefore, it is not surprising that smart transportation that relates to intelligent and green transportation solutions represents a key dimension of smart cities research.

Cycling is considered one of the most sustainable and green forms of transportation, a fact acknowledged by the smart transportation initiatives in particular and smart cities initiatives in general. As such, promoting cycling is listed as main objective by both the European Initiative on Smart Cities and European Innovation Partnership for Smart Cities and Communities [4], [5]. Cycling can be the answer to many problems of the nowadays society such as greenhouse gas emissions, traffic congestion, parking, etc. Therefore, promoting cycling was not only an initiative in the new context of smarter cities, but also many governments had this on their agenda long before the concept of smart cities was introduced. National policies were proposed and adopted with the sole aim of promoting cycling in quite a considerable number of countries more than ten years back [6] – [8]. As a result, cities have started to become more cycling friendly, new cycling facilities have been built. One of the consequences of these policies was the successful adoption of bike sharing schemes that currently is spread in cities all over the world [9], [10]. One of the actions enlisted by these cycling policies is the development of advanced telematics, namely systems that bring benefits to cyclists, thus encouraging them to cycle more. As noticed by the European Innovation Partnership for Smart Cities and Communities, but also when evaluating the implementation of these cycling policies (e.g. the audit of Dublin City Council on the Ireland National Cycle Policy Framework in Dublin [11]) not many steps have been made in the direction of providing such technical solutions. In European Innovation Partnership for Smart Cities and Communities one of the listed goals is to employ ICT in designing technical solutions that promote and support cycling.

[4] European Commission, Smart Cities project website, http://ec.europa.eu/eip/smartcities/index_en.htm

Vehicular ad hoc networks (VANET) or simply vehicular networks are considered to play a crucial role in supporting the creation of smarter cities. VANETs are based on “smart” inter-vehicle communications and information exchange with the infrastructure via so called V2X communications (i.e. V2V – vehicle-to-vehicle and V2I/I2V – vehicle-to-infrastructure/infrastructure-to-vehicle). VANETs demonstrated their huge potential when designing not only intelligent transportation solutions and traffic management systems [1], but also green transportation solutions [12], [13]. The latter category was mostly focused on V2X communications aiming to reduce fuel consumption and gas emissions. With the increased popularity of electric vehicles (EV), the focus has been recently moved on how V2X communications can help EVs save energy. So far, these solutions targeted exclusively the electric cars, but there are also electric bicycles. Moreover, most VANET research is dedicated to cars, although bicycles are also important in the context of vehicular networks. Therefore there is a need to integrate the research in this space [14], [15]. Ongoing efforts are made in this direction in the context of the EU FP7 VRUITS[5] project for instance. To the best of our knowledge the only VANET-based application that focuses on bicycles and that was proposed so far is the cooperative automatic emergency breaking system that help the driver in avoiding possible collisions with the cyclists – in emergency cases and if the driver does not take any action, the car is stopped automatically –, while warning the cyclist at the same time. This is proposed in the context of VRUITS and some preliminary tests/trials were released in this project in 2015 [16].

1.2. Problem Statement and Proposed Solutions

As seen in the previous section, there is an ongoing effort of promoting cycling, one of the greenest mean of transportation. However, the efforts put so far are not enough to meet for instance the targets imposed by national policies [11], and therefore other approaches are strongly encouraged. One gap identified is the lack of technical solutions aiming to address the needs of the cyclists, especially regarding the factors that demotivates people from cycling. There are several such factors demotivating people from cycling among which are: safety, weather conditions, road conditions, etc. [17].

Lately, a modern form of cycling which uses electric bicycles has gained popularity. Research reports show that there is and there will be a worldwide increase of electric bicycles in the next years [18], [19]. Electric bicycles have many benefits. Like traditional bicycles, electric bicycles are environmentally friendly and are associated with very low gas emissions when compared to other vehicles. According to a study performed in 34 major cities in China [20], the CO_2 emissions of electric bicycles are between 14-27g/pkm (passenger kilometre), about 10 times less than when compared to conventional vehicles and 9 times less than when compared to electric

[5] EU FP7 project VRUITS website, http://www.vruits.eu/

cars. In a top of 7 greenest vehicles[6], electric bicycles are situated second with 5-30g CO_2e/km depending on the type of fuel used for the generation of electricity, just after the traditional bicycles that have also associated some CO_2 emissions if their production is considered. Electric bicycles improve the traditional riding experience, especially for the people who are not so fit, require significantly less effort and decrease time travel. In comparison with other green vehicles, electric bicycles have lower energy cost per distance travelled [21] and avoid other additional costs (e.g. parking, insurance, registration, etc.). Consequently, it is not a surprising fact that electric bicycles are the most popular of all EVs and their popularity is increasing.

Similar factors that affect traditional cycling (i.e. safety issue, weather conditions, road conditions) also affects electric bicycles. Weather conditions are affecting the cyclists the most among the traffic participants, bad weather conditions, such as wind and rainfall, are not pleasant for the rider and they are main de-motivators for cycling [17]. Moreover, electric bicycles have a weak point related to the same aspect that makes them capable of providing some of the already mentioned advantages to the cyclists: the battery. Due to the battery, electric bicycles are in general heavier than traditional bicycles with a varying extra-weight of 2 to 5kg that corresponds to the battery weight. Furthermore, the battery has a relative short autonomy, in the 16km-50km range (i.e. affected in time by the number of charges) and a long charging cycle between 2 and 6 hours [21]. This relative long period makes performing research to find power-saving solutions for electric bicycles of very high interest.

In order to address these challenges this thesis proposes a VANET-based ***Intelligent Advisory Solution for Bicycle Eco-riding and Eco-routing*** that makes use of a **Speed Advisory System for Electric Bicycles (SAECy)**, and an **Energy Efficient Weather-aware Route Planner solution for Electric Bicycles (eWARPE)** that provides off-route assistance to the cyclists. Moreover a **Fuzzy Logic-based Clustering Scheme (FuzzC-VANET)** over VANETs is proposed. SAECy provides on-route assistance to the cyclists aiming to improve the cycling experience and reduce the cyclists' effort in the case of bicycles in general and the energy consumption in the particular case of electric bicycles. SAECy is based mainly on the traffic light to vehicle (i.e. I2V) communication for obtaining traffic light related information, but also weather information. eWARPE provides off-route assistance to the cyclists supporting them to avoid the adverse weather conditions and to save the battery in the case of electric bicycles. FuzzC-VANET is a generic VANET clustering scheme that can be employed for information dissemination in SAECy's context in order to enhance its performances and to increase the accuracy of weather information. As mentioned in the previous section, VANET is an emergent technology with huge potential in the context of smart transportation in general and in supporting cycling in particular. However, before

[6] Shrink That Footprint website, http://shrinkthatfootprint.com/7-green-vehicles

the full adoption and deployment of this technology, there are several challenges that need to be addressed. Scalability and stability are among these main challenges [1], [22], [23]. FuzzC-VANET is also a response to these issues.

Thus, the proposed Intelligent Advisory Solution for Bicycle Eco-riding and Eco-routing provides on-route and off-route assistance for electric bicycles in particular and bicycles in general, energy savings being translated into reduced effort on behalf of the cyclist. FuzzC-VANET extends its benefits on all the other classes of users (i.e. drivers of different types of vehicles) in vehicular networking.

A more detailed overview of the solutions proposed which represent contributions of this work, is given in the next section.

1.3. Contributions to the State of the Art

The contributions of the research work presented in this thesis are as follows:

- **SAECy** represents a novel vehicular communications-based speed advisory system dedicated to electric bicycles. The solution belongs to the class of Green Light Optimal Speed Advisory (GLOSA) systems based on the I2V communications and is the first GLOSA system dedicated to electric bicycles. The proposed solution recommends strategic riding (i.e. the appropriate speed) when bicycles are approaching an intersection to avoid high power consumption scenarios. Moreover, the approach also includes an innovative Fuzzy Logic-based wind-aware speed adaptation policy as among all the other vehicles, bicycles are mostly affected by the wind. The benefits of the solution translate not only in energy-efficiency, but also in an increased user quality of experience, as the waiting times at traffic lights are reduced or even avoided.

- **eWARPE** represents a step forward for the cycling route planners, going beyond planning the route itself (how to get from point A to point B). eWARPE is planning the optimal departure time for the route: when to leave from point A towards point B on the previously planned route. The solution makes use of the weather information in order to recommend the departure time that allows the cyclist to avoid the adverse weather conditions as much as possible and to increase the energy savings of the electric bicycle.

- **FuzzC-VANET** is a general Fuzzy Logic-based CH-based clustering scheme over VANETs, the first, to the best of our knowledge, to employ Fuzzy Logic as the main decision tool in choosing the CH. Creating a clustered network model, this scheme solves some of the main issues of VANET, namely scalability and stability issues. As a consequence of this, the V2V communications in the network will have an increased throughput as it will be seen in the next chapters. Solving the scalability and stability issues, the clustering scheme creates a base for MAC, routing protocols and information dissemination. In the latter case, it is known that clusters contain local-based and highly up-to date information. In disseminating weather information, these two characteristics are highly important and in this context it is to be mentioned that there are quite a few applications that are based on VANETs capacity of providing weather information, one of this being our previously proposed SAECy. An instantiation of FuzzC-VANET is also proposed that is based on a hybrid VANET architecture.

1.4. Structure of the thesis

The thesis was structured in six chapters as follows:

- **Chapter 1** presents motivation for the research work conducted, identifies the problem and gives a short overview of the proposed solutions. The chapter also presents the main contributions of this thesis.
- **Chapter 2** introduces technical background for the work presented in this thesis.
- **Chapter 3** presents a comprehensive survey of the current works from the areas of research that relate to the work presented in this thesis: green transportation solutions, FL-based solutions and clustering algorithms, all in the context of VANETs.
- **Chapter 4** presents the solution architecture and algorithms of SAECy, eWARPE and FuzzC-VANET, the solutions proposed in this thesis.
- **Chapter 5** presents performance evaluation of the proposed solutions.
- **Chapter 6** draws conclusions of the thesis and presents possible future work directions.

Chapter 2

TECHNICAL BACKGROUND

This chapter describes the main technical concepts that are used throughout the thesis. It starts by introducing the vehicular networks, their architectures, main characteristics and applications. Then, it introduces Fuzzy Logic and clustering concepts in the context of vehicular networks, as the contributions of this thesis are based on them.

2.1. Vehicular Networks

2.1.1. Vehicular Networks Overview

Vehicular Ad-hoc Networks (VANET) or simply vehicular networks are a novel class of mobile ad-hoc networks (MANET), where the mobile nodes are in fact vehicles. Although VANETs are a class of MANETs, they have specific characteristics that differentiate them, characteristics that will be discussed in section 2.1.4. Vehicular networks are very important as they play a crucial role in supporting the creation of smart cities. Smart cities represent a hot research topic both for academia and industry. The main purpose of smart cities is to make use of city facilities (e.g. buildings, infrastructure, transportation, energy, etc.) in order to improve people's quality of life while creating a sustainable environment. In this context, smart transportation, as a fundamental dimension of smart cities, relates to both intelligent and "green" transportation solutions. VANETs demonstrated their huge potential when designing not only intelligent transportation systems (ITS) and traffic management systems [1], but also green transportation solutions [12], [13].

VANETs are based on communications between vehicle to vehicle (V2V), vehicle to infrastructure (V2I) or infrastructure to vehicle (I2V), generally referred as V2X communications. This type of communications is supported by a novel type of wireless access paradigm called Wireless Access for Vehicular Environment (WAVE). WAVE includes a suite of standards dedicated to communication in vehicular environments which will be presented in the next section. Note that V2X communications refer exclusively to the communications supported by WAVE (e.g. if the communication between a vehicle and another vehicle is done via another type of access technology such as WiFi for instance, this communication will not be referred to as V2V communication). In addition to V2X communications, other types of technologies are also used in supporting vehicular communications. Depending on how these VANET enabling technologies are employed in vehicular communications, three types of VANET architectures are defined: pure ad-hoc, pure WLAN/cellular and hybrid [23], [24]. Figure 2.1(a) presents the ad-hoc architecture, while Figure 2.1 (b) and Figure 2.1(c) present pure WLAN/cellular, respectively hybrid architecture.

In the ***ad-hoc architecture***, there are V2V communications only, without any infrastructure support. This situation is widely encountered as infrastructure and wireless access points are not everywhere and their deployment is limited by the cost or geography.

In the ***WLAN/cellular architecture***, cellular base stations and WLAN access points facilitate vehicles' connection to the Internet and provide support for vehicular communications-based applications. In this type of architecture the vehicles do not have support for direct communication with each other in a distributed manner with few exceptions. An exception to this direct communication approach is the case when the vehicles are travelling in groups and they can inter-communicate via WiFi. However, if the vehicles do not have similar mobility pattern, communication via WiFi is not possible. Cellular networks have a centralized architecture so they do not provide any native support for direct communication between vehicles and all communications are done via infrastructure. Among the cellular technologies, Long Term Evolution (LTE) appears to be the most promising enabling technology for vehicular applications due to the high data rates provided, support for high mobility (up to 350km/h) and high market penetration – LTE was confirmed as the fastest developing mobile system technology ever[7]. However, so far, it appears likely that LTE alone is not capable of supporting vehicular applications, especially applications that require an intense exchange of information between vehicles or during rush hours [26]. Moreover, there are already many studies showing that the huge growth in data will be almost impossible to be

[7] GSM/3G Stats, GSA website, http://www.gsacom.com/news/statistics, accessed November 2015

supported by the cellular networks at all in the near future [27]. Therefore, even if cellular technology is used in the VANET architecture, it is preferably to limit the amount of communication via these technologies as much as possible. In this context, solutions such as clustering can be successfully employed to limit cellular network communications.

In the ***hybrid architecture*** all types of communications are present. Vehicles can talk to each other and exchange information (V2V communications), but also can communicate with the fixed infrastructure that is potentially deployed alongside the road and is referred to as *road side units* (RSU) (V2I) or with access points, or wireless towers (WLAN, cellular). This is the most complex architecture and provides support for complex applications, for instance infotainment applications which require rich content are based on this type of architecture, but also complex traffic management systems.

In VANET architectures the communication capabilities of a vehicle are provided by an in-vehicle component referred to as the *on-board unit* (OBU) that can have multiple network interfaces (e.g. V2X, UMTS, LTE, etc.). Note that this component was envisioned to be integrated in cars by the car manufactures, but OBU can stand for different devices with wireless capabilities such as the driver's smartphone because WAVE technology can be enabled in smartphones as stated in [137]. This aspect is facilitating the integration of two-wheel vehicles (i.e. bicycles, electric bicycles, motorcycles) in VANET-based solutions.

OBU also supports intra-vehicle communication needed to collect data from the vehicle's sensors and devices, data that is then used in the applications enabled by VANET. Data processing or any other computational processing is done by a unit called *application unit* (AU). Note that these units can be physically integrated or separately (or they can consist of a group of components) being defined as such from a conceptual point of view. Most VANET applications assume that the position of the vehicle is known, so a GPS or other positioning system is considered to be either integrated in OBU or complement OBU.

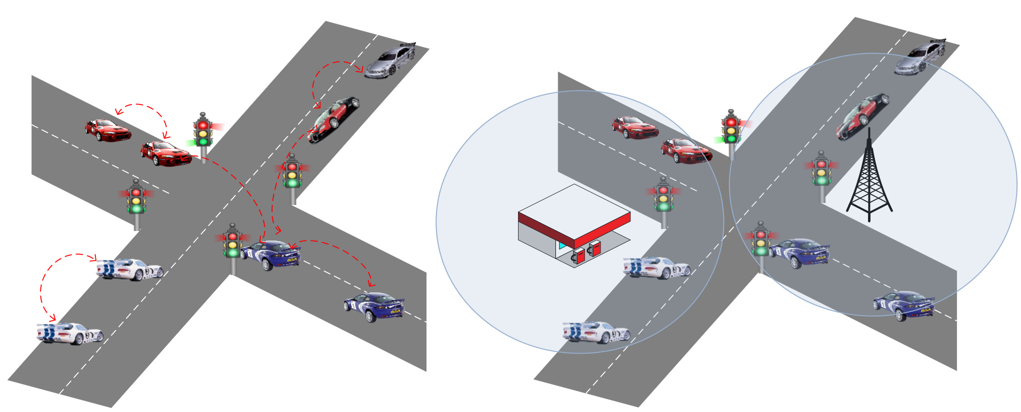


Figure 2.1. (a) Ad-hoc Architecture (b) WLAN/cellular Architecture

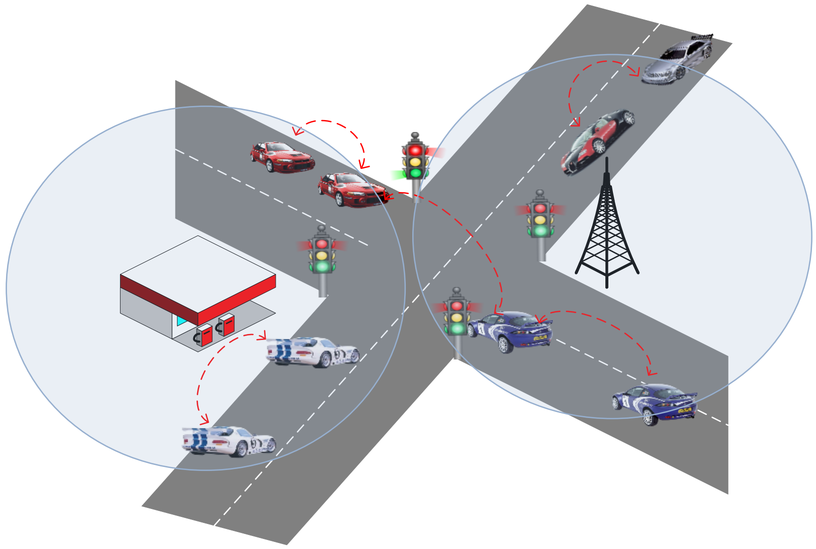


Figure 2.1. (c) Hybrid Architecture

2.1.2. Supporting Standards

V2X communications are supported by many standards and among the main and the most important are the ones developed by IEEE under the name: Wireless Access for Vehicular Environment (WAVE). WAVE contains the following standards dedicated to vehicular environment [28]: IEEE 802.11p and IEEE P1609.x standards. IEEE 802.11p, developed to provide wireless access to vehicles, is a new amendment of IEEE 802.11 that was ratified in July, 2010 [29]. Its aim was to make IEEE 802.11 suitable to the ever-changing transportation environment and able to deal with very short latencies. IEEE 802.11p was based upon the allocation of the Dedicated Short Range Communications (DSRC) spectrum band. This initiative started in USA in 1999 and has allocated dedicated spectrum of frequency to be used exclusively by V2X communications. Seven channels of 10MHz in the 5.9GHz range are allocated for use in DSRC/IEEE 802.11p standard. Figure 2.3 presents the channel assignment in North America: out of the 7 channels, 6 are service channels (SCH), while the one left is the control channel (CCH) which is reserved for system control and safety messages. One SCH channel is dedicated to safety messages as well, whereas the rest of SCHs are mainly used to exchange non-safety and larger data. The next steps were the allocation of dedicated spectrum in Europe and Japan. Figure 2.4 shows the DSRC spectrum allocation across the world.

In Europe, the European Telecommunication Standards Institute (ETSI) defined a profile standard of IEEE 802.11p [138] in order to adapt the 802.11p to the European spectrum. Moreover, ETSI defined in a standard the ITS station and communication reference architecture that covers the whole network stack [139]. Figure 2.2 presents this reference network stack. Note a new layer is introduced, **Facilities layer**, in comparison to traditional network layers such as Network and Transport, Access Technologies. This provides a collection of functions in order to support ITS Applications. It provides data structures for storage and aggregates and maintains data that is either local (e.g. from vehicle's sensors), or received from the network.

There are two other components in Figure 2.2 : **ITS Management** and **ITS Security**. The first one is in charge with ITS station configuration and with the cross-layer exchange of information, while the second one is in charge with security and privacy issues over the layers.

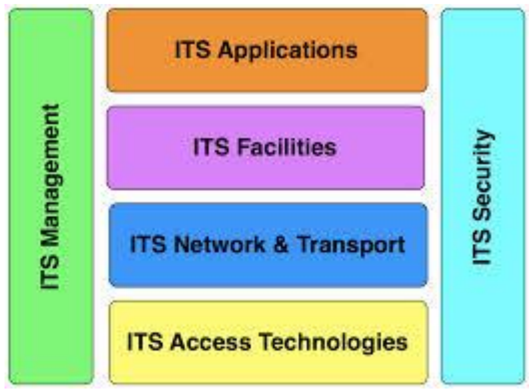


Figure 2.2. Reference Protocol Stack of an ITS Station [139]

Similar to ETSI, IEEE WAVE developed the IEEE P1609.x suite of standards covering the entire VANET scope of services from application down to the MAC layer, as IEEE 802.11p already covers the Physical and MAC layers. These are as follows:

- IEEE P1609.0 [135] is the WAVE *Architecture* describes the WAVE architecture and services necessary for multi-channel DSRC/WAVE devices to communicate in a mobile vehicular environment.
- IEEE P1609.1 [30] is the WAVE *Resource Manager* standard, defining the interfaces and services of WAVE applications and the format of data messages.
- IEEE P1609.2 [31] is the WAVE *Security Services* for Applications and Management Messages standard that defines the WAVE security: anonymity, authenticity and confidentiality and also the exchange of messages.
- IEEE P1609.3 [32] is the WAVE *Networking Services* that defines routing and transport services. It provides description and management to the protocol stack, network configuration management and also provides the transmission and reception of WAVE short messages.

- IEEE P1609.4 [33] is the WAVE *Multi-channel Operations* that provides enhancements to the IEEE 802.11p MAC in order to support frequency band coordination and management.

- IEEE P1609.6 [133] is the WAVE *Remote Management Services* that will provide inter-operable services in order to manage WAVE devices that comply to IEEE Std. P1609.3.

- IEEE P1609.11 [134] is the WAVE *Over-the-Air Electronic Payment Data Exchange Protocol for ITS* that defines the services and secure messages formats dedicated to secure electronic payments.

- IEEE P1609.12 [136] is the WAVE *Identifier Allocations* that provides identifier values allocated to WAVE systems.

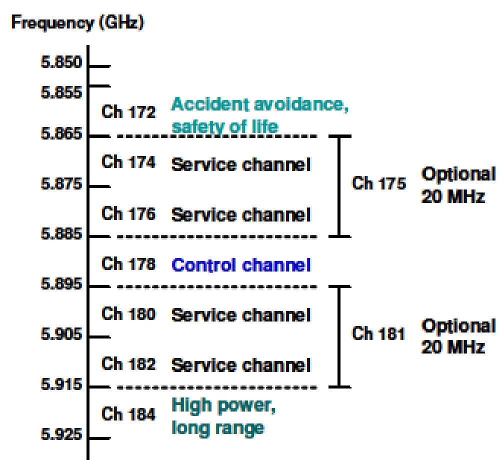


Figure 2.3. DSRC Channel Assignment in North America [132]

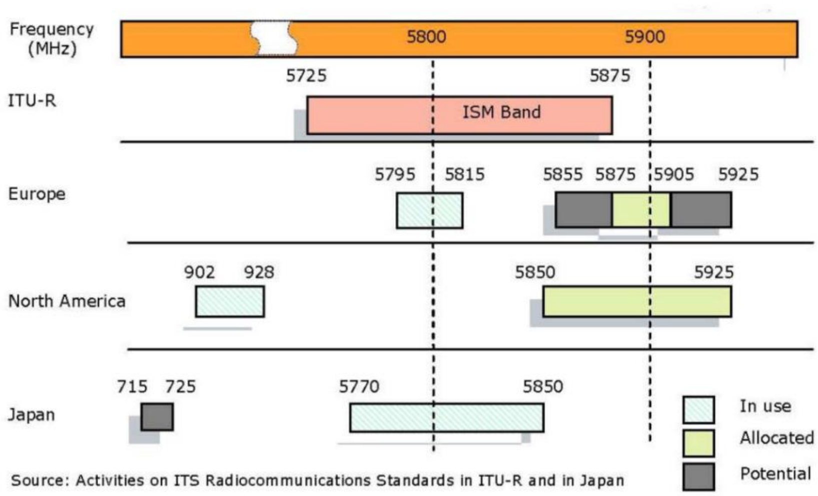


Figure 2.4. DSRC Spectrum Allocation Worldwide

There are also specifications in the proposed standards for various types of messages such as cooperative awareness messages (CAM) [139], signal phase and timing messages (SPAT) [140],

[142], and mechanisms for dissemination of these messages such as the publish-subscribe mechanism specified in CEN/ISO 17429 [143]. The SPAT message details are further presented as it is of particular interest in the context of this thesis.

SPAT is specified both by IEEE and ETSI, but its format was standardized by SAEJ2735 [140] and accepted by the previous standardization bodies in their proposed architectures for VANET/ITS. SPAT message reflects the current traffic light state of all lanes, in all approaches, in a single intersection. Figure 2.5 presents the simplest type of a SPAT message that consists of 2 states: one for the current active lanes (green light) and one for the current inactive (red light). Lanes 1 and 2 are the active ones as for these the *SignalState* parameter of the SPAT message indicates the green light. Note that in the figure the parameters are not assigned the specific values (i.e. as they are in the message), but the values are explained in detail. *TimeToChange* parameter indicates the time until the current colour is changing, in the given example, for lanes 1 and 2 is 12.3 seconds till the green light will change.

SPAT messages are usually accompanied by map data (MAP) [141], [142], also known as Geometric Intersection Description (GID) when refers to the dynamic description of an intersection that provides map data information useful to the applications using SPAT messages.

```
SPaT Message
    Msg id =  0x0c  (indicates a SPaT message)
    SPaT id = TBD   (indicates a unique value for this intersection)
    States
            State #1
                    Lane Set                (list of lanes this applies to)
                            1, 2
                    Movement State (signal state or pedestrian state)
                            SignalState   = Green light
                    TimeToChange          =  12.3 seconds
                    YellowSignalState     =



         State #2
                 Lane Set u(list of lanes this applies to)
                         3,4.5.6, etc...
                 Movement State (signal state or pedestrian state)
                         SignalState   = Red light
                 TimeToChange          =  Indeterminate for this state
                 YellowSignalState     =
```

Figure 2.5. SPAT message example

2.1.3. VANET Applications

A large plethora of applications have been proposed for VANETs. These can be categorized in three big classes [34]: active road safety applications, traffic efficiency and management applications and infotainment applications.

Active road safety applications aim to provide a safer driving environment by reducing the probability of accidents and preventing the loss of lives. Such applications are traffic signal violation warning, emergency electronic brake light, pre-crash sensing, lane change warning, cooperative forward collision warning, etc. [35] – [40]. These are all pro-active approaches that are trying to avoid accidents. Reactive safety approaches based on VANETs can be developed in the context of emergency systems. "Green" routes for emergency vehicles can lead to saving human lives. In a recent survey, Martinez *et al.* [41]emphasized both the great potential of V2I/V2V communications in enhancing the emergency services and the need of designing systems based on this type of communications to ensure efficient emergency service delivery. The architecture and principles of a complete solution, a VANET-based traffic management system ensuring "green" routes for emergency vehicles has been proposed in [42].

Traffic efficiency and management applications aim to improve the overall efficiency of transportation by managing the navigation of the vehicles via cooperative co-ordination (e.g. cooperative adaptive cruise control [43]). Also, they aim to improve not only the overall efficiency, but also the efficiency per vehicle via speed management applications (e.g. avoiding stopping to the intersection, [48], [49]). This latter type of applications are situated somewhere at the border between safety and infotainment applications.

Infotainment applications are applications that are not directly related to traffic safety or efficiency, but they are designed for the needs and comfort of vehicular users. These applications can be split into two big classes: entertainment applications and driver assistance applications.

Entertainment applications include solutions for different service delivery such as multimedia delivery and live video streaming over VANETs [44], [45], file-sharing and gaming platforms over VANETs [46], [47].

Driver assistance applications comprise many VANET-based solutions. This type of applications provide the driver with information useful in driving process, but not only (e.g. applications that provide valuable information for driver, such as price of fuel or closest charging station, are also included). Example of such applications are routing applications [50] - [53], free parking discovery applications [54], [55], tolling applications [56], [57], applications that give driving advice based on certain criteria (e.g. how to drive in certain conditions in order to reduce gas emissions fuel or energy consumption, [48], [49]), etc.

2.1.4. VANET Characteristics and Challenges

VANETs have specific characteristics that differentiate them from any other type of ad-hoc networks. Some of these characteristics are very attractive for the researchers, while others are creating new technical challenges that need to be addressed. To the first mentioned class subscribe the following aspects:

Theoretical unlimited power, due to the fact that the vehicle-node is capable of generating itself power while moving. In the case of the mobile nodes of classic MANETs, power is a very serious issue. However, this characteristic is not applicable in the case of EVs, where energy preservation is vital for increasing the range of the EVs.

High computational and storage capabilities: unlike the handheld devices in the classic MANETs, vehicles can afford significant computational, storage and communication capabilities. This capability is partially made possible by the previous mentioned characteristic (i.e. power) and is not applicable in general for two-weels vehicles where OBU is replaced usually by a handheld device.

Predictable mobility is possible in VANETs due to the fact that vehicle movement is constrained by the roads, traffic regulations and driver behavior. So, given parameters such as the current position, current speed, route, average speed and/or learning driver behavior, it is possible to predict the next position of the vehicle. On the contrary, the mobility of the classic MANETs nodes is very hard to predict.

To the challenging class of VANET characteristics subscribe the following issues:

High mobility. Vehicle-nodes have very high speed compared to the nodes in MANETs. In highway scenarios speeds of up to 300km/h may occur, while in city scenarios speeds of up to 50-70km/h are encountered.

Rapidly changing topology. The aforementioned high mobility of VANET nodes leads to a frequent link disconnection between the vehicle-nodes and consequently to a rapidly changing technology of VANETs.

Diversity of conditions that mainly refer to the diversity of the network density that can be very sparse or on the contrary, very dense. In a city scenario, especially during rush hours, the network is extremely dense, while in a highway scenario the network can be very sparse.

Frequently disconnections in the network that are mainly caused by the two previous mentioned characteristics. Road dead-ends is another factor that can produce frequent disconnections in VANETs.

Potentially large scale. VANETs are networks with a potential high number of nodes. There is no limitation in terms of number of nodes, as it is in the case of other networks, so vehicle-nodes can potentially expand over the entire road network.

Diversity of applications. As presented in the previous section, a large plethora of applications have been envisioned for VANET: traffic safety applications, traffic management and efficiency applications, infotainment applications from multimedia applications to driver assistance applications. The requirements of these applications are as diverse as their range is. Consequently, VANETs dedicated technology needs to be designed so these networks can cope with all this diversity of applications.

In the presence of these characteristics, some of the main technical challenges of VANETs are imposed in the context of MAC protocols, security, routing and data dissemination protocols and service delivery architectures due to the frequent disconnections [23]. MAC protocols are considered to be a key issue in the design of VANETs [34]. Efficient MAC protocols need to be specifically designed for VANETs in order to cope with the high dynamic environment caused by the high mobility and rapidly changing topology. In addition, MAC protocols designed for VANETs need to fulfill the requirements of all the diverse application types. As such, they need to be able to provide quality of experience (QoS) for non-safety applications (such as infotainment applications) and reliability for safety applications. Routing protocols in VANETs need to be able to cope with their rapidly changing topology, but also with the different types of networks densities and diversity of applications. Data dissemination in the conditions of potentially large number of nodes must take into account the efficient usage of the available bandwidth. Data aggregation addresses this issue in the context of data collection, avoiding the dissemination of the similar information in the network. In such a dynamic environment, security protocols become a challenge as an optimal trade-off should be found between safety and complexity.

Socio-economic challenges (i.e. the cost of the infrastructure needed for the deployment of VANET solutions and market penetration of V2X communication technologies) were included as well among the main challenges of VANETs [58]. Two solutions were proposed for the latter problem that either enforce a regulative order, or deploy user-oriented applications in order to advertise the added value of the technology [23], [59]. While steps towards imposing the adoption of the V2X communication technologies for new vehicles through regulation have been made (e.g. in USA), it is likely there will be long periods of time during which the penetration rate of these technologies will be very low (e.g. in USA alone it will take more than 10 years to acquire 100% penetration rate if a new technology will start to be deployed on vehicles now) [59]. In this context, the deployment of user-oriented applications appears to be the best solution to fasten the market penetration of these technologies.

2.2. Fuzzy Logic and Vehicular Networks

2.2.1. Introduction to FL

FL, first introduced in 1965 by Prof. Zadeh, was defined as an "*attempt to mimic human control logic*" [60]. FL, an extension to the binary logic, uses continuous values in the [0, 1] interval and allows for the usage of linguistic variables. Thus, FL is the only mathematical framework able to do the reasoning based on linguistic information and simulate the human reasoning. Moreover, FL is able to deal with uncertainty and imprecision, model non-deterministic problems and also deal with multiple parameters that describe the problem modeled. This makes it suitable to modeling and solving a wide-range of real-world problems. FL had a huge impact, being used on a wide scale and in many domains, including engineering, medicine, science, and business. Applications of FL include: intelligent control systems, decisional systems, information systems, pattern recognition systems (e.g. image processing, machine vision), prediction systems or systems that detect different conditions (e.g. faults detection systems in automotive or process diagnostics systems). There are many well-known, successful FL applications deployed in the industry. Examples of such applications are: Sendai subway control (Hitachi), aircraft control (Rockwell Corporation), cruise control (Nissan vehicles), self-parking model car (Tokyo Technology University), elevator scheduling (Hitachi, Mitsubishi, Fujitech), stock market analysis (Yamaichi Securities), TV picture adjustment (Sony), video camera autofocus (Sanyo, Canon) [61]. Besides these well-known industrial applications, many of current research works in different areas further explore the huge potential of FL. Telecommunication area subscribes to these various areas where FL was successfully employed in solving various challenges and its potential is still being explored. Some of the first FL applications in telecommunications have been analyzed in [62] and include: queue modeling, faults and other conditions detection in the telephone networks, decision making in an automated very high frequency, dynamic assignments of radio channels and control applications in the context of asynchronous transfer mode networks such as congestion or call admission control. Since then, the telecommunication field evolved, but FL continued to be applied in new areas of Telecommunications. The newest applications are in the context of wireless networks (e.g. decision making in network selection [63]), wireless sensor networks (e.g. decision making in energy-aware routing and clustering or security protocols [64]) and more recently in the context of cognitive radio networks (e.g. FL reasoning can be employed in inferring the "cognition"/knowledge of cognitive radio networks [65], [66]) and vehicular networks.

2.2.2. Fuzzy Logic Concepts

2.2.2.1. Linguistic Variables

Cognitive scientists state that humans tend to think in terms of conceptual patterns and mental images rather than in terms of any numerical quantities. In consequence, natural language is perhaps the most powerful form of describing a problem that requires solving or reasoning. To sustain this idea, the authors from [67], give an interesting example: a fragment from "A Tale of Two Cities" of Charles Dickens is translated into a mathematical language. The fragment *"It was the best of times, it was the worst of times…"* became *"The time interval x was the period exhibiting a 100 percent maximum of possible as measured along some arbitrary social scale, [and] the interval x was also the period of time exhibiting a 100 percent minimum of these values as measured along the same scale"*, which is a mathematical paradox. This power of natural language has largely remained untapped in mathematical paradigms until the introduction of FL that allows reasoning in terms of natural language expressed by means of linguistic variables. Zadeh, the founder of FL, defines linguistic variables as "*variables whose values are not numbers, but words or sentences in a natural or artificial language*" [68]. The values of linguistic variables are called *(atomic) terms*.

Example 1: In the vehicular networks space, s*peed* is one of the most common linguistic variables. So far it was used to refer either to the speed of the vehicle (e.g. [69], [70]) or to the difference in speed of the vehicles (e.g. [71]). The terms of speed linguistic variable in [71] are *low*, *medium* and *high*.

In linguistics, often to a fundamental term is associated a modifier like: *very, extremely, almost, approximately, slightly*. FL allows for the usage of modifiers in association to the atomic terms of a linguistic variable. In FL, the modifiers are named *linguistic hedges.*

Example 2: Following *Example 1*, in [69], the linguistic variable speed has as main terms *slow*, *medium* and *fast*. Additionally, a linguistic hedge is applied to slow: very slow. Thus the speed is either: *very slow*, *slow*, *medium* or *fast*.

2.2.2.2. Fuzzy Sets and Membership Functions

A fuzzy set is a fundamental concept in FL and represents a generalization of an ordinary set, called in FL, *crisp set*. A crisp set is defined either by listing its elements or by defining the condition that makes an element x member of the set. For any value x there are only two possible statuses: member of the set or non-member. Thus, the statement: "x is member of set A" can be either false, either true. In binary logic the false is assigned value 0 and true is assigned value 1. Taking into considerations all the aforementioned aspects, a function that illustrates the membership relation (i.e. membership function) can be defined. Let A be a crisp set and $\mu_A(x)$ its membership function described by eq. (2.1).

$$\mu_A(x) = \begin{cases} 0, & \text{if x is not a member of set A} \\ 1, & \text{if x is member of set A} \end{cases} \tag{2.1}$$

A fuzzy set F, is described exclusively by its membership function $\mu_F(x)$ that unlike a membership function describing a crisp set, can take more than 0 and 1values, can take any values in the interval [0, 1]. In this case, $\mu_F(x)$ shows the degree of truth of the element x being member of the set F. This is how fuzzy sets extend the crisp sets and fuzzy logic extends the binary logic from {0, 1} to [0, 1]. The formal definition of fuzzy set F is expressed as a set of pairs (x, $\mu_F(x)$) as in eq. (2.2).

$$F = \{(x, \mu_F(x)) | \, x \, €X, \, \mu_F(x){:}X \rightarrow [0,1]\} \tag{2.2}$$

where X is called in FL the *universe of discourse* and defines all the possible values that x can take.

A fuzzy set describes what the atomic term of a variable signifies in a mathematical language.

Example 3: In *Example* 1, the linguistic variable *speed* is used in the solution presented in [71] having the following atomic terms: *low, medium* and *high*. The interpretation of these terms is defined by the fuzzy sets associated to these terms. In Figure 2.6 the fuzzy sets of *low* and *medium* terms are presented as described in [71]. Note that trapezoidal membership functions are used.

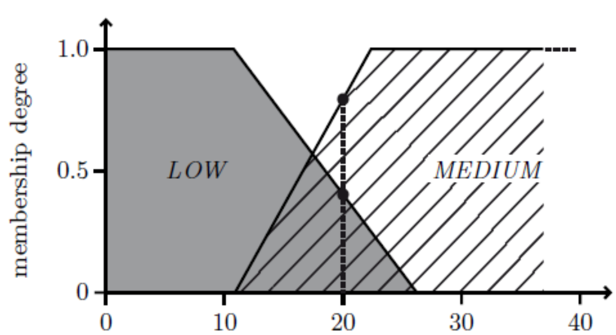


Figure 2.6. The Fuzzy sets of low and medium terms of *speed* ([71])

The most popular membership functions in FL are: singleton, triangular, trapezoidal and Gaussian, being named after the geometric figures that the pairs (x, $\mu_F(x)$) are shaping. Note that less computational complex membership functions beside the singleton are triangular and trapezoidal functions. These are the most common functions used in engineering applications [61].

Solving problems in FL implies a *FL inference system* or a *FL controller*, concepts introduced by Mamdani in 1975 [72]. Note that these two terms, FL inference system or simply FL system (FLS) and FL controller (FLC) are basically defining the same concept. On a common basis, these terms are not differentiated, but on a strict basis there is a difference: a FLC is a specific type of FLS that is designed for control purpose. There are several types of classic FLSs that were imposed as models therefore they are also called fuzzy models. Among these, the one introduced by Mamdani in 1975 is the most popular. Other fuzzy models were introduced in time, such as Sugeno fuzzy model [73], also known as Takagi and Sugeno fuzzy model, Tsukamoto fuzzy model [74] and Larsen fuzzy model [75]. FLSs can have multiple(M)/single(S) inputs(I) and multiple(M)/single(S) outputs(O) in any combination (i.e. FLSs can be MIMO, MISO, SIMO, SISO systems). MISO FLSs are the most common ones and therefore this type is further considered for exemplification. The architecture of a classic MISO FLS is presented in Figure 2.7, but its components and their description adapted to the number of inputs/outputs are valid for all FLSs, independent of the number of inputs and outputs. In the next paragraphs, each of the components of a FLS is presented, indicating the particularities of each fuzzy model.

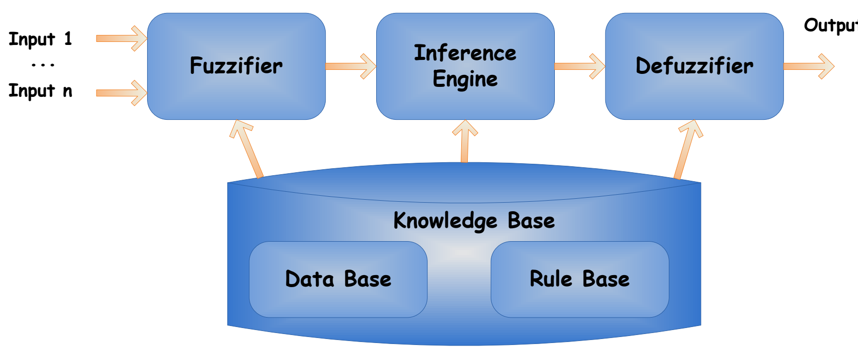


Figure 2.7. Classic FLS architecture

The ***Fuzzifier*** component takes the crisp values of input parameters and gives as output their corresponding fuzzy degree of membership based on the defined membership functions for each input.

The ***Inference Engine*** component does the mapping of the input fuzzified values on the output fuzzy set described by the output membership function. The mapping is done based on the rules contained in the ***Rule Base***. FL operators are applied on these rules. Different operators are applied depending on the type of fuzzy model that is followed by the FLS: Mamdani, Sugeno, Larsen or Tsukamoto. Therefore, in the FL terminology there are the following terms used in relation to inference: Mamdani, Sugeno, Tsukamoto, and Larsen inferences, respectively. The descriptions of the operators used in the inference is quite complex and it is not in the scope of this work; however a thorough presentation could be found in [61], [67], [76].

Knowledge Base contains the rule base, which is a collection of “IF-THEN” rules expressed linguistically, and the fuzzy sets of the inputs and output that are represented in the data base. An example of rule formalization is given in eq. (2.3).

$$R^{(l)}: \textbf{IF } u_1 \textit{ is } F_1^l \textbf{ AND } u_2 \textit{ is } F_2^l \textbf{ AND } \ldots$$
$$u_p \textit{ is } F_p^l \textbf{ THEN } v \textit{ is } G^l \tag{2.3}$$

where l is the index of the rule in the rule base, $u_{1 \ldots p}$ and v are linguistic variables, p is the number of input variables considered in the rule, $F_{1 \ldots p}^l$ are fuzzy sets. G^l has different interpretations depending on the types of inference: it is a fuzzy set in Mamdani and Larsen inferences, a crisp function depending on the numerical values of $u_{1 \ldots p}$, a singleton in the Sugeno inference, or a fuzzy set with a monotonic membership function in the Tsukomoto inference.

Example: 4 In [69], the authors built a FLS for congestion detection. In their rules they expressed linguistically the dependency of the congestion on the speed of the vehicle and density of the vehicles around. Speed as described in *Example 1* and *Example 2* is a linguistic variable having as terms: *very slow*, *slow*, *medium* and *fast*. Traffic density is a linguistic variable as well having as terms *low*, *medium*, *high* and *very high*. An example of a rule is as follows:

If speed is very slow and traffic density is medium then the level of congestion is moderate.

The ***Defuzzifier*** performs the defuzzification process that is the opposite of fuzzification: a fuzzy set (that resulted in the inference) is mapped to a crisp value. This value is the output of the FLS. There are various defuzzification methods, including: centre of area (COA) also called centre of gravity or centroid, bisecter of area, weighted average, maximum, mean of maximum. In Sugeno and Tsukamoto fuzzy models the defuzzification used is the weighted average method, while in Mamdani and Larsen the defuzzification method is not imposed by the fuzzy model applied, being decided when designing the FLS. The most popular defuzzification method is COA. Further discussions on defuzzification methods are included in [61], [67], [76].

2.2.3. Designing a Fuzzy Logic System Step-by-Step

This section aims to outline the steps that should be followed in the design of a FLS. These steps are general and can be used for developing FLSs in any context, but specific particularizations to the vehicular networks context are also presented. Also, for a better understanding, examples are provided and these are taken exclusively from the FL solutions existent in the vehicular networks.

Usually, when designing a FLS, a fuzzy model from the ones presented in the previous section is followed and this is also considered here. However, these fuzzy models can be combined to obtain a unique design for the FLS. However, this scenario considers there already is certain experience in designing FLSs.

1) The first step in designing a FLS is to clearly define the inputs and the outputs of the systems and their range. Translated into the FL terminology this means that the linguistic variables representing the inputs and outputs are defined and their universe of discourse is identified. Moreover, in this step, the terms of the linguistic variables and their numerical range are identified.

2) Second step is to decide the type of membership functions of the fuzzy sets describing the fuzzy terms of the inputs and outputs. As presented before, there are various types of membership functions. After selecting the type of membership functions, their parameters need to be set. First, the setting of the parameters has to be based on the “expert knowledge” of the designer or lessons learnt from the literature. Further, these parameters can be modified by performing either manual tuning, or automatic tuning.

3) Third step in the design of a FLS is to define the rules. This is a very important step for the performance of the system. Good rules can be defined based on the knowledge about how the system is supposed to work, its context and conditions. Thus, like in the case of the membership functions, initial rules should be based on the expert knowledge or lessons learnt from the literature. Further, manual tuning can be performed in order to adjust the rules. The manual tuning is a good choice for small FLSs that do not have many rules, but if the number of rules is high, this would be a very difficult task. Automatic tuning techniques are available as well, but these are not as used as the automatic tuning techniques for the membership functions [67].

Example 5: In *Example 4*, a rule was presented as an example from a FLS designed to detect road congestion. The rule base for this FLS was built based on some levels of congestion estimated by the Skycomp [77] following the analysis on the data collected through aerial surveys of different freeways. The levels of congestion are described linguistically and they depend on speed and density, the same parameters that are considered as inputs of the FLS. Thus, this solution represents an example of a rule base build based on lessons learnt from the literature.

4) Fourth step is to define the inference type and the defuzzification method. This step can be influenced by the second step. For instance, if the output terms are represented through singletons, then a Sugeno inference might be suitable, but not necessary. Also, if the fuzzy sets of the output are monotonic, we have an incentive to apply a Tsukomoto inference, but again this is not a must. However, Tsukomoto or Sugeno inferences are conditioned by the sets describing the outputs of the system, as presented in the previous section. Moreover, these two inference types also come with their own defuzzification method, so this is not a design decision anymore. However, in the case of Mamdani or Larsen inferences, selecting a defuzzification method is a design decision. COA is the most popular defuzzification method. It is more complex, thus more computational effort is needed when compared to other defuzzification methods. However, if the complexity is not an issue, COA is highly recommended. In VANETs we need high performance, thus complexity is an issue. So a trade-off between accuracy and reduced computation complexity should be made when designing a FLS in this space. For instance, the membership functions selected should be less computationally complex (e.g. singleton, triangular, trapezoidal functions) in order to compensate the complexity of the COA defuzzifier.

5) Last step in the design of a FLS is represented by assessing and tuning of the system. Tuning the system can refer to reviewing the range of the inputs/outputs and their terms, revising the fuzzy sets and maybe defining additional ones, tuning the membership functions – revising their parameters or shape –, tuning the rules – adding, removing, assigning/modifying their weights –, and experimenting with different types of inferences.

Most of the actions performed in tuning a FLS are done manually. However, for tuning the membership functions and the rules, automatic techniques were designed. These techniques are based on learning algorithms. In tuning FLSs two types of learning are used: supervised and reinforcement learning. In both cases, there are a variety of techniques used and most commonly these are borrowed from the fields of Neural Networks and Genetic Algorithms, but in engineering common techniques are also borrowed from Particle Swarm Optimization and H ∞ filtering [78].

In **automatic tuning based on supervised learning** there is a need for a so called training set. This is represented by numerical data for inputs and the corresponding outputs. In some fields where FL is applied, the FLS is basically created starting from such a data set. But in the context of vehicular networks obtaining this training data set is not straightforward. One way to obtain it is through simulation, considering for certain inputs what the best output that can be obtained is. For the final evaluation of the system a different simulation scenario must be considered than the one used for obtaining the training set. This is valid in the case of any manual tuning that is performed: it is highly required that after tuning the FLS in a certain scenario the final tests to be performed on different scenarios. The supervised-based learning is performed off-line, before the deployment of the FLS.

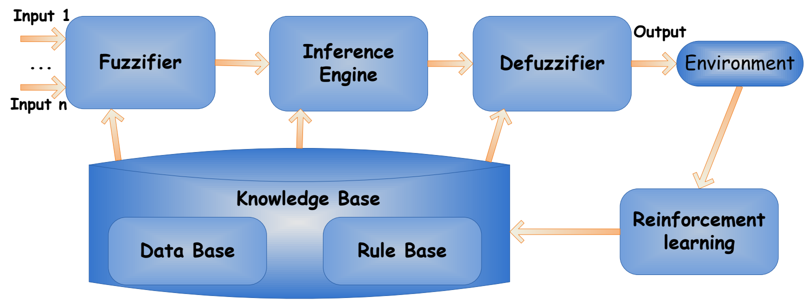


Figure 2.8. Reinforcement learning-based real-time adaptive FLS architecture

In **automatic tuning based on reinforcement learning** there is no need for a training set. The FLS is able to adapt its parameters dynamically, on-line or, in other words, at run-time based on the output of the system and the impact of the output on the environment (e.g. process controlled, impact of the decision taken on the modeled problem, etc). Thus, this is based on run-time recursion and consequently the architecture of a FLS that has reinforcement learning (Figure 2.8) is slightly different compared to the architecture of a classic FLS. There is another type of real-time adaptive architecture used in the design of FLSs and this is presented in Figure 2.9. The adaptive mechanism is very simple and is not based on any kind of learning algorithms, but on the impact of the output on the environment only. This is a typical architecture used by FL controllers, thus FLSs designed for control. These two architectures together with the classic architecture of a FLS that is displayed in Figure 2.7 represent the main architectures of FLSs

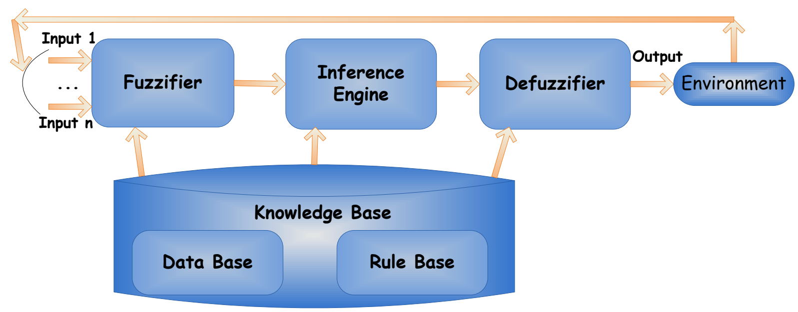


Figure 2.9. Simple real-time adaptive FLS architecture

Note that the design of a FL inference system is a loopback process, manual/automatic tuning being required to adjust especially the parameters of the membership functions and the rules. Moreover, experimenting with different shapes of membership functions or different inference engines can be very useful in designing the best FLS as there are no rigorous rules when designing a FLS, only recommendations. Thus it cannot be said that by using a Larsen inference better performance would result than by employing a Mamdani inference in certain conditions. As such, the design process is quite difficult, if not performed with the support of a specialized FL tool. Experimenting and tuning the system is very much limited unless a huge extra-implementation effort is done. Moreover, in-depth knowledge about the FL operators and their implementation and complex Mathematics is required to be implemented. Using a FL tool allows for defining each component in a descriptive language and then to treat them like black boxes. This is the reason why there is a dedicated section to the FL tools.

2.2.4. Fuzzy Logic Tools

FL is very powerful for solving a wide-range of problems, but there is a considerable complexity required for implementing FLSs from scratch and a huge added complexity if the system is planned to be thoroughly tuned. This is the reason why the author dedicated this section to the tools or libraries that offer support for designing FLSs. There is a considerable number of FL software tools developed by both industry and academia. We focused on the preferable open-source software provided in the academia that meets the following requirements:

- At least the most common membership functions, singleton, triangular, trapezoidal, gaussian, are built-in.
- At least Mamdani inference, known to be the most popular, is supported.
- In case of open-source software, this is maintained and a minimum user support is provided.
- The implementation is general and not specific (e.g. FuzzyPLC is dedicated to programmable logic controllers [79]).

The requirements were chosen to give flexibility and enough options in designing the FL systems. After considering the aforementioned requirements, the following FL tools were selected for this comparative study: Matlab FL toolbox [80], Octave FL toolkit [82], *R* FL toolbox [81], jFuzzyLogic [83], JFuzzyQT (i.e. a C++ clone of the jFuzzyLogic) [84] and XFuzzy3 [85]. Among these, Matlab FL toolbox is the only proprietary software, but was chosen due to its wide popularity in the academic environment.

The selected FL tools are further analyzed considering new criteria that enlarge their flexibility and easiness of use, such as: inference types supported in addition to Mamdani, linguistic hedges support, automatic tuning support, graphical user interface (GUI) provided for building FL systems and if the FL toolbox is able to generate embeddable code.

As emphasized in the previous sections, there are no rules for a perfect FLS design, only recommendations. Moreover, the performance of a FLS is dependent on the context. Thus, for instance it has been notice that Sugeno FLS are more efficient in some control problems than Mamdani ones. Therefore, more options provided for the inference allows for experimenting with multiple inference types and consequently allows for choosing the best inference for the designed

FLS. Except for the R FL Toolbox, all the others FL tools have at least Sugeno inference supported in addition to Mamdani. jFuzzyLogic and its C/C++ clone, jFuzzyQT, provide all the inference types.

Linguistic hedges support allows for a better and immediate granularization of the fuzzy sets. This is very helpful especially in manual tuning, as it is fastening the process of modifying the fuzzy sets used for inputs or outputs. Except for Matlab and *R* FL toolboxes all the other FL tools provide support for linguistic hedges.

Automatic tuning support is highly desirable, as implementing learning algorithms leads to a considerable increase in the complexity of the FLS implementation. Octave and *R* FL toolboxes do not provide any support for automatic tuning, while Matlab FL toolbox provides very limited support, namely it allows for the automatic tuning of single output Sugeno's FLSs only. The automatic tuning is based on supervised learning and targets only the membership functions parameters. Supervised learning-based automatic tuning is enabled by the XFuzzy3 tool as well, but no limitations regarding the type of FLSs that can be applied on are imposed. Moreover, XFuzzy3 provides the biggest variety of supervised learning algorithms from all the analyzed FL tools, and these address both membership functions tuning and rules automatic tuning. However, XFuzzy3 does not enable reinforcement learning, this feature being under development at the moment. Although jFuzzyLogic and jFuzzyQT do not have the same variety of supervised learning algorithms, they do provide minimum support for creating real-time adaptive FLSs.

Having a GUI option reduces even more from the complexity of developing a FLS. Designing a FLS in FL toolboxes supposes a description of each of the design block presented in the previous sections in a specific language, Matlab, Octave (similar as Matlab) or *R* in the case of their corresponding FL toolboxes, or a pre-defined descriptive language in the case of other FL toolboxes. A GUI eliminates the need of learning the specific description language for describing the FL components. Thus the GUI option makes easier the design and development process of a FLS. From the FL toolboxes analyzed Matlab and XFuzzy3 only provide the GUI option.

In addition, taking into consideration that FL is analyzed in the context of its applicability in VANETs, the inference system designed is most probable required to be embedded into a simulation platform. Computer simulation is the method most used to test and validate the solutions proposed for VANETs. This is mostly due to the fact that test-beds are most of the times prohibitively expensive. However, even if there is the possibility of creating the test-beds for validation, still simulation is usually used as a first step in testing and validation. A variety of simulation tools are used from Matlab or its clone Octave, to the more specific simulators for

VANETs such as STRAW[8], Veins[9], iTETRIS, etc. A comprehensive survey on the latter category of simulation tools is presented in [86]. These simulators are designed in C, C++ or Java languages. Thus the FL inference system could be required to be integrated in C, C++ or Java code. In case that FL system cannot be directly embeddable, additional effort is required to develop communication interfaces between the simulation platform and FL tool. A summary of the FL tool comparison under the considered criteria is presented in TABLE 2.1.

TABLE 2.1. FL TOOL COMPARISON

FL Tool	Inference Types	Linguistic Hedges Support	Automatic Tuning Support	Generating Embeddable Code	GUI for building the FL system
Matlab FL toolbox	Mamdani, Sugeno	no	automatic tuning based on supervised learning only for Sugeno FLSs with several constraints;	C code	yes
Octave FL toolkit	Mamdani, Sugeno	yes	no	no	no
R FL toolbox	Mamdani	no	no	no	no
jFuzzyLogic	Mamdani, Sugeno, Tsukamoto, Larsen	yes	Automatic tuning based on supervised learning + support for reinforcement learning also	Java code	no
jFuzzyQT	Mamdani, Sugeno, Tsukamoto, Larsen	yes	Automatic tuning based on supervised learning + support for reinforcement learning also	C/C++ code	no
XFuzzy3	Mamdani, Sugeno	yes	automatic tuning based on supervised learning	Java, C or C++ code	yes

2.3. Clustering and Vehicular Networking

2.3.1. Introduction to Clustering

Clustering is a division technique that creates groups of similar objects [87] mainly with the purpose of dealing with scalability. The similarity between objects is built upon one or more clustering metrics that are extremely varied and highly dependent on the context of clustering. Clustering is widely used in data analysis, data mining, statistics, text mining, information retrieval, etc. It has been widely adopted in MANETs, as it provides support for good system performance, good management and stability of the networks in the presence of mobility and large number of terminals [88]. Thus, clustering helps solve some of the main issues in MANETs: scalability and stability [87].

In MANETs, clustering involves dividing the nodes into virtual groups based on some rules that establish if a node is suitable to be within a cluster or not. These rules are defined based on some clustering metrics. In MANETs these metrics can be node type, battery energy level, mobility pattern, etc.

[8] STRAW simulator website, http://www.aqualab.cs.northwestern.edu/projects/144-straw-street-random-waypoint-vehicular-mobility-model-for-network-simulations-e-g-car-networks
[9] VEINS simulator website, http://veins.car2x.org/

In general, based on the node membership and task associated to the node, a clustering scheme considers that a node can be in one of the following states [88]:

- *unclustered*, also known as non-clustered or independent, when it does not pertain to any cluster
- *cluster member* or clustered when the node is within a cluster
- *cluster head* (CH) when the node has extra-responsibilities in a cluster. Usually, CH is the main controller of the cluster, the main coordinator of the communication within the cluster (i.e. intra-cluster communication) and has a main role in the functionality that is supposed to be provided by the cluster.
- *gateway node* is the node that ensures the communication between the clusters, also called inter-cluster communications.

Figure 2.10 graphically represents these states.

A general classification of MANET clustering schemes is based on the existence (or not) of a CH as follows: *CH-based clustering*, if there is a CH in the clusters created or *non-CH-based clustering*, if there is no CH in the cluster created. Note that in CH-based clustering, the performance of clustering is highly dependent on CH election, as this node has the main responsibilities in its cluster. Therefore in this type of clustering algorithms the focus is mainly on CH selection algorithms. Another general classification of clustering is based on the number of hops between node pairs in the cluster: *1-hop clustering* or *multi-hop clustering*.

Successfully applied in MANETs to address stability and scalability, clustering was adopted in VANETs, where these issues are even more augmented. At the beginning, MANET clustering algorithms were adopted and directly applied to VANETs without any modifications, but as this research direction evolved, new clustering algorithms dedicated to VANETs were designed to address their specific characteristics [121].

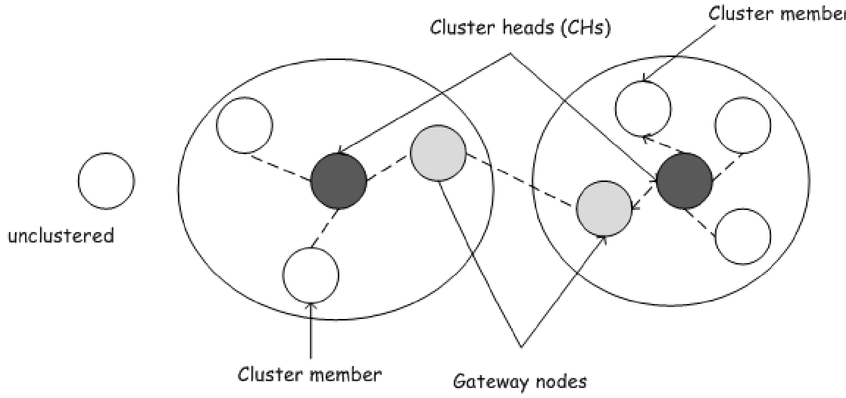


Figure 2.10. Illustration of node states in MANET clustering

2.3.2. Clustering in VANETs

The clustering concepts presented in the context of MANETs are valid in VANETs context as well, especially given the fact that clustering in VANETs has evolved from MANETs. There are only some additional aspects that need to be mentioned and that derive from the adaptation of clustering to VANET-specific conditions.

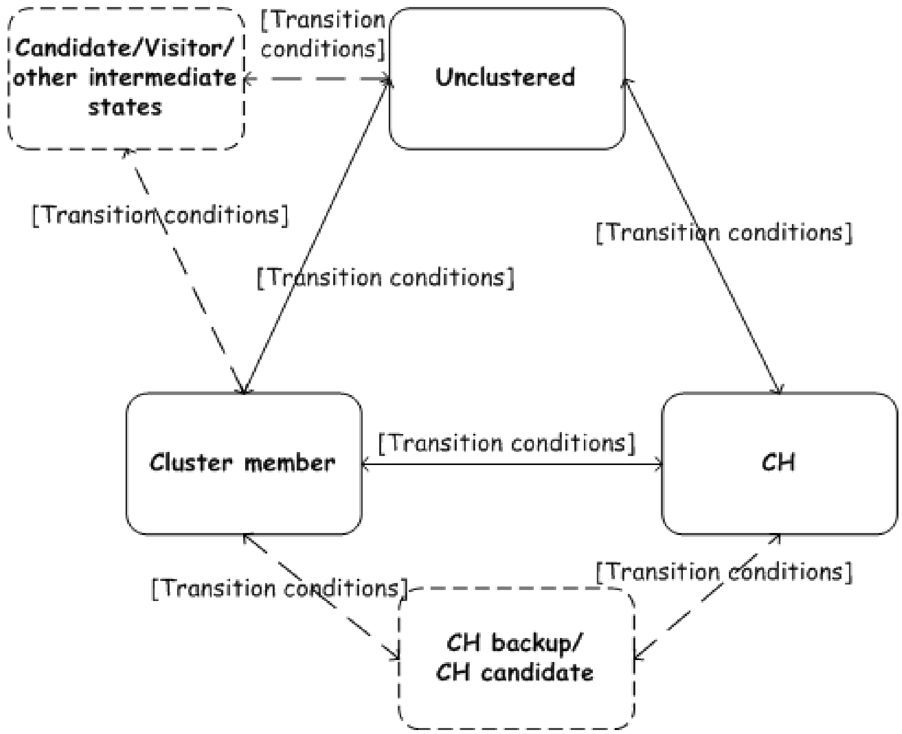


Figure 2.11. State machine representing CH-based clustering in VANETs

Clustering metrics were adapted not only to address VANET challenges imposed by their specific characteristics such as high mobility, rapidly changing topology and diversity of conditions, but also to take advantage of some of these characteristics, such as predictability of their movement. Therefore, clustering in VANETs is based upon more metrics than in MANETs that need to describe the complexity of VANET environments. Among the most common metrics in VANET clustering are direction, vehicle’s relative speed in comparison to other neighbouring vehicles, vehicle’s relative position, but also traffic flow, the lane in urban scenarios (e.g. right lane, left lane, and ahead lane), predicted future speed and position, density of vehicles (sparse or dense), etc.

Additional states of the nodes have been added in VANETs in the effort to cope with their more dynamic environments. We called these intermediate states as they are in between the well-known states. These intermediate states include the *candidate node* and *CH backup* or *CH candidate states*. The candidate state was introduced by some approaches in order to obtain a better stability of the cluster. A node is not immediately given the cluster member state; it goes into the candidate state until it proves that it has certain stability in the cluster. The CH backup/CH candidate (quasi-CH in other approaches) state was introduced to make faster and smoother the process of changing the CH. Clustering in VANETs can be represented as a state machine, where the machine is the vehicle-node that can be in one of the following states: *unclustered*, *cluster member*, *CH* (in the case of CH-based clustering) and, optionally, in an intermediate state *candidate* and *CH backup/candidate* as previously defined. Figure 2.11 presents this state machine. The transitions between states are controlled by different conditions that relate to different algorithms. For instance, in the context of CH-based clustering, CH selection algorithm is one of the main algorithms. CH selection is based on different metrics that are combined based on different rules that can be simple or more complex (e.g. based on complex Mathematical models) depending on the clustering solution.

Once adopted in VANETs, clustering gained popularity mostly due to their efficiency in addressing network stability issues. Clustering algorithms were implemented in the design of a large variety of VANET solutions: MAC protocols [89] – [92], routing protocols [93] – [99], data aggregation [100], security [101], [102], inter-vehicle communication [103] – [105] and data and infotainment dissemination solutions and architectures [106], [107]. In addition, various generic clustering algorithms were defined for VANETs [108] – [115].

Independent of the type of VANET solution the clustering algorithm is designed for, one of the main purpose of clustering is to achieve network stability. Therefore, clustering metrics are focusing mainly on this aspect and they relate to VANET's dynamic environment. Thus independently of the context in which clustering is applied (e.g. MAC protocols, routing protocols, etc), clustering metrics focus on the same issues and they are similar to each other. They are only dependent on the ingeniously modeling of the VANET environment and they are different from solution to solution as researchers are experimenting in trying to find the best clustering metrics to express the dynamicity of the VANETs. Similarly, in clustering performance assessment, usually first the network stability achieved is measured and then, the overall assessment of the clustering solution is performed (the overall solution where clustering is integrated; e.g. MAC protocol, data aggregation, etc). All these considerations allow for a uniform analysis of clustering algorithms in VANETs, independent of the type of solution/application in which they are integrated.

Although there is a considerable number of clustering solutions in VANETs, this research direction is still not mature. A closer analysis of the existent solutions in the literature reveals some major issues that relate to the performance assessment of clustering solutions in VANETs. So far, no analysis on this topic was provided in the literature and this is reflected by the fact that the existing clustering solutions use intuitively-defined performance assessment metrics or re-defined metrics similar with already existing ones mostly because researchers were not aware that such metrics have already been proposed in the literature. This resulted in metrics having various name versions. In particular this is the case for the metrics used to measure the stability of the clusters, which contributes to network stability. These metrics are a direct measure of the performance of clustering algorithms in the context of VANETs, where the performance of clustering algorithms is reflected in how well clustering algorithms perform in achieving good network stability. It can be therefore concluded that the aforementioned major issues directly relate to the metrics used to evaluate the performance of clustering algorithms in VANETs.

In the absence of a study on performance assessment metrics of VANETs clustering solutions and in the absence of the standardized metrics, we performed a survey of the performance evaluation of clustering solutions in VANETs and of clustering algorithms designed for these solutions. This survey resulted in the identification and comprehensive definition, including in mathematical terms, of generic metrics that can be used in the evaluation of clustering algorithms in VANETs. Next sections describe in details the results of this study that can be considered a guide for the performance assessment of VANETs cluster-based solutions in general and VANET clustering algorithms in particular. This guide was used when assessing the performance of the clustering solution proposed in this thesis.

2.3.3. Performance Assessment of Clustering in VANETs

As a result of the analysis performed on the VANET clustering schemes proposed in the literature, three major classes of performance assessment metrics for clustering solutions were identified: network-specific metrics, application-specific metrics and topology-based metrics. These are presented in Figure 2.12 and are next described.

Network-specific metrics are well-known metrics applied in network communications, evaluating the performance of the clustered network mainly in terms of data transfer: throughput, loss, delay, data delivery ratio, overhead, etc. For instance, the overhead imposed by the clustering messages is a commonly used metric in the performance assessment of clustering schemes.

Application-specific metrics depend on the type of the cluster-based solution employed. As emphasized above, clustering algorithms were implemented in the design of a large variety of VANET solutions: data aggregation solutions, MAC and routing protocols, security, etc. Therefore, this class includes a large variety of metrics as well. For instance, a data aggregation cluster-based solution is evaluated by measuring the size of data that needs to be disseminated, as the goal of a data aggregation scheme is to reduce the size of the data that needs to be disseminated. Note that these metrics and network-specific metrics are evaluating the performance of the overall solution based on clustering.

Topology-based metrics (hashed in Figure 2.12) evaluate the stability and robustness of the resulted clusters. Cluster stability translates into network stability, thus topology-based metrics are measuring the network stability. Network stability is emphasized as an important issue in VANETs due to their rapidly changing topology. Therefore, topology-related metrics are of great importance, fact acknowledged by researchers: the majority of proposed clustering solutions are using topology-based metrics in the performance assessment.

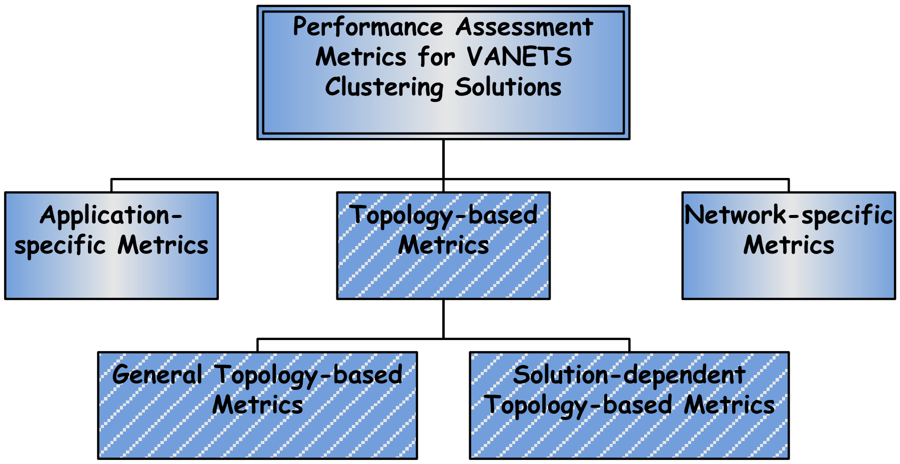


Figure 2.12. Classification of the performance assessment metrics used in VANETs clustering solutions

Independent on the type of VANET solution clustering algorithm is designed for, the general aim of a clustering algorithm is to achieve network stability. In this context, the performance of clustering algorithms is not seen from a computational point of view (e.g. complexity of the algorithms). The focus is on how well clustering algorithms perform in achieving good network stability. Based on these considerations it can be said that in the context of VANETs, topology-based metrics are a measure of clustering algorithms performance. The overhead imposed by the clustering messages, metric included in the first category, is equally a measure of the performance of these algorithms. The aim is for clustering algorithms to achieve network stability while not imposing a big overhead.

2.3.4. Analysis of Topology-based Metrics for Clustering Algorithms Assessment in VANETs

In the previous section, topology-based metrics were identified as the metrics suitable for the evaluation of clustering algorithms in VANETs. Therefore, this section contains an in-depth analysis of the topology-based metrics.

The topology-based metrics are classified in two classes: general metrics and solution-dependent topology-based metrics. The first class is represented by general metrics that have their definition and interpretation independent of any kind of particular characteristics of the clustering algorithms they were defined to assess. General topology-based metrics can be used to evaluate the performance of any type of clustering algorithms. A single restriction was defined in case of these topology-based metrics, namely some of them can be applied to cluster head-based algorithms only. However, these metrics maintain their generality in that context.

The solution-dependent topology-based metrics are dependent on the clustering solutions they were defined to assess. The most common dependency is related to the interpretation of the metric: their interpretation depends on some particular characteristics of clustering solutions.

Next the topology-based metrics identified during the analysis of VANETs clustering algorithms are presented. Several aspects regarding the form of presentation need to be mentioned before getting to the presentation of the topology-based metrics. These aspects relate to several issues revealed during the study performed on the clustering solutions in VANETs.

TABLE 2.2. NOTATIONS USED IN TOPOLOGY-BASED METRICS DEFINITION

Notation	**Explanation**
$\|V(t)\|$	Total number of vehicles at time t
$\|C(t)\|$	Numbers of clusters at time t
$\|CH_i(t)\|$	Number of cluster members in cluster i at time t
$\overrightarrow{SP_i}(t)$	Velocity vector of vehicle i at time t
$\|\overrightarrow{SP_i}(t)\|$	Speed of vehicle i at time t
S	Simulation time
$CHL_i(t)$	Time period between time t, when vehicle i becomes a cluster head, and the moment of time vehicle i changes to another state.
$CL_i(t)$	Time period between time t, when cluster i is formed and the moment of time cluster i is dismissed.
$CMT_j^i(t)$	Time period between time t, when vehicle j becomes a cluster-member of cluster i, and the moment of time vehicle j leaves cluster i.

An important issue is the inconsistency in naming the metrics. As already mentioned some of the metrics have been re-defined and consequently bare different names. In some cases of re-defining, general metrics are constrained by particular conditions/characteristics of the algorithm they are used to assess: general metrics are defined either using particular conditions/assumptions of the algorithm or using particular parameters of the current algorithm. This is not necessary wrong as it is perfectly applicable for that particular algorithm, but can lead to misconceptions and misusing in case of other clustering algorithms. In addition, there are a considerable number of metrics that are not provided with a mathematical definition at all, being described in words only. So far, a single work in the literature [89] has provided mathematical definitions for some general metrics used in the evaluation of clustering.

Due to the aforementioned aspects, when presenting the metrics in the next sections, all naming versions are provided. The most representative name was chosen based on either the popularity or the degree of match with the metric description. In addition, general mathematical definitions for each of the general metrics were provided. The mathematical formulas were proposed based on the textual definitions of the metrics and the in-depth analysis of the results. The general mathematical definitions provided to some of the general metrics by Su et al. [89] were taken in the same form as presented in their work. Together with these metrics were taken some of the notations used in their definitions, notations that are listed in TABLE 2.2 which also lists the additional notations used in this work to define the topology-based metrics.

2.3.4.1. Topology-based Metrics or General Metrics for the Evaluation of Clustering Algorithms in VANETs

Average cluster head lifetime ($\overline{\mathrm{CHL}}$) (average cluster head duration, cluster head time) is one of the most popular topology-based metrics. It is applicable for the cluster head-based clustering algorithms only. It was used for the evaluation of VANET clustering algorithms in [89], [107] – [109], [116], [125], [127] – [130]. The popularity of this metric is explainable: the importance of cluster heads lifetime is crucial as, usually, the cluster head is the main controller and content forwarder in the cluster head-based clustered networks [109]. Smaller cluster head lifetime affects the overall performance of the clustering algorithm. Take for example a data transfer between the cluster head and a member of the cluster. Larger cluster head lifetime implies more stable cluster topologies, leading to a decrease in number of re-clusterings, and consequently avoiding the waste of system resources and excessive computation time [89], [106].

The mathematical definition of $\overline{\mathrm{CHL}}$ metric, as taken from [89], is given in eq. (2.4).

$$\overline{\mathrm{CHL}} = \sum_{t=0}^{S} \frac{\sum_{i=1}^{|V(t)|} \mathrm{CHL}_i(t)}{\sum_{i=1}^{|V(t)|} \mathrm{BC}_i(t)} \tag{2.4}$$

In eq. (2.4), $\mathrm{BC}_i(t)$ represents a function that defines the transition of a node (vehicle i) to the cluster head state as described by eq. (2.5).

$$\mathrm{BC}_i(t) = \begin{cases} 1, & \text{if vehicle i changes to cluster head at time t} \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

$\overline{\mathrm{CHL}}$ is one of the metrics that were re-defined in the literature. In [108], this metric is given a mathematical definition dependent on the particularity of the clustering algorithm (eq. (2.6)).

$$\overline{\mathrm{CHL}} = \frac{1}{\mathrm{L}} \sum_{\mathrm{i}=1}^{\mathrm{L}} \mathrm{C}_{\mathrm{i}}^{\mathrm{life}} \tag{2.6}$$

where L is the number of clusters created throughout the session and $\mathrm{C}_{\mathrm{i}}^{\mathrm{life}}$ has the same meaning as $CHL_i(t)$. This is not a general definition, as the clusters are dismissed when the cluster head is changed. There are clustering algorithms [92] where clusters have back-up cluster heads. Often, when the vehicle cluster head changes its state, its role is taken by its back-up and the cluster is not dismissed. In this situation, eq. (2.6) cannot be used in all cluster head-based clustering algorithm evaluation as eq. (2.4) can be.

Average number of clusters ($\overline{\boldsymbol{NoC}}$) is a very popular metric used to assess the performance of a large number of clustering algorithms, [108] – [111], [100] – [101], [117] – [118], [126], [127], [129], [130]. It can be stated that same metric is used in [111], although instead of number of clusters it measured the number of cluster heads. This is due to the fact that in this particular solution, a cluster is dismissed when its cluster head is dismissed. In consequence, the number of cluster heads is equal to the number of clusters created. $\overline{NoC}$ is a general metric that can be applied to assess the performance of all the categories of clustering algorithms. $\overline{NoC}$ is a measure of network stability. Usually, when there are fewer clusters, better network stability is obtained [109]. Eq. (2.7) is proposed as a general mathematical definition of $\overline{NoC}$.

$$\overline{NoC} = \frac{1}{S} \sum_{t=0}^{S} |C(t)| \tag{2.7}$$

Average cluster size ($\overline{\boldsymbol{CS}}$) metric (average number of cluster members) measures the average size of the clusters throughout the session. This size of a cluster is considered to be determined by the number of vehicles in the cluster. $\overline{\mathrm{CS}}$ is a metric applicable to all clustering algorithms and was used in [89], [92], [93], [112], [113], [127]. Its general definition, as taken from [89], is given in eq. (2.8).

$$\overline{CS} = \frac{1}{S} \sum_{t=0}^{S} \sum_{i \epsilon C(t)} |CH_i(t)| \tag{2.8}$$

Average cluster head change rate *($\overline{\boldsymbol{CHCR}}$)* [109], [112] – [114], [117], [125], [128], [129] is seen as a measure of cluster stability. In more stable clusters, nodes in general and cluster heads in particular change their cluster membership or their state less often [113], [117]. It is a metric that can be used for the performance assessment of cluster head-based algorithms. Eq. (2.9) is proposed as the general mathematical definition of $\overline{CHCR}$.

$$\overline{CHCR} = \frac{1}{S} \sum_{t=0}^{S} \sum_{i=1}^{|v(t)|} CHD_i(t) \tag{2.9}$$

where $CHD_i(t)$ is the cluster head dismissed function of vehicle i, that was defined in eq. (2.10) to express the transitions from cluster head to another type of node.

$$CHD_i\,(t) = \begin{cases} 1, & if\ vehicle\ i \\ & changes\ from\ cluster\ head \\ & to\ another\ type\ of\ node\ at\ time\ t \\ 0, & otherwise \end{cases} \tag{2.10}$$

Average cluster member lifetime ($\overline{CML}$) (*average cluster member duration, cluster residence time*) was used as a performance metric in [94], [106], [109], [125], [128], [129] and is a general topology-based metric measuring the overall stability of clustering [109]. $\overline{CML}$ is a similar metric to the average cluster head lifetime just that the lifetime is computed for all the nodes in the cluster members not only for the cluster head. Same as for $\overline{CHL}$, longer average cluster member lifetime indicates a more stable clustering topology. A general mathematical definition of $\overline{CML}$ is provided in eq. (2.11).

$$\overline{CML} = \frac{1}{|C(t)|}\sum_{t=0}^{S}\sum_{i\in C(t)}\frac{\sum_{j\in CH_i(t)} CMT_j^i(t)CMD_j^i(t)}{\sum_{k=0}^{S}\sum_{j\in CH_i(t)} CMD_j^i(k)} \tag{2.11}$$

where CMD_j^i(t) is the cluster member dismissed function of vehicle *j* from cluster *i* being defined in eq. (2.12).

$$CMD_j^i\text{(t)} = \begin{cases} 1, & if\ vehicle\ j\ , cluster\ member\ of \\ & cluster\ i\ is\ dissmissed \\ & from\ the\ cluster\ i\ at\ time\ t \\ 0, & otherwise \end{cases} \tag{2.12}$$

Cluster changes per node ($\overline{CC}$) *(average number of cluster switches per node [110], [118])* , [125] measures the number of transitions of a vehicle between clusters and is used in order to measure cluster stability. The less number of transitions indicates better cluster stability. In [110], the metric is called improperly cluster stability and was measured through average number of cluster switches per node.

Based on the descriptions provided in [110] and [118] and on the evaluation of the results, eq. (2.13) is proposed as the general mathematical formula for $\overline{CC}$, applicable to all clustering algorithms.

$$\overline{CC} = \frac{1}{S}\sum_{t=0}^{S}\sum_{i\epsilon C(t), j\in CH_i(t)} CMD_j^i(t) \tag{2.13}$$

where CMD_j^i(t) was defined in eq. (2.12).

Cluster stability is stated as a metric in [108] and [110]. In [110], it is described and measured in terms of the average number of cluster switches per node during the simulation, but it is actually $\overline{CC}$. In [108], the authors sustain that the cluster stability metric depends on the change rate of the cluster and provide a formula that is not generic, but solution-dependent and uses some undefined terms. However, we consider that no metric can be called *cluster stability* until is not able to comprise all the metrics that were defined so far for its measurement. In addition, cluster stability is a property of the vehicular networks which is assessed by most of the topology-related metrics defined and not a metric.

Average cluster lifetime ($\overline{CL}$) [108] *(average cluster lifecycle [115])* it is another general metric that can be used for performance assessment of any type of clustering algorithm. It is a measure of cluster stability: larger average cluster lifetime translates into more stable clusters, thus a more stable network. In [108], the authors consider the average cluster lifetime equal to the average cluster head lifetime, as the clusters are dismissed whenever their cluster head changes. The authors define $\overline{CL}$ through eq. (2.6) which was considered not to be an appropriate general formula for cluster head lifetime. It is obvious that eq. (2.6) cannot be also considered a general formula for the $\overline{CL}$ as the cluster duration is not always dependent on the cluster head lifetime. This is valid in non-cluster head schemes and even in cluster head-based schemes. Eq. (2.14) is proposed as a general mathematical definition for $\overline{CL}$.

$$\overline{CL}=\frac{\sum_{t=0}^{S}\sum_{i\epsilon C(t)} CL_i(t)}{\sum_{t=0}^{S}|C(t)|} \tag{2.14}$$

Cluster reconfiguration rate ($\overline{CRR}$) [119] *(number of re-clusterings* [106]*)* is a metric defined to measure cluster stability based on the fact that a good clustering algorithm should be stable and it should not change the cluster configuration too drastically when a few nodes are moving and the topology changes rapidly. This metric does not have a mathematical definition in none of the solutions that use it. Moreover, these solutions [106], [119] are both cluster head-based algorithms and they state that the re-clustering/reconfiguration happens when cluster head changes. Described like this, the metric becomes identical to $\overline{CHCR}$. However, eq. (2.15) proposed a general mathematical description for the $\overline{CRR}$ applicable to all clustering algorithms.

$$\overline{CRR}=\frac{1}{S}\sum_{t=0}^{S}\sum_{i=1}^{|C(t)|} CD_i(t) \tag{2.15}$$

where $CD_i(t)$ is the cluster dismissed (*CD*) function for cluster *i* as defined in eq. (2.16).

$$CD_i(t)=\begin{cases}1, if\ cluster\ i\ was\ dissmissed \\ \quad at\ moment\ t \\ 0, \quad otherwise\end{cases} \tag{2.16}$$

Average relative speed compared to the cluster head within a cluster ($\overline{RSWC}$) [89] *(average cluster stability factor [92])* measures the topology stability of clusters. It is a general topology-based metric for all the cluster head-based algorithms. In all cluster head-based algorithms, a smaller average speed of the cluster member compared to that of the cluster head is translated into an increased stability of the cluster. However, $\overline{RSWC}$ is not a very common metric. It was defined and used as in eq. (2.17) [89]. In [92], this metric was re-defined as the *average cluster stability factor* and is described depending on some particular parameters of the solution. A deeper analysis reveals that average cluster stability factor is identical to $\overline{RSWC}$.

$$\overline{RSWC} = \frac{1}{|C(t)|S} \sum_{t=0}^{S} \sum_{i \epsilon C(t)} \frac{\sum_{j \epsilon CH_i(t)} |\overrightarrow{SP_i}(t) - \overrightarrow{SP_j}(t)|}{|CH_i(t)|} \tag{2.17}$$

Average relative speed among cluster heads ($\overline{RSCH}$) [89] is a general topology-based metric for cluster head-based algorithms. It measures the global topology of the network. Eq. (2.18) represents the general mathematical definition of this metric as taken from [89].

$$\overline{RSCH} = \frac{1}{S} \sum_{t=0}^{S} \frac{\sum_{i,j \epsilon C(t) \wedge i \neq j} |\overrightarrow{SP_i}(t) - \overrightarrow{SP_j}(t)|}{|C(t)|^2} \tag{2.18}$$

$\overline{RSWC}$ and $\overline{RSCH}$ are more complex metrics for cluster head-based algorithms that are indicators of both cluster head and network stability, as this type of metrics are measuring better the cluster stability in general. This statement also made in [112], where clustering algorithms defined are first assessed using $\overline{CS}$ and $\overline{CHCR}$. Then, the authors define a relative measure ($\overline{CS}/\overline{CHCR}$) arguing that this is a better measurement of cluster stability.

All the general metrics presented and defined (except the cluster stability which is not considered a metric) are summarized in TABLE 2.3. This section and the table-based summary provided represent a good guide for the performance assessment of VANETs clustering algorithms in particular and of VANETs cluster-based solutions in general. Evaluating the performance of clustering algorithms via these general metrics can greatly facilitate the comparison between the clustering algorithms, independent from their type: generic algorithms or integrated in a specific solution (e.g. clustering algorithm implemented in a MAC protocol).

2.3.4.2. Solution-dependent Topology-based Metrics

Node re-clustering time [94], [120] / **Re-affiliation frequency** [116] are metrics very differently named, but both defined as the time between cluster associations for a given node. Solutions using this metric consider that this is a measure of the stability of a cluster membership and shorter node re-clustering time/re-affiliation frequency means an increased stability of a cluster membership. However, the average cluster membership lifetime and average cluster head lifetime (in case of cluster head-based algorithms) are better indicators of the stability of cluster membership and this statement is based on the following considerations. There are clustering algorithms (e.g. [113]) where not all the nodes are clustered. They have different roles like candidate, visitor and they are not clustered until they demonstrate their future stability in the cluster, meaning a long lifetime as a cluster member. This translates into longer period of re-clustering/re-affiliation frequency, but this does not mean that the topology is less stable. These considerations represent also the motivation of including this metric in the class of solution-dependent topology-based clustering metrics. In this class, the **average percentage of vehicles (nodes) clustered nodes** [113] metric is also included. A bigger average percentage of clustered nodes it is usually translated into a better stability of the network topology. However, in the aforementioned particular clustering algorithms, this metric is not applicable because at some moments of time there can be a considerable number of nodes not-clustered (candidate or visitors).

Although the average percentage of vehicles clustered is not a general metric, when comparing similar clustering algorithms this metric is very useful to be analyzed in conjunction with the other general clustering metrics such as $\overline{NoC}$ and $\overline{CS}$ metrics. Sometimes the average number of clusters can be smaller due to the fact that a clustering scheme is not that successfully in clustering a higher number of vehicles. For instance, if a clustering algorithm has higher $\overline{NoC}$, but also higher average percentage of vehicles clustered and similar $\overline{CHL}$ or $\overline{CML}$ it is more successful than the clustering scheme being compared against. As a conclusion, in the performance assessment of clustering solutions, in order to measure the stability, more metrics should be used and the results should be corroborated and not analyzed independently for each metric.

2.4. Chapter Summary

This chapter presented the main technical concepts that are used throughout this thesis. First, vehicular networks also known as VANET are introduced as they represent the general context in which the contributions of this thesis have been designed. They represent a novel type of communications that enables ITS and cooperative ITS in what we call smart cities.

Then FL concepts were introduced as FL was used in all the proposed solutions. The presentation of these concepts has both general and also particular character, as the concepts were put in the context of VANET through the given examples.

In the end, clustering concepts were introduced in the context of VANET as this relates to the last contribution that is presented in this thesis. Emphasize is made on the clustering evaluation and an in-depth study was made in the absence of some guidelines in the literature referring to this aspect in order to ensure a proper performance evaluation for the proposed algorithm.

TABLE 2.3. GENERAL METRICS FOR THE EVALUATION OF CLUSTERING ALGORITHMS IN VEHICULAR NETWORKS - SUMMARY

Metric	Mathematical Definition	Popularity	Restriction
Average Cluster Head Lifetime($\overline{CHL}$)	$\overline{CHL} = \sum_{t=0}^{S} \frac{\sum_{i=1}^{\lvert V(t)\rvert} CHL_i(t)}{\sum_{i=1}^{\lvert V(t)\rvert} BC_i(t)}$, where $BC_i(t) = \begin{cases} 1, & if\ vehicle\ i\ changes\ to\ cluster\ head\ at\ time\ t \\ 0, & otherwise \end{cases}$	[89], [107] – [109], [116], [125], [127] – [130]	Cluster head-based algorithms only
Average number of clusters ($\overline{NoC}$)	$\overline{NoC} = \frac{1}{S} \sum_{t=0}^{S} \lvert C(t)\rvert$	[108] – [111], [100] – [101], [117] – [118], [126], [127], [129], [130]	-
Average cluster size ($\overline{CS}$)	$\overline{CS} = \frac{1}{S} \sum_{t=0}^{S} \sum_{i \epsilon C(t)} \lvert CH_i(t)\rvert$	[109], [112] – [114], [117], [125], [128], [129]	-
Average cluster head change rate ($\overline{CHCR}$)	$\overline{CHCR} = \frac{1}{S} \sum_{t=0}^{S} \sum_{i=1}^{\lvert v(t)\rvert} CHD_i(t)$, where $CHD_i(t) = \begin{cases} 1, & if\ vehicle\ i\ changes\ from\ cluster\ head\ to\ another\ type\ of\ node\ at\ moment\ t \\ 0, & otherwise \end{cases}$	[94], [106], [109], [125], [128], [129]	Cluster head-based algorithms only
Average cluster member lifetime ($\overline{CML}$)	$\overline{CML} = \frac{1}{\lvert C(t)\rvert} \sum_{t=0}^{S} \sum_{i \in C(t)} \frac{\sum_{j \in CH_i(t)} CMT_j^i(t) CMD_j^i(t)}{\sum_{k=0}^{S} \sum_{j \in CH_i(t)} CMD_j^i(k)}$, where $CMD_i^j(t) = \begin{cases} 1, & if\ vehicle\ j\ ,cluster\ member\ of\ cluster\ i\ is\ dissmissed\ from\ the\ cluster\ i\ at\ moment\ t \\ 0, & otherwise \end{cases}$	[110], [118], [125]	-
Cluster changes per node ($\overline{CC}$)	$\overline{CC} = \frac{1}{S} \sum_{t=0}^{S} \sum_{i \epsilon C(t), j \in CH_i(t)} CMD_j^i(t)$, where *CMD* defined as above	[108], [115]	-
Average cluster lifetime ($\overline{CL}$)	$\overline{CL} = \frac{\sum_{t=0}^{S} \sum_{i \epsilon C(t)} CL_i(t)}{\sum_{t=0}^{S} \lvert C(t)\rvert}$	[106], [119]	-
Cluster reconfiguration rate ($\overline{CRR}$)	$\overline{CRR} = \frac{1}{S} \sum_{t=0}^{S} \sum_{i=1}^{\lvert C(t)\rvert} CD_i(t)$, where $CD_i(t) = \begin{cases} 1, & if\ cluster\ i\ was\ dissmissed\ at\ moment\ t \\ 0, & otherwise \end{cases}$	[89], [92]	-
Average relative speed compared to the cluster head within a cluster ($\overline{RSWC}$)	$\overline{RSWC} = \frac{1}{\lvert C(t)\rvert S} \sum_{t=0}^{S} \sum_{i \epsilon C(t)} \frac{\sum_{j \epsilon CH_i(t)} \lvert \overrightarrow{SP_i}(t) - \overrightarrow{SP_j}(t)\rvert}{\lvert CH_i(t)\rvert}$	[89]	Cluster head-based algorithms only
Average relative speed among cluster heads ($\overline{RSCH}$)	$\overline{RSCH} = \frac{1}{S} \sum_{t=0}^{S} \frac{\sum_{i,j \epsilon C(t) \wedge i \neq j} \lvert \overrightarrow{SP_i}(t) - \overrightarrow{SP_j}(t)\rvert}{\lvert C(t)\rvert^2}$	[109], [112] – [114], [117], [125], [128], [129]	Cluster head-based algorithms only

Chapter 3

RELATED WORKS

This chapter presents and discusses prior art in the areas of research that are related to the work presented in this thesis. There are three main such areas: green transportation solutions, FL-based solutions, and clustering algorithms, all in the context of VANETs.

3.1. VANET-based Green Transportation Solutions

First, VANETs were considered very promising in supporting especially safety in traffic (e.g. preventing accidents) and traffic congestion issues. Consequently the focus was on designing VANET-based applications to solve these issues. Recently, due to the generally acknowledged pollution problem and due to the advances made in VANET research that demonstrated their huge potential, the focus shifted towards designing "green" solutions based on VANET. Therefore, vehicular communications are at the core of some successful designs of both intelligent and green transportation solutions [1], [12], [13]. Two main classes of green transportation solutions based on vehicular communications can be identified: eco-routing and eco-driving solutions.

3.1.1. VANET-based Eco-routing Solutions

VANET-based eco-routing approaches subscribe to the major class of vehicle routing solutions. Vehicle routing aims to find the most convenient path from start to destination based on certain criteria. In vehicle eco-routing the criteria include lower gas emissions, fuel or energy consumption. Vehicle routing is well represented in the literature and a large plethora of solutions have been proposed. V2X communications capabilities allowed for advanced dynamic and real-time routing solutions based on more accurate information regarding real-time traffic conditions and events or road characteristics and conditions. Such solutions are proposed in [50], [51], [228], [226], [202] and are dedicated to internal combustion engine-powered vehicles. The authors show how they reduce fuel consumption and gas emissions.

In [51] and [228], Collins et al. proposed both an eco-routing solution, and also a traffic management system enabling the routing solution. The best route decision is taken based on three factors: travel time of the road, the estimated fuel consumption and road congestion. In the approach proposed by Doolan et al. [50] time is not considered in taking the best route decision, but as a novelty, the road characteristics, namely the road roughness and gradient, are considered together with road congestion. In both aforementioned approaches, vehicular communications are employed in data collection and for both approaches the principle is similar: utility functions based on the factors considered are employed to compute a rating associated to each road segment and then the routing follows the Dijkstra algorithm [225].

In [226], Jabbarpour et al. proposed a very complex approach that considers multiple factors in their routing decision: travel time, speed, distance, vehicle density (i.e. congestion) and road topology and characteristics. The purpose is to reduce fuel emissions by determining the least congested shortest paths to the destination. Similar to the solutions previously discussed, VANET is employed in data collection. As a novelty, this solution involves computational intelligence in the best route decision: an ant-based algorithm is employed for route selection. Extensive simulations considering various scenarios have shown considerable reduction in fuel consumption and gas emissions.

The solution proposed by Souza et al. in [202] has a different approach and it can be said that is an event-driven eco-routing solution. The vehicles are following their regular routes until an event warning message (e.g. accident, congested road ahead) is received via vehicular communications. Based on this information the vehicles are re-routed if needed in order to avoid congestion that may determine increased fuel consumption and gas emissions.

Vehicular communications-based eco-routing solutions dedicated to EVs are in early stages. However, solutions have been proposed for EVs too, such as the one presented by Demestichas et al. in [203]. In this solution, machine learning techniques are employed in the computation of the most energy efficient route that integrates static map information and database information containing previous driving experience: road conditions and characteristics, traffic conditions, and charging stations. The data collection process is done via V2X communications. It is expected that the solutions proposed for internal combustion engine-powered vehicles to be also studied in the context of EVs as basically same external factors that influence fuel consumption and gas emissions, also affect the energy consumption in case of EVs.

3.1.2. VANET-based Eco-driving Solutions

Solutions in this class advise on how to drive in order to reduce fuel consumption, gas emissions or energy consumption. These solutions employ communications between traffic light and vehicles combined or not with V2V communications. Some of these solutions [204], [205] adapt the traffic light phases to the flow of vehicles approaching the intersection. When employing these approaches, the waiting times and the number of vehicles stopped at the intersection reduce and consequently the fuel consumption and gas emissions also decrease. In both cited approaches the information regarding the density of the vehicles approaching the intersection is gathered via V2V communications and is further transmitted to the traffic light.

However, most approaches adapt the speed of the vehicles to the traffic light phases by exploiting the traffic light – to – vehicle communication (I2V communications) in order to avoid stopping at the signaled intersections, having inadequate speeds or performing maneuvers that lead to increased fuel/energy consumption and/or gas emissions. These approaches can work both with static traffic signals and as shown in [227] adaptive traffic signals and are also known in the literature as Green Light Optimal Speed Advisory (GLOSA) solutions. GLOSA approaches are presented in [48], [49], [206] – [208], [229], and are dedicated to internal combustion engine-powered vehicles.

In [206], the focus is not on the mechanisms behind the speed advisory system, but on studying the factors influencing the reduction of the fuel consumption and gas emissions when such a GLOSA system is employed. The main results of the study reveal two such important factors: the gear choice and the distance from the traffic light where the message containing the information from the traffic light is received by the vehicles. For fuel consumption and gas emission measurements, the authors employ the Passenger car and Heavy duty Emissions Model (PHEM), developed at the Institute of Internal Combustion Engines and Thermodynamics of Graz University of Technology Austria, which is highly used in various R&D projects [209], [210].

In [48], [49], [207] and [208] the focus is on the speed advisory algorithm of GLOSA systems: finding the appropriate speed that will prevent stopping at the intersection if possible and minimize the fuel consumption and gas emissions. The approaches proposed in [49] and [208] do not consider in computing the appropriate speed any fuel consumption or gas emissions model, and employ these models when evaluating the performance of the GLOSA systems proposed only.

These solutions consider the vehicle's different maneuvers only (e.g. acceleration/deceleration), and from this point of view, the approach presented in [208] is the most complex in the literature so far, as it considers all possible maneuvers. It also includes complex testing, the performance of GLOSA system being evaluated against penetration rate variations and using an almost realistic scenario. In [48] and [207] the authors do consider a fuel consumption and emission model when computing the appropriate speed, namely the Virginia Tech Microscopic (VT-Micro) model. The goal is to find the optimum speed, especially in [207] where a very complex algorithm is employed for finding this optimum. Specific to [207] when compared to the other presented GLOSA approaches is the fact that V2V communication is also employed in sending the traffic light phasing messages in a multi-hop architecture in addition to I2V (traffic light-to-vehicle communication). In the GLOSA solutions presented, the benefits in terms of fuel savings and gas emissions reduction vary in a range of 8% - 22%, the higher ranges being associated with simple testing scenarios that consider a single intersection. However, the lower range can be even lower at small penetration rates as shown in [208].

In the most recent eco-driving solution described in this section, Xiang et al. [229] did the first steps in considering driver reactions in the context of GLOSA. A novel model for GLOSA that is adaptive to the driver behavior was proposed. Benefits in terms of fuel consumption are proven against the baseline for a vehicle in a single intersection. However, before imposing as a reference model, the model will need to be tested in the field which is a goal specified by the authors in their future works along with more extensive simulations for more complex scenarios.

In [211], Tielert et al. performed a study in order to demonstrate that similar GLOSA systems proposed for reducing fuel consumption and gas emissions can be employed for reducing energy consumption of electric cars. The focus is not on computing the appropriate speed, simple policies are implemented in computation. These policies are based on the strategies used for internal combustion engine-powered vehicles. Instead, while showing benefits in terms of energy consumption reduction, the authors of the study underline the need for GLOSA systems for EVs to take into consideration EV specific characteristics as compared to internal combustion engine-powered vehicles.

3.1.3 Discussion – Future Directions

There are very few energy-efficient or aware solutions based on VANETs dedicated to EVs are clearly very few and in their very beginning. Currently, battery models for microscopic simulation of EVs are being proposed and steps necessary to integrate EVs in VANET-specific simulators are undertaken [230], [231]. The lack of simulation tools for EVs is one of the reasons there are very few VANET-based eco-solutions dedicated to EVs. As EVs are a hot spot in both research and industry and they have as main issue energy and the relative small range imposed by energy limitations, energy-efficient solutions are and will be of high interest. Moreover, a class of EVs, electric bicycles, was completely neglected so far in the context of VANETs. Note that electric bicycles have different energy models, different traffic conditions, and different facilities than electric cars that were so far considered in the solutions. Consequently, dedicated energy-efficient solutions must be developed for electric bicycles that are not only the most popular among the electric vehicles [220], but also their popularity is on rise[10]. Cyclist smartphones could replace OBU, as there are already solutions for deploying the IEEE 802.11p communication stack on mobiles. Therefore eco-routing and eco-riding solutions dedicated to bicycles that aim to improve energy efficiency in the case of electric bicycles or reduce cyclist effort based on vehicular communications could be proposed. These can provide better service or enhance the service of existing cycling planners. Existing cycling route planners are static and they assist the rider only in creating a cycling route a-priori to their trip. Some cycling route planners allow for a single-criterion routing as [221] or as the cycling planner recently incorporated in Google Maps[11]. This new feature of Google Maps is further used as a support for building other cycling route planners and applications such as Bike Route Planner (&tracker)[12] application for Android. The more complex ones allow for multi-criteria routing such as [222], [223].

In [222], the authors proposed a web-based cycling route planner meant to promote cycling in Vancouver, Canada. This solution enables users to find a cycling route from start to destination based on their preferences. Users can select one of the following preferences: shortest path route, restricted maximum slope, least elevation gain, least traffic pollution and most vegetated route. Separately, users can select between two road types: cycling-friendly roads that encompasses cycling lanes and roads considered safe by the cyclists and municipal planners, and all roads. The preferences were decided based on the survey that determines main cycling motivators and de-motivators [17].

BBBike [223] is a cycle route planner originally developed for Berlin, Germany but then extended to 200 cities worldwide based on OpenStreetMap[13] data. It can be used either online or as desktop/Android application and allows for a route selection based on multiple criteria: shortest route (default setting), road surface, street category (similar to road type but more than two types are considered), avoidance of unlit streets, avoidance of traffic lights and green routes preference. The latter two criteria are available for Berlin only. Due to the recent interest shown in sustainable transportation and especially in cycling, a special branch of OpenStreetMap was dedicated to

[10] ZDNet website, http://www.smartplanet.com/blog/cities/popularity-of-electric-bikes-on-the-rise/4413
[11] Google Maps, https://www.google.ie/maps
[12] Google Apps Bike Route Planner, https://play.google.com/store/apps/details?id=net.bikerouteplanner&hl=en
[13] OpenStreetMap website, http://www.openstreetmap.org

cycling: OpenCycleMap[14]. Based on this, cycling route planners have been developed such as web-based BikeHub[15], BikeHub for smartphones (for Android[16], for iPhone[17]) that provides two options: the fastest road or the quietest one.

The solutions proposed in the thesis come to complement and enhance these approaches. One of the contribution represents a step forward for the cycling route planners, going beyond planning the route itself (how to get from point A to point B) and planning the departure time for the route: when to leave from point A to get to point B on the planned route in order to avoid as much as possible the adverse weather conditions and to increase the energy savings in the particular case of electric bicycles. Moreover, the first VANET-based GLOSA system for the bicycles is being proposed in this thesis that aims to increase cycling/user experience and decrease energy consumption in the particular case of electric bicycles.

3.2. Fuzzy Logic Solutions in VANETs

This section presents and discusses most important FL-based solutions proposed in the area of VANETs. As FL has recently started to be used in VANETs, the number of solutions is not high and there is still place for improving and experimenting in this direction. However, the FL has been employed in various types of VANET solutions such as routing, MAC protocols, handover, data aggregation, traffic (understood both as data traffic and road traffic), congestion detection, mobility models, and trust systems. The most representative solutions from numerical point of view are so far designed in routing, MAC protocols, handover and data aggregation. As emphasized in Chapter 2, the design of FLS in VANETs context is highly dependent on the solutions they are designed for. Therefore, FL solutions are next described in the context they were designed for.

3.2.1. FL-based and Routing Protocols in VANETs

3.2.1.1. Routing Protocols in VANETs

A routing protocol defines the way two communication entities exchange information in a network and, as described in Figure 3.1, includes path/route selection in the network, data forwarding and action in maintaining the route or recovering from route/link failure [144].

In the literature there are several surveys about routing protocols in VANETs such as: [144] – [148]. Based on the aforementioned surveys a general VANET-based routing classification is presented in Figure 3.2. Routing protocols are grouped into three major classes: unicast, multicast/geocast and broadcast [145]. Furthermore, VANET unicast routing protocols are categorized into two big classes: topology-based routing protocols and geographic-based (position-

[14] OpenCycleMap website, http://www.opencyclemap.org/
[15] BikeHub website, http://routes.bikehub.co.uk/
[16]Google Apps BikeHub, https://play.google.com/store/apps/details?id=com.bikehub.journeyplanner&hl=en
[17] Iphone App, BikeHub, http://www.bikehub.co.uk/iphone-app/bike-hb-iphone-app-now-in-review/

based) routing protocols [144], [147]. Similar, multicast routing is classified in topology-based and position-based [148]. The geographic-based routing protocols represent a new type of routing dedicated to VANETs that are exploiting the VANET characteristic of having a geographic type of communication. FL is applied in a variety of routing protocols that subscribe to most of the classes presented. The classes of routing solutions in which FL is applied are hashed in Figure 3.2. Moreover, FL is employed in two from the three phases of routing (as illustrated with yellow in Figure 3.1.).

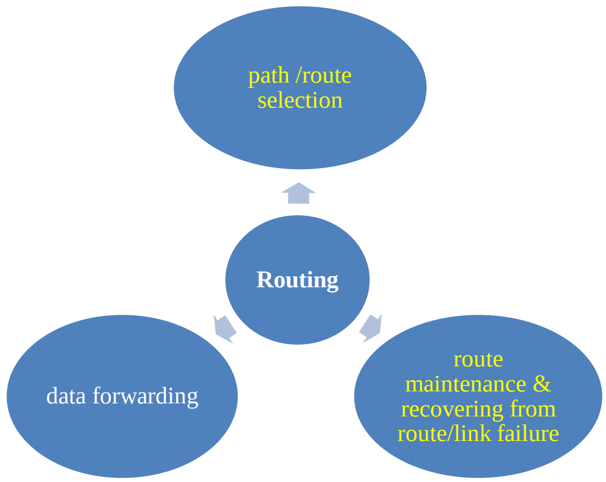


Figure 3.1. Routing Protocol Description

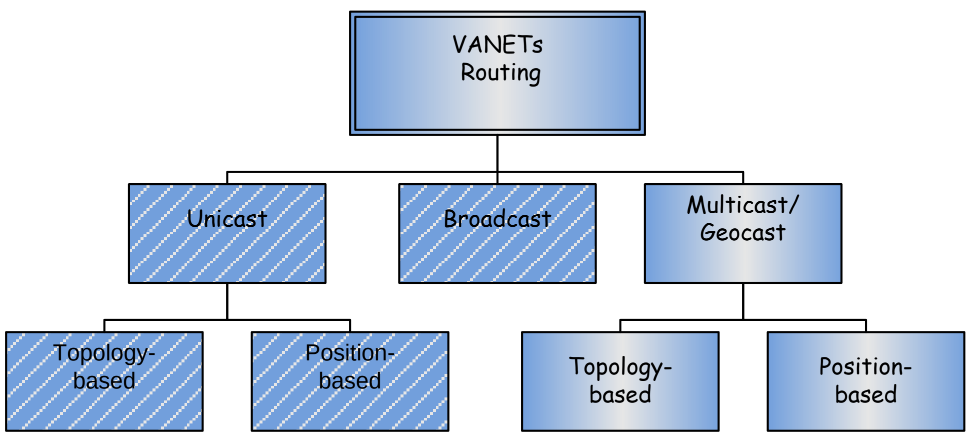


Figure 3.2. VANET-based Routing Classification

Routing is not trivial in VANETs because of their specific challenging characteristics which include: rapid change in topology, relative high speed of nodes, dynamic information exchange, frequently disconnected networks. Consequently, it is of high importance to consider these characteristics in VANET routing protocols in order to create stable routes. Moreover, VANETs are also associated with several particular characteristics that can be very helpful such as predictability of the movement. In developing routing protocols, taking advantage of these useful particularities often increases their performance. Thus, it is clear that a routing protocol in VANETs should consider multiple parameters that should also include VANET specific characteristics. However, so far there are no deterministic models that describe with precision the influence of these parameters in routing. FL is known as a perfect mathematical framework for handling multiple parameters and dealing with imprecision and non-deterministic problems. Therefore, FL seems to be a perfect candidate to solving routing challenges in VANETs and the following section demonstrates this idea.

3.2.1.2. FL-based Routing Protocols in VANETs

Wang et al. [149] proposed the Fuzzy control-based AODV routing (Fcar), which enhances the performance of the classical ad hoc on-demand distance vector routing (AODV) by taking into consideration VANET specific criteria. Simulations performed show that Fcar improves the routing performance in comparison with AODV in a VANET context. Several criteria are incorporated into the input parameters of the FLS designed for route selection: percentage of same-directional vehicles and route lifetime. Route lifetime is computed based on vehicle speed and distance between vehicles related to the effective communication range between vehicles. A FLS is very flexible and allows for multiple inputs which lead to multiple path/route selection criteria. This aspect was considered by the authors when choosing to employ FL in their solution. The best route is indicated by the output parameter of the FLS which is the selection probability. FLS has trapezoidal membership functions, the authors motivating their decision based on the fact that these are simple and computational efficient. In the design of the rule base, manual tuning was adopted ("simulation by a mass of tests and adjustments" [149]). The authors do not provide any details about the inference type implemented in the FLS or about the defuzzification process.

Huang et al. [150] proposed a FL-based load balancing and congestion-avoidance routing protocol aiming to satisfy the stringent Quality of Service (QoS) requirements of real-time traffic over VANETs. The proposed protocol is derived again from AODV which is considered to be suitable for bandwidth constrained routing and provides some minimal control to enable nodes to specify QoS parameters. In addition, VANET specific criteria are taken into consideration in order to address the rapidly changing topology. These parameters were incorporated into the inputs of a FLS that determines as output the appropriateness of the vehicle as an intermediate node in the route. Namely, the inputs are: minimum bandwidth required for satisfying the real-time traffic (R)/length of the queue for the non-real-time traffic (Q), expected remaining connection time between the vehicle and its neighbors (L) and the bandwidth currently used (B). The traffic is differentiated between real-time and non-real-time and dependent of this, the FLS has R, L and B as inputs or Q, L and B, respectively. The VANET specific criteria are incorporated into the L input which is dependent on vehicle's speed, position and vehicle's mobility model. The other two inputs are considered in such way to avoid congestion and to provide load balancing. In this way, the protocol proposes to satisfy the stringent QoS requirements in VANETs, especially in the case of real-time traffic [34]. In the design of the FLS, triangular functions are the authors' option for the membership functions. The membership functions are automatically tuned using H-infinity filtering in order to adapt to the volatile characteristics of VANET. Thus, the FLS designed is adapting in real-time and subscribes to the generalized architecture illustrated in Chapter 2, Figure 2.8. The authors experimented both neural networks (NN)-based tuning and genetic algorithms (GA)-based tuning, but the performance tests have shown that while the accuracy is comparable, NN approach performs slightly better, while GA is visible worse. The learning time is also much longer for these latter techniques. In VANETs, due to the multiple and repeated network topology changes, frequent training is required, so NN and GA based learning techniques are not suitable for real-time applications in VANETs. However, this last statement cannot be generalized, as not many details are provided about the specific learning algorithms implemented and multiple NN and GA-based learning algorithms can be implemented. The rule base of FLS resulted from "expert knowledge" on the deep understanding of the influences of each parameter in the network. The Inference Engine is based on the Tsukamoto inference and consequently, the defuzzification process is also Tsukamoto.

Ghafoor et al. [151] proposed a FL-based delay and reliability-aware routing protocol aiming to overcome the disadvantages of the existent protocols for mobile ad hoc networks that do not take into consideration the specific characteristics of VANETs or the fact that the communication channel between vehicles is prone to radio frequency interference and shadowing as well, as different forms of fading including multi-path and slow fading. The FLS designed has as inputs signal to noise ratio (SNR) as an estimator of the behavior of time varying channels, channel delay and velocity vector difference. This latter parameter is a mobility-related parameter representing the difference in speed between the current vehicle and its neighbors. The output of the FLS is the Fuzzy Cost that indicates the probability of the vehicle being chosen in the routing path. FLS is deployed in each vehicle and whenever the source or an intermediate node selects a routing path FLS are activated in each of the source or intermediate node neighbors. The neighbor having the highest Fuzzy Cost is selected in the routing path. In the design of the FLS, the authors opted for triangular functions motivating that these are extensively used in real-time applications and they are computationally efficient. The functions are initially chosen based on experience and then automatically tuned using H-infinity filtering technique to adapt in real-time to the rapid changing topology of VANETs. The automatic tuning is based on reinforcement learning and the FLS has the real-time adaptive architecture illustrated in Chapter 2, Figure 2.8. The rule base benefitted from expert knowledge and "careful understanding of the philosophy behind metrics and vehicular networks behaviour". As inference type and defuzzification, the authors have chosen the Mamdani inference, respectively COA.

Ghafoor et al. [152] proposed a FL-based routing protocol aiming to tackle the problem of the lossy wireless channels, but considering other parameters. In addition, a mechanism is implemented to deal with data forwarding in VANET environment that is prone to network disconnection. FL is implemented only in the route selection mechanism. One of the inputs of the FLS that is deployed in each vehicle is the relative distance between the source node/intermediate node and its neighbors based on the consideration that lossy wireless channels lead to signal attenuation. The other input is related to vehicles' mobility and is the relative direction. The output, similar to the previous discussed approach, is a Fuzzy Cost computed for each neighbor of the source/intermediate node. The membership functions selected are triangular and their parameters are selected based on experience and manual tuning – trial and error. The rule base is again based on "expert knowledge". Similar to the previous approach, the inference type is the Mamdani inference, while the defuzzification is performed via COA. There is no comparison between the two FL-based routing protocols proposed by the same authors. Both have the same main purposes and both employ a FLS to decide the best path in routing. However, the parameters considered as inputs for FLS are different and it would have been very useful to see which of the two solutions is better and in what conditions.

Khokhar et al. [153] proposed a routing protocol dedicated to urban vehicular environment that employs novel concepts in order to address security challenges in VANETs. The novel aspect is the rationale behind the routing protocol. The authors state that there are some social behavior patterns developed in urban environments and these should be exploited in order to make secure routing decisions. A friendship mechanism is developed that is used in taking routing decisions at intersections. The decision making is based on a FLS that has the following inputs: Friends, Friends of Friends and Non-friends and gives as output the path fuzzy cost. The list of friends of a vehicle is made based on the social behavior pattern developed in traffic: if there was a V2V communication between the vehicle and another vehicle, the latter is its friend. In addition, the social networks of the drivers are considered. For instance, if the drivers are friends on Facebook, their vehicles are friends as well. Regarding the FLS design decisions, the authors opted for triangular functions as they are computationally efficient. Membership functions' parameters and rule base were established based on expert knowledge and manual tuning. The COA defuzzification method is the authors' option for defuzzification, while the inference type is not revealed.

Wu et al. [154] introduced a FL decisional system to select the nodes where to relay the broadcast messages in the context of a new broadcast protocol for VANETs. The technique of using only a few neighboring nodes for relaying the broadcast messages ensures the efficiency of the proposed broadcast protocol. The FLS uses multiple parameters in the relay node selection which ensures the high reliability of the protocol. These parameters are distance factor (inter-vehicle distance), mobility factor that considers both the current position and the next moment position and received signal strength indicator. They are used as inputs by the FLS that outputs the rank of a node. The node with the higher rank is selected as relay node. The FLS has trapezoidal and triangular functions, but the authors do not give any details about the rationale behind this design decision. The selection of functions' parameters is not motivated either, nor it is the rule base design. Information is provided regarding the inference type (i.e. Mamdani) and defuzzification method (i.e. COA). In [232], this FL-based solution is combined with network coding in order to reduce the packets retransmission in case of failure. However, the characteristics of FLS are not changed.

In [155], Huang et al. designed and employed two FL-based decisional systems for the selection of relay nodes in an infotainment dissemination scheme over vehicular heterogeneous networks. The design decisions of the FLSs are very poorly described. The first FLS is used to decide upon the most appropriate vehicle from a list of candidates to become the relay node that is storing and forwarding the infotainment data to the requesting nodes. A second FLS is designed to decide from these requesting nodes if they are suitable to become relay nodes when needed. The first FLS has as inputs bandwidth, overhead and lifetime, the latter parameter representing a combination of VANET specific criteria: vehicle velocity, distance between vehicles and direction of vehicles. These criteria define the connectivity in the VANET dynamic and ever-changing environment. The output is represented by the appropriateness of the node to be a relay node. The membership functions are triangular and the FLS has a Tsukamaoto inference type and defuzzification method. None of the design decisions are motivated. Regarding the second FLS, no details are provided except for the inputs that are represented by the capabilities of the requesting node such as computation capability, buffer size and the stability of the signal strength and the output is the suitability of the node to become a relay.

In [156] and [157], Wu et al. proposed FL and Q-learning [158] based approaches for unicast routing. Both approaches derive from the AODV protocol. The routing protocol proposed in [156] is called FQLAODV (Fuzzy Q-Learning AODV-based protocol), while the one proposed in [157] is called PFQ-AODV (Portable Fuzzy Q-learning AODV-based protocol). Both protocols have the same basic principle: they use a FLS for evaluating the link that is possible to be used in the routing path and based on the ranking provided by the FLS, Q-learning is applied for selecting a route that ensures multi-hop reliability and efficiency. The differences between FQLAODV and PFQ-AODV are in the inputs considered for the FLS and in the fact that the last one does not make the assumption of existence of any GPS or other positioning system, making it portable and more practical. The FLS in FQLAODV has as inputs: bandwidth, mobility factor and received signal strength indicator. The last two inputs are exactly the same computed and have the same membership functions as the inputs of the FLS proposed for the broadcasting protocol in [154]. PFQ-AODV has more refined inputs, considering again bandwidth, mobility factor and a new input link quality. Mobility factor is computed different than in FQLAODV, as it does not make the assumption that the vehicles know their positions. Link quality is a more complex input than received signal strength, taking into consideration in its computation a network metric, packet loss, and a topology metric – number of neighbors. In the design of both FLSs, same decisions are applied: triangular and trapezoidal membership functions for their efficiency, Mamdani inference type and COA defuzzification method. The membership functions' parameters selection is entirely based on the authors' experience and knowledge, thus expert knowledge, although the authors do state that automatic tuning is possible and brings an advantage to FLS in VANETs – adaptability to any kind of conditions (e.g. sparse or dense network). Moreover, the benefits of employing FL in the proposed routing protocols are demonstrated against AODV and QLAODV [159] – AODV modified with Q-learning via both simulations and real testbed. However, the authors do not perform a comparison between their two FL-based approaches.

Huang et al. [160] introduced a FL-based proactive recovery from link failure mechanism. This mechanism, deployed in each vehicle, involves two components: an alternative link construction component and a prediction component. The prediction component predicts both the congestion or link failure and either if congestion is detected or link failure is predicted, it activates the alternate link construction component. The prediction component has two modules: Fuzzy speed prediction module and Fuzzy congestion detection module. Basically, these modules are represented by two FLSs. The FLS predicting the speed incorporates some knowledge about driver's age as there is a connection between driver's age and driver's behavior. Thus the inputs of this FLS are driver's age, distance between vehicle and the front vehicle and current speed. The output is the predicted velocity. The FLS for congestion detection has as inputs: the queue length, the hop counts that the packet travel through the vehicles and the expected number of the vehicles within radio range of the vehicle during next time period, and as output the congestion indicator. Both FLS have similar design: trapezoidal membership functions chosen to reduce computational complexity, and Tsukamoto inference and defuzzification method. The design of the FLS is a real-time adaptive design that allows for automatically tuning the parameters of the membership functions based on Particle Swarm Optimization techniques. The authors do not provide any details about how the rule base was built.

3.2.1.3. Discussion

Routing is considered among the main challenges in VANETs as it faces an ever-changing environment. However, as emphasized before, VANETs do not impose challenges only, they also have attractive characteristics such as the fact that the movement of the nodes can be more easily predicted as the vehicles follow the road rules and regulations, topology, etc. Thus, selecting the best route/path is a complex decision that should be based on multiple criteria that incorporate also the characteristics of the VANET complex environment. In almost all the analyzed solutions, FL is applied in decision making based on multiple metrics that are fed as inputs into FLSs that provide as outputs a ranking of best choices: a list of fuzzy costs, appropriateness, selection probabilities, ranks, etc. In the unicast routing, the FLSs provide the ranking that indicates which is the best link/path in the route ([149] – [153], [156], [157]). In broadcasting, the FLS provides the ranking that indicates the best choice for the relay nodes supposed to store the information to be broadcasted ([154], [155]). A single routing protocol from the analyzed approaches employs FL in prediction in order to design a mechanism for proactive recovery from failure [160]. Two FLSs are used: one to predict the future speed and the other one to predict the network traffic congestion.

Regarding the inputs selected for the FLSs, most of the approaches search for the successful combination of parameters to describe the high mobility and rapidly changing topology of VANETs and to take advantage of the predictability of the movement. However, so far the similar FL-based routing solutions (e.g. unicast topology-based solutions) have not been compared among themselves and consequently we cannot state with accuracy regarding the most successful combination of parameters used for describing the VANET environment in order to make the best decisions regarding the best route. The lack of comparison among the FL-based solutions also prevents us from learning some lessons regarding the performance of different fuzzy models or designs in the context of VANET routing. So far, Tsukamoto model and Mamdani with COA as defuzzification method were employed in the FL-based routing protocols. A comparison could result in some conclusions regarding the Tsukamoto vs. Mamdani fuzzy models performance in VANET routing. Thus comparison between FL-based routing protocols is a "must" in order to be able to decide the best approaches and the best design decisions that can be at the basement of future improved works in this direction. However, there are some lessons regarding the design of FLSs in the context of VANETs routing that can be learnt even in the absence of these comparisons. First lesson is that triangular and trapezoidal functions provide good performance in this context and due to their reduced computational complexity and efficiency they are perfect candidates when choosing to design any FLS used in VANET routing. The Rule base is built in all the approaches presented based on expert knowledge and in very few approaches is further refined through manual tuning. These rules can be further used and improved by tuning when designing new approaches.

FL is successfully employed and presented in the literature, in various domains to address both problems: decision making and prediction. The FL-based routing protocols analyzed demonstrate that FL can be successfully employed in the context of VANET routing in both decision making and prediction.

3.2.2. FL-based MAC Protocols in VANETs

3.2.2.1. MAC Protocols in VANETs

MAC protocols are considered to be a key issue in the design of VANETs [34] and they are identified among the main technical challenges imposed by VANETs [23]. In a VANET context, efficient MAC protocols need to be designed in order to cope with the highly dynamic environment. In addition they need to be able to provide high quality of service (QoS) levels for non-safety applications and reliability for safety applications. It is clearly stated the need of designing new MAC protocols designed for VANETs, as the existent ones are not suitable [161].

In [162], the MAC protocols for VANETs are classified in two broad classes: contention-based protocols and scheduled-based protocols. The first class has the advantage of not being influenced by the ever-changing topology of VANETs, but their main problem relates to the random delay introduced in order to regulate the access to the medium so that the chances of collisions are reduced. This delay is bounded by an interval, also called "backoff interval" that is statically increased/decreased, approach that is not the best for a dynamic environment such as VANET. Thus, the delay is controlled through the increase/decrease of this backoff interval regulated by so called backoff schemes. An inappropriate control of this delay can cause serious issues especially in the case of safety applications. The second class of MAC protocols does not have this delay issue, but it is influenced by the topology change – slot relocation may often occur due to the rapidly changing topology of VANETs. So far, FL has been implemented in VANET MAC protocols pertaining to the first class (Figure 3.3) in order to address their main drawback.

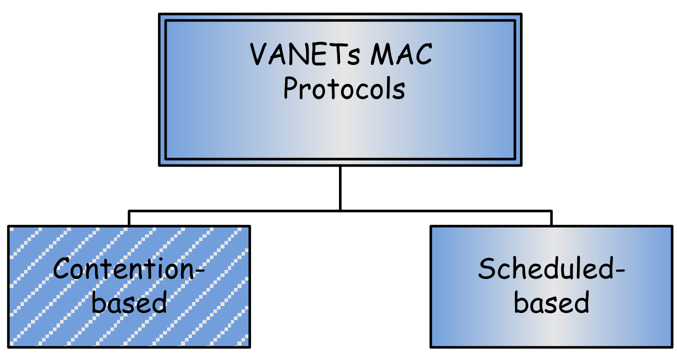


Figure 3.3. VANETs MAC Protocols Classification

3.2.2.2. FL-based MAC Protocols in VANETs

Abdelkader et al. [163] proposed a feedback FLS that controls the changes in the backoff interval (i.e. the level of increase/decrease). They started from an observation that a constant increase/decrease of backoff interval is not appropriate for the dynamic environment of VANETs and that this process should take into consideration the dynamic network conditions. So, each node in the network should monitor network conditions and based on this to control their backoff interval increase/decrease. However, there is no direct mathematical mapping between network conditions and backoff interval computation therefore an exact model cannot be built. Thus, FL known for its capabilities to deal with imprecise information is selected for modelling the relationship between the network conditions and backoff interval computation. The network conditions monitored are successful transmission ratio (S) and the last backoff interval value (Blast). S measures the fraction of delivered packets to the total generated packets per node, while Blast is the measure of network load.

The FLS designed for controlling the increase/decrease in the backoff interval uses S and Blast as inputs and gives as output the normalized amount of decrease/increase of the interval, dB. The actual value to modify the backoff interval is obtained by multiplying dB with Blast. Note that this is an adaptive FLS as the output processed in Blast is fed back into the FLS as input. The membership functions for all inputs and output are triangular and their parameters are described in the simulation scenario, but the authors do not explain how their selection has been done. The rule base is designed based on "expert knowledge" – authors knowledge regarding the influence of the network conditions taken into consideration and backoff interval – and manual tuning – "trial experiments". The inference type is not specified, but from the description of the FLS it appears to be Mamdani. The COA method is applied in defuzzification.

It is demonstrated that the FL-based backoff scheme outperforms wide-used backoff schemes in MAC protocols in terms of throughput and fairness and gets closer to the performances of what is considered to be an ideal backoff scheme [164].

In [165], Abdelkader et al. proposed four FL-based backoff control schemes: three are built upon SISO FLSs and the other one is built upon a MISO FLS. All these FLSs are designed based on the previously proposed FLS [163]. The MISO FLS in [165] has the same inputs (successful transmission ratio – S – and last backoff interval value – Blast) and output (change in backoff interval – dB) as in [163]. The changes are in the membership function of the output, that now has 5 fuzzy terms instead of 3 and consequently the rule base is extended with new rules, but it is built based on the same considerations – "expert knowledge" – as in [163]. Also, the parameters of the input membership functions used in the simulation, again their selection is not motivated, are different than the ones used in [163].

Regarding the three SISO FLSs proposed, they have only one input, S, and same output as the MISO FLS and each of them follows a certain policy:

- Selfish policy, where the node objective is to access the network regardless of its limited share. The backoff scheme based on the FLS implementing this policy is called Fuzzy1-DS.

- Generous policy, where the channel is given to other nodes if found busy. The backoff scheme based on the FLS implementing this policy is called Fuzzy-1GS.

- Cautious policy or a fair policy where each network node has the objective of a fair access to the medium. The backoff scheme based on the FLS implementing this policy is called Fuzzy-1CS.

The rule base for each of the systems is designed based on the same “expert knowledge” that is the understanding of the authors on how the input influence the output. This knowledge can be summarized as follows: if the success ratio is very low this means that the channel is busy so the probability of collision is very high, therefore the recommendation is to increase the backoff interval. This recommendation is reflected differently by the three FLS in their rule bases depending on the policy adopted: selfish, generous or fair. For instance, the selfish scheme, Fuzzy1-DS, will tend to rather decrease the backoff interval than to increase it in most of the circumstances. All FLSs have triangular functions and their parameters are specified in the simulation scenario, but no explanation of how these were selected is given. All the four FL-based backoff schemes are compared against a widely-used backoff scheme for MAC protocols and the one considered optimum [163]. Most of FL-based schemes outperform the first one and get closer to the optimum in terms of throughput, but in terms of fairness, Fuzzy1-DS scheme suffers the most.

Souza et al. [166] proposed a FL-based adaptive mechanism for a better control of access to the medium. Similar to the previous discussed approaches (i.e. [163] and [165]), the backoff is controlled depending on the network conditions. The parameter monitored is the density of the vehicles within the transmission range of each vehicle. This parameter is the only input of the FLS that is deployed in each of the vehicles for controlling the access to the medium. The output is the backoff value. The membership functions are gaussian functions and their parameters are chosen based on previous studies in the literature that relates to the traffic density [167] and backoff values [168]. The rule base is designed based on “expert knowledge” that can be summarized as follows: “If the density is low, the fuzzy system returns a small backoff value and close to the optimum value in order not to underutilize the network. On the other hand, if the network is dense, the returned backoff value is great and near to optimal value, enough to provide a good cost-benefit relationship between throughput and collisions or loss.” [166]. No information is provided about the inference and defuzzification method implemented in the designed FLS.

The improved performance of the FL-based adaptive backoff scheme is demonstrated against the one implemented in the IEEE 802.11p standard that is non-adaptive based on network conditions. It is interesting to mention that one of the metrics used for assessment is the success ratio that is used as input in the previous discussed FL-based backoff schemes proposed in [163], [165].

The authors should have considered a comparison against the scheme proposed in [163], as their approach was probably developed in parallel with [165]. However, the lack in details regarding the design of the FLS makes the comparison between these schemes very difficult. For instance in [163], we only assumed based on the description a Mamdani inference type was used, but this was not clearly specified.

Chrysostomou et al. [169] proposed a FLS to control the wireless access in an adaptive QoS-aware MAC protocol. The proposed MAC protocol presents a different approach for controlling the backoff value. It keeps the IEEE 802.11p basic principle of updating the backoff interval based on the contention window (CW) value, but controls the CW based on network conditions reflected in channel traffic occupancy (CTO). Keeping the IEEE 802.11p basic mechanism allows for the differentiation and prioritization of different traffic types in the proposed solution. This combined with the FL-based control scheme of CW lead to providing an improved QoS compared to the classic IEEE 802.11p. The FLS ensuring the control of CW has as inputs CTO values for consecutive sampling periods. The membership functions are triangular, design decision motivated by the authors based on their computational simplicity. Their parameters were selected based on the "qualitative understanding of the system". Similarly, the rule base was built on the understanding of the system, "expert knowledge", and manual tuning. The philosophy behind the knowledge base is on the one hand aggressive response when the density of the channel is very high for two consecutive periods of time and on the other hand smooth response when the density is low. No information about inference and defuzzification processes is provided.

In [170], the same authors proposed a FL-based mechanism for controlling the *min* and *max* values of CW. According to the authors, this is the first scheme in the literature in which *CW min* and *CW max* values are adapted based on network conditions. The FLS has the same inputs as the FLS proposed in [169] and as output the value controlling the increase/decrease of *CW min* and *CW max* values called *factor*. Similar design principles are adopted for designing the proposed FLS as those employed in [169].

Hafeez et al. [171] described their new VANET MAC protocol which makes use of an adaptive FLS for predicting the speed and location in order to adjust the protocol to driver's behaviour on the road. In this solution, FL is not employed directly in controlling the access to the medium, the challenge of the contention-based protocols being addressed in this approach through clustering. The FLS is used in the cluster maintenance process and based on the predictions made, the structure of the cluster is updated or not. The inputs of the FLS are the speed and inter-distance and the output predicts if the driver is going to accelerate or decelerate. Based on this prediction, the vehicle's speed and position in the near future are further predicted. The membership functions of inputs and output are triangular. A basic reinforcement learning algorithm is implemented for automatic tuning of the parameters of speed membership function. Basically this mechanism adapts the FLS to the driver's behaviour. The rule base of the FLS is fully described and it is based on "expert knowledge". Also, fully information is provided about inference – Mamdani inference type is implemented – and defuzzification method adopted – COA. This is probably the only FLS proposed in the context of MAC protocols that has a complete description.

3.2.2.3. Discussion

FL is well-known for its popularity in control applications, especially but not exclusively in the automotive industry. The solutions analyzed and their results prove that FL can be successfully employed for control purposes in a VANET context as well, namely in controlling the access to the medium in VANETs. Except for the SISO FLSs [165], [166], the rest of the solutions designed for control have a simple real-time adaptive FLS architecture: a typical architecture of a FLC illustrated in Chapter 2, Figure2.9. From the protocols analyzed, a single one employs FL in prediction, namely in predicting if the driver accelerates or decelerates and based on that the future position and speed of the vehicle is predicted.

The FLSs proposed in the context of MAC protocols lack in details regarding the design decisions. Moreover, the same issue that was outlined in the case of FL-based routing protocols can be outlined here as well: FL-based MAC protocols are not compared against themselves, excepting the approach where the authors proposed four FL-based schemes for controlling the backoff interval and a comparison was made between these four [165]. However, in most of the approaches the rule base is completely described, and therefore these rules can be further used in designing new FLSs in a similar context. Moreover, the rules describe a very detailed dependency, in a linguistic language, between the inputs used and the outputs. Thus, although comparisons are not made between solutions, and therefore it is not possible to state which is the best combination of inputs in the control solutions designed, the rules can lead towards an in-depth understanding on the influence of the inputs considered. This provides a support for better decisions regarding the parameters that can be considered as inputs when a new FLS is designed for controlling the medium access in VANETs. From the solutions analyzed, a single one uses gaussian membership functions, the rest are using triangular functions. Thus, we can emphasize on the conclusion drawn in the section referring to the FL-based routing protocols, that triangular functions are perfect candidates for designing new FLSs in the context of VANET MAC protocols in particular and in the context of VANET solutions in general.

3.2.3. FL-based Data Aggregation Solutions in VANETs

3.2.3.1. Data Aggregation in VANETs

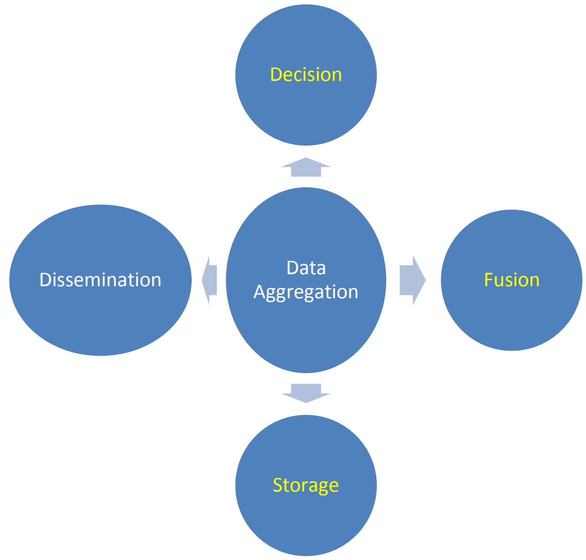


Figure 3.4. Data Aggregation – Functional Components

One of the major challenges in VANETs, networks with potential large numbers of vehicle-nodes, is the efficient usage of the available bandwidth [172]. Data aggregation addresses this issue in the context of data collection, avoiding the dissemination of similar information in the network. Thus data aggregation is used to combine correlated information from different nodes before redistributing the information in the network. Data aggregation process is illustrated in Figure 3.4 and basically consists of the following functional components: decision, fusion, storage and dissemination [172]. In the decision component the data that is collected locally from the vehicle sensors or received from other vehicles from the network is analyzed to see whether there is any correlation between atomic data items and a decision is taken accordingly. If the decision component detects a correlation then the data is fused (i.e. fusion component). Data, fused or not depending on the correlation, is disseminated in the network (i.e. dissemination component). Regarding the storage component this is placed either before decision phase in which case it stores all the collected data, or after the fusion, in which case it stores the aggregated data before disseminating it in the network.

Data aggregation schemes in VANETs have several limitations and challenges. One common problem of data aggregation schemes is the lack of flexibility in the criteria of similarity between data. The information is correlated based on fixed or structured segmentation of the road assuming that two information items fit into the same spatial segment. The data aggregation schemes based on these decision components are not flexible, being static as they depend either on fixed road segments (e.g. [173], [174]) or predefined structures as trees (e.g. [175]) or clusters (e.g. [176]).

Another challenge that arises in data aggregation is the security. The integrity of aggregated data is harder to be verified, opening new opportunities for attackers. In order to address this challenge, some data aggregation solutions (e.g. [177]) rely on a tamper-proof service in each vehicle that requests for integrity proofs from the randomly chosen original records, but this service can be easily by-passed by attackers. Other approaches employ security mechanisms that are dependent on node reputation (e.g. [178]) or on some fixed structures (e.g. fixed road segments [179]). Security schemes based on node reputations are very hard to employ in the self-organized networks such as VANETs [180], while the security schemes based on fixed structures are demonstrated to have scalability issues [181].

Most of the data aggregation schemes proposed so far are application-oriented and are not able to cope with different types of data or even less with simultaneous applications. In this context, European Telecommunications Standard Institutes (ETSI) underlined the need of data aggregation standardization [182]. Few steps in this direction have been done to date [172], [183], and the identification of the four functional components is one of the results of these steps.

3.2.3.2. FL-based data aggregation solutions in VANETs

This section proposes to exhaustively discuss the FL-based data aggregation schemes in VANETs. As emphasized in Figure 3.4, FL was employed in three of the four phases of the data aggregation process so far.

Caballero-Gil et al. [178] proposed a data aggregation solution that employs FL in the decision component. The FL decisional system is exemplified with two inputs: space – the approximate location of the provenience of information, and time – the persistence of information. The output parameter is the correlation between the pieces of information to be aggregated and has two possible values: YES and NO. The solution is open to extensibility and generalization by considering other input parameters that might be dependent on the application type (e.g. speed, distance between vehicles etc.). In the given example, the membership functions used for inputs are triangular, while the output has a singleton function. The inference, as it appears to be from the rules description, is of Sugeno type. However, the focus is not on the FLS design, but instead on the benefits of using FL in the decision component of a data aggregation scheme: flexibility and extensibility in the set of criteria used for correlating the information for aggregation. This approach also proposes a FL-based selection scheme to be implemented in the storage component that is placed in this solution immediately after data collection. This scheme aims to select the most relevant data items for aggregation in order to avoid the overloading of communication channel that could lead to restrictions in data that is sent in the network to the vehicles. The scheme is not detailed, and only some of the parameters that could be considered in the selection are mentioned, such as severity or antiquity of information.

Dietzel et al. [71], [184] proposed a FL-based decision component in their data aggregation approach that does a step forward in showing the flexibility, extensibility and generality that can be reached via FL. The output parameter of the FL decisional system is the same as in the aforementioned approach, but the input parameters are generalized and called influences: *Influence 1*, …, *Influence n*. These are the parameters that represent an influence on deciding upon data similarity. Examples of influences are speed difference [71], [184] and location difference between vehicles [184]. In exemplifying the fuzzification of the inputs, the authors use trapezoidal membership functions for the speed difference input. However, the focus is again not on the internal design decisions of the FL decisional system, but on its generalization. As already emphasized, all the proposed FL-based decision components ensure flexibility and extensibility in the set of criteria used for correlating the information for aggregation. This resulted in structure-free and dynamic aggregation approaches unlike the others data aggregation approaches existent in the literature that do not rely on FL.

The flexibility and extensibility provided by the FL-based approaches are of great importance in the development of a data aggregation solution. As such, in the early stages of the development, such as the architectural description or design of the architectural components, there is no need of knowing all or even none of the parameters that influence the correlation between the pieces of information supposed to be aggregated. Moreover, as claimed in [71] and [178], it is actually more likely that in these early stages of the design these parameters not to be even known. By comparison, consider the other aggregation schemes that rely on structures where the decision component follows the following rule: if the pieces of data pertain to the same structure then these are similar, so consequently they are aggregated. This means that when designing such an approach we need to be absolutely sure about the parameters that define the similarity of data in order to create corresponding structures. If later on it is decided that there are other parameters that define the similarity, the impact on the solution is huge. Most probably the designed structures would not be valid anymore, as they would not reflect these new parameters. Basically a complete re-design would be required.

Based on the aforementioned arguments that show the advantages brought over the non-fuzzy solutions and based on its generality, a FL decisional system could be imposed as a design model for the decision component of the data aggregation schemes. This could represent a step forward in the modeling and the desired standardization of data aggregation schemes.

In [185], Dietzel et al. enhanced the fusion component that resulted in a FL-based trust fusion component aiming to address the security challenge imposed by data aggregation in vehicular networks. The FL-based security mechanism is data-centric. A probabilistic scheme is employed to achieve a selective attestation of the aggregates. The selective attestation process results in clues leading to trust in the correctness of an aggregate. The clues are used as input parameters of a FLS that has as output parameter the *Trust* with values in the {0%-100%} range. The FL-based design allows for extensibility and flexibility in the considered clues depending on the type of the application, been presented as a generalization. Thus the inputs of the FLS are represented as *Clue 1*, …, *Clue n*. Some examples of inputs are provided together with their fuzzification for which trapezoidal membership functions has been chosen.

However, as in the previous presented FL-based approaches, the focus is not on the internal design decisions of the FLS, but on its flexibility and generalization. Basically in each vehicle the parameters influencing the trust can be different, the clues being selected locally through the probabilistic scheme. Moreover, this generalized design of the security scheme can be employed in any type of data aggregation solution independent of the application type. In addition, the FL-based approach addresses some of the aforementioned limitations of the other security mechanisms existing in VANET data aggregation solutions. Thus, it removes the need for a tamper-proof service in each car, and it is neither dependent on node reputation as the scheme is data-centric nor on any kind of structures. In the case of structure-based trust mechanisms implemented in data aggregation solutions more or less same disadvantages are met as in the case of the data aggregation approaches that have their decision component dependent on such structures. The important disadvantage is the lack of flexibility in the correlation criteria, here, the lack of flexibility in the trust criteria. The vehicles pertaining to the same structure would either have to agree on the common views on the environment (i.e. view that is defining what is secure – trustful – and what is not), or would have this view "hardcoded" by design in the structure they pertain to. In the first case, there are great scalability issues as it was mentioned in the description of the data aggregation concept and challenges, due to the fact that vehicles need to exchange messages for sharing the common view. Changing in the trust criteria can be acquired easier compared to the second case, but this only augments the scalability issue, as more messages need to be exchanged. In the second case, changing in trust criteria is impossible. The whole data aggregation solution should be re-designed as it is highly coupled by the structures that are defined in this case by the parameters influencing the trust.

3.2.3.3. Discussion

Flexibility (structure-free), extensibility and generality are some of the requirements that data aggregation schemes based on FL have to meet. None of the solutions is bounded to specific inputs, thus none of the solution is bounded to any specific application requirements. The solutions presented can be considered architectural models for VANET data aggregation schemes based on FL. Two models can be outlined: a model for the decision component and a model for secure or trust fusion component. These models can represent steps forward in the potential process of standardization of data aggregation schemes in VANETs. One of the main benefits of using these architectures in developing data aggregation schemes for VANETs is the extensibility: the solutions are not bounded to any structures and late modifications to the solutions have a minimum impact on the design. It was emphasized during the presentation of the solutions that this impact is very high in the case of the other approaches in the literature. Taking into consideration the complexity of VANET environment, it is very optimistic to consider that the parameters that are defining the similarity in data for instance are unchangeable.

The FL-based architectural models can be further followed in proposing structure-free data aggregation solutions based on different and multiple criteria. Moreover, the solutions also suggest other possible applications of FL in the context of data aggregation schemes in VANETs. The solutions presented give the incentives necessary for applying FL in data aggregation. What needs to be done is to decide the inputs and to take the internal decisions regarding the design of the FLS. Regarding this last aspect, not many lessons can be learnt, as the solutions are focusing more on the flexibility and generalization, rather than on presenting particular solutions. However, some examples are presented and from these can be concluded that triangular, trapezoidal and singleton membership functions are perfect candidates in designing FLSs in the context of data aggregation solutions in VANETs.

3.2.4. FL-based and Handover in VANETs

3.2.4.1. Handover in VANETs

This section analyzes FL employment in handover (HO) solutions in the context of vehicular heterogeneous networks. HO in vehicular heterogeneous networks subscribes to the HO problem in heterogeneous networks. However, vehicular networks, due to their specific characteristics, need new approaches.

The HO process has three phases (Figure 3.5):

- monitoring (i.e. collecting the information related to network conditions based on which the HO decision phase is triggered)
- HO decision (i.e. selecting the most suitable access network – network selection – and deciding whether to switch to this network)
- HO execution (connecting to the pre-selected network).

HO decision phase has an overwhelming importance in the HO process as its performance highly depends on how the targeted network is selected in order to be best possible. In consequence, it is not a surprising fact that in the literature a large number of solutions were dedicated to solve the network selection issue [186], [187] and researchers employed a variety of mathematical models in solving this issue: utility function, multiple attribute decision making (MADM), game theory and FL [63], [188].

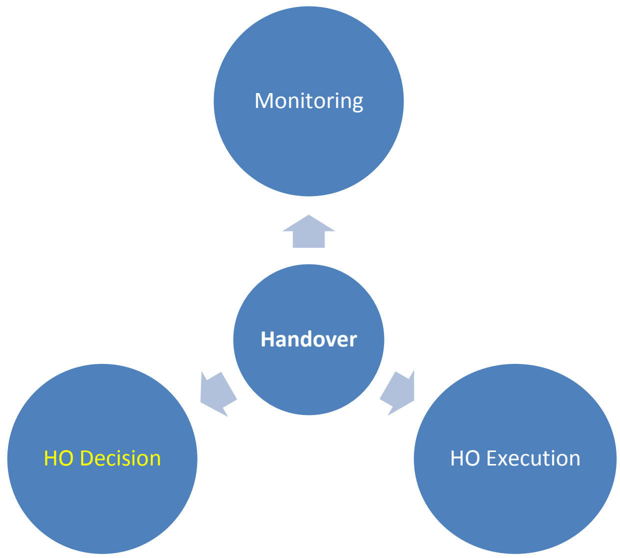


Figure 3.5. HO Phases

An effective network selection takes into account multiple criteria that can subscribe to the following classes: network metrics, device related, application requirements and user preferences [188]. Deciding with precision the influence of these parameters on the degree of electability of a network is impossible. Thus, in network selection there is a need of combining multiple parameters with a level of uncertainty. Therefore, FL, known for its capability of dealing with imprecision and with multiple parameters, provides a robust framework for HO decision. A considerable number of HO solutions for heterogeneous networks have employed FL in the network selection, such as: [189] – [192]. Some of the solutions are simply applying FL without any automatic techniques, tuning or learning, such as [189], [191], whereas other solutions, combine FL with other techniques such as MADM [190] or neural networks [192]. Note that these are not applied for building or tuning the rule base or membership functions. For instance, in [192] the neural network module is used to estimate the number of users in a network, result that is fed as input in the FLS.

The focus in this section is on the vehicular heterogeneous networks that, as underlined before, have different characteristics and requires dedicated solutions. Consequently, FL-based network selection solutions for vehicular heterogeneous networks are further analysed.

3.2.4.2. FL-based Handover Solutions in VANETs

This section analyzes FL employment in handover (HO) solutions in the context of vehicular heterogeneous networks. HO in vehicular heterogeneous networks subscribes to the HO problem in heterogeneous networks. However, vehicular networks, due to their specific characteristics, need new approaches.

Chantaksinopas et al. [193] proposed a general framework for network selection transparency on vehicular networks considering multiple decision criteria from each of the following classes: network metrics, device related, user preferences and application requirements. For the decision making process the authors propose FL as a Mathematical model. The authors provide only architectural details of the framework, where decision making is one of the architectural blocks, leaving the implementation details out of the picture.

Bechler et al. [194] employed FL in the network selection in a vehicular communication scenario. The particularity of the FLS designed to solve the decision problem is that it has actually two FL controllers connected: AbstractionController and DecisionController. The first one has multiple inputs representing network metrics and application requirements. The outputs of Abstraction Controller, link quality and load of network, are the inputs for the DecisionController. The information regarding the design of the FLS is very poor. The type of the membership functions, triangular, is specified and an example of rule from the rule base of DecisionController is given only. However, although the authors claim that their solution is for vehicular networks none of these networks' specific characteristics is taken into consideration.

Ma et al. [195] proposed a speed adaptive HO algorithm for vehicular heterogeneous networks based on FL. The HO decision is based on a FL decisional system that decides the best available network and whether the HO should be done. The input parameters of the system are a combination of network metrics and application requirements: bandwidth capacity, power charge, received signal strength (RSS) and delay. No VANET specific characteristics are directly considered in the inputs of FLS. These are incorporated in the speed adaptive strategy that takes into consideration the high mobility of the vehicles. This strategy is applied in order to identify the candidate networks that are then ranked by FLS. The membership functions chosen for the designed FLS were triangular and trapezoidal based on the fact that these are known for their good performance especially in real-time systems. Except for specifying the type of the membership functions, their parameters are not specified either, no other design decisions are revealed about the FLS. The efficiency of the solution proposed is demonstrated against a classic FL-based solution for heterogeneous networks that does not take into consideration VANET specific characteristics.

Dhar et al. [196] proposed an intelligent scheme based on FL aiming to ensure the always best connectivity for intelligent transportation systems. In the best network selection, the FL-based scheme considers multiple criteria that subscribe to all the aforementioned categories: network metrics, device related, user preferences and application requirements. In addition, another criterion considered relates to VANETs: the speed of vehicles. The network selection is triggered by received signal strength (RSS) and application required bandwidth. Further, network traffic load, affordable cost and vehicle speed are inputted in a FLS that gives as output the network rank. Based on this rank the best network is selected. The FLS has its triangular membership functions fully described in the paper. However the selection of the parameters is not motivated. The rule base is built based on the analysis done in order to prioritize the influence of the three inputs considered in ranking the network. This analysis is performed via Analytic Hierarchy Process (AHP). AHP was developed by Saaty [197] and its aim is to determine and prioritize the relevance of each criterion that has influence in a multi-criteria decisional process. The authors do not provide any details about the inference type implemented in FLS or about the defuzzification process.

3.2.4.3. Discussion

The FL-based HO solutions for vehicular networks, except for the solution proposed in [196] that combines FL with AHP for building the rule base, are quite basic compared to the complexity of the solutions proposed in general in the wireless heterogeneous networks space. They simply apply FL without any other techniques. Consequently, there is a lot of space for improvement in this research direction. However, even if the solutions analyzed are not that complex, the results obtained demonstrate that FL can be successfully employed in solving the network selection problem in the context of vehicular networks. It can be concluded that this is another area in VANETs where FL is proven to have applicability, area that joins to the large diversity of areas and domains where FL is used in decision making.

Regarding the design of FLSs, we emphasize again the suitability of using triangular and trapezoidal functions when developing FLSs in the context of HO in particular and VANETs in general. The description of FL decisional systems lacks in details and therefore best design practices cannot be underlined.

3.2.5. FL Solutions in VANETs – Classification and Lessons Learnt

This section provides a general classification of the FL-based solutions in VANETs, classification that is not bounded anymore to the context FL is applied in (whether is routing or handover, etc.). The classification is based upon the analysis performed in this chapter on the FL-based solutions in VANETs, and also on other FL-based solutions existent in the literature that are debated in this section. At the end, lessons learnt based on all the analysed solutions are presented.

3.2.5.1. Classification

FL applications in VANETs can be classified in:

- **decision making systems**
- **control systems**
- **prediction and detection systems**

The most popular class of solutions is represented by the **FL-based decision making solutions**. Thus, FL is applied to decide upon the best route choice, or best network to do the handover to, decide the similarity between the data or decide if the data is trustful in the context of data aggregation.

The next most popular FL-based solutions in VANETs are the **control systems**. In the solutions previously analysed, FL control systems or FL controllers were designed for controlling the access to the medium in the context of VANETs MAC protocols. Another example of FL in control process is provided by Ghafoor et al. in [198], where a FLS is used to control the beacon rate in the vehicular network, depending on the traffic conditions: in dense traffic conditions the beacon rate is required to be low, in sparse traffic conditions the beacon rate is required to be higher to increase the cooperative awareness. The inputs of the FLS designed to control the amount of beacons are the percentage of the same directional vehicles and the vehicle emergency status. The first parameter is chosen based on the traffic flow theory of Kerner [199] that states this parameter is an indicator of traffic density, while the latter is imposed by the fact that an emergency vehicle has to continue sending its status in the network. Thus the amount of the beacons is dependent not only on the traffic density, but also on the emergency status of the vehicle. This is one of the considerations that together with the expert knowledge is at the basis of the FLS rule base. The output of the FLS is represented by the beacon rate. Both, the inputs and the output have triangular membership functions. The inference type of the FLS is Mamdani, while the defuzzification method is COA.

In [200], Milanés et al. proposed a FL-based crossroad-traversing system for autonomous cars that aims to improve the traffic flow. The FL controller is used to control the speed of the car without right of way in accordance to the speed of the car with right of way. The input information of the FLS is based on the information provided from the vehicular network: the speed and positions of other vehicles. The FLS has a Mamdani inference type and is a MIMO type system, having 3 inputs and 2 outputs the throttle (T) and break (B). The membership functions of the inputs are trapezoidal and triangular, while the outputs have singleton membership functions. The decisions of the FLS design are not motivated. Defuzzification process is based on COA method.

In [201], Ghafoor et al. proposed a FL redundancy controller for controlling the amount of redundant packets depending on the traffic density and SNR of the channel. These latter parameters are the inputs of the FLS, while the output is called the coding density: the ratio between the encoded packets and the whole amount of packets received. Triangular and trapezoidal functions are chosen for inputs and outputs considering their computational efficiency and their large scale success in real-time systems. A Mamdani inference type is used in the FLS and a COA method for defuzzification. The option for FL employment in controlling the amount of redundant packets in order to improve the network load is motivated by the authors based on the capacity of FL of dealing with the uncertainty and imprecision, characteristics of the ever-changing VANET environment. Moreover, the authors emphasize on the advantages of the FLSs/FL controllers that are the modifiability: it is easy to tune rules, membership functions or even changing the parameters of the system in order to enhance the performance.

The **prediction and detection solutions** presented so far are used to predict acceleration and speed on one hand and to detect the network congestion on the other hand. In addition to these, another example of FL-based detection solution is described in [69] and [70]: Bauza et al. designed a FL-based system for road traffic congestion detection starting from the premise that FL is a powerful tool to address complex nondeterministic problems such as the identification of the traffic congestion. The FLS developed to determine the level of congestion is designed following some rules of congestion developed by Skycomp [77]. These rules express in a linguistic manner the level of congestion based on density of the vehicles and their speed. Thus, FL appears as the natural tool in solving this problem. A FLS is designed to be deployed on each vehicle for detecting the level of congestion around. The inputs of the FLS are the speed of the vehicle and the density that is determined based on the number of neighboring vehicles detected through V2V communications. The membership functions of the inputs are triangular and trapezoidal. The output of the FLS is the level of congestion and its membership function is a singleton. Although not specified, from the description and the results detailed, it is clear that the FLS has a Sugeno inference type.

TABLE 3.1 presents a summary of all the analysed solutions with the emphasis on the class the solutions subscribe to, problem FL solves and design decisions regarding the FLS.

3.2.5.2. Lessons Learnt

Employed in a variety of VANET solutions, FL has been demonstrated as a powerful tool that is able to solve a variety of issues in VANETs. The following lessons can be learnt from the study performed on FL solutions in VANETs:

- FL is a powerful mathematical tool for dealing with imprecision and uncertainty of VANETs dynamic environment. FL is also able to deal with multiple parameters that are necessary in order to describe the complexity of VANET environments.

- The two previous considerations plus the fact that FL is a powerful decisional tool results in the suitability of FL in being employed in making complex decisions in the context of VANETs. Therefore, it is not surprising that the most popular class of solutions is represented by the FL-based decision making solutions in VANETs.

- FL is a powerful technique for controlling processes that are difficult to model and this is exactly the case in the VANET environment.

- A FLS has a design that allows for flexibility and generality at a conceptual, structural and architectural level when this characteristic is needed as it is the case in VANETs data aggregation schemes.

- FLS are suitable to be applied in VANETs as they have predefined automatic tuning techniques for adjusting membership functions and rules, thus for adjusting the entire system to the ever-changing network environment. In a VANET context, real-time tuning (i.e. reinforcement learning-based tuning) appears to be more suitable than off-line tuning (supervised learning-based tuning); except for the cases where VANET solution is designed for certain network architecture. Otherwise, in order to adapt to the diversity of conditions imposed by VANETs, it is most appropriate that FLS to be real-time adaptive. However, there are not enough studies related to this aspect that allows us to say with certainty that a reinforcement learning-based tuning is better than a supervised learning-based tuning. It is still not clear how much the learning time affects the performance of the FLS in such a dynamic environment. Regarding this aspect, the solutions proposed so far impose the $H\infty$ filtering technique in learning as it appears to be faster. In this context, a better option is to choose a simple real-time adaptive FLS architecture as this eliminates the complexity of learning algorithms and the time needed for learning, but it might still provide a good adaptation.

- Regarding the other design decisions related to FLS, Mamdani is the most popular inference type, together with its specific COA method for defuzzification. The complexity of this is compensated by opting for singleton, triangular or trapezoidal membership functions that prove to provide good performance in this context and due to their reduced computational complexity and efficiency they are perfect candidates when choosing to design any FLS used in VANETs.

TABLE 3.1. FL-BASED VANET SOLUTIONS – CLASSIFICATION

Class	Motivation for FL	Objective	FLS design decisions					Ref
			μ functions	**Rule base building method**	**Inference Type**	**Defuzz.**	**FLS Arch.**	
FL-based Decision Making Solutions	FL is known as a perfect mathematical framework for decision making based on multiple parameters and for dealing with imprecision and non-deterministic problems.	**Route/path selection** in a unicast topology-based routing protocol	Trapezoidal functions; Parameters' selection method not specified	Manual tuning	-	-	Classic FLS	[149]
		Route/path selection in a unicast topology-based routing protocol	Triangular functions; Automatic tuning – H-infinity	Expert knowledge	Tsukamoto	Tsukamoto-specific	Reinforcement learning-based real-time adaptive FLS	[150]
		Route/path selection in a unicast geographic-based routing protocol	Triangular functions; Automatic tuning – H-infinity	Expert knowledge	Mamdani	COA	Reinforcement learning-based real-time adaptive FLS	[151]
		Route/path selection in a unicast geographic-based routing protocol	Triangular functions; Manual tuning	Expert knowledge	Mamdani	COA	Classic FLS	[152]
		Route/path selection in a unicast geographic-based routing protocol	Triangular functions; Manual tuning	Expert knowledge	-	COA	Classic FLS	[153]
		Relay nodes selection for storing the messages to be broadcasted in a broadcast protocol	Triangular and trapezoidal functions; Expert knowledge	-	Mamdani	COA	Classic FLS	[154]
		Relay nodes selection for storing the messages to be disseminated in an infotainment dissemination scheme	Triangular functions; Parameters' selection method not specified	-	Tsukamoto	Tsukamoto-specific	Classic FLS	[232]
		Route/path selection in a unicast topology-based routing protocol	Triangular and trapezoidal functions; Expert knowledge	Expert knowledge	Mamdani	COA	Classic FLS	[155]
	Above reason + automatic tuning is possible and brings an advantage to FLS in VANETs – adaptability to any kind of conditions (e.g. sparse or dense network)	**Route/path selection** in a unicast topology-based routing protocol	Triangular and trapezoidal functions; Expert knowledge	Expert knowledge	Mamdani	COA	Classic FLS	[156]
	FL is able to take decisions based on multiple criteria A FL-based design allows for flexibility, extensibility and generalization	**Deciding upon the data similarity** in data aggregation	Triangular functions; Parameters' selection method not specified	-	Sugeno	Sugeno-specific	Classic FLS	[157] [178]
		Selection of most relevant data from the aggregates to be further disseminated	-	-	-	-		
		Deciding upon the data similarity in data aggregation	Trapezoidal functions; Parameters' selection method not specified	-	-	-	Classic FLS	
		Deciding upon the data trustfulness in	Trapezoidal functions;	-	-	-	Classic FLS	[71], [184]

		data aggregation	Parameters' selection method not specified					
	FL is an excellent math framework for complex decision making based on multiple parameters that describe the environment with a degree of imprecision	HO Decision: **network selection**	-	-	-	-		[185]
		HO Decision: **network selection**	-	-	-	-	Classic FLS	[193]
		HO Decision: **network selection and decide if the HO should be performed**	Triangular and trapezoidal functions; Parameters' selection method not specified	-	-	-	Classic FLS	[195]
		HO Decision: **network selection**	Triangular functions	Built upon an analysis performed with AHP	-	-	Classic FLS	[194]
FL-based Control Solutions	The capability to qualitatively capture the attributes of a control system based on observable phenomena is a main feature of FL control. The capacity of modelling imprecision that characterize the network conditions influence on backoff and contention window	**Controlling the backoff interval** in a MAC protocol	Triangular functions Parameters' selection method not specified	Expert knowledge	Mamdani	COA	Simple real-time adaptive FLS	[196]
		Controlling the backoff interval in a MAC protocol	Triangular functions Parameters' selection method not specified	Expert knowledge	Mamdani	COA	Simple real-time adaptive FLS	[163]
		Controlling the backoff interval in a MAC protocol	Gaussian functions; Lessons learnt from the literature	Expert knowledge	-	-		[165]
		Controlling the contention window in a MAC protocol	Triangular functions; Expert knowledge	Expert knowledge	-	-	Simple real-time adaptive FLS	[166]
		Controlling the contention window in a MAC protocol	Triangular functions; Expert knowledge	Expert knowledge	-	-	Simple real-time adaptive FLS	[169]
	FL is applied to control the beacon rate based on intelligently combined metrics and to deal with the uncertainty that characterize the relationship between these parameters and the value of beacon rate	**Controlling the amount of beacons in the VANET**	Triangular functions	Lessons learnt from the literature and expert knowledge	Mamdani	COA	Simple real-time adaptive FLS	[170]
	FL is a powerful technique for controlling processes that are difficult to model and linearize	**Controlling the speed of the car without right of way** in accordance to the speed of the car with right of way in a crossroad traversing system	Singleton, triangular and trapezoidal functions; Parameters' selection method not specified	-	Mamdani	COA	Classic FLS	[198]
	1) FL is able to deal with the uncertainty and imprecision, characteristics of the ever-changing VANET environment. 2) Modifiability and adaptability of FLSs controllers: it is theoretically easy to	**Controlling the amount of redundant packets in the VANET**	Triangular and trapezoidal functions Parameters' selection method not specified	-	Mamdani	COA	Classic FLS	[200]

	tune rules, membership functions or even changing the parameters of the system in order to enhance the performance.							
FL-based Prediction and Detection Solutions	FLS with real-time adaptive architecture able to deal with the volatile characteristics of VANETs	**Prediction of speed and detection of network congestion** in a pro-active recovery from failure mechanism for a unicast topology-based routing protocol	Trapezoidal functions; Automatic tuning – PSO techniques	-	Tsukamoto	Tsukamoto-specific	Reinforcement learning-based real-time adaptive FLS	[201]
	1) FL is capable of dealing with the uncertainty (driver's behavior in this case) 2)FLS is adaptable to external changes when combined with learning techniques	**Acceleration prediction**	Triangular functions; automatic tuning using a basic reinforcement learning algorithm	Expert knowledge	Mamdani	COA	Reinforcement learning-based real-time adaptive FLS	[150]
	FL is a powerful tool to address complex nondeterministic problems such as the identification of the traffic congestion	**Road traffic congestion detection**	Singleton, triangular and trapezoidal functions; Lessons learnt from the literature	Lessons learnt from the literature	Sugeno	Sugeno-specific	Classic FLS	[171]

3.3. Clustering Algorithms in VANETs

Initial approaches of clustering in VANETs used clustering algorithms designed for MANETs. Lowest Id [217] is a state-of-the-art clustering algorithm in ad-hoc networks and was borrowed in VANETs from MANETs. Its principle is very simple. The nodes have assigned a unique fixed id which is broadcasted periodically in the network. The clusters are formed around the node with the lowest id among them, which is chosen as CH. Although the principle is very simple, Lowest Id is a very efficient algorithm, more efficient than other clustering schemes, such as Highest-Degree [88], that take into consideration more factors [112]. Highest-Degree is another state-of-the-art clustering algorithm in the area of ad-hoc networks. Its principle is similar to the Lowest Id algorithm, but the clusters are formed around the node with the highest number of neighbors. These two algorithms, as state-of-the-art algorithms in the area of ad-hoc networks, are very often used in the comparison-based assessment of the VANETs clustering algorithms and served as source of inspiration for many VANETs clustering approaches.

As emphasized before, although VANETs represent an instantiation of MANETs, they have unique features that need to be considered in order to design appropriate clustering algorithms for vehicular networks. On one hand some of the VANET characteristics need to be overcome by the clustering schemes, such as their rapidly changing topology, high mobility and scalability, while on the other hand clustering schemes can make use of other characteristics such as predictable mobility due to the road topology, traffic regulations and driver's behavior. Researchers acknowledged these facts and VANET-dedicated clustering solutions have been proposed.

As discussed in Chapter 2, clustering algorithms were implemented in the design of a large variety of VANET solutions: MAC protocols, routing protocols, data aggregation, security protocols, inter-vehicle communication, and data and infotainment dissemination solutions and architectures. In addition, a considerable number of generic clustering algorithms were defined for VANETs. Independent of the type of VANET solution the clustering algorithm is designed for, one of the main purpose of clustering is to achieve network stability. Therefore, the clustering metrics are focusing mainly on this aspect and they relate to VANET's dynamic environment. Thus, independently of the context in which clustering is applied (e.g. MAC protocols, routing protocols, etc), clustering metrics focus, in the majority of cases, on the same issues and they are similar to each other. They are only dependent on the ingeniously modeling of the VANET environment and they are different from solution to solution as researchers are experimenting in trying to find the best clustering metrics to express the dynamicity of the VANETs. Similarly, in clustering performance assessment, usually first the network stability achieved is measured and then, the overall assessment of the clustering solution is performed (the overall solution where clustering is integrated; e.g. MAC protocol, data aggregation, etc). All these considerations allows for a uniform analysis of clustering algorithms in VANETs, independent of the type of solution/application in which they are integrated.

After an overview of VANET clustering solutions in the literature, a very broad classification is provided here and several approaches are presented for each class for exemplification. The classification is made based on the cluster formation criterion: is the cluster formation dependent on some fixed structures such as road segments, grids, etc, or is it independent on any kind of structure and it is just following the traffic flow, vehicle movement? In the first case, vehicles from the same structure (road segment, grid, etc.) are grouped into a cluster. Thus static clusters are created bounded by this structure. Therefore, we called this type of VANET clustering algorithm under the generic name of static clustering algorithms. In the second case, cluster formation does not depend on any type of structures. Clusters are created by following the movement of the vehicles: vehicles with similar mobility patterns such as neighboring vehicles are grouped into clusters through exchange of clustering messages. In this type of approaches there is usually a beaconing message (a periodically broadcasted message in the network) sent either by the unclustered vehicle, either by a CH or a node with extra-responsibilities in the cluster. In the absence of predefined structures, this is necessary in order to announce the availability of joining the cluster or the availability of a cluster in zone so that a vehicle can join a cluster. The clusters created following this approach are mobile clusters, following the mobility of the vehicles and therefore we name this class of VANET clustering algorithms, mobile clustering algorithms.

3.3.1. Static Clustering Algorithms

Cherif *et al.* [115] propose a CH-based clustering algorithm where the cluster formation is depended on fixed road segments. The communication area where vehicles can be reached by RSU via multi-hop communication is called extended communication area. This area is split into fixed length segments, vehicles located in the same segment forming a cluster. Beside CH and simple cluster member, nodes can have another status inside a cluster, called super-member. This is a node that has been a CH and is yielding the job to another node. Inside the cluster, a main area of interest is conceptually partitioned in the centre of the segment. This area is called central zone and has the radius equal to the transmission range. Central zone has an important role in the distributed election of the CH. Initially, each member in the cluster estimates the time period it is going to spend in the central zone. The main principle behind CH election algorithm is to choose as CH the vehicle with the highest probability to spend the longest duration in the central zone. The speed and the position of the vehicle are also taken into consideration. All these parameters are used in the computation of each vehicle's electing factor, based on which the CH is selected. After that, each vehicle periodically examines its status and, by using the laws of uniform motion from Physics ($distance = speed\ x\ time$) predicts its future position in the immediate next moment of time. If a CH determines that it will be leaving the central zone in this moment of time, it will resign as CH, and a new CH is elected following the same procedure.

The proposed algorithm takes into consideration the high mobility of VANET nodes and movement predictability. Algorithm assessment is performed both via general topology-based metrics –$\overline{CL}$– and network metrics – *overhead*, *end to end delay* and *delivery ratio*. These are evaluated in relation to network density, but it is to be mentioned as a limitation the fact that the solution is not compared against any other clustering scheme.

Luo *et al.* [95] propose a CH-based clustering algorithm where the cluster's formation is based on square grids. The geographical area is divided into a subset of square grids. All the vehicles pertaining to a grid form a cluster. The vehicle having the closest position to the centre of the grid is elected as CH. This clustering scheme is implemented in a cluster and position-based routing protocol dedicated to VANETs and claims to reduce the overhead and packet delivery delay. CHs are the main data forwarders, a packet is sent from CH to CH until it gets to the CH that governs in the cluster where the destination node is positioned. The performance assessment is not very thorough, the authors presenting just a small analysis where they make some observations about their algorithm in comparison with state-of-the-art routing algorithms. Moreover, the clustering scheme neither tries to address any of VANET challenging characteristics, nor takes advantage of any of VANET characteristics. Thus, the clustering scheme by itself is not VANET dedicated, but the routing protocol is taking advantage of the vehicle's knowledge about their own positioning via the GPS integrated in their OBU.

Ramakrishnan *et al*. [103] adopt a similar approach in their proposed CH-based clustering algorithm to the one previously discussed: cluster formation is based on road segments called clustering areas. However, these clustering areas are not assigned with a fixed length value. Their size varies depending on the average speed of the vehicles within them. If the average speed is small then the cluster size is smaller, otherwise bigger. However, it is not mathematically described what smaller or bigger means. If an RSU is inside a cluster, then this is elected as CH. Otherwise, CH election is based on a single metric that is the *velocity*. As the clusters are static, the vehicle with the lowest speed in the cluster is going to spend the more time inside the cluster. Thus this vehicle is elected as CH. However, although the $\overline{CHCR}$ is reduced, is not clear how the fact that the position of CH related to the other cluster members is not taken into consideration is affecting the communication between CH and cluster members. Performance assessment is done via topology metrics only, which are quite different than the ones typically used. Instead of measuring directly the rate of changes in CH or clusters, the times of creation of clusters or the time of electing CH is measured. However, these are not good measurements of the stability in clusters; instead these assess the initial performance of clustering.

Tung et al. [124] proposed a clustering algorithm designed in the context of an intersection collision avoidance service. This clustering algorithm is employed in the design of a novel VANET WLAN-cellular architecture. This architecture is based on a heterogeneous network: LTE and WiFi. The communication messages inside the cluster are done via WiFi and they are called beacons, while CHs only are using the LTE interface for communicating with the base stations. This algorithm bridges the two types of clustering algorithms as it uses both static and mobile approach. On one hand, the clustering is bounded by the so called service region, region that is placed in the nearby of the intersection, but on the other hand it follows the mobility of the vehicles taking into account their direction. The proposed clustering algorithm is very specific to the solution built within. However, it indicates an efficient modality of bringing LTE in the vehicular networking context, as at this moment it appears to be more likely that LTE cannot handle the multiple messages that can be generated in VANET, especially during rush hours and in the traffic collision related applications when a huge number of messages can be generated. This solution was preceded by the one proposed in [213] that has employed a clustering algorithm to design a LTE-WAVE network architecture dedicated to multimedia delivery. This latest work uses a similar principle as in [214] and [215], where a generic VANET UMTS-WAVE architecture based on clustering is designed, but instead of 3G brings 4G in the VANET context. The principle of these three works differ from Tung et al. solution [124] by delegating the responsibility of communicating to infrastructure (via 3G or 4G) to another node, a gateway node, while CH is the main forwarder of messages inside the cluster. Multiple metrics are involved in both selection procedures: CH and gateway node as both states are of great importance. Independently of the type of node that has the responsibility of communicating with the infrastructure, CH or gateway, there is a single node in each cluster that is accessing the cellular network interface. This leads to an optimized architecture that is also proven to be reliable even for applications that require rich content such as multimedia applications.

In [124], the procedure of selecting the CH is based on a single metric: the proximity to the base station. The algorithm is evaluated in the context of the overall solution using solution-dependent metrics. Although WiFi standard is chosen for the inter-vehicle communications, the authors suggest that this can be replaced with V2V communication (IEEE 802.11p). Such architecture is used in the clustering solution proposed in [127]: IEEE 802.11p – V2V communication – for intra-cluster communication and LTE for the communication between CH and base station. This algorithm is a general CH-based clustering algorithm for VANETs. The clustering metrics are not clearly stated, but the CH is selected following the same policy as above: minimum distance to the base station. The algorithm is evaluated in terms of both network-specific performance metrics and topology-based performance metrics, namely: $\overline{CHL}$, $\overline{CS}$ and $\overline{NoC}$.

3.3.2. Mobile Clustering Algorithms

Ucar et al. [128] proposed a mobile clustering algorithm in the context of a safety message dissemination solution. Similar to the last approaches discussed in the Static Clustering Algorithm section, this algorithm is used to create a hybrid architecture that is an LTE – IEEE 802.11p-based architecture. The clustering is based on the exchange of hello packets containing the clustering information. The vehicles are clustered in a multi-hop fashion around a cluster head that is elected based on the relative mobility metric – the average relative speed with respect to the neighbouring vehicles. The novelty as compared to other clustering schemes is the fact that it allows for fast connection with the neighbour that is already a cluster member or a cluster head rather than a multi-hop connection with the cluster head. Moreover, LTE is employed if IEEE 802.11p is not available or overloaded, this being a novelty as compared to the other hybrid architectures, where there is no offload into LTE in case IEEE 802.11p is overloaded. The algorithm is evaluated in terms of both network-specific performance metrics, such as overhead, delay and data packet delivery ratio that are also a measure of the overall solution, and topology-based performance metrics, namely: $\overline{CHL}$, $\overline{CML}$ and $\overline{CHCR}$. The evaluation is comparison-based against the clustering scheme proposed in [118] that was adapted to have an IEEE802.11p-LTE hybrid architecture and the previously mentioned solution [215] that originally has an IEEE802.11p-UMTS architecture, but in the evaluation it was modified to an IEEE802.11p-LTE hybrid architecture. The proposed algorithm outperforms these schemes in terms of solution/network specific metrics and the authors explain the success of their algorithm based on the fact that it is most successful in achieving a better clustering stability that was measured through the topology metrics.

Su *et al.* [89] proposed a CH-based clustering algorithm in the design of a dedicated VANET MAC protocol. The cluster formation is based on beaconing messages (an initial message periodically broadcast in the network either by a vehicle recently entered in the network, or by CHs) and other cluster messages among the same-direction neighbours. Thus in cluster formation the main criterion considered is the direction of the vehicles based on the assumption that vehicles flowing in the same direction have similar speeds and moving patterns that are regulated by the traffic rules. Another criterion considered in cluster formation is signal strength and its role is revealed in the next paragraph.

The possible states of a vehicle-node in this clustering algorithm are: CH, quasi-CH, cluster member and quasi-cluster member. Each vehicle is seen from the moment of entering on the road a potential CH, so it receives the quasi-CH state. If after a predefined period of time it does not receive any valid *invite-to-join* beaconing message from a CH, the vehicle elects himself as a CH, otherwise the vehicle joins the cluster and its state changes to cluster member. Note that valid *invite-to-join* message must have the signal strength greater than a predefined threshold. Thus the size of the cluster is determined by the signal strength threshold.

This algorithm is among the first mobile VANET clustering algorithms. Its principle is simple, the only clustering metrics considered are *direction* and *signal strength* and the CH election is very simple, no decision process based on multiple metrics is involved. However, it is the first that considered direction metric in clustering the vehicles. In addition, this is the first approach in the literature that thoroughly defined some of the most popular general topology metrics in VANETs: $\overline{CHL}$ and $\overline{CS}$. Also, they defined two relative topology metrics, previously discussed: $\overline{RSWC}$ and $\overline{RSCH}$. These metrics are used to illustrate the performances of the clustering algorithm, but no other clustering algorithm is used as reference. The focus of the authors is on testing the MAC protocol where the clustering solution has been integrated. Tests show that this MAC protocol outperforms the standard IEEE 802.11p.

Kuklinski *et al.* [113] proposed a mobile clustering algorithm where mobile clusters are formed by the neighbouring vehicles through beaconing and other message exchange. Multiple clustering metrics are considered in creating stable clusters such as: connectivity level that is actually measuring the density, link quality estimated by SNR, relative nodes position and the prediction of this position in the future (based on speed and position) and nodes reputation built upon the history of node connections. The prediction of vehicle positions aims on one hand to avoid situations like clustering the vehicles that are moving in different directions with high speed. On the other hand, it allows for clustering the vehicles that are moving in different directions but with a low speed (e.g. vehicles in traffic jam). This approach leads to a greater stability of the clusters. Moreover, in order to avoid a high rate of re-clusterings, a node is given three possible states, excepting the CH state: member, candidate and visitor. Vehicles must prove they are potentially stable members of the clusters before they can join. First, a vehicle is in the visitor state, then after a time threshold is given the candidate state and only after applying the other clustering metric (connectivity, future position, etc), its state is changed into a member. Candidate and visitor nodes do not have the same rights as members do. They are not provided with the services that are provided in the cluster and they only have the right to exchange clustering messages. CH election algorithm is not described, although in each cluster a vehicle is assigned with this role. In addition, it is not clear what the CH responsibilities are. The proposed solution is compared against the state-of-the-art algorithm, Highest Degree and proves better performances in terms of $\overline{CS}$ and $\overline{CHCR}$ topology metrics.

Almalag *et al.* [114] introduced a CH-based clustering algorithm designed mainly for urban scenarios that uses traffic flow in cluster formation. The authors focus on the CH election algorithm as it is a well-known fact that stable CHs conduct to stable clusters. This algorithm is based on multiple clustering parameters: *density*, *distance between vehicles*, *speed* and *lane of travelling*. This last parameter is a new parameter considered so far in the clustering schemes and the key novelty of the algorithm. The rationale behind considering this parameter is that CH should be selected from a lane that the majority of vehicles are travelling in. Each vehicle first determines its own lane. Then each lane, referred as traffic flow, is given a weight. It is not explained what is the rationale behind weights' assignment for each traffic flow. Then for each vehicle it is determined on one hand the number of vehicles it is connected to (density), the comparison of its speed compared to others within its range and the comparison of its distance from all other vehicles within its range and on the other hand all these parameters but within their own traffic flows. The first group of parameters are multiplied with the traffic flow weights and then added to the second group in order to obtain the CH level of each vehicle. The vehicle with the highest CHL is selected as CH.

The proposed algorithm is compared against other three algorithms: the well-known Lowest Id, Highest Degree and against what authors generic named the Utility Function algorithm for VANETs. The latter clustering approach was proposed by [112] having as models Lowest Id and Highest Degree and is probably the first clustering scheme proposed for VANETs. The focus in this scheme is fully on the CH election that is suggested to be chosen for VANETs as the vehicle having the speed closest to the average and the distance between vehicles closest to the average. Although the authors do not provide details about what closest to the average means, they state that simulation results show better performance of their approach compared to Lowest-Id and Highest Degree. In the performance assessment of the traffic flow based algorithm, the authors use their own understanding of what closest to the average means for both speed and distance parameters. This is the same understanding that they used for implementing their own algorithm with respect to speed and distance metrics. The traffic flow-based algorithm outperforms all three algorithms (i.e. Lowest Id, Highest Degree and Utility Function) in terms of the topology metric used, $\overline{CHCR}$.

In [109], the authors introduced another mobility-based clustering algorithm for VANETs with focus on the stability of the resulted clustered network. The novelty of the algorithm consists in employing affinity propagation [212], a clustering technique that is borrowed from the data clustering field. Same pattern for clustering formation is followed as in the other structure-free discussed algorithms: exchange of clustering messages between vehicles in 1-hop neighbourhood. Direction is the first parameter considered in clustering formation: the vehicles form clusters with their 1-hop same-direction neighbours. The focus is again on the CH election algorithm where the affinity propagation technique applies. This technique is based upon a similarity function that is tailored for VANETs. Thus it is based on the Euclidean distance between the position of the node and the positions of its same-direction neighbours and the Euclidean distance between the next position of the node and the next positions of its same-direction neighbours. The efficiency of the algorithm is demonstrated against the previously discussed clustering algorithm proposed in [89] applying the most popular topology-based metrics: $\overline{CHL}$, $\overline{CML}$, $\overline{NoC}$ and $\overline{CHCR}$. In [121] the efficiency of this algorithm is demonstrated against two more clustering algorithms [117] and [216].

Goonewardene et al. [94] proposed a mobile clustering algorithm based on exchange of clustering messages between 1-hop neighbours designed with a robust adaptability to mobility – RMAC (i.e. the robust mobility adaptive clustering). The algorithm is designed to support geographic routing, although no routing protocol is proposed. An unclusterd node first makes a list of its 1-hop neighbours that answer to its beaconing messages with a message containing their speed, location and direction of travelling. Based on these metrics, the list is then sorted so that the most appropriate neighbour of the unclustered node to be selected as its CH. The appropriateness is decided as follows. First the *position* parameter is considered. Based on this the *Euclidean distance* is computed between the node and its neighbours. If the distances are comparable, then the next parameters, *speed* and *location* are considered. Based on these two parameters the next locations of the node and its neighbours are computed. The first neighbour in the list, the most appropriate to become the CH, is the one closer in the current moment of time and in the next one. This is quite a new approach in the literature, as usually a CH is elected in the cluster based on some values (id, computed weight using different techniques) that applies globally. The clustering algorithm proposed here is node-oriented – node precedence algorithm – as each node elects its own CH. If the first node in its 1-hop neighbours list is already a CH then the unclustered node becomes a member of its CH cluster. Otherwise, the vehicle selects this node as its CH and e new cluster is formed. Thus, beside cluster member and CH, a node can be in a dual state that is when it is a CH of a cluster and a member of another cluster. This leads to overlapping neighbouring clusters and no message overhead in case of a cluster member transition to a neighbouring cluster. Another novel concept introduced by this algorithm is the zone of interest that enables each vehicle to keep an updated table of its neighbours that goes beyond their transmission range. Zone of interest' radius is established as two times their transmission range. Thus vehicles have prior knowledge about the network while they are travelling into the neighbourhood which is translated into an optimized and smoother process of re-clustering. The algorithm is compared against an algorithm proposed by Basagni [120] that is shortly called DMAC (Distributed and Mobility Adaptive Clustering). DMAC is a generalised clustering algorithm designed for MANETs where the CH election is done globally and is not node-oriented. Each vehicle has a weight associated. The clustering process begins with each node examining the weights of all nodes within its own transmission range. The node with the highest weight becomes the CH. This algorithm can be tailored for VANETS where the weight of a vehicle is calculated using metrics such as distance/speed/acceleration. RMAC outperforms DMAC in terms of $\overline{CML}$, and in terms of another topology metric: *node re-clustering time*.

Hafeez et al. [171] introduced a clustering algorithm in the context of a new MAC protocol. Vehicles are organized in clusters on the basis of the beaconing and clustering messages they exchange in their neighbourhood. The focus is again on the CH election as CH is assigned with the main organizing and communication roles inside its cluster. The vehicles can have 5 different states: lone (not clustered), member, temporal CH, backup CH and CH. Temporal CH and backup CH roles aim to provide on one hand a stable CH in the cluster, a temporal CH must prove that it is the most stable selection, and on the other hand to ensure a smoother CH re-election, backup CH is ready to take over the CH role. CH election algorithm is based on a weighted stability factor that is built upon the exponential-weighted moving average of the previous stability factors. Stability factor is computed for each vehicle and it is based on the relative movement between the neighbouring vehicles reflected in the average speed difference between the vehicle speed and its neighbours' speed. The novelty of this clustering scheme consists in the technique implemented in order to provide a smoother CH re-election and consequently to improve the cluster stability. Basically, this technique states how backup CH is taking over the CH role. The implementation of this technique is based on a FLS that aims to predict and learn driver's behaviour. Based on this the next position and speed of the vehicles are computed. If in this next moment of time if not all the members of its cluster are in its range anymore, but they are still in the range of the backup CH, then CH hands over its role to the backup CH.

The proposed clustering solution is assessed using a large variety of metrics from all the classes presented. Thus, the MAC protocol integrating the clustering solution is assessed as well. However as our interest relays in the clustering schemes we mention the topology-based metrics employed in assessment: $\overline{CHL}$, $\overline{CS}$, $\overline{CML}$. The proposed clustering algorithm is demonstrated to overcome the performance of another cluster algorithm designed for a VANET MAC protocol [89] that was previously discussed. Moreover, the MAC protocol based on the proposed clustering solution shows better performance than the protocol used as comparison.

Wahab et al. [122] is a very recently proposed two-hop CH-based clustering scheme for VANETs that is also incorporating computational intelligence. This is one of the proposed clustering algorithm novelties: the employment of Ant Colony Optimization in selecting the nodes having a state called multi point relay. These nodes are selected by CH for inter-cluster communication. The other novelty of this approach consist in building 5 new clustering metrics models with the focus on QoS. The most complex one combines bandwidth, connectivity and mobility metrics specific to VANETs (i.e. relative speed and distance). The output of the model is a QoS factor that is computed for each vehicle. Each of these models can be employed in further QoS-oriented clustering algorithms for VANETs, the authors describing each model's recommended scenario. This QoS factor is used to elect the most suitable CH and the multi point relays, while the clustering formation is done only on the basis of 2-hop neighbouring.

The clustering algorithm is designed in the context of a new routing protocol dedicated for VANETs that derives from a MANET routing protocol designed to improve QoS, QoS-OLSR that on its turn derives from the state-of-the-art routing protocol for MANETs, QLSR (Optimized Link State Routing). Thus the performance assessment aims to demonstrate on one hand the efficiency of the clustering algorithm in terms of network stability and on the other hand to prove the efficiency of this algorithm in the designed routing protocol. In the first case, the performance is tested using a solution-dependent topology metric defined by the authors that is generic entitled stability. The metric is highly dependent on the clustering parameters. The authors show the performance of the algorithm in terms of stability and network metrics (e.g. packet delivery ratio) in five different cases corresponding to the five different clustering metrics models. No comparison is done with other clustering schemes. The comparison-based assessment is done only when showing the performances of the overall solution. The cluster-based routing protocol outperforms QoS-OLSR and OLSR both.

3.3.3 Discussion – Future Directions

The start of a roadmap of clustering algorithms in the VANET research area can be roughly placed in 2005 when the first studies have been performed by employing Lowest Id and Highest Degree in VANETs scenarios and by suggesting new approaches that relate to these (e.g. [112]).

Actually, most of the VANET mobile clustering algorithms are using the basic principles that fundament the Lowest Id and Highest Degree algorithms. Back in 2005 some of the main clustering challenges in VANETs have been outlined as well: rapidly changing topology of VANETs, scalability, multiple services to be provided with different requirements [218]. Since then, clustering algorithms evolved, many approaches have been proposed to tackle especially the rapidly changing topology of VANETs. More and more mobility parameters have been considered in clustering: *direction*, *lanes*, *speed*, *position*, *predicted speed* and *position* and combined with other parameters such *bandwidth*, *connectivity* (or *density of the vehicles*) and *signal strength*. These parameters are mainly considered in selecting the nodes with extra-responsibilities in the cluster, especially CH. CH election algorithms are of high importance and some of the researchers are actually focusing on this aspect of clustering only. Usually CH has the main responsibilities in the cluster and therefore a stable CH is required. In addition, the stability of CH is highly influencing the stability of the cluster itself, as most of the times when CH is re-elected a cluster reconfiguration is required, too. Therefore, researchers employed all kinds of techniques to combine the clustering parameters and to decide which is the most suitable CH. The predominant techniques are utility functions and weight-based techniques, but recently more innovatory techniques such as Affinity Propagation have been employed [109], [121]. One of the newest trends in VANET clustering algorithms is the employment of computational intelligence such as FL [171] and Ant Colony Optimization [122] in some secondary roles in clustering such as predicting the future positions of the vehicles. FL decisional systems, known as very powerful decisional systems, could be further employed as main player in CH election algorithms. FL is the perfect mathematical framework for dealing with imprecise information such as the one used in clustering (it is impossible to define precisely how each of the clustering parameters influence the stability of CH) and with multiple parameters.

Another trend in VANET clustering is the employment of clustering algorithms in designing reliable and efficient VANET architectures that bring together multiple access technologies. The need to converge multiple types of technologies in VANET context in order to enable the diversity of vehicular applications is underlined more and more in the literature and is also enforced by the low penetration rate of the WAVE technology. Thus, design techniques of VANET architectures that bring together multiple access technologies are of high interest. First, VANET cluster-based 3G-WAVE architectures were envisioned and very recently, VANET cluster-based architectures using 4G together with other technologies (WLAN or WAVE) were proposed. Due to its superior suitability for VANET environment compared to 3G technologies, the interest in bringing LTE in vehicular networks is far higher and considerable research efforts are done in this direction [26]. Although, LTE has high data rates and supports high mobility, it does not support direct communication between vehicles and from the studies performed so far it appears that it also cannot support the high number of messages exchanged by vehicles in certain scenarios (e.g. rush hours, applications that request an exchange of many messages such as collision avoidance applications). In addition, statistics have shown that the huge growth in mobile data will be almost impossible to be accommodated in cellular networks [219]. Therefore, if brought in the context of VANETs, either 3G or 4G (LTE) technologies, these technologies need to be used in an optimum manner and they are mostly suitable for the communication between vehicles to infrastructure. In [128], LTE is used in CH-CM communication if IEEE 802.11p is not available or if it is overloaded. This is a novel idea of offloading.

The cluster-based solutions presented in this chapter demonstrated that clustering is suitable for designing multiple access technologies-based VANET architectures with the aforementioned recommendations. Two instances of this type of architecture can be outlined: one in which the communication with the infrastructure is done via CHs and the other one in which the communication with the infrastructure is done via gateways.

In conclusion, the future directions in this research area of clustering in VANETs can be summarized as follows:

- More mathematical frameworks incorporating computational intelligence should be experimented in trying to find the most appropriate method of combining the clustering parameters in order to obtain stable CHs and stable networks;
- A direction to be followed is the one that relates to clustering capabilities of conducting to a reliable and optimized design of VANET architecture based on multiple access technologies that is able to support the diversity of VANET applications. A comparison between the two outlined instances of VANET cluster and multiple access technologies-based architectures in diverse scenarios and application types would be of high interest. This research direction could lead to a generalized VANET cluster and multiple access technologies-based architecture or to guidelines in designing this type of architectures that are application-oriented, adapted to the type of application aiming to support.
- Performance assessment as underlined in Chapter 2 is a big issue in VANET clustering algorithms. The analysis of the VANET clustering algorithms conducted in this work, leads to the idea that there is a need of standardizing clustering performance metrics, especially the general topology metrics. In addition, there is a huge need of traffic and mobility models for performing the testing of VANET clustering algorithms.

3.4. Chapter Summary

This chapter presented and discussed related works in three main areas: green transportation solutions based on VANET, FL applications in VANET and clustering algorithms based on VANET. VANET-based green transportation solutions are classified in two main categories: eco-routing solutions and eco-driving solutions. Most of these solutions are dedicated to internal combustion engine-powered vehicles, but the current trend of designing such solutions for EVs a class of vehicles that would highly benefit from energy efficient solutions is also identified. In this class, electric bicycles are mentioned as having different characteristics when compared to electric cars. Although these are the most popular among EVs, they have been neglected so far in the context of VANET research.

FL solutions in VANETs are thoroughly analyzed with the emphasis on how FL is employed in solving the issues in a large variety of VANET applications, but also on the design of FLS for each of the solutions. Although the analysis of the solutions is made in their specific context, a general classification of FL applications in VANET is outlined: decision making systems, control systems, respectively prediction and detection systems. The most popular class of solutions is represented by the FL-based decision making solutions. Thus, FL is applied to decide upon the best route choice, or best network to do the handover to, decide the similarity between the data or decide if the data is trustful in the context of data aggregation. Lessons learnt are presented among which the fact that FL is a powerful mathematical tool for dealing with imprecision and uncertainty of VANETs dynamic environment. FL is also able to deal with multiple parameters that are necessary in order to describe the complexity of VANETs environment. The two previous considerations plus the fact that FL is a powerful decisional tool results in the suitability of FL in being employed in making complex decisions in the context of VANETs. Therefore, it is not surprisingly that the most popular class of solutions is represented by the FL-based decision making solutions in VANETs. A summary of all these solutions and their distribution in the identified classes is presented in TABLE 3.1.

At the end of the chapter, VANET-based clustering algorithms are analysed. Clustering is employed in a large plethora of solutions in VANET having many applications. From this point of view there are two main classes: generic clustering solutions, but also application-based or solution-integrated clustering algorithms. Independent of the type of VANET solution the clustering algorithm is designed for, one of the main purpose of clustering is to achieve network stability. Therefore, the clustering metrics are focusing mainly on this aspect and they relate to the VANET dynamic environments. This allows for a uniform analysis of clustering algorithms in VANETs, independent of the type of solution/application in which they are integrated. The analysis is focused on the cluster formation techniques, but also on the performance assessment as there are no standardization or general guidelines in the literature to cover this aspect. A broad criterion in the cluster formation was used to classify these algorithms in two main classes: static clustering algorithms, respectively mobile clustering algorithms.

Chapter 4

PROPOSED ARCHITECTURE AND ALGORITHMS

*This chapter presents the architecture and algorithms of the proposed Intelligent Advisory Solution for Bicycle Eco-riding and Eco-routing over vehicular networks. This solution encompasses the major contributions of this thesis**: a Speed Advisory System for Electric Bicycles (SAECy)** that consists of a Green Optimal Speed Advisory solution for Electric Bicycles and a FL-based Weather-Aware Speed Adaptation Policy for Electric Bicycles, an **Energy Efficient Weather-aware Route Planner solution for Electric Bicycles (eWARPE)** that provides off-route assistance to the cyclists and **a FL-based Clustering Scheme over VANETs (FuzzC-VANET)**. SAECy exploits the traffic light to vehicle (I2V) communication aiming to provide on-route assistance to the cyclists in order to both reduce the energy consumption of the bicycles and to improve their cycling experience. FuzzC-VANET can be employed for information dissemination in the Speed Advisory System in order to enhance the performances of the system.*

4.1. Overall Architecture and Network Model

This thesis proposes an Intelligent Advisory Solution for Bicycle Eco-riding and Eco-routing which provides on-route or on-trip assistance and off-route or pre-trip assistance for bicycles in general and electric bicycles in particular. The aim of the solution proposed is to increase user/cyclist's riding experience and reduce the cyclist's effort in the general case of bicycles, respectively to decrease the energy consumption in the particular case of the electric bicycles.

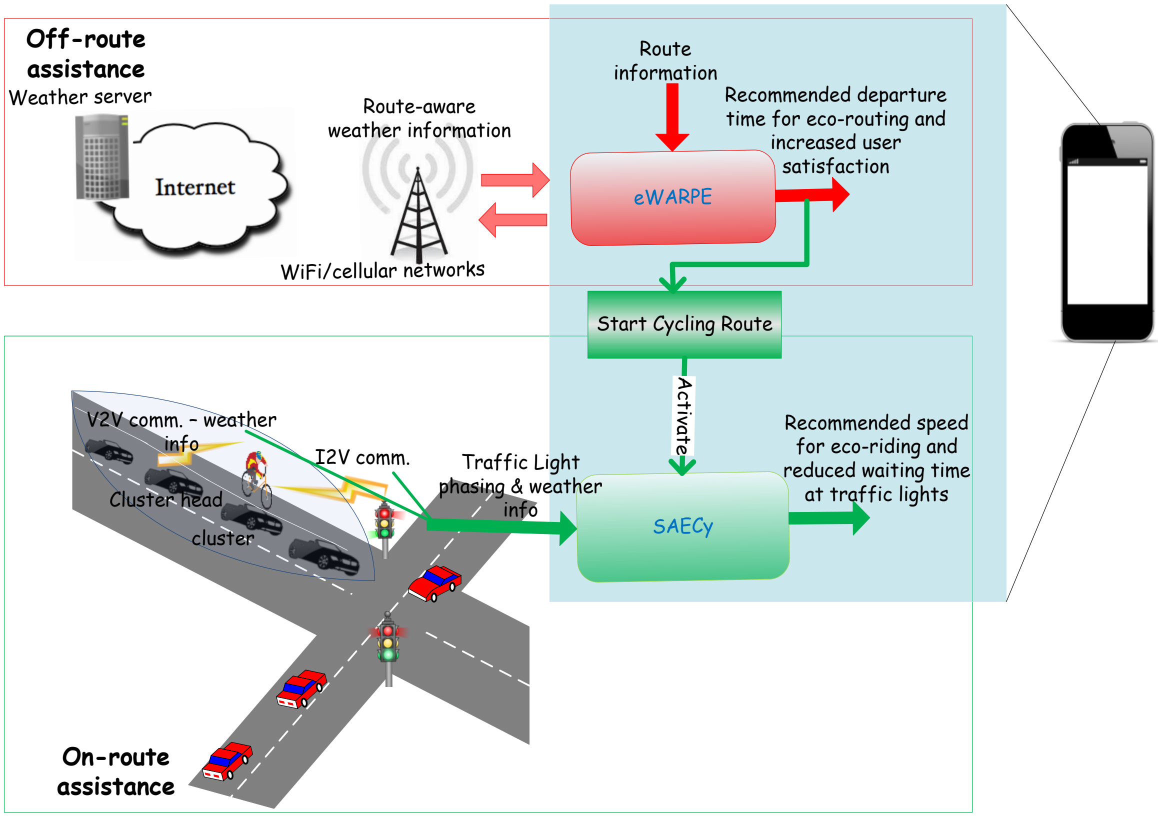


Figure 4.1. Overall Solution Architecture and Network Model

The on-route assistance is provided during the cycling through a **S**peed **A**dvisory System for **E**lectric Bi**Cy**cles (SAECy). SAECy exploits the traffic light to vehicle (I2V) communication aiming to reduce the energy consumption of the bicycles and to improve riders' cycling experience. Based on traffic light phasing obtained via I2V communications, the solution recommends strategic riding (i.e. the appropriate speed) when bicycles are approaching an intersection to avoid, if possible, stopping and starting due to red traffic light signals, which are high power consumption scenarios. The proposed approach includes a FL-based speed adaptation policy independent of the traffic light phases which is wind-aware, as among all the vehicles, bicycles are the most affected by the wind. This policy provides a better speed adaptation to the wind conditions leading to increased savings in energy. The wind information can be received from traffic light aggregated with the traffic light phasing information, if available, or via V2V communication from clustered vehicles. Clusters are used to optimize information dissemination and solve scalability and stability issues in VANETs. They are also known as being able to provide highly up-to-date and localized information of all types. Such a type is the weather information. The performances of SAECy can be enhanced by obtaining local-based wind information in a clustered network obtained through the proposed **Fuzz**y Logic-based **C**lustering scheme for **VANET** (FuzzC-VANET). FuzzC-VANET is a general clustering scheme that extends its benefits on all types of vehicles and that has as main objectives to solve some of the biggest challenges of VANETs scalability and stability issues.

The off-route assistance of the proposed solution is provided by a novel **e**nergy-efficient **W**eather–aw**AR**e route **P**lanner solution for **E**lectric bicycles (eWARPE). eWARPE makes use of weather information in order to recommend optimal departure time that allows a cyclist to avoid adverse weather conditions (such as wind, rain fall) and increase the energy savings of the electric bicycle. Note that the departure time is in a user-configurable time interval. Wind in particular has a great influence on the electric bicycle power consumption. Therefore reduced wind speed and a more favorable wind direction are preferable not only to improve the cyclist experience, but also to save the bicycle battery power. eWARPE represents a step forward for the cycling route planners, going beyond planning the route itself (how to get from point A to point B). eWARPE is planning the optimal departure time for the route: when to leave from point A to get to point B on the previously planned route in order to avoid the adverse weather conditions and to increase the energy savings. Moreover, to the best knowledge of the authors, this is the first route planner that is targeting electric bicycles. However it can be used by all cyclists in general, energy saving being translated into cyclist effort reduction.

The system architecture of the overall Intelligent Advisory Solution for Eco-riding and Eco-routing solution is presented in Figure 4.1. It can be seen from the figure that prior the cycling journey, eWARPE recommends a departure time based on the information received from the weather server and route information in order to help cyclist avoid as much as possible adverse weather conditions. Once the cycling journey starts, the on-route assistance is activated. SAECy is responsible with providing on-route assistance and FuzzC-VANET can be also involved in this. SAECy is communicating with the traffic lights in order to retrieve traffic light phasing information and weather information that is referring to the wind. Highly accurate wind information can be also retrieved from the clusters created based on FuzzC-VANET. Figure 4.1. shows how the cluster head from such a cluster communicates the weather information to the bicycle. As shown in the figure, this solution can be deployed on the cyclist smartphone that can be easily mounted on the bicycle handlebar. The smartphone is considered to be configured as a vehicle on board unit and has IEEE 801.11p Wireless Access Vehicular Environment (WAVE) support [137]. This configuration enables the smartphone to receive messages from the Road Side Units (RSU) associated to the traffic lights via IEEE 802.11p communication interface and messages from other vehicles.

Next, each of the aforementioned contributions, SAECy, eWARPE and FuzzC-VANET, are described in more details, with the focus on their architecture, functionality and algorithms.

4.2. Speed Advisory System for Electric Bicycles - SAECy

4.2.1. SAECy Overview

This section presents an overview of the proposed speed advisory system, depicted in Figure 4.2, its main inputs and its functionality. SAECy recommends the cyclist the appropriate speed when approaching a signalled intersection in order to avoid stopping at the traffic light. This is the main function of any GLOSA system designed for vehicles in general. In addition to this main function, SAECy includes a secondary function which increases the benefits in terms of energy efficiency. This second functionality of the system is providing an additional recommendation to the cyclist: the advised speed is communicated in terms of maximum speed, and the cyclist should not exceed the indicated speed limit in order to increase the energy savings. This secondary function is provided by a FL-based wind-aware speed adaptation policy, is weather dependent, and does not relate to the traffic light phasing. Its aim is to provide a better adaptation of the bicycle speed to the wind conditions with the purpose of reducing the energy consumption. Among all the vehicles, the bicycles are mostly affected by the wind.

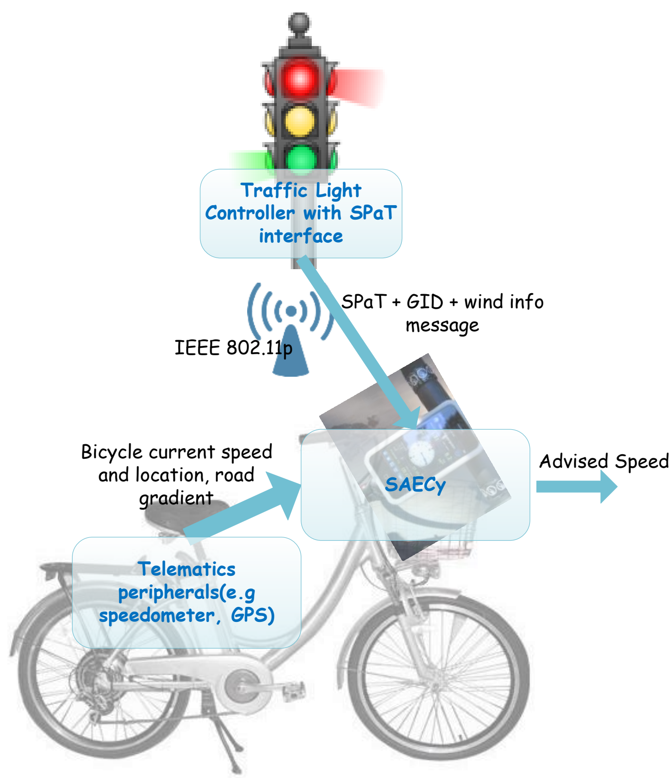


Figure 4.2. SAECy – Overview

Figure 4.2 presents an overview of SAECy that can be deployed on the cyclist's smartphone that can be easily mounted on the handlebar as shown in the figure. It is also shown the communication of SAECy with two other components that are providing its inputs: *Traffic Light Controller* and *Telematics Peripherals*. More details about this are next presented. The smartphone SAECy's deployed on is considered to be configured as a vehicle on board unit and has IEEE 801.11p Wireless Access Vehicular Environment (WAVE) support being able to receive messages from the RSUs associated to the traffic lights via the IEEE 802.11p communication interface. These messages are generated by the Traffic Light Controller component that is associated to each traffic light. The Traffic Light Controller is considered to have a SPaT interface, thus able to generate and transmit the standardized SPaT and other associated messages such as GID messages.

Moreover in vehicular networking, the infrastructure, such as traffic lights, can disseminate updated and relevant weather information that can be easily obtained from local weather stations or through V2I communications [236]. In this approach the Traffic Light Controller is also in charge with providing wind information (i.e. wind speed and wind direction). Traffic Light Controller encapsulates all the information, SPaT, GID and wind information, into a single message. Alternatively, the wind information can be obtained via intra-cluster communications (V2V communications), the clusters being known in VANETs as able to provide highly up-to date and local-based information.

The main fields of interest from SPaT messages are the following ones:

- *timeToChange*, time until the current traffic light colour changes
- *signalState* , indicating the current traffic light phase

These message fields are stated for each lane and possible direction that can be taken at the intersection. The GID message provides the coordinates of the position of the intersection. Thus, the speed advisory system is receiving from Traffic Light Controller via I2V communications the necessary information related to the traffic light phasing and the position of the intersection.

From the Telematics Peripherals SAECy is receiving the coordinates of the current position of the bicycle, the current speed, direction and the road gradient. Telematics Peripherals can be a set of sensors/tools/etc. within an external device (e.g. speedometer, cycling computer) or a set of integrated/single application(s) on the smartphone (e.g. the integrated GPS).

4.2.2. SAECy Architecture

This section presents the architecture of the speed advisory. Figure 4.3 describes this architecture that is based on two main architectural components: Data Collector and Processor and Computation and Recommendation Module. Each of these architectural components and their interactions are detailed next.

4.2.2.1. Data Collector and Processor

This component collects local information that relates to the bicycle and network information received via I2V communications. The local information comprises the bicycle current speed and location. The network information is represented by a message that encapsulates SpaT, GID and wind information. The Data Collector and Processor component extracts the following information: *timeToChange*, *signalState*, intersection location coordinates and wind speed and direction (v_w, D_w). Based on the intersection location and the bicycle current location coordinates, the component computes the distance to the intersection, d. This parameter together with *timeToChange*, *signalState*, bicycle speed (v_i), direction (D_B), and v_w are fed as input to the Computation and Recommendation Module.

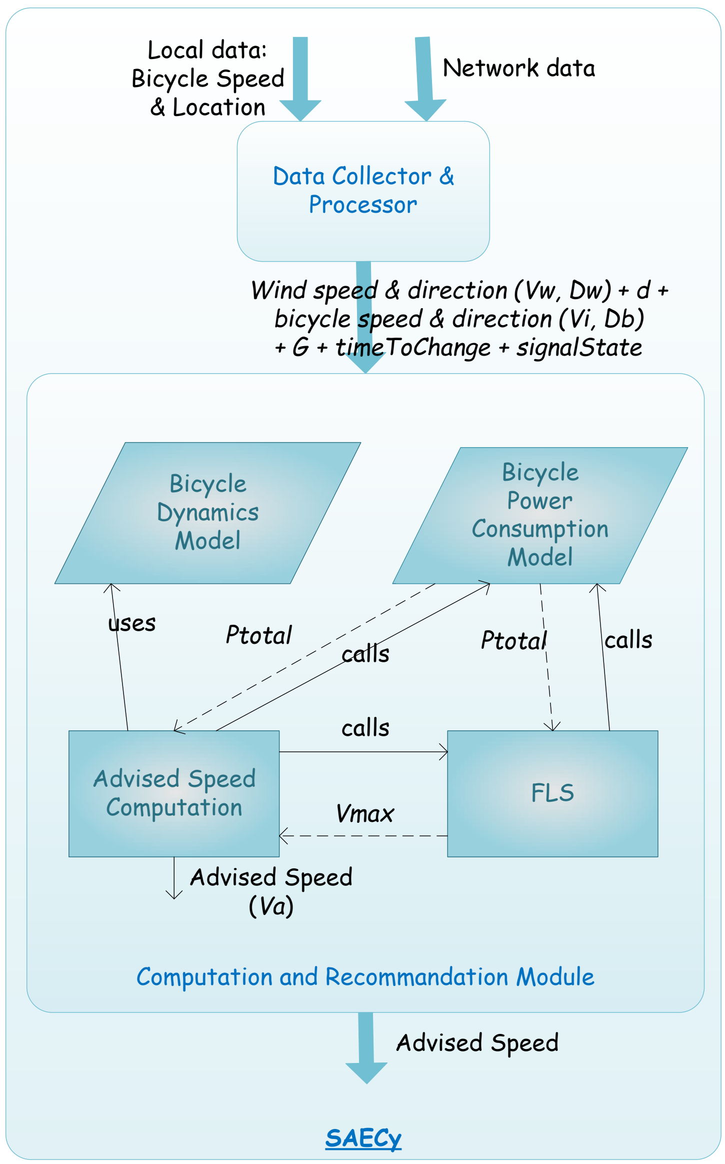


Figure 4.3. SAECy – Architecture

4.2.2.2. Computation and Recommendation Module

Computation and Recommendation Module is the core component of SAECy, its inputs being the aforementioned parameters fed by the Data Collector and Processor, while the output is the advised speed. Computation and Recommendation Module has four internal components: Advised Speed Computation, a Fuzzy Logic System (FLS), Bicycle Power Consumption Model and Bicycle Dynamics Model. The next sections present in detail these four components.

A) Bicycle Power Consumption Model

SAECy uses the power consumption model that was employed in computing the theoretical power consumption of an electric bicycle in [21]. According to this model the total power consumption (P_{total} – eq. (4.1)) is the sum of three terms: the power needed to overcome the air drag (P_{drag} – eq. (4.2)), the power needed to overcome the slope (P_{hill} – eq. (4.3)) and the power needed to overcome the surface resistance ($P_{friction}$ – eq. (4.4)). The notations employed in the equations are explained in TABLE 4.1.

$$P_{total} = P_{drag} + P_{hill} + P_{fricition} \tag{4.1}$$

$$P_{drag} = \left[0.5 \cdot C_d \cdot D \cdot A \cdot (v_g + v_w \cdot \cos(D_w - D_B))^2\right] \cdot v_g \tag{4.2}$$

$$P_{hill} = (g \cdot G \cdot m) \cdot v_g \tag{4.3}$$

$$P_{fricition} = (g \cdot m \cdot R_c) \cdot v_g \tag{4.4}$$

TABLE 4.1. POWER CONSUMPTION MODEL NOTATIONS

Notation	**Explanation**
C_d	Drag coefficient
D	Air density (kg/m^3)
A	Frontal Area (m^2)
v_g	Ground speed of the bicycle (m/s)
D_B	Bicycle direction (degrees)
v_w	Wind speed (m/s)
D_w	Wind direction (degrees)
g	Gravitational acceleration = 9.81 (m/s^2)
G	Slope grade – taken from the map data
m	The overall weight: cyclist + bicycle + additional equipment (kg)
R_c	Rolling coefficient

Most of the parameters used in the power consumption model have typical values in urban environments [21], therefore they can be preset in the system and allowed to be changed through an user interface if necessary. The same user interface can be used to set the *m* value which is dependent on the cyclist and that can be changed when the bicycle is used by another user. The variable parameters are v_g, D_B, D_w, v_w and G. v_g is provided by the components triggering the functionality of this model, while the rest of the parameters are provided by the Data Collector and Processor component.

B) Bicycle Dynamics Model

Equations of motion eq. (4.5) and eq. (4.6) – for uniformly accelerated/decelerated motion –, and eq. (4.7) – for constant motion – were used to model the bicycle dynamics [237] as follows:

$$v_a = v_i \pm a \cdot t_{ia} \tag{4.5}$$

$$v_a^2 = v_i^2 \pm 2 \cdot a \cdot (d - x_c) \tag{4.6}$$

$$x_c = t_c \cdot v_a \tag{4.7}$$

$$t_c + t_{ia} = t \tag{4.8}$$

The equations were adapted to our solution (see TABLE 4.2 for more detailed explanations) and further computations were made. From eq. (4.5), eq. (4.6), eq. (4.7) and eq. (4.8) the value of v_a is deducted in eq. (4.9).

$$v_a = \begin{cases} \frac{2 \cdot d}{t} - v_i \text{ , } if\, t_c = 0 \\ v_i + a \cdot t - \sqrt{|a \cdot (a \cdot t^2 + 2 \cdot t \cdot v_i - 2 \cdot d)|}, if\, t_c \neq 0 \\ \text{and it is accelerated motion} \\ v_i - a \cdot t + \sqrt{|a \cdot (a \cdot t^2 - 2 \cdot t \cdot v_i + 2 \cdot d)|}, if\, t_c \neq 0 \\ \text{and it is deccelerated motion} \end{cases} \tag{4.9}$$

TABLE 4.2. BICYCLE DYNAMICS MODEL NOTATIONS

Notation	Explanation
d	The distance to the intersection
t	The time in which distance d is to be travelled
v_i	Initial speed of the bicycle (current speed of the bicycle before advised speed is recommended)
v_a	The advised speed
t_{ia}	The time passed till the moment the bicycle speed v_i becomes v_a
x_c	The distance the bicycle is assumed to be travelling at a constant speed after reaching the recommended speed necessary to cross the intersection without stopping
t_c	The time distance x_c is travelled
a	Acceleration or deceleration required so that v_i becomes v_a

C) FLS and Advised Speed Computation components

FLS is a FL system that implements the FL-based wind-aware speed adaptation policy. Its functionality is triggered by the Advised Speed Computation component. The FLS has a single input and a single output and makes use of the Bicycle Power Consumption Model. The design of the FLS is focused on reduced computation complexity following design principles from [61]. It follows a zero-order Sugeno model known for its efficiency, reduced complexity and suitability for real-time systems [73]. The structure of the system is classic for a FLS. The full description of this structure is presented in section 4.2.3.2., as it is intrinsically connected to its functionality which is described in that section.

Advised Speed Computation component is the core of the Computation and Recommendation module. It has as inputs all the inputs of the Computation and Recommendation module, while the output is the advised speed (v_a). The component makes use of the other components of the module as it can be seen from Figure 4.3 and it implements the main logic behind the computation of the advised speed, namely the SAECy algorithms that are described in the next section.

4.2.3. SAECy Algorithms

The algorithm employed in the computation of the advised speed is presented in two stages. As described in SAECy overview, SAECy's complete functionality is provided by: a Green Optimal Speed Advisory function, the main function of any GLOSA system designed for vehicles in general, and a FL-based wind-aware speed adaptation policy. In the first stage, the algorithm describes only the Green Optimal Speed Advisory function (Algorithm 4.1) that can work standalone. In the second stage, the FL-based wind-aware speed adaptation policy is added to this function and included in the description of the algorithm (Algorithm 4.3). The latter represents the complete SAECy algorithm and encompasses all the functionality of the proposed speed advisory system. This latter algorithm uses Algorithm 4.2 for computing the membership functions employed in the context of FL-based wind-aware speed adaptation policy.

4.2.3.1. Green Optimal Speed Advisory Function

The computation of the advised speed is described in details in Algorithm 4.1. It includes an initialization phase that assigns the initial values to some of the parameters and the main procedure.

After the initialization phase, as long as the cyclist has not crossed the intersection yet, the speed of the bicycle is monitored continuously. Action is taken in two situations. First, if *signalState = "red" && timeToChange > t* , a new, decreased speed is recommended. Second, if *signalState = "green" && timeToChange < t && timeToChange* $\geq d/v_{max}$, a new, increased speed is recommended. The parameters *t* and *d* are explained in TABLE 4.2., while the parameter v_{max} is explained in the next paragraph. The advised speed is computed based on the Bicycle Dynamics Model (eq. (4.9)).

Algorithm 4.1: Green Optimal Speed Advisory

INITIALIZATION PHASE:

BEGIN

maxSpeed = 6.95 //*the maximum speed set for bicycle*

v_{max} = *maxSpeed*

$v_w = 0$

END

GREEN OPTIMAL SPEED ADVISORY PROCEDURE:

-triggered when a message is received from a traffic light and then triggered in the monitoring cycle while the bicycle has not crossed the intersection yet

BEGIN

update v_w

compute v_{max} for P_{total} = *maxPower //eq. (4.1), where* v_{max} *replaces* v_g *in eq. (4.2), (4.3), (4.4)*

get the distance to the intersection, *d*

get the current speed of the bicycle, v_i

$t = d/ v_i$

if (*signalState* = "red" && *timeToChange* > *t*)

  compute v_a *: eq. (4.9) for decelerated motion*

endif

if (*signalState* = "green" && *timeToChange* < *t* && *timeToChange* ≥ d/v_{max})

  compute v_a *: eq. (4.9) for accelerated motion*

endif

//the computed advised speed, v_a*, is recommended to the rider*

END

Note that this algorithm, although it does not include the FL wind-aware speed adaptation policy, it does consider the wind speed, too. The algorithm is designed to take into account the characteristics of the bicycles in general as compared to other vehicles, and as it was underlined before, the bicycles are the only class of vehicles highly affected by the wind. In most of the GLOSA systems designed for vehicles in general, the maximum speed considered in computation, v_{max} is equal to what in our algorithm is represented by *maxSpeed*, which is the maximum speed or the speed limit of the vehicles. In the case of the vehicles in general this is given by the road rules and regulations. In the case of the bicycles, we chose as *maxSpeed* a safety value of 25km/h

(6.95m/s). However, v_{max} is not always equal to this *maxSpeed* as in the case of the other vehicles and this is due to the special characteristics of the bicycles in general and electric bicycles in particular. First, the wind factor is highly affecting the power consumption and second, every electric bicycle has a power limit that can be sustained while riding. This power limit is associated to the *maxPower* parameters in our algorithm. Thus, v_{max} is computed based on the Bicycle Power Consumption Model considering all these factors. If v_{max} is not computed by taking into consideration these special characteristics of the bicycles, the advised speed recommended in the case: *signalState* = "green" && *timeToChange* < *t* && *timeToChange* $\geq$ *d/vmax* (the advised speed would be a new increased speed) could be too high in order to be sustained by the bicycles. Therefore, the Green Optimal Speed Advisory functionality could be affected to a certain degree.

4.2.3.2. FL-based Wind-aware Speed Adaptation Policy

This policy is implemented as it was previously mentioned by the FLS component of the Computation and Recommendation Module. The FLS has a single input, the wind speed, v_w, and a single output, v_{max}.

The structure of the FLS includes a *Fuzzifier*, an *Inference Engine*, a *Defuzzifier* and a *Knowledge Rule Base* and is typical for a FLS.

Fuzzifier takes the crisp value as input and gives as output the corresponding Fuzzy degree of membership based on the defined membership functions.

Inference Engine maps the input fuzzified value to the output based on the "IF-THEN" rules contained in the ***Knowledge Rule Base***. The *Knowledge Rule Base* is the one that also contains the membership functions.

The membership function of v_w is trapezoidal and is described in eq. (4.10) and eq. (4.11). A trapezoidal function was used for the input parameter due to its suitability to real-time systems, as it has reduced computation complexity [61].

$$\mu_{trapezoidal} = \begin{cases} \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & otherwise \end{cases} \tag{4.10}$$

$$\mu(v_w) = \{(a, b, c, d) \mid a, b, c, d \text{ are the coefficients for the corresponding fuzzy sets}\} = \{(0, 0.2v_{w1}, 0.8v_{w1}, v_{w1}), (0.8v_{w1}, 0.6v_{w2}, 0.8v_{w2}, v_{w2}), \ldots, (0.8\, v_{wi}, 0.2v_{windmax}, \infty)\} \tag{4.11}$$

$v_{windmax}$ represents the maximum wind speed considered to be safe for cycling which is approximately 10m/s[18]. However this parameter can be changed according to the needs and

[18] BikeRadar, WindSpeed – What's safe?, http://www.bikeradar.com/forums/viewtopic.php?f=40020&t=12903729

capabilities of the cyclist. The membership function is parameterized and its parameters are computed in real-time based on the bicycle power consumption model. Thus, $v_{wi,}$ parameters are dynamically computed following a very simple and fast iterative procedure described in Algorithm 4.2. This process takes place at system initialization, when the power consumption model is set.

Algorithm 4.2: Computing the membership function parameters and rule base parameters

$maxSpeed$ = 6.95 //*the maximum speed set for bicycle[21]*

$v_{max} = maxSpeed$

$v_w = 0$

$v_w = v_w + windUnit$

$v_{windmax} = 10$

$i = 1$

while ($v_w < v_{windmax}$)

  compute P_{total} //*eq. (4.1), where* v_{max} *replaces* v_g *in eq. (4.2), (4.3), (4.4)*

  while ($P_{total} < maxPower$)

    $v_w = v_w + windUnit$

  endwhile

  $v_{wi} = v_w$

  $v_{maxwi} = v_{max}$

  $v_{max} = v_{max} - windUnit$

  i++

endwhile

Being a zero-order Sugeno FLS, the IF-THEN rules have as consequents crisp values. This is the reason why the output was not associated with a membership function. The crisp values taken by the output v_{max} are described in eq. (4.12) and they are designed to correspond to each of the input's Fuzzy set described in eq. (4.11). Consequently these crisp values were parameterized, too. An example of an "IF-THEN" rule is given in eq. (4.13).

$$\{|v_{maxw1}| - windUnit, |v_{maxw2}| - windUnit, \ldots\} \tag{4.12}$$

,where $|v_{wi}|$ represents the value of the v_{w1} rounded to the speed measurement unit used by the speed advisory system to present the recommendations to the cyclist, while *windUnit* represents one single unit of the same measurement. The choice of these outputs has been made mainly for practical purposes.

$$\textbf{\textit{IF}}\ v_w\ \textit{is Low}\ \textbf{\textit{THEN}}\ v_{max}\ \text{is}\ |v_{maxw1}| - windUnit \tag{4.13}$$

,where *Low* is the Fuzzy set described by *(0, 0.2*v_{w1}*, 0.8*v_{w1}*,* v_{w1}*)* in eq. (4.11).

Defuzzifier's role in a FLS is to give the crisp value of the output applying different defuzzification methods on the output of the *Inference Engine*. In this case, the *Defuzzifier* uses the weighted average defuzzification method that is specific to the Sugeno fuzzy models. However, being a single-input single-output controller, the defuzzifier can be bypassed as the value of the output is given in crisp value directly by the *Inference Engine.*

The algorithm describing the full functionality of SAECy, the Green Optimal Speed Advisory function and the wind-aware speed adaptation policy is presented in Algorithm 4.3. Note that in this case, the solution will make two different recommendations depending on the context: a recommendation of the advised speed before intersections, where the cyclist is recommended to ride at a certain speed (the advised speed), and the second whenever the wind information is made available and involve recommending the cyclist a speed limit (now the advised speed is communicated as a maximum speed). The Green Optimal Speed Advisory procedure is the same as presented in section 4.2.3.1), with a single modification, the v_{max} is now computed by the FLS.

Algorithm 4.3: Green Optimal Speed Advisory + FL-based Wind-aware Speed Adaptation Policy

INITIALIZATION PHASE:

BEGIN

maxSpeed = 6.95 //*the maximum speed set for bicycle*

v_{max} = *maxSpeed*

if (wind_info available from a weather server)

    init v_w

    call FLS => v_{max}

    $v_a = v_{max}$

//advise the cyclist to ride at a maximum speed of v_a

else

    $v_w = 0$

endif

END

GREEN OPTIMAL SPEED ADVISORY PROCEDURE:

-triggered when a message is received from a traffic light and then triggered in the monitoring cycle while the bicycle has not crossed the intersection yet

BEGIN

    update v_w

call FLS => v_{max}

get the distance to intersection, *d*

get the current speed of the bicycle, v_i

$t = d/ v_i$

if (*signalState* = "red" && *timeToChange* > *t*)

compute $\boldsymbol{v_a}$**:** *eq. (4.9) for decelerated motion*

endif

if (*signalState* = "green" && *timeToChange* < *t* && *timeToChange* $\geq$ d/v_{max})

compute $\boldsymbol{v_a}$**:** *eq. (4.9) for accelerated motion*

endif

//the computed advised speed, v_a*, is recommended to the rider*

END

FINAL PHASE:

-intersection is crossed

BEGIN

//advise the cyclist to ride at a maximum speed of v_a

END

4.3. Energy Efficient Weather-aware Route Planner solution for Electric Bicycles – eWARPE

4.3.1. eWARPE Architecture and Functional Principle

This solution represents a step forward for the cycling route planners. eWARPE goes beyond planning the route itself, how to get from point A to point B. eWARPE is planning the optimal departure time for the route: when to leave from point A to get to point B on the previously planned route. The departure time is chosen from a user configurable time interval so that the cyclist will get the best possible cycling experience in that interval (avoid the adverse weather conditions) and will increase the energy savings of the electric bicycle. The architecture of eWARPE (Figure 4.4) with the focus on this latter functionality is detailed next.

User Interface is the front-end that enables the cyclist to introduce the input information represented by the route information: the path (how to get from start to destination) and the departure time interval [$T_{initial}$, T_{final}] (the time interval when cyclist is available to take the selected path). *User Interface* allows for planning the path, route itself, based on the following information: start, destination and route preferences (e.g. more cycling facilities, fastest route, etc.) or for directly loading a preferred route. This functionality of eWARPE can be ensured by integrating an open source route planner such as BBBike[19] for instance. If the cyclist has a preferred route this can be loaded directly in the *User Interface*. The route information is sent to the *Recommendation Module* that feeds back the departure time not long before the smallest edge of the departure time interval. The departure time can be designed to be announced by the *User Interface* as an alarm message. Details about weather conditions and energy consumption for the route when leaving at the recommended departure time (i.e. $time_{departure}$) – (route, $time_{departure}$) – are displayed as well by *User Interface*. If requested by user, these details can be displayed for each (route, T_i), where T_i is in departure time interval and the difference T_i - T_{i-1} is user configurable. These details are provided to the *User Interface* by the *Recommendation Module.*

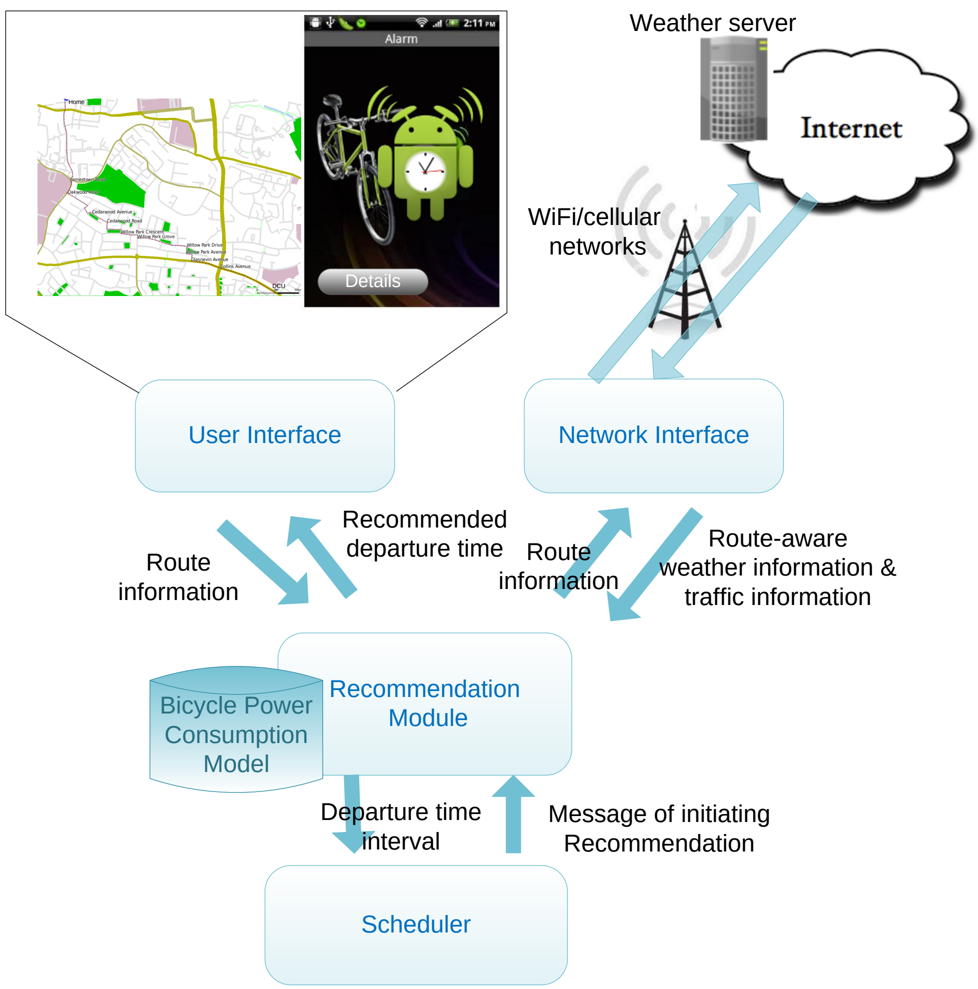


Figure 4.4. eWARPE Architecture

Network Interface is the component responsible with retrieving route-aware weather information from a weather server. There are many weather services based on the HTTPRequest/Response paradigm, providing APIs and able to deliver the information needed such

[19] Cycling Route Planner, http://www.bbbike.org/

as OpenWeatherMap[20], AccuWeather[21]. This information is stored in a list of 5-tuples, W_{info}, that covers the user defined interval. The 5-tuple has the following fields:

- *time*, the starting time of predicted weather which is valid for a time interval, *t*, that represents the granularity the weather server is capable to provide weather information with. Commonly this granularity is hourly (t = 60min) or every half an hour (t = 30min).
- *precipitation chances*, given in percentages
- *wind speed*, given in m/s
- *wind direction*, given in degrees
- *weather warnings*, denotes special weather conditions (e.g. strong thunderstorm, strong wind, etc.).

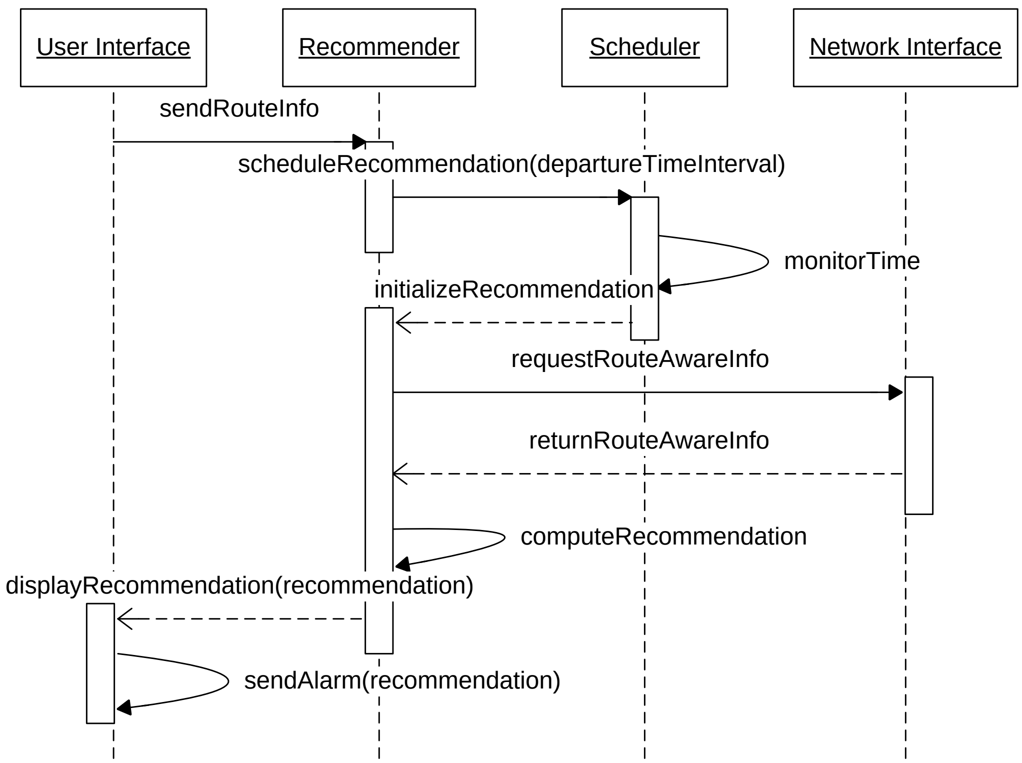


Figure 4.5. Interaction between Functional Blocks

The information about the route is provided by the *Recommendation Module* when it requires the aforementioned information from the *Network Interface*. The route-aware weather information is the weather forecasted by the closest meteorological station to the route. This increases the accuracy of the solution.

[20] OpenWeatherMap website, openweathermap.org
[21] AccuWeather website, http://apidev.accuweather.com/developers/

Recommendation Module is the component that, based on the information it is provided with, determines $time_{departure}$ (in the given time interval) so that the cyclist will be provided with the best weather conditions possible in terms of cycling experience and energy-efficiency. The computation of the optimum departure time, $time_{departure}$, is described in Algorithm 4.4 and employs the *Bicycle Power Consumption Model* and a zero-order Sugeno FLS that is described in the next section. The *Recommendation Module* is also responsible with computing the energy consumption of every (route, T_i). The energy consumption of the route is determined based on the *Bicycle Power Consumption Model*, the same as the one employed in SAECy, and the estimated travel time that is given when the route is established or loaded in the route planner.

The computation of $time_{departure}$ is controlled by a *Scheduler*. The *Scheduler* checks the departure time interval and ensures that *Recommendation Module* requires the information from *Network Interface* not long before the smallest edge of this interval. As such, the accuracy of the information received from the *Network Interface* will increase and consequently the accuracy of the computation will increase, too. The *Scheduler* timings can be configured by the users via the *User Interface.* For a better understanding, the interaction between the functional blocks as presented in their description is also summarized and illustrated in Figure 4.5 in form of a sequence diagram. Note that in this diagram, the *Recommender* block is actually *Recommendation Module.*

4.3.2. eWARPE Algorithm

The algorithm presented in this section, Algorithm 4.4, describes the logic embedded in the *Recommendation Module* in order to provide the user with the best suitable departure time. For this, the weather-aware information is analyzed for all the 5-tuples imposed by t and the user-defined interval [$T_{initial}$, T_{final}]. The number of tuples is defined by the formula given in eq. (4.14).

$$number\_of\_tuples = \frac{[T_{final} - (T_{final}\ mod\ t) + t] - [T_{initial} - (T_{initial}\ mod\ t)]}{t} \quad (4.14)$$

As aforementioned, *Recommendation Module* encapsulates a FLS that follows the zero-order Sugeno fuzzy model-based and it is aimed to help in the decision making process. The FLS is giving a *priority*, the output of the FLS, to each of the 5-tuples from W_{info} based on the *precipitation chances* and *wind speed* that represent the input of the FLS. Then the recommended departure time, $time_{departure}$, is selected from the time interval defined by the tuple that has the maximum *priority* and minimum estimated energy consumption, *Ec* that is computed as described in eq. (4.17).

The FLS employed for computing the *priority* has the same components as the one designed in SAECy context as it follows the same zero-order Sugeno-model. Thus the components have the same role, their behavior being based on the ***Knowledge Rule Base*** that contains the membership functions of the two inputs and the ***Rule Base***. The membership functions of the *precipitation chances* (Figure 4.6) and *wind speed* (Figure 4.7) are trapezoidal and are generally described by eq. (4.11) as all the trapezoidal functions and specifically in eq. (4.15) and eq. (4.16) respectively, while the ***Rule Base*** is defined in TABLE 4.3. The membership functions and rules are built so that the chances of bad weather conditions are reduced, as the aim is to assign higher *priority* to better weather conditions.

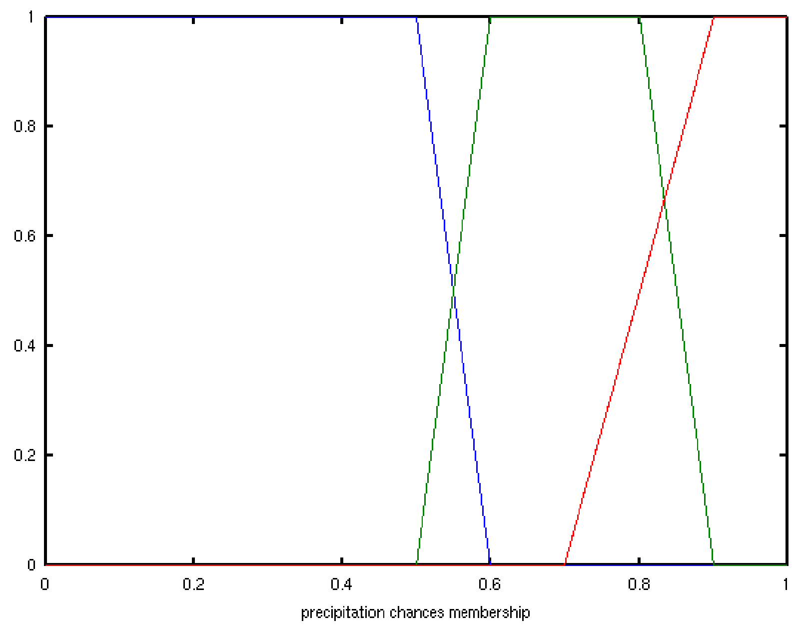


Figure 4.6. Precipitation chances Membership

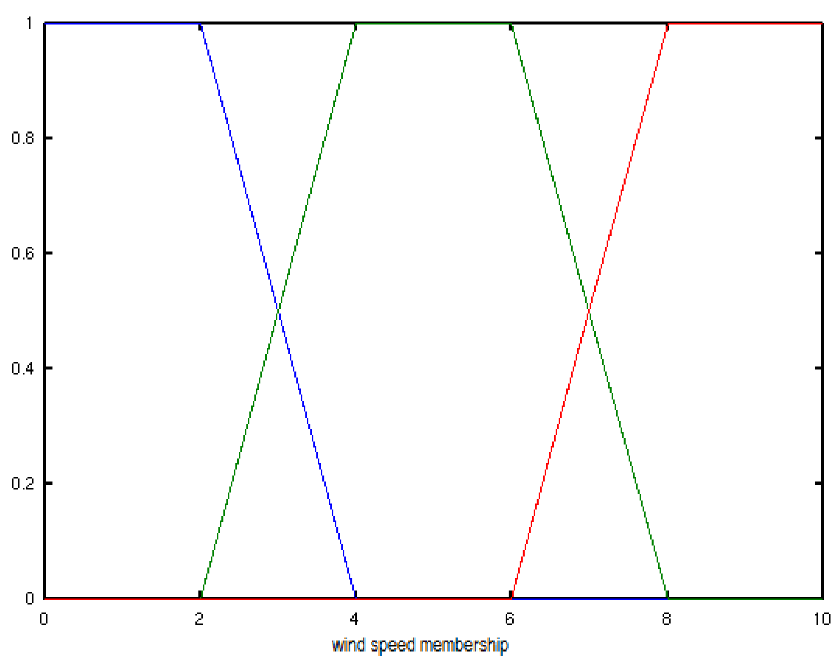


Figure 4.7. *Wind speed* Membership

Algorithm 4.4: eWARPE Algorithm

if *weather warnings* in [$T_{initial}$, T_{final}]

display warning

remove all *tuples* from W_{info} with *weather warnings*

end if

for all *tuples* **in** W_{info}

call FLS => *priority*

compute *Ec //using eq. (4.17)*

end for

select $time_{departure} = time_i$, where $time_i$ is in $tuple_i$ that has $priority_i = \underbrace{max}_{tuple_j\ in\ W_{info}}\ priority_j$ **and** $Ec = \underbrace{min}_{tuple_j\ in\ W_{info}}\ Ec_j$

μ(precipitation chances) = {(a, b, c, d)| a, b, c, d are the coefficients for the Low, Medium, respectively High Fuzzy sets} = {(-∞, 0, 0.5, 0.6), (0.5, 0.6, 0.8, 0.9), (0.7, 0.9,1, ∞)} (4.15)

μ(wind speed) = {(a, b, c, d)| a, b, c, d are the coefficients for the for the Low, Medium, respectively High Fuzzy sets } = {(-∞, 0, 2, 4), (2, 4, 6, 8), (6, 8, 10, ∞)} (4.16)

TABLE 4.3 EWARPE FLS RULE BASE

Rule	***Rain chances***	***Wind speed***	***priority***
1	Low	Low	3
2	Low	Medium	3
3	Low	High	2
4	Medium	Low	3
5	Medium	Medium	2
6	Medium	High	1
7	High	Low	1
8	High	Medium	1
9	High	High	1

The computation of *Ec* is performed based on the *Bicycle Power Consumption Model*, eq. (4.1), as described in eq. (4.17), where $P_{total}^{road\_segment}$ is the power corresponding to a certain road segment pertaining to the route, while $time_{road\_segment}$ is the estimated time the road segment is travelled in. The road segments are the result of the total route segmentation in correspondence to the direction of the bicycle.

$$E_c = \sum P_{total}^{road\_segment} \times time_{road\_segment} \tag{4.17}$$

4.4. FuzzC-VANET - a Fuzzy Logic-based Clustering Scheme

4.4.1. FuzzC-VANET Principle and Network Model

Network scalability and stability issues are identified among the main challenges of VANETs [22]. Unlike most ad-hoc networks that usually assume a limited network size, VANETs can be extended on the entire road network which involves a potentially great number of vehicle-nodes. The stability issues in VANETs are imposed by the the high mobility of the vehicles. Clustering addresses these two issues as it provides good system performance, good management and stability of the networks in the presence of mobility and large number of nodes [88]. FuzzC-VANET is a general clustering scheme that represents a response to stability and scalability issues, the main problems that MAC, routing, data dissemination protocols have to address in VANETs. Thus, FuzzC-VANET can be instanciated further on for specific MAC, routing protocols or information dissemination. Further on, in VANETs, the clusters are well-known to be able to store very up-to-date location-based information. Considering this, in the context of SAECy, the clusters formed with FuzzC-VANET can be employed to retrieve the location-based weather-information.

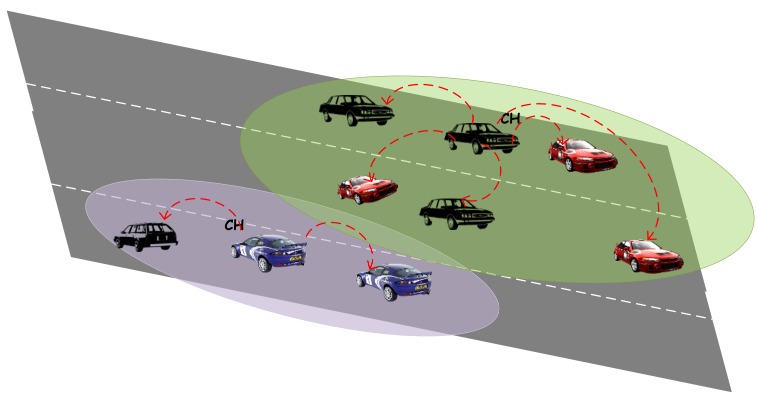


Figure 4.8. Clustered Network Model

The proposed distributed clustering scheme organizes the vehicles into clusters based on neighbouring and direction of travelling creating a clustered network (Figure 4.8). This scheme is included in the category of the CH-based clustering solutions, the vehicles are self-organizing themselves into clusters around a CH. CH selection is based on multiple clustering metrics such as: location, velocity, direction, provided by each vehicle's OBU that has a GPS integrated, and number of neighbours. The clustering process takes place through message exchange using the IEEE 802.11p technology, messages that contain all necessary information, including the clustering metrics. The intra-cluster communication, excluding here the exchange of clustering messages, takes place only between CH and the cluster members and not between the cluster members themselves.

Figure 4.9 presents the finite state machine that describes the clustering schemes. As it can be seen in the figure, three possible states of the vehicles are defined: independent, not included in any cluster, clustered, a cluster member, and CH. A CH will maintain a list of all the cluster members in its cluster. The transitions between these states are regulated by the algorithms involved in the FuzzC-VANET clustering scheme that are detailed in the next section.

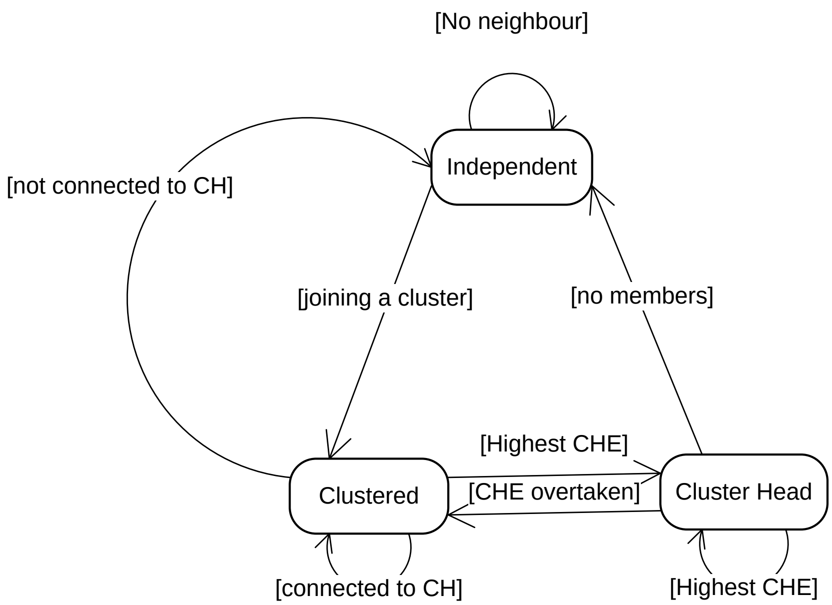


Figure 4.9. Clustering Scheme Finite State Machine

4.4.2. FuzzC-VANET Algorithms

Each vehicle advertises itself as part of the network by transmitting a beacon message. The message has the format presented in Figure 4.10, where the *id* is a unique number assigned to each vehicle for identification, *direction* is the direction of the vehicles expressed as an angular value, *speed* represents the current speed of the vehicle expressed in m/s, *position* represents the standard GPS coordinates of the vehicles, *CH status* is a true or false flag to indicate whether the vehicle is a CH, *CH id* is a unique number identifying the CH of the cluster the vehicle belongs to and *CHE* is a measure of how suitable is the vehicle to become a CH.

id	*direction*	*speed*	*position*	*CH status*	*CH id*	*CHE*

Figure 4.10. Clustering beacon message

Based on these messages, each vehicle, v_i, establishes its neighbors that travel in the same direction and retains in a list, N_i, each neighbor-related information and triggers the computatation of the *CHE* as described in section 4.4.2.1. At the receipt of a beacon message, each vehicle triggers the Algorithm 4.6 that is basically building the aforementioned list of neighbours. Two vehicles, v_i and v_j, are considered to have the same direction if the difference between their directions, θ_{ij}, is not greater than θ_{MAX} grades, where θ_{MAX} was set to 45 grades as in [118]. If v_i is the receiving vehicle and v_j the sender and if v_j is not already in the list if v_i 's neighbours, N_i, vehicle v_j is added in N_i. However if θ_{ij} is greater than θ_{MAX} grades and v_j is not included in N_i then v_j is ignored. If it is in the list, the vehicle is not being removed immediately from the list of neighbours as is the case for the rest of the clustering schemes designed for VANETs that consider direction as a clustering metric. This design decision is based on the fact that greater angle than θ_{MAX} grades does not necessary imply a change in the direction of the vehicle v_j as compared to v_i . It might represent for instance a change to a same-direction lane, avoiding an obstacle or an irregularity in the shape of the road. The vehicle v_j will be removed from N_i if next no beacon message will be received by v_i from v_j.

Moreover, in the maintenance process of the neighbour list, N_i, a *cleanup* is triggered every second by each vehicle, similar to other VANET clustering approaches [109], [121]. The *cleanup* function (Algorithm 4.5) analyzes the *expTime* field of every neighbour, to identify neighbours that have not been in communication within the required time frame and are further removed from the list. Setting the *expTime*, which is the aforementioned time frame, is one of the actions taken in the Algorithm 4.6. Its value is updated in the defined algorithms in accordance with the broadcasting interval of the beacon messages, meaning that this time is increased with the value of the broadcasting interval. This interval was set to one second as in the most VANET clustering algorithms based on beacon advertising (e.g. [109], [121]).

Algorithm 4.5: Cleanup function

```
//triggered every second by each vehicle, v_i
for each neighbour in N_i
   if expTime timeout
       remove neighbour from N_i
       if CHStatus_i = true  and neighbour in CM_i
          remove neighbour from CM_i
       endif
    endif
end for
```

Algorithm 4.6 may have a further impact than only building the list of neighbours of a vehicle and triggering the computation of *CHE*. If the receiving vehicle, v_i is a CH then this algorithm contributes to the creation and maintenance of the v_i 's cluster member list, CM_i. Note that the aforementioned *cleanup* function also contributes to the maintenance of this list. When a neighbour is deleted from the neighbour list, N_i, of vehicle v_i that is CH, if this neighbour is included in the CM_i, then this is also being removed from CM_i. Morevover, Algorithm 4.6 triggers Algorithm 4.7, which is the algorithm responsible with CH election or re-election, in two cases:

- Both vehicles, the sender (v_j) and receiver (v_i) are CHs, $CHStatus_i$ = true and $CHStatus_j$ = true, and $\theta_{ij} < \theta_{MAX}$ grades. This is the case of merging two clusters, one vehicle will loose its CH state (*CHStatus* = false) and it will become a simple cluster member (clustered state) in the cluster governed by the other vehicle. The cluster members of the cluster governed by the vehicle that lost its CH will most probably become cluster members of this cluster, too.
- Vehicle v_j is the CH of v_i ($CHId_i = v_j$), but v_j has lost its CH state ($CHStatus_j$ = false). In this case, v_i needs to re-elect a CH.

Algorithm 4.6: Receipt of beacon message

```
//Vehicle v_i receives the beacon message from vehicle v_j
compute θ_ij
if 0 <= θ_ij <= θ_MAX then
   if v_j not in N_i
      add v_j to N_i
   endif
   update expTime
   if CHStatus_i = true and CHStatus_j = true
      call Algorithm 4.7
   endif
endif
if θ_ij > θ_MAX and v_j in N_i
    update expTime
endif
if v_j in N_i
    if CHId_i = v_j and CHStatus_j = false
       call Algorithm 4.7
    endif
    if CHStatus_i = true and CHId_j = i
        if v_j not in CM_i
           add v_j to CM_i
       endif
    endif
    compute CHE_i //triggering the FLS (see section 4.4.2.1 )
endif
```

The CH (re)election process described in Algorithm 4.7, determines if a vehicle should select a CH, become a CH, remain or resign as CH. The starting point in this sequence is the CH status flag (*CHStatus*), which specifies whether the vehicle is a CH or not. It can be seen from Algorithm 4.7 instructions that the vehicle with maximum *CHE* among its neighbours is to become CHs, neighbours that become cluster members of the cluster gorverned by this CH. Thus the aim is that in a cluster, the CH to have the highest *CHE* among the all other cluster members. The procedure described by this algorithm is triggered by Algorithm 4.6 or periodically on a time interval (TI) by each vehicle.

Algorithm 4.7: Cluster head (re)election

---

if $CHStatus_i$ = false
  search in N_i for v_k where $CHE_k = \underbrace{max}_{v_l \; in \; N_i} CHE_l$
  if $CHStatus_k$ = true
    $CHId_i = k$ //vehicle v_k is selected as CH for vehicle v_i
  else if v_k unclustered **and** $CHE_i > CHE_k$
    $CHStatus_i$=true //auto-elect as CH
  endif
else
  search in N_i for v_k where $CHE_k = \underbrace{max}_{v_l \; in \; N_i} CHE_l$

  if $CHStatus_k$ = true **and** $CHE_i < CHE_k$
    $CHStatus_i$=false // vehicle v_i resigns as CH
  else if $CHStatus_k$ = false
    $CHStatus_i$=false // vehicle v_i resigns as CH
  endif

4.4.2.1. CHE Computation

The clustering metrics used in CH election and consequently in computing the CH eligibility factor, *CHE*, are average distance (AD), average velocity (AV), described in eq. (4.18) and eq. (4.19), and the number of neighbours (N).

AD_i represents the overall average absolute distance between v_i and the neighbors from its neighbours list N_i.

$$AD_i = \frac{\sum_j \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}}{|N_i|} \quad (4.18),$$

where (x_i, y_i), (x_j, y_j) are the coordinates of the positions of v_i and v_j respectively.

AV_i is the average of the differences between the velocity of the v_i and those of the neighbors from N_i.

$$AV_i = \frac{\sum_j |S_i - S_j|}{|N_i|} \quad (4.19),$$

where S_i, S_j represents the velocity of v_i and v_i, respectively. In each of the previous equations, j takes the values of each id from the (N_i), while $|N_i|$ is the cardinality of the neighbors list.

The smaller the average of the differences in speed, AV_i,, is the closer is the speed of vehicle v_i to the majority of its neighbours, thus the closest to the average. Similar reasoning applies in the case of AD_i, while the number of the neighbours, $|N_i|$, is desirable to be high.

Based on the fact that FL is a powerful math tool to deal with the imprecision of VANETs environment modeled through the clustering metrics considered and, also is well known for its ability of dealing with multiple parameters, we have developed a FL-based approach. This design decision was also influenced by the fact that as we presented before, FL has been successfully used before in making complex decision in VANETs. However, so far, to the best of our knowledge, no FL decisional system for CH election has been implemented in VANETs clustering. The FLS designed plays a major role in CH election as it computes *CHE* for each vehicle.

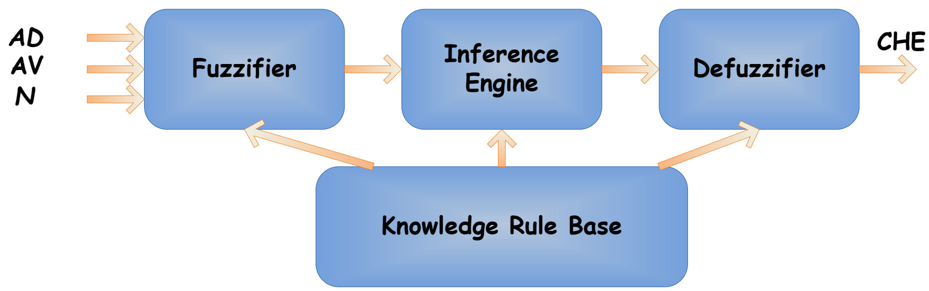


Figure 4.11. FuzzC-VANET FLS

The proposed FLS (Figure 4.11) follows the Mamdani fuzzy model, and has a typical structure for such a FLS including a *Fuzzifier*, an *Inference Engine*, a *Defuzzifier* and a *Knowledge Rule Base*. The design of the presented FLS is a trade-off between accuracy and reduced computation complexity and follows design principles from [61] and [234]. Next paragraphs describe each of this FLS components.

Fuzzifier takes the crisp values of *AD*, *AV* and *N* as inputs and gives as output their corresponding Fuzzy degree of membership based on the defined membership functions. The membership functions of *AD*, *AV* and *N* are trapezoidal and they are described in equations (4.11), (4.21), (4.22) and (4.23), while the membership function of *CHE* is triangular and is described in equations (4.24) and (4.25). Figure 4.12 presents the membership functions of *AD*, while Figure 4.13 illustrates the membership functions of *CHE*. The membership functions of *AV* and *N* have the same shapes as the membership functions of *AD*.

All the membership functions are stored in the ***Knowledge Base***. Trapezoidal and triangular membership functions were used for the input/output parameters due to their suitability to real-time systems, as they have reduced computation complexity. The parameters of membership functions were selected using manual tuning and "expert knowledge" based on the reasoning previously made related to the *AV*, *AD* and *N*. The membership functions are parameterized starting from their maximum possible values. The maximum possible value for *AV* is the maximum speed allowed for the corresponding road segment, v_{max}. The maximum value of *AD* is the transmission range of the technology, d_{max}. The maximum number of neighbours of a vehicle, n_{max}, can be described through the eq. (4.20), where *averageCarLength* represents the length of the cars on average, *minGap* represents the minimum distance between a car and its lead car and *numberOfLanes* represents the current number of same direction lanes for the corresponding length of the road given by the transmission range, d_{max}. If there are different number of lanes on the distance $2 \times d_{max}$, the equation is broken in multiple components accordingly. The multiplication with 2 is needed as there are neighbours both in the front of the car and behind. The values for *averageCarLength* (*averageCarLength* = 5m) and *minGap* (*minGap* = 2.5m) are predefined and they are taken from Krauss car-following model [239].

$$n_{max} = 2 \times \frac{d_{max}}{(averageCarLength + minGap)} \times numberOfLanes \tag{4.20}$$

$$\mu(AV) = \{(a, b, c, d) \mid a, b, c, d \text{ are the coefficients for } F_{AV}^{L}, F_{AV}^{M}, F_{AV}^{H}\}$$
$$= \{(-\infty, 0, 0.1v_{max}, 0.2v_{max}), (0.1v_{max}, 0.2v_{max}, 0.4v_{max}, 0.6v_{max}), (0.4\,v_{max}, 0.6\,v_{max}, v_{max}, \infty)\} \tag{4.21}$$

$$\mu(AD) = \{(a, b, c, d) \mid a, b, c, d \text{ are the coefficients for } F_{AD}^{L}, F_{AD}^{M}, F_{AD}^{H}\}$$
$$= \{(-\infty, 0, 0.1d_{max}, 0.2d_{max}), (0.1d_{max}, 0.2d_{max}, 0.4d_{max}, 0.6d_{max}), (0.4d_{max}, 0.6\,d_{max}, d_{max}, \infty)\} \tag{4.22}$$

$$\mu(N) = \{(a, b, c, d) \mid a, b, c, d \text{ are the coefficients for } F_{N}^{L}, F_{N}^{M}, F_{N}^{H}\}$$
$$= \{(-\infty, 0, 0.1n_{max}, 0.2n_{max}), (0.1n_{max}, 0.2n_{max}, 0.4n_{max}, 0.6n_{max}), (0.4n_{max}, 0.6n_{max}, n_{max}, \infty)\} \tag{4.23}$$

$$\mu_{triangular} = \begin{cases} \frac{x-a}{m-a}, & a < x \le m \\ \frac{b-x}{b-m}, & m < x < b \\ 0, & otherwise \end{cases} \tag{4.24}$$

$$\mu(CHE) = \{(a, b, c) \mid a, b, c \text{ are the coefficients for } \mathcal{F}_{\mathcal{CHE}}^{\mathcal{VL}}, \mathcal{F}_{\mathcal{CHE}}^{\mathcal{L}}, \mathcal{F}_{\mathcal{CHE}}^{\mathcal{M}}, \mathcal{F}_{\mathcal{CHE}}^{\mathcal{H}}, \mathcal{F}_{\mathcal{CHE}}^{\mathcal{VH}}\}$$
$$= \{(-\infty, 0, 0.25), (0, 0.25, 0.5), (0.25, 0.5, 0.75), (0.5, 0.75, 1), (0.75, 1, \infty)\} \tag{4.25}$$

Inference Engine maps the input fuzzified values to the output Fuzzy set described by the output membership function. The mapping is done based on the "IF-THEN" rules contained in the ***Rule Base*** that is included in the ***Knowledge Base*** and described in eq. (4.26).

$$\mathcal{R}^{(l)}: \textbf{\textit{IF}}\ N \text{ is } \mathcal{F}_{\mathcal{N}}^{\ell}\ \textbf{AND}\ AD \text{ is } \mathcal{F}_{\mathcal{AS}}^{\ell}\ \textbf{AND}\ AV \text{ is } \mathcal{F}_{\mathcal{AV}}^{\ell}\ \textbf{\textit{THEN}}\ CHE \text{ is } \mathcal{F}_{\mathcal{CHE}}^{\ell} \tag{4.26}$$

where $\mathcal{R}^{(l)}$ is the index of the rule in the *Rule Base* and $\mathcal{F}_{\mathcal{AC}}^{\ell}$, $\mathcal{F}_{\mathcal{AS}}^{\ell}$, $\mathcal{F}_{\mathcal{AV}}^{\ell}$, $\mathcal{F}_{\mathcal{CD}}^{\ell}$ and $\mathcal{G}^{\ell}$ are the corresponding Fuzzy sets of *AC*, *AV*, *AD* and *CHE*, respectively (TABLE 4.4). The full description of the Rule Base, containing all the rules, is presented in TABLE 4.5.

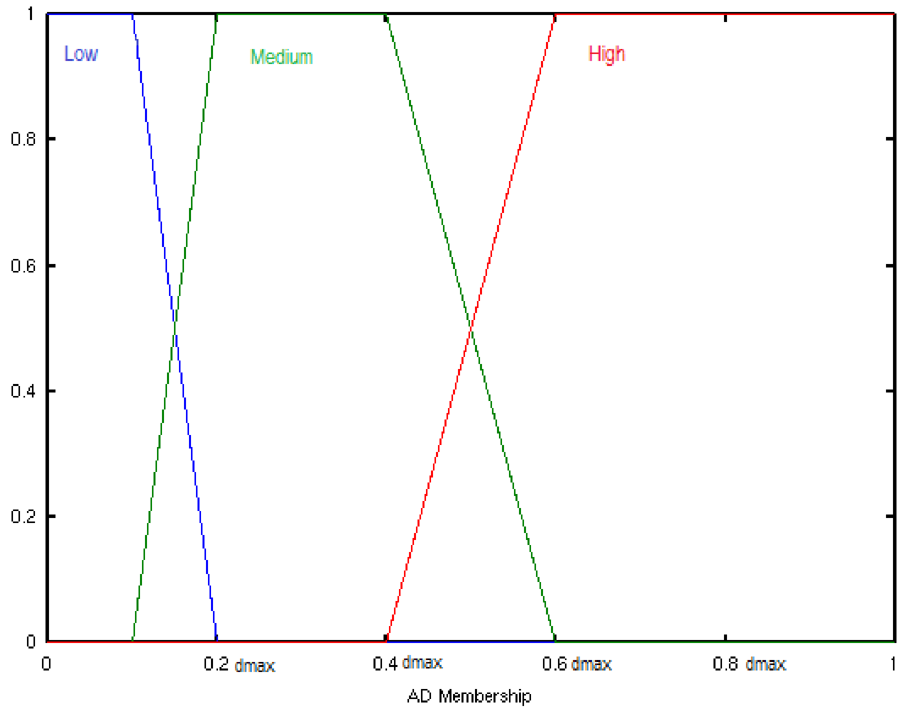


Figure 4.12. *AD* Membership Functions

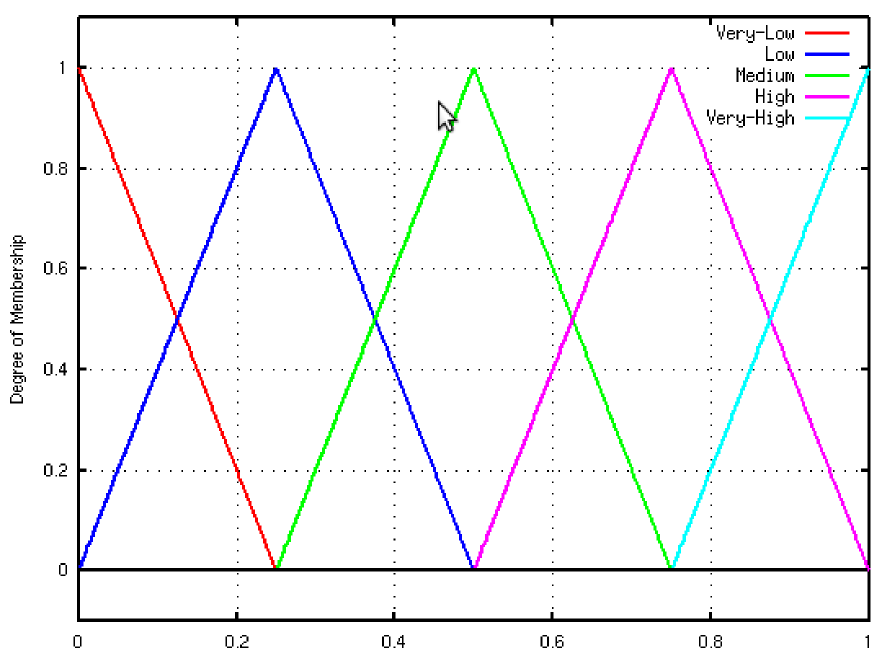


Figure 4.13. *CHE* Membership Functions

TABLE 4.4. INPUTS/OUTPUT LINGUISTIC VARIABLES

Linguistic Variables		**Fuzzy Sets**
Input linguistic variables	Average Velocity (*AV*)	{Low ($\mathcal{F}_{\mathcal{AV}}^{\mathcal{L}}$), Medium ($\mathcal{F}_{\mathcal{AV}}^{\mathcal{M}}$), High ($\mathcal{F}_{\mathcal{AV}}^{\mathcal{H}}$)}
	Average Distance (*AD*)	{Low ($\mathcal{F}_{\mathcal{AD}}^{\mathcal{L}}$), Medium ($\mathcal{F}_{\mathcal{AD}}^{\mathcal{M}}$) , High ($\mathcal{F}_{\mathcal{AD}}^{\mathcal{H}}$)}
	Number of neighbours (*N*)	{Low ($\mathcal{F}_{\mathcal{N}}^{\mathcal{L}}$), Medium ($\mathcal{F}_{\mathcal{N}}^{\mathcal{M}}$) , High ($\mathcal{F}_{\mathcal{N}}^{\mathcal{H}}$)}
Output linguistic variable	CH Eligibility (*CHE*)	{Very Low($\mathcal{F}_{\mathcal{CHE}}^{\mathcal{VL}}$) Low($\mathcal{F}_{\mathcal{CHE}}^{\mathcal{L}}$), Medium($\mathcal{F}_{\mathcal{CHE}}^{\mathcal{M}}$), High($\mathcal{F}_{\mathcal{CHE}}^{\mathcal{H}}$), Very High($\mathcal{F}_{\mathcal{CHE}}^{\mathcal{VH}}$)}

TABLE 4.5. FUZZ-C VANET FLS RULE BASE

Rule	**AD**	**AV**	**N**	**CHE**
1	Low	Low	Low	Low
2	Low	Low	Medium	High
3	Low	Low	High	Very High
4	Low	Medium	Low	Low
5	Low	Medium	Medium	High
6	Low	Medium	High	Very High
7	Low	High	Low	Very Low
8	Low	High	Medium	Low
9	Low	High	High	Medium
10	Medium	Low	Low	Low
11	Medium	Low	Medium	High
12	Medium	Low	High	Very High
13	Medium	Medium	Low	Low
14	Medium	Medium	Medium	Medium
15	Medium	Medium	High	High
16	Medium	High	Low	Very Low
17	Medium	High	Medium	Low
18	Medium	High	High	Medium
19	High	Low	Low	Very Low
20	High	Low	Medium	Low
21	High	Low	High	Medium
22	High	Medium	Low	Very Low
23	High	Medium	Medium	Low
24	High	Medium	High	Low
25	High	High	Low	Very Low
26	High	High	Medium	Low
27	High	High	High	Low

Defuzzifier takes the Fuzzy set given as output by the inference process and transforms it to the crisp value of *CHE*. The defuzzification is performed based on the COA defuzzification method. Basically the crisp value of *CHE* is the center of the area represented by the Fuzzy set given as output by the inference process. The formula for the center of area is given by the eq. (4.27), where A is the Fuzzy set given as output by the inference process.

$$z_{COA} = \frac{\int \mu_A(z) z \, dz}{\int \mu_A(z) \, dz} \tag{4.27}$$

4.4.3. User-oriented Fuzzy Logic-based Clustering Scheme

This is an instantiation of the clustering scheme proposed above that can be implemented in an infotainment system whose architecture is further described. The focus is on the stability of the network, but also on increasing the chances of users being provided with information of their interest. In the next sections, the architecture of the infotainment solution is presented and then the additions brought to the main clustering scheme are described.

4.4.3.1. Architecture

The architecture of the infotainment system that implements our proposed user-oriented FL-based clustering scheme is presented in Figure 4.14. The proposed solution uses a client-server architecture based on a hybrid VANET architecture. Consequenlty, the vehicles have an OBU supporting IEEE 802.11p technology and a location-aware mechanism (e.g. GPS) which determines vehicle speed, direction of travel and location. Next, the architectural components and the interaction between these in order to provide the system functionality are described.

Clustering Module implements the user-oriented FL-based clustering scheme. The proposed clustering solution makes use of IEEE 802.11p to exchange clustering specific messages, the beacon messages, and to deliver information.

Application Server provides information based on the dominant interests of a cluster that are communicated via RSU or base stations. Thus, the application server has information management capabilities being able to retrieve the information based on interest and location. The information stored in the server can come from different sources (e.g. Internet) or vehicles themselves. However, this is not in the scope of our work.

Context-aware Model Framework, a general configurable framework, such as the one proposed in [238], that is deployed on AU of each vehicle. The framework is able to model the user profile and based on this, to provide recommendations regarding user preferences. These recommendations are in form of a vector of user interests. Consequently each vehicle k has associated a vector of interests: $[p_{i1}, p_{i2}, \ldots, p_{ij}, \ldots p_{in}]$, where p_{ij} represents the percentage of user interest in topic j.

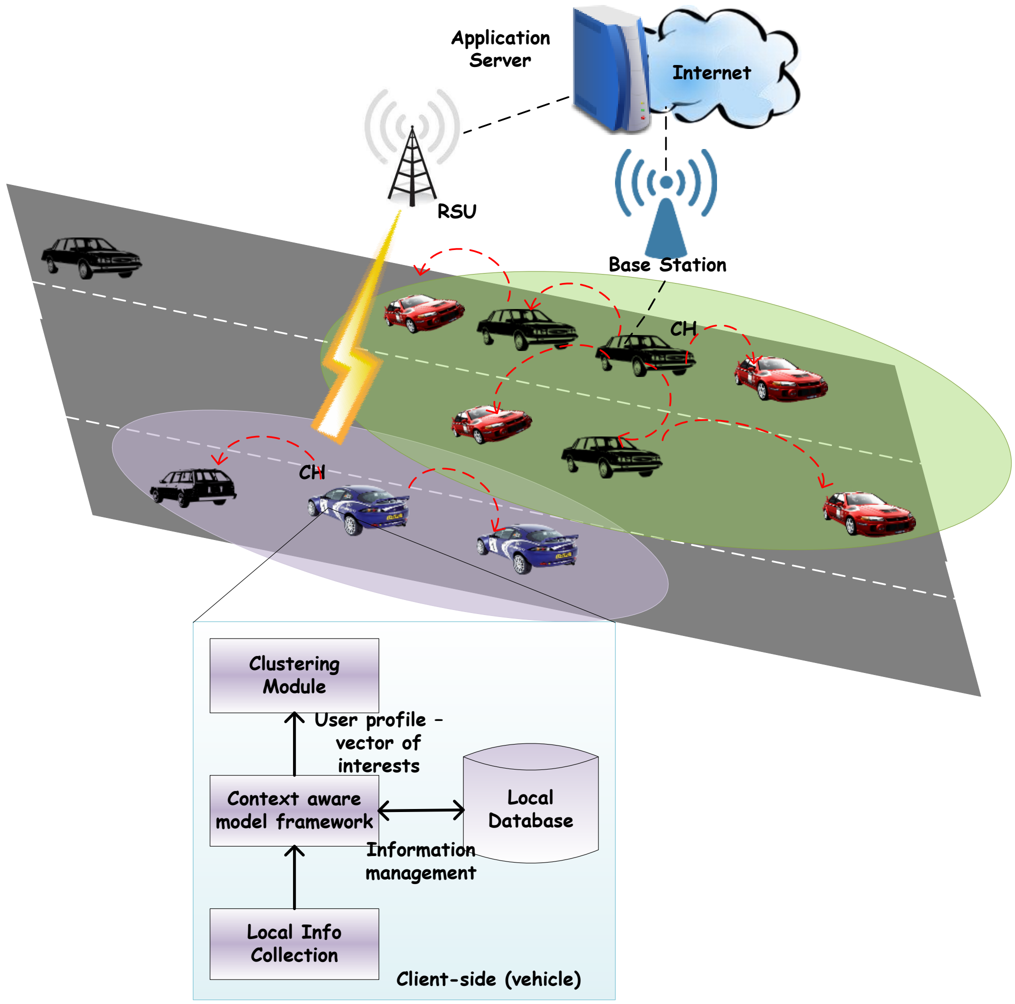


Figure 4.14. System Architecture

The vehicles are organized in clusters based on user interest level in certain information, location, direction of travel and velocity. Each cluster has a cluster head (CH) that deals with information according to the user interests in the cluster. CH is the only cluster node that communicates directly to RSU. CH sends the cluster user interest and location and other cluster-related information to the server via RSU and receives information from the server via RSU. In case of a pure ad-hoc architecture, the server's place can be taken by the CH that can provide information from its *Local Database* that stores information retrieved locally via *Local Info Collection* module or from the Application Server.

CH further disseminates the information to all its cluster members. Each cluster member holds a list with the same-direction neighbors that were registered during cluster formation process. When the information transmission is initiated by the CH, each cluster member will disseminate the information to all its neighbors. The cluster-based communication is independent from RSU. In addition, RSUs communicate with CHs only, resulting in decreased overload and energy saving at RSU level.

4.4.3.2. Additions to the Clustering Scheme

A) Additions to the Clustering Beacon Message Format

The beacon format is adapted in order to also contain the vector of interests of each vehicle (Figure 4.15). Based on this information, the receiving vehicle computes the *interest compatibility* (*IC*) that is an indicator of the compatibility in interests between the sender and receiving vehicle. The *IC* between vehicle *x* and *y* is measured by applying cosine similarity between the two vector of interests associated with vehicle *x* and *y* as presented in eq. (4.28).

id	*direction*	*speed*	*position*	*vector of interests*	*CH status*	*CH id*	*CHE*

Figure 4.15. Modified Beacon Structure

$$IC_{x,y} = \Phi(x,y) = \frac{\sum_{i=1}^{n} p_{xi} p_{yi}}{\sqrt{\sum_{i=1}^{n} p_{xi}^2 \sum_{k=1}^{n} p_{yi}^2}} \tag{4.28}$$

B) Alterations to CHE Computation

A new clustering metric is added that accounts only for CH election, namely average compatibility (AC). AC_i is the average of the interest compatibilities computed for vehicle *i* and each of its neighbors from N_i as described in eq. (4.29), where *j* takes the values of each id from the neighbors list N_i.

$$AC_i = \frac{\sum_j IC_j}{|N_i|} \tag{4.29}$$

AC is added as an input to the FLS that is employed in the computation of *CHE*. Consequently there are few modifications at the level of the FLS that are described next.

Fuzzifier is having now an additional crisp input, *AC*, to transform in the corresponding Fuzzy degree of membership based on a defined membership function. Thus a new membership function is defined in the *Knowledge Base* that corresponds to *AC*. Similar to the other inputs this is trapezoidal and it is described by eq. (4.21) and eq. (4.30).

$\mu(AC)$ = *{(a, b, c, d)| a, b, c, d are the coefficients for AC's Fuzzy sets:* Low ($\mathcal{F}_{\mathcal{AC}}{}^{\mathcal{L}}$), Medium ($\mathcal{F}_{\mathcal{AC}}{}^{\mathcal{M}}$), High ($\mathcal{F}_{\mathcal{AC}}{}^{\mathcal{H}}$)*} = {(-∞, 0, 0.2, 0.4), (0.2, 0.4, 0.6, 0.8), (0.6, 0.8,1, ∞)}* (4.30)

Inference Engine uses a modified *Rule Base* in order to map the input fuzzified values to the output Fuzzy set described by the output membership function. Only the rules modified or added as compared to the baseline version are listed in TABLE 4.6.

TABLE 4.6. ALTERATIONS TO THE RULE BASE

Rule	AD	AV	N	AC	CHE
1	Low	Low	Low	Low	Very Low
1	Low	Low	Low	Medium	Low
1	Low	Low	Low	High	Medium
2	Low	Low	Medium	Medium	High
2	Low	Low	Medium	High	Very High
4	Low	Medium	Low	Low	Very Low
4	Low	Medium	Low	Medium	Low
4	Low	Medium	Low	High	Medium
5	Low	Medium	Medium	Medium	High
5	Low	Medium	Medium	High	Very High
9	Low	High	High	Medium	Medium
9	Low	High	High	High	High
10	Medium	Low	Low	Low	Very Low
10	Medium	Low	Low	Medium	Low
10	Medium	Low	Low	Medium	Medium
11	Medium	Low	Medium	Medium	High
11	Medium	Low	Medium	High	Very High
13	Medium	Medium	Low	Low	Very Low
13	Medium	Medium	Low	Medium	Low
13	Medium	Medium	Low	High	Medium
14	Medium	Medium	Medium	Low	Medium
14	Medium	Medium	Medium	Medium	High
14	Medium	Medium	Medium	High	High
15	Medium	Medium	High	Medium	High
15	Medium	Medium	High	High	Very High
17	Medium	High	Medium	Low	Very Low
17	Medium	High	Medium	Medium	Low
17	Medium	High	Medium	High	Medium

4.5. Chapter Summary

This chapter presents the overall architecture and network model of the Intelligent Advisory Solution for Bicycle Eco-riding and Eco-routing over vehicular networks that is proposed in this thesis. This solution encompasses three major components that each can be considered independent and constitutes a separate contribution of this thesis. The architecture, functional principles and algorithms are presented for each of these contributions that are summarized below:

- SAECy, a novel vehicular communications-based speed advisory system dedicated to electric bicycles. The solution subscribes to the class of GLOSA systems based on the traffic light to vehicle communications (i.e. I2V communication) and is the first GLOSA system dedicated to electric bicycles. The proposed solution recommends strategic riding (i.e. the appropriate speed) when bicycles are approaching an intersection to avoid high power consumption scenarios. Moreover, the approach also includes an innovative FL-based wind-aware speed adaptation policy as among all the other vehicles, bicycles are mostly affected by the wind. The benefits of the solution translate not only in energy-efficiency, but also in an increased user experience, as the waiting times at traffic lights are reduced or even avoided.

- eWARPE represents a step forward for the cycling route planners, going beyond planning the route itself (how to get from point A to point B). eWARPE is planning the optimal departure time for the route: when to leave from point A towards point B on the previously planned route. The solution makes use of the weather information in order to recommend the optimal departure time that allows the cyclist to avoid the adverse weather conditions and to increase the energy savings of the electric bicycle.
- FuzzC-VANET is a general FL-based CH-based clustering scheme over VANETs, the first to employ FL as the main decision tool in choosing the CH. Creating a clustered network model, this scheme solves some of the main issues of VANET, namely scalability and stability issues. As a consequence of this, the V2V communications in the network will have an increased throughput as it will be seen in the next chapters. Solving the scalability and stability issues, the clustering scheme creates a base for MAC, routing protocols and information dissemination. In the latter case, it is known that clusters contain local-based and highly up-to date information. In disseminating weather information, these two characteristics are highly important and in this context is to be mentioned that there are quite a few applications that are based on VANETs capacity of providing weather information, one of this being our previously proposed SAECy. An instantiation of FuzzC-VANET is also proposed, under the name of user-oriented FL-based clustering scheme that is based on a hybrid VANET architecture.

Chapter 5

PERFORMANCE EVALUATION

This chapter presents the performance evaluation of the proposed solutions: SAECy, eWARPE and FuzzC-VANET. The performance evaluation is done using multiple assessment techniques: experimental testing based on a real test-bed, interviews and online questionnaire to measure the need for the solution proposed and the interest of users and simulations based on highly realistic scenarios.

5.1. SAECy's Performance Evaluation

This section presents the performance evaluation of the proposed speed advisory solution, SAECy, which is assessed both through experiments using a real-life test-bed and via simulations, using realistic scenarios. For validation purposes, the scenarios used for experimental testing are also implemented in the simulation environment used for the assessment. Comparable results are obtained, thus leading to the validation of our simulation model. However, in order to perform extensive testing, more complex scenarios are also tested via simulation.

5.1.1. Experimental Testing and Simulation Model Validation

5.1.1.1. Experimental Test-bed Description

This section presents the experimental test-bed, as captured in Figure 5.1. As it can be seen from Figure 5.1, the test-bed consists of an electric bicycle enhanced with additional equipment.

The main components of the electric bicycle are: a battery and an electric motor. The additional equipment consists of a meter for measuring the power consumption, a speedometer, a GPS device and a video recorder for monitoring the power meter. Moreover, the test-bed also includes the cyclist smartphone that is supposed to support V2X communications capabilities and has deployed SAECy.

The battery of the electric bicycle is a Lihium Ion battery with the following characteristics: 10Ah capacity, nominal voltage of 36V, charging time ~ 6h, full charge capacity ~ 300Wh and weight of 5kg. The claimed battery range is of about 36-40km (20 miles), but in real life testing the range was measured to be around 25-30km, approximately 1h – 1h 20 min autonomy.

The electric motor, mounted on the front wheel, has a wired connection with the battery and the pedal so that the motor is engaged by applying pressure to the pedal. The bicycle subscribes to the category of electric bicycles with assistance at start [240].

A Garmin Edge 500[22] bike computer incorporates the functionalities of both speedometer and GPS-based location device. Garmin Edge 500 features a high-sensitivity GPS receiver that allows for accurate positioning and also for an accurate output of instantaneous speed. The power meter was connected according to the requirements to the battery and the electric motor. The outputs of the power meter are instantaneous power, voltage, current and the total power consumption per hour (i.e. energy consumption in Wh). A video recorder is used to monitor both the functionality of the meter and the instantaneous power output. The recorded values were then used in the result analysis. This method was preferred, as the serial output-based logging designed in the absentia of a built-in recording functionality is very sensitive to the motion.

Other variables that affect the power consumption are shortly described next. The tire pressure, important parameter for a better rolling, was correspondingly adapted to the city roads scenario and to the weight of the cyclist (80kg), its value being 100psi. The total weight of the bicycle with all the equipment was 25 kg. Moreover, the wind speed during the tests was negligible (it was checked using an electronic anemometer) and the tests were performed on a relative straight road with normal roughness, in excellent weather conditions.

5.1.1.2. Scenarios Description

The testing scenarios considered an electric bicycle that is approaching a signaled intersection. If the current speed is maintained, the cyclist will not be able to cross the intersection without stopping, and will be enforced to stop at the traffic light. There are two possible scenarios

[22] Garmin Systems: https://buy.garmin.com/en-US/US/into-sports/cycling/edge-500/prod36728.html

in which the proposed solution takes action when a bicycle is approaching an intersection and these are considered in these tests.

In the first scenario (Scenario 1), when the distance between bicycle and traffic light is equal to a predefined distance *d*, the color of the traffic light is *green*. The time to traffic light changes (*timeToChange*) to *red* is 70s and the duration of *red* is 50s. This scenario corresponds to the condition expressed in the SAECy's algorithms (described in Chapter 4) as: *signalState = green && timeToChange < t.*

In the second scenario (Scenario 2), when the distance between bicycle and traffic light is equal to the predefined distance *d*, the color of the traffic light is *red*. The time to traffic light changes (*timeToChange*) to *green* is 50s and the duration of *green* is 70s. In the tests performed, the role of the traffic light is taken by a timer. This scenario corresponds to the condition expressed in the SAECy's algorithms as: *signalState = red && timeToChange > t.*

Figure 5.1. Experimental Test-bed

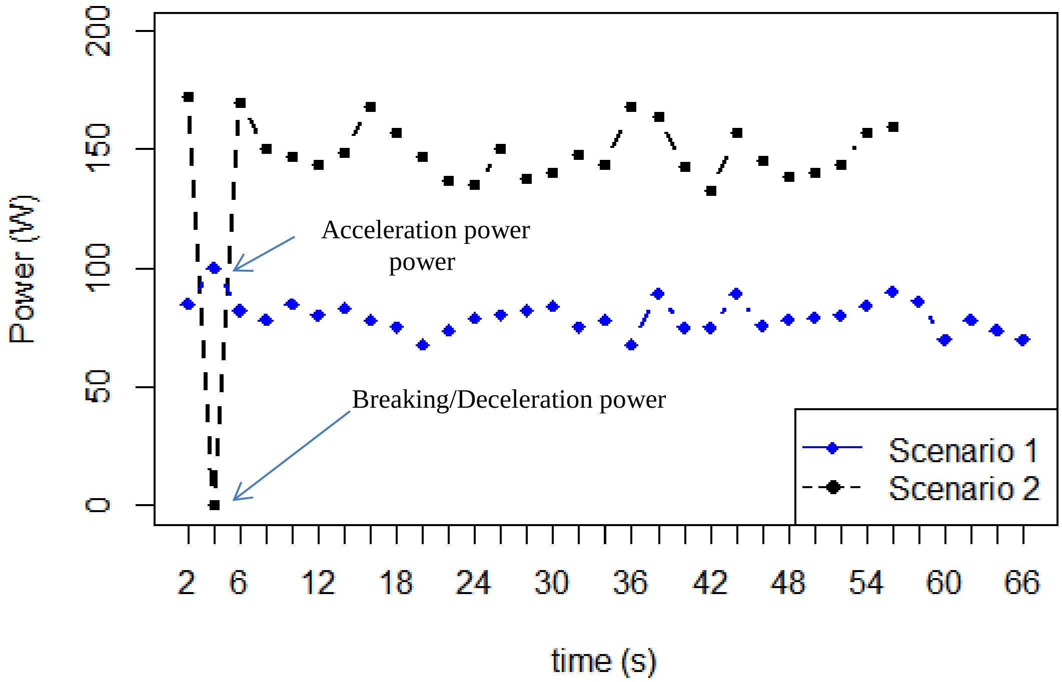


Figure 5.2. Power consumption for bicycle equipped with speed advisory system – test-bed results

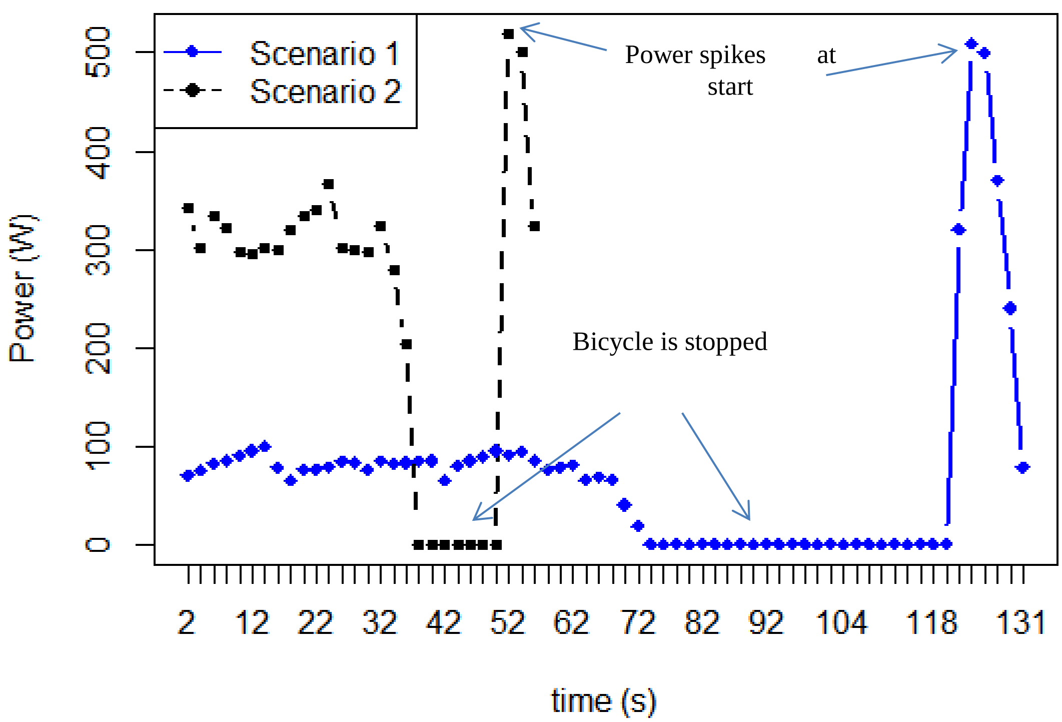


Figure 5.3. Power consumption for non-equipped bicycle – test-bed results

Power consumption is recorded for each of the two above scenarios in two cases: in the first case the bicycle is equipped with SAECy, the proposed speed advisory system, while in the second case the bicycle is not equipped. The tests were done five times for each testing scenario in each case. In the first case for each testing scenario, the rider makes use of the speed advisory system and consequently avoids stopping at the traffic light. It is assumed that the information regarding traffic light phasing and positioning is received via V2X communications when the distance between bicycle and traffic light is equal to *d*. The value of *d* is set in the test scenarios to 200m, typical value for transmission ranges in 802.11p. The rider ensures that the current speed when receiving the recommended speed is 10km/h for Scenario1, and 20km/h for Scenario 2. Power consumption is monitored on the distance *d*.

In the second case for each testing scenario, the electric bicycle is not equipped with the speed advisory system. The electric bicycle is in motion having the corresponding speed for each of the two scenarios: 10km/h for Scenario 1 and 20km/h for Scenario 2. This time, the rider does not receive any information while traveling the same distance (*d*) towards the traffic light and keeps the speed constant. When it gets at the traffic light, the rider stops. Power consumption is monitored for the same distance *d*.

5.1.1.3. Results Analysis – Experimental vs Simulation Results

In both testing scenarios, significant benefits in terms of power consumption were obtained when using the proposed speed advisory system, SAECy. In the first testing scenario, Scenario 1, SAECy reduces the energy consumption by 46% when compared to the classic case of a non-equipped bicycle, while in Scenario 2 the energy consumption is reduced by 44%. These results were obtained using the previously described test-bed.

In Figure 5.2 it can be seen that the electric bicycle used in the tests is confirmed to be included in the category of electric bicycles with assistance at start, as power spikes are noticeable when starting the bicycle. The power consumption model used in the implementation of the speed advisory system does not consider these power spikes, as it is a generic model suitable for all types of bicycles. However, these power spikes have no influence in deciding the recommended speed as they appear only when starting the bicycle and our main goal is to avoid when possible stopping at the traffic light.

Same scenarios, Scenario 1 and Scenario2, were implemented in the simulation model that is next described. However, in order to correspond to the real tests performed with the test-bed, the communication modelling between traffic light and bicycle was removed from the simulation model for these 2 scenarios, being considered that the bicycle receives the message from the traffic light exactly at distance *d* = 200m from traffic light. Moreover, the starting power spikes were introduced in computation in the simulation model in order to obtain a fair comparison between the test-bed results and simulation results. Comparable results were obtained, resulting in benefits of 45% for Scenario 1, and 41% for Scenario 2 in terms of energy consumption.

Assuming that the electric bicycle used in the test-bed has no assistance at start, the power spikes from start were removed, and the simulation model was left unaltered (without the power spikes previously introduced based on the experimental results). Again comparable results were obtained in terms of energy consumption reduction for the two testing environments, test-bed and simulation as can be seen in TABLE 5.1 that summarizes all the results related to Scenario 1 and Scenario 2.

TABLE 5.1. RESULTS SUMMARY EXPERIMENTAL VS SIMULATION TESTING

Testing environment		**Energy consumption reduction – equipped bicycles vs non-equipped bicycles**	
		Scenario 1	Scenario 2
Electric Bicycle with assistance at start	**Test-bed**	46%	44%
	Simulation	45%	41%
Electric Bicycle without assistance at start	**Test-bed**	24%	32%
	Simulation	19%	28%

5.1.2. Simulation-based Testing

This section presents SAECy's performance evaluation via simulations. First the simulation model, the testing scenarios and assessment methodology are introduced. The results and their in-depth analysis are presented at the end.

5.1.2.1. Simulation Model

A) Simulation Platform

The simulations are performed using iTETRIS[23] simulation platform, an open-source simulator designed in the context of a European FP7 project. The platform couples the traffic simulation capabilities of SUMO[24] and the network communication capabilities of NS3[25]. This simulation platform subscribes to the class of two-way communication simulators, the most desirable class of simulators in the context of vehicular networks as it allows feedback from the mobility simulator to the network simulator and vice-versa. Other types of simulators used in vehicular networks allow only for one-way communications (e.g. SWANS++[26] [86]) which may affect the quality of the simulations, especially in terms of their realistic character. This was the main reason of selecting iTETRIS as a simulation platform, but other factors contributed to this decision, factors that individualize iTETRIS in the context of the other two-way communication simulators. These are listed below:

---

[23] iTETRIS website: www.ict-itetris.eu
[24] SUMO website: www.sumo-sim.org
[25] NS3 official website: www.nsnam.org/
[26] SWANS++ project website, http://sourceforge.net/projects/straw/

- iTETRIS is highly scalable mostly due to the usage of NS3 simulator known for its good scalability when compared to NS2 or other network simulators. Simulators before iTETRIS were coupling NS2 simulator (e.g. TraNS[27]) and SUMO. Other simulators couples SUMO with other network simulators such as OMNET++ (e.g. Veins[28]) that have lower support in terms of protocols when compared to NS3
- It is based on SUMO, an open-source traffic simulator highly supported and continuously improved
- It has a flexible architecture allowing for de-coupling the applications' code from the main code of the simulation platform.

The simulation platform has been extended to support the simulation of the solutions proposed in this thesis. There was a need for a Fuzzy Logic toolkit for implementing the Fuzzy Logic systems that are integrated in these solutions. For this purpose, Octave FL Toolkit was used. Figure 5.4 presents the architecture of the extended simulation platform iTETRIS + Octave FL toolkit, with the proposed solutions integrated. Our contributions are marked in orange. As it can be seen in the figure, there are four core architectural components of iTETRIS platform: SUMO, NS3, iCS and Applications. The iCS component synchronizes the two components that are represented by the traffic simulator, SUMO, and the network simulator, NS3. The Applications component is the component where the entry point of every designed application by the simulation platform's users is placed. For the communication purposes between Octave and iCS, a new module was designed that is named Octave <–> iCS Communication Module in Figure 5.4.

[27] TraNS simulator website, http://lca.epfl.ch/projects/trans/

[28] Veins simulator website, http://veins.car2x.org/

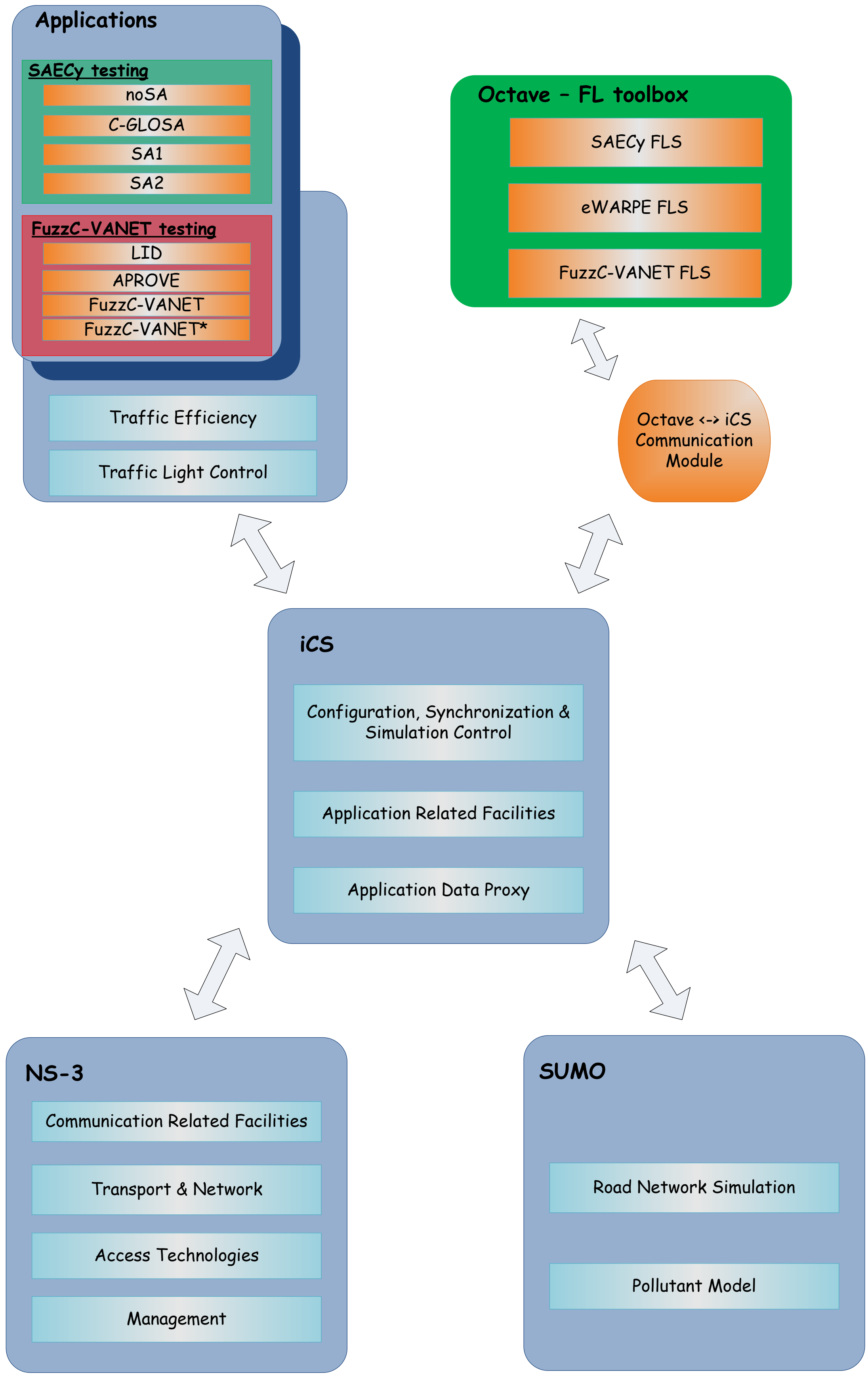


Figure 5.4. Simulation Platform

B) Simulation Settings

There were several parameters that needed to be considered when setting the simulation model and these mainly relates to the bicycle power consumption model. These parameters are: the frontal area *(A*), the drag coefficient (C_d), the wind speed (v_w), the air density (*D*) and the cumulate weight of the cyclist and the bicycle (*m*). The last two parameters were set to correspond to the real tests performed, m = 105 kg (cyclist 80kg, bicycle 25kg), while D = 1.247 kg/m^3, the sea value level. Note that unless there is a considerable altitude difference this is the commonly used value [21]. In the TABLE 5.2. are also listed the values for 1500 m and 3000 m altitudes.

The values for the frontal areas vary between 0.32 and 0.7m^2 and depend mainly on the cyclist position on the bicycle, but also upon the cyclist weight and height. For instance an upright cyclist, which is the typical position for cycling in the city is usually considered to be 0.7 m^2 [21], [244]. The drag coefficient interrelates to the frontal area parameter, for upright cyclist is set to 1. In our simulation settings we considered these values as the simulation scenarios are city scenarios. Moreover, in the real tests performed the cyclist was tall and had a considerable weight. Other possible values associated to these parameters are presented in TABLE 5.2.

TABLE 5.2. SIMULATION PARMETERS – LIST OF POSSIBLE VALUES

Parameter	**Values**
Frontal area (*A*)	typical values between 0.32 and 0.7[21], [244]; 0.7 for upright cyclist [244]
Drag coefficient (C_d)	recumbent bicyclist 0.77, upright cyclist 1 [21]
Rolling coefficient (R_c)	values between 0.0016-0.0066 [245] 0.004 for smooth asphalt roads the most used value [21][245]
Air density (*D*)	Sea level value 1.227kg/m^3 1500m value 1.056 kg/m^3 3000m value 0.905 kg/m^3

The wind speed was varied during the tests between 0 to 10m/s, the latter being considered the maximum wind speed that is safe for cycling, as explained in Chapter 4.

Rolling coefficient is a parameter that depends on the type of the surface. There are several values that can be considered, the most common one is 0.004, the common value for a smooth asphalt road [21]. This value can vary depending on the type of roads between 0.0016-0.0066 [245].

Tests were performed for these values of rolling coefficient. The other simulation parameters were kept constant to the values chosen for the simulation model. The testing scenarios used for testing were the same scenarios used in the real testing that were entitled Scenario 1 and Scenario 2. The results are shown in TABLE 5.3. It can be seen that the energy consumption reduction do not vary excessively depending on this parameter, the variation is less than 10% for the two edges of the 0.0016-0.0066 interval, only in the case of Scenario 1 for the bicycles with assistance at start being 14%. However, values less than 0.003 characterizes very smooth surfaces that are not the case for the real roads. Considering this and the fact that 0.004 is quite in the middle of the interval, that is the most commonly used value, but also the similar results with the real tests, this value was chosen in the simulation model.

Two other parameters needed to be set that relate to the SAECy's algorithms. These are *maxSpeed* and *maxPower*. The first parameter was set to the safety value of 6.95m/s according to [21]. The second one, *maxPower* was set to 400W which is the maximum instantaneous power while riding provided by most of the electric bicycles [21]. This was also the value that corresponded to the test-bed.

IEEE 802.11p was used to model the communication between the infrastructure (traffic light) and the smartphone attached to the bicycle – transmission range: 250m. Each Traffic Light Controller broadcasts the information message every second, value chosen as in [206], [208]. The message size varies depending on the number of possible movements at the intersection, in the simulation performed the message size was between 31 bytes to 48 bytes.

TABLE 5.3. RESULTS SUMMARY FOR SCENARIO 1 AND 2 WHEN VARYING ROLLING COEFFICIENT

Rolling Coefficient	**Energy consumption reduction – equipped bicycles vs non-equipped bicycles (bicycles with assistance at start)**		**Energy consumption reduction – equipped bicycles vs non-equipped bicycles (bicycles without assistance at start)**	
	Scenario 1	**Scenario 2**	**Scenario 1**	**Scenario 2**
0.0016	53%	46.5%	23%	32%
0.002	52%	45.5%	22%	31%
0.003	48%	43%	20%	29%
0.004	45%	41%	19%	28%
0.005	42%	39%	18%	26%
0.006	40%	37.5%	17%	24.5%
0.0066	39%	36.5%	16.5%	24%

5.1.2.2. Scenarios Description

This section presents the scenarios used to evaluate the performance of the speed advisory system. Basically, there are three testing scenarios, each represented by a different route with a different topology and different numbers of traffic lights on the way with phases between 55s and 85s. These are real routes established on the real map of Dublin, Ireland. The destination of each of the routes is Dublin City University (DCU, GPS coordinates: -6.26263, 53.38507) as it can be seen on the maps. Figure 5.5 and Figure 5.6 display these routes.

First route (Route 1) starts at [-6.26263, 53.38507], has a simple, quite straight topology and has 6 traffic lights on the way (Figure 5.5). This is a simple scenario as the traffic lights from a straight route tend to be synchronized, thus for a relative constant speed, the number of stops at traffic lights are relatively low.

Second route (Route 2) starts at [-6.28407, 53.40603] and has the most complex topology among the three routes, including more turns and having 9 traffic lights on the way (Figure 5.6).

Third route (Route 3) starts at [-6.25725, 53.40098] and has only 4 traffic lights on the way (Figure 5.5). Route 1 and Route 3 were chosen to be able to measure the benefits of the solution proposed for routes having less traffic lights and also for routes with simple topologies.

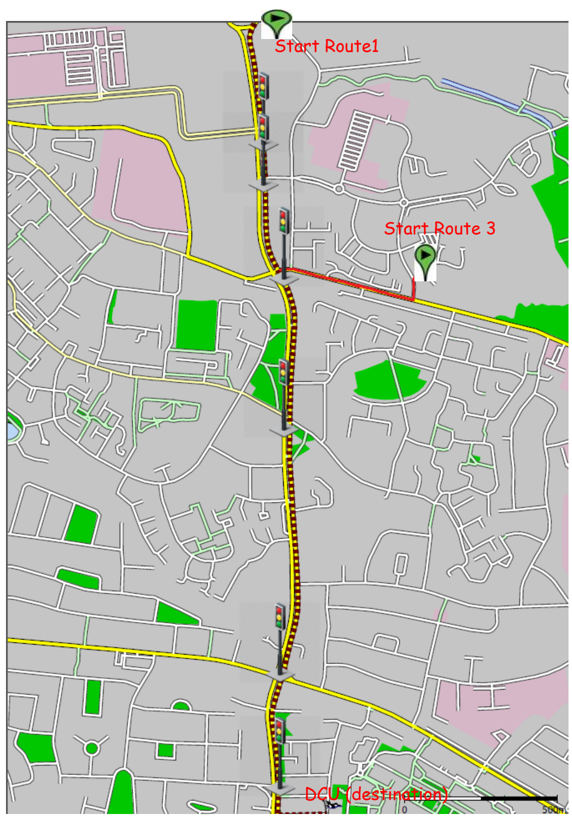


Figure 5.5. Route 1 and Route 3

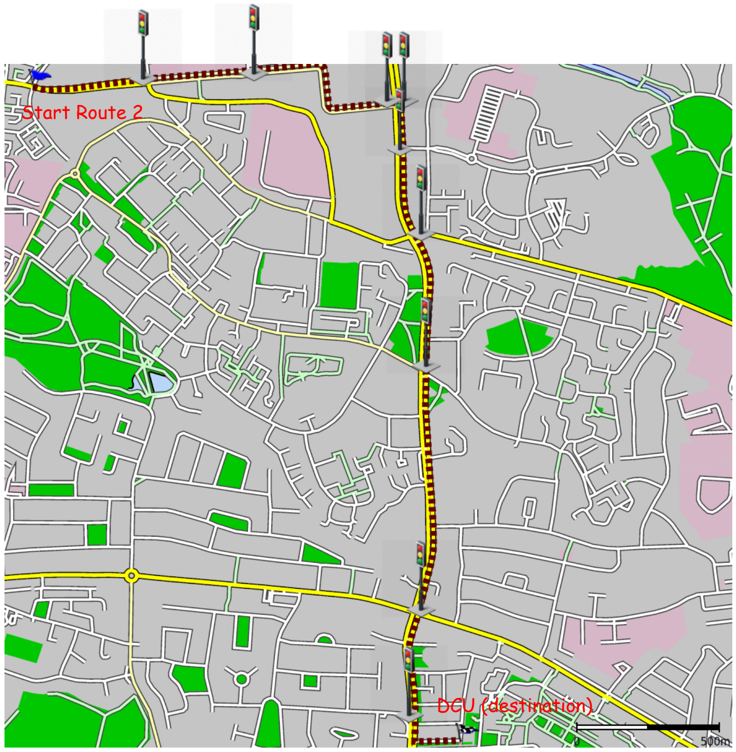


Figure 5.6. Route 2

5.1.2.3. Comparison-based Assessment

Two versions of SAECy are considered for performance assessment. The first version, SA1, implements the proposed approach having deployed the Green Light Optimal Speed Advisory function only which was described by Algorithm 4.1 in Chapter 4. The second version, SA2, has the Fuzzy Logic-based wind-aware speed adaptation policy added to the Green Light Optimal Speed Advisory function and encompasses the whole functionality provided by SAECy. This was described by Algorithm 4.3.

In the simulation model is also implemented a classic approach of a GLOSA system (C-GLOSA) proposed in [48]. The C-GLOSA approach was such implemented in order to correspond to the bicycle dynamics.

All these approaches are compared among themselves and also against the baseline which is represented by the common case when the bicycles are not equipped with any type of speed advisory system (noSA).

5.1.2.4. Results and Analysis

The proposed speed advisory system is evaluated in terms of energy consumption reduction and two comfort-related metrics: *number of stops* and *waiting times* at traffic lights cumulated over each route. In addition, the impact of the speed advisory system on the total travel time is analyzed, known as a highly important metric for assessing the quality of a travel [242], [243]. These metrics are studied against the variation of the wind speed from 0 to 10m/s.

For the energy consumption reduction metric, two sets of results were obtained for each of the three routes. In the first set of results, the power consumption model was left unaltered, as it is described in section 4.2.2.2.A. This model corresponds to an electric bicycle without assistance at start.

For the second set of results, we considered that the bicycle is with assistance at start and as the parameters used in computation are compliant to the conditions in which the tests with the test-bed were performed, we introduced in computation the power spikes experimentally determined. The other metrics are not affected by this modification in the power consumption model, thus they are the same for the bicycle with or without assistance at start.

The scenarios chosen allow for testing the impact of the number of traffic lights and different route topologies on the performance of SAECy.

A) Impact of the Traffic Lights Number and Topology

Energy consumption reduction metric evaluates the percentage of energy savings of the three approaches C-GLOSA, SA1 and SA2 against the baseline noSA allowing then for a comparison between them. Figure 5.7 – Figure 5.12 present the results in terms of this metric. Following a general analysis on these results it can be said that SA2 clearly outperforms the other two approaches, C-GLOSA and SA1, for all the routes, and the energy consumption reduction is more significant in the case of the bicycles with assistance at start. SA1 approach also outperforms C-GLOSA. Another observation that can be made is that the energy savings are more significant for Route 2, as this has the largest number of traffic lights and a more complex topology. A higher number of traffic lights and a more complex topology of the route determined an increased number of stops at the traffic lights along the way as can be seen in Figure 5.13 as compared to the other routes, Route 1 and Route 3. For Route 2, the energy consumption reduction for the bicycles without assistance at start can reach 15% for SA2 and 9% for SA1, while for bicycles with assistance at start the energy consumption reduction reaches 18% for SA2 and 13% for SA1.

It can be seen that there are some fluctuations in the plots that represent the energy consumption reduction. There are basically two reasons for these fluctuations. The first reason can be exemplified by the first fluctuation in the SA2 vs noSA curve corresponding to the change in wind speed from 0m/s to 1m/s (Figure 5.7 – Figure 5.12) and represents the increase in the energy consumption reduction due to bicycle's speed adaptation to the wind speed. The second reason of fluctuation represents the increase/decrease of the energy consumption reduction due to the numbers of stops avoided (e.g. the fluctuation of SA2 vs noSA or SA1 vs noSA curves in Figure 5.9).

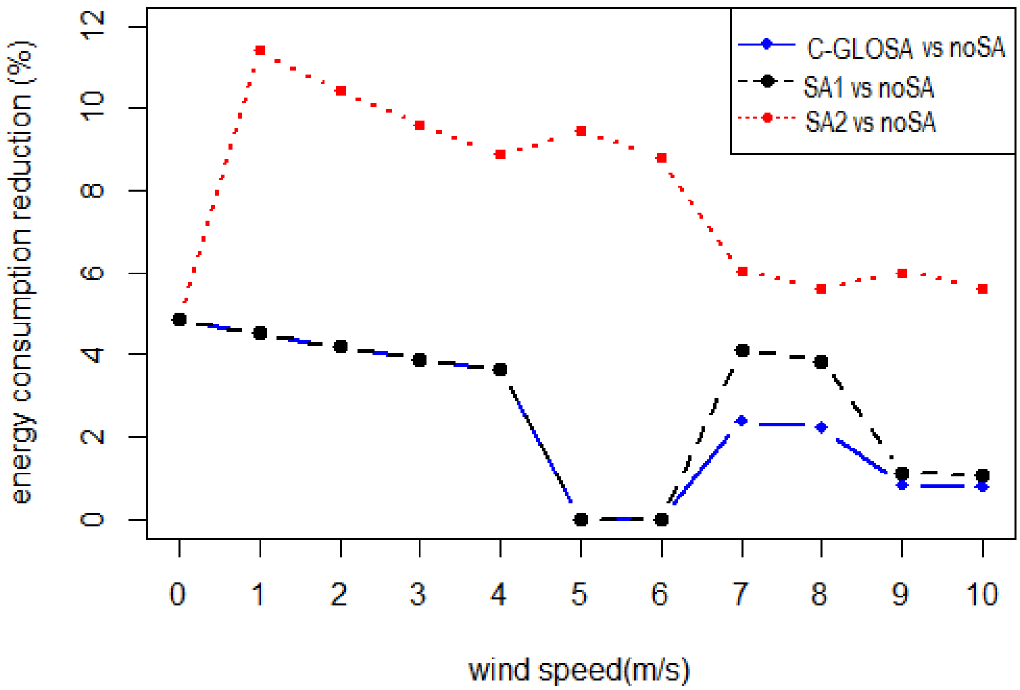


Figure 5.7. EC reduction – Route 1 (bicycle without assistance at start)

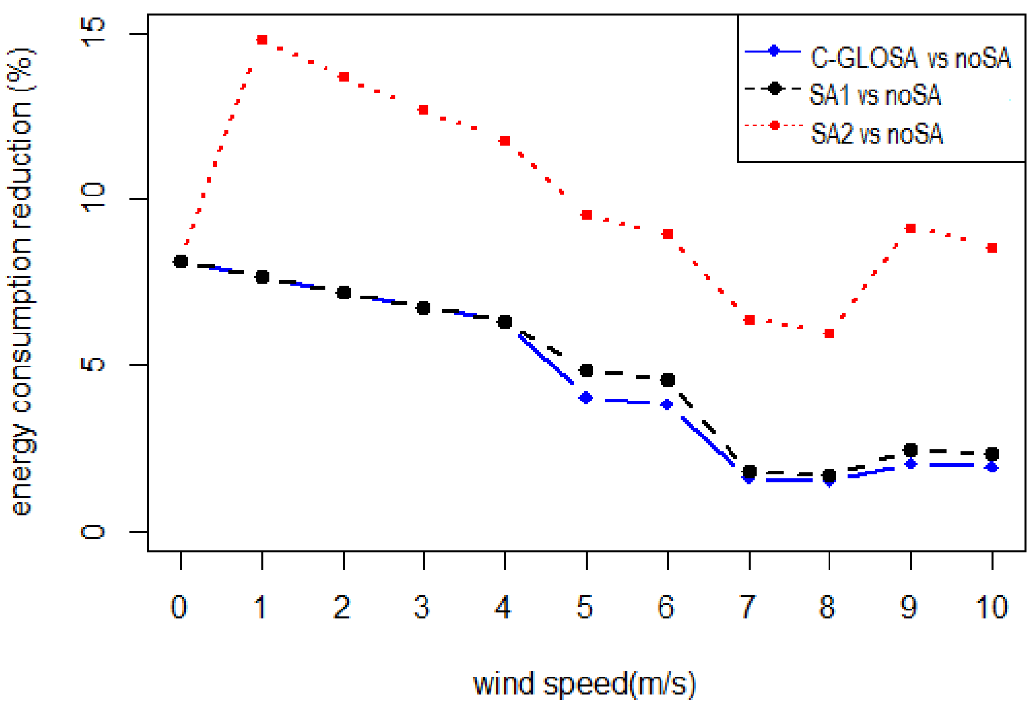


Figure 5.8. EC reduction – Route 2 (bicycle without assistance at start)

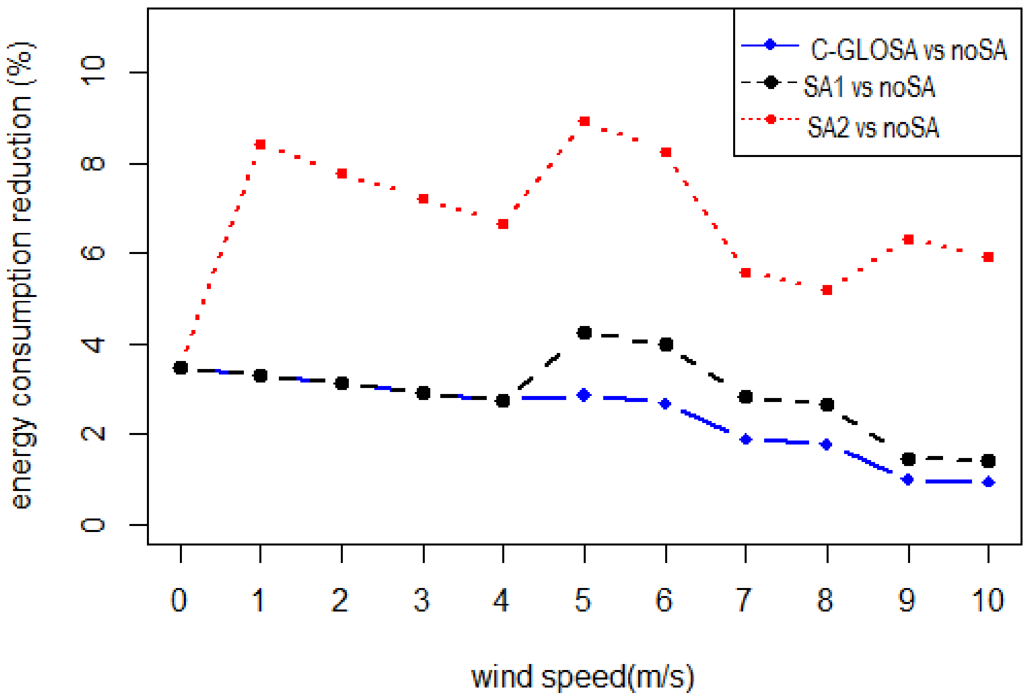


Figure 5.9. EC reduction – Route 3 (bicycle without assistance at start)

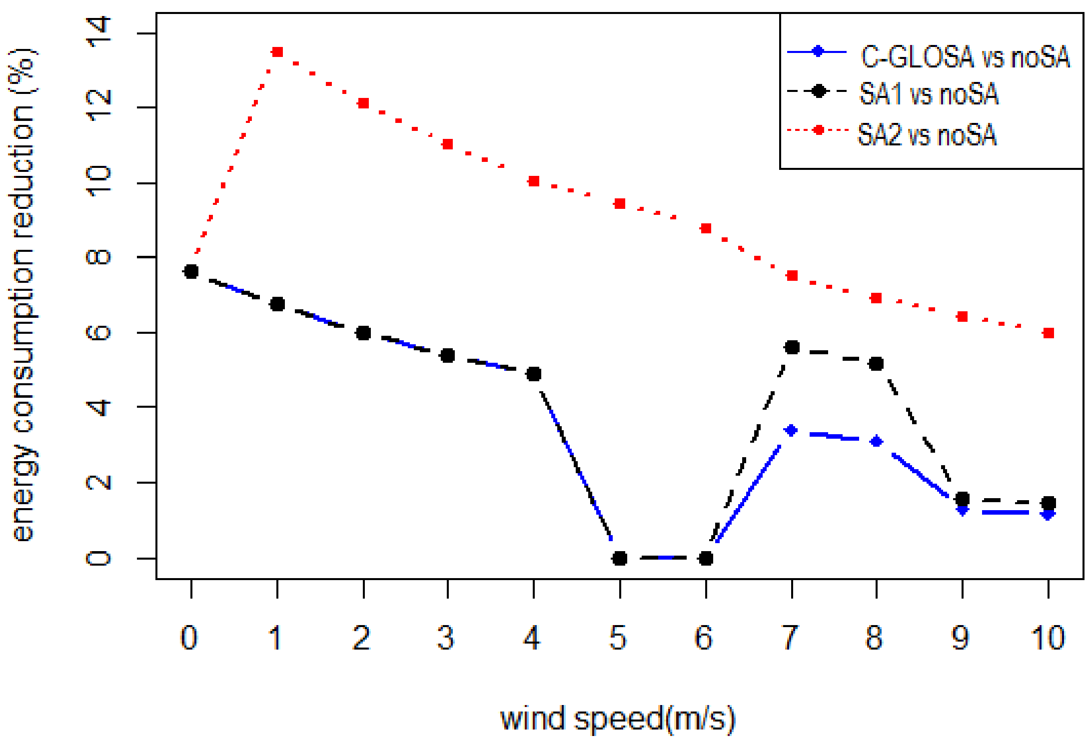


Figure 5.10. EC reduction – Route 1 (bicycle with assistance at start)

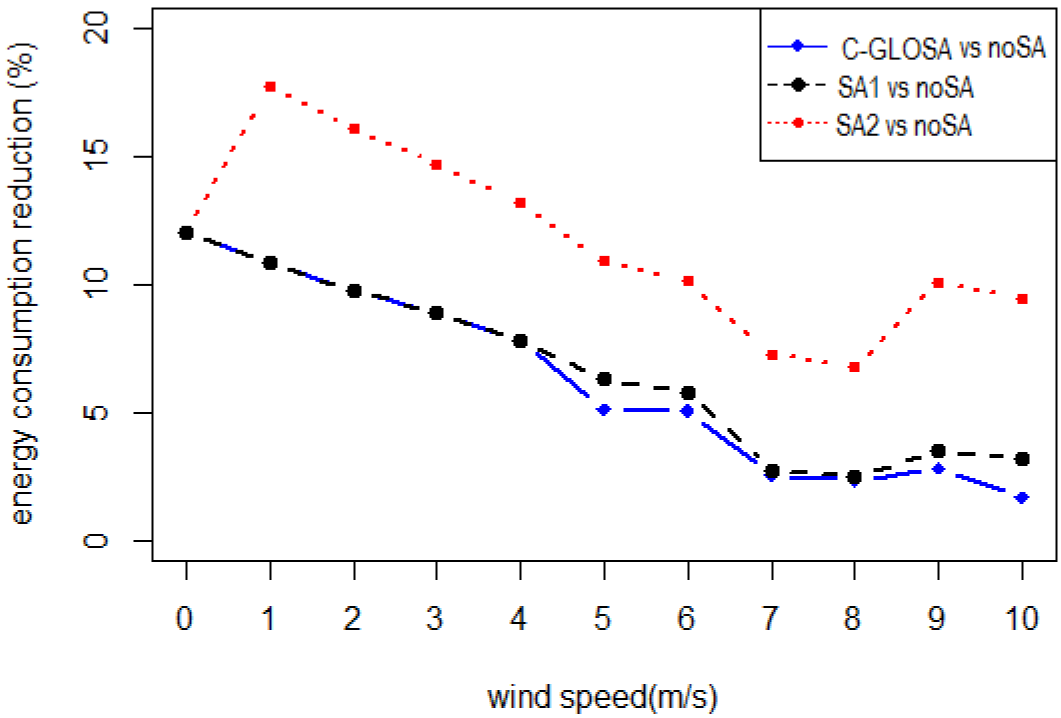


Figure 5.11. EC reduction – Route 2 (bicycle with assistance at start)

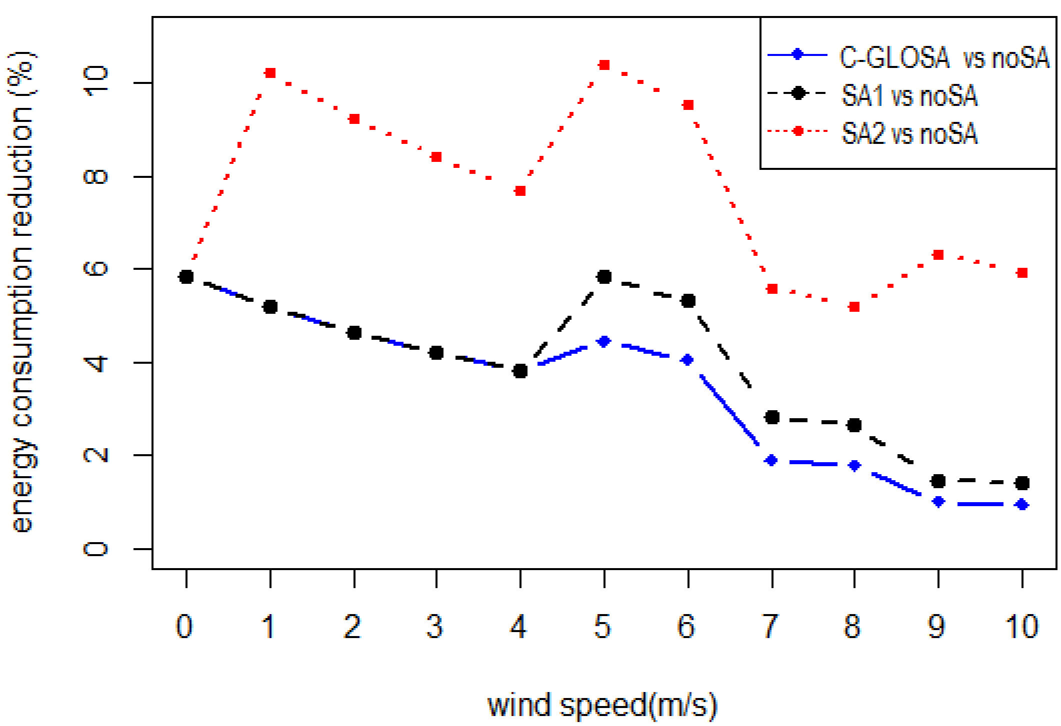


Figure 5.12. EC reduction – Route 3 (bicycle with assistance at start)

The number of stops to be avoided is presented in Figure 5.13 for non-equipped bicycles, and in Figure 5.14 for equipped bicycles. This number, in case of equipped bicycles, vary/fluctuate due to the fact that the average speed of the bicycle is also varying with the wind speed imposed by the limitations in power. This is one of the reasons of waiting times curves variation, that are presented in Figure 5.15 – Figure 5.17, in addition to the efficiency of the speed advisory systems of avoiding the stops at the intersections.

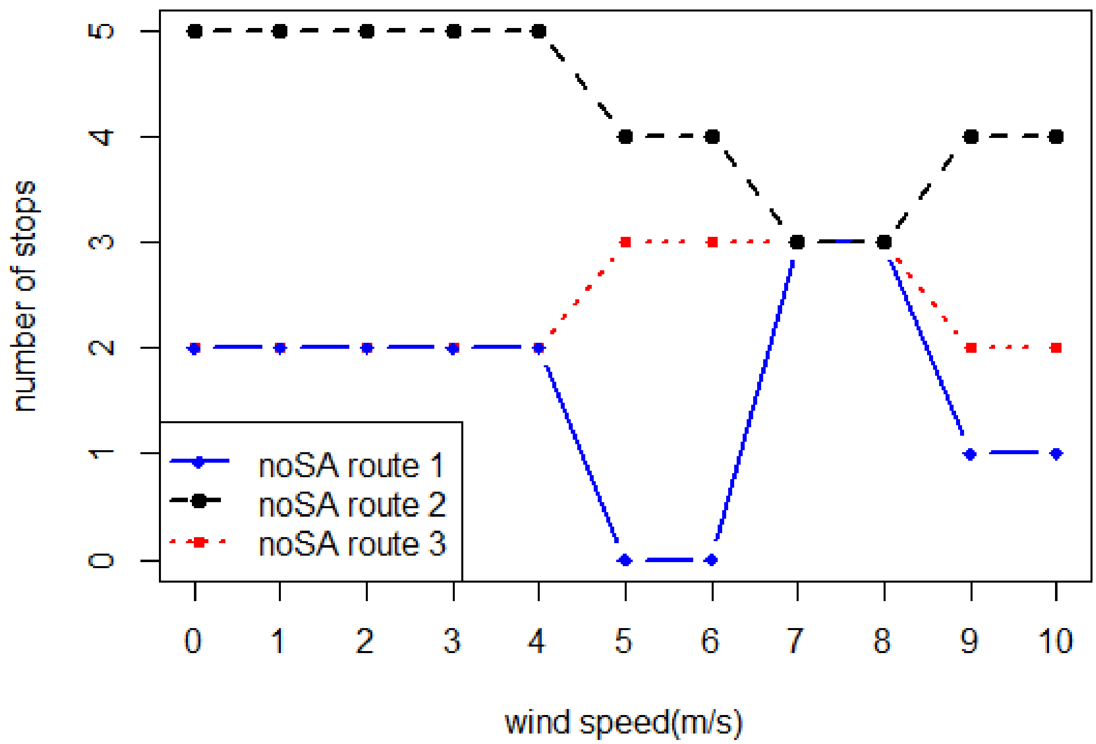


Figure 5.13. Number of stops for the non-equipped bicycle for the three routes

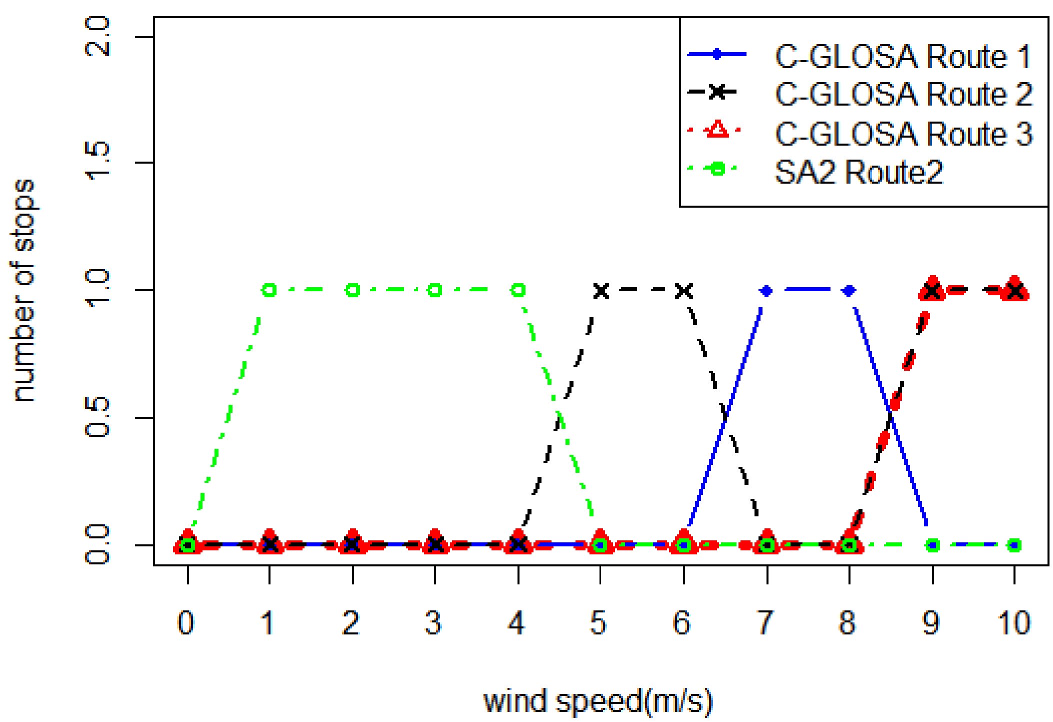


Figure 5.14. Number of stops for the equipped bicycle for the three routes

Regarding the number of stops and waiting times, the proposed speed advisory system implemented in both forms SA1 and SA2 reduces these to 0, respectively 0s with one exception. This exception is caused by the fact that 2 traffic lights on the Route 2 are much closer to each other than usual and there is not enough time to adapt the speed to avoid stopping at the second traffic light. However, the waiting time is very low of 3 s only (Figure 5.16). This situation can happen for any type of green light optimal speed advisory system such as the classic approach represented by C-GLOSA. It can be seen that C-GLOSA fails in a bigger proportion in reducing completely the number of stops at the traffic lights (Figure 5.13) and this is not caused by the positioning of the traffic lights. The failure is due to the fact that the advised speed to avoid stopping at the traffic light does not take into consideration the wind speed and recommends speeds that are not adapted to this important factor for the bicycles. Consequently, these speeds cannot be sustained by the bicycles and the bicycles end up stopping at the traffic lights. Additional more power is consumed and higher waiting times are introduced (Figure 5.15 – Figure 5.17).

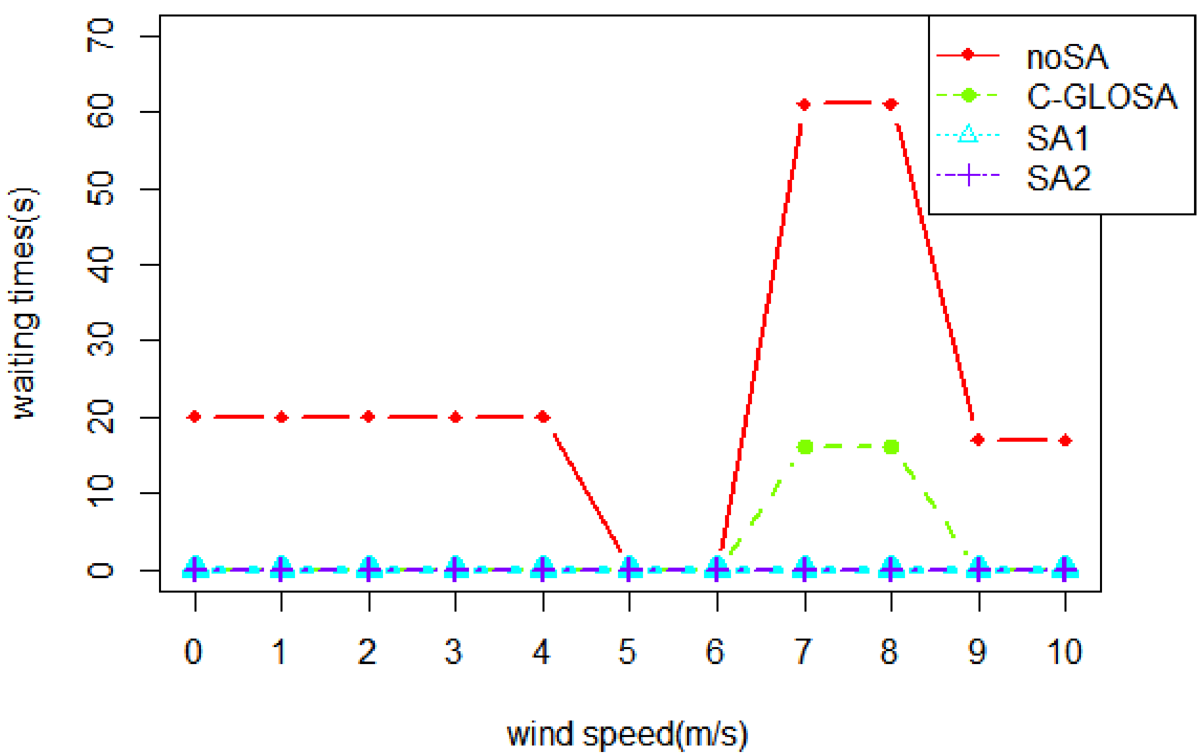


Figure 5.15. Waiting times – Route 1

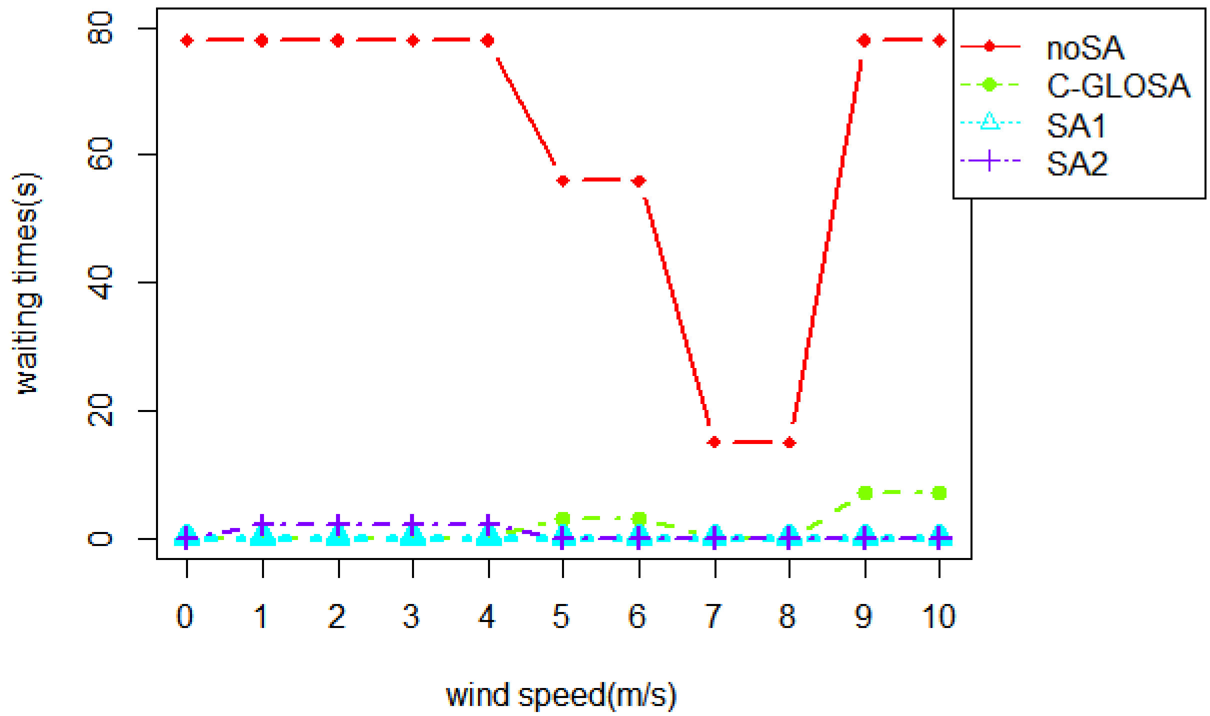


Figure 5.16. Waiting times – Route 2

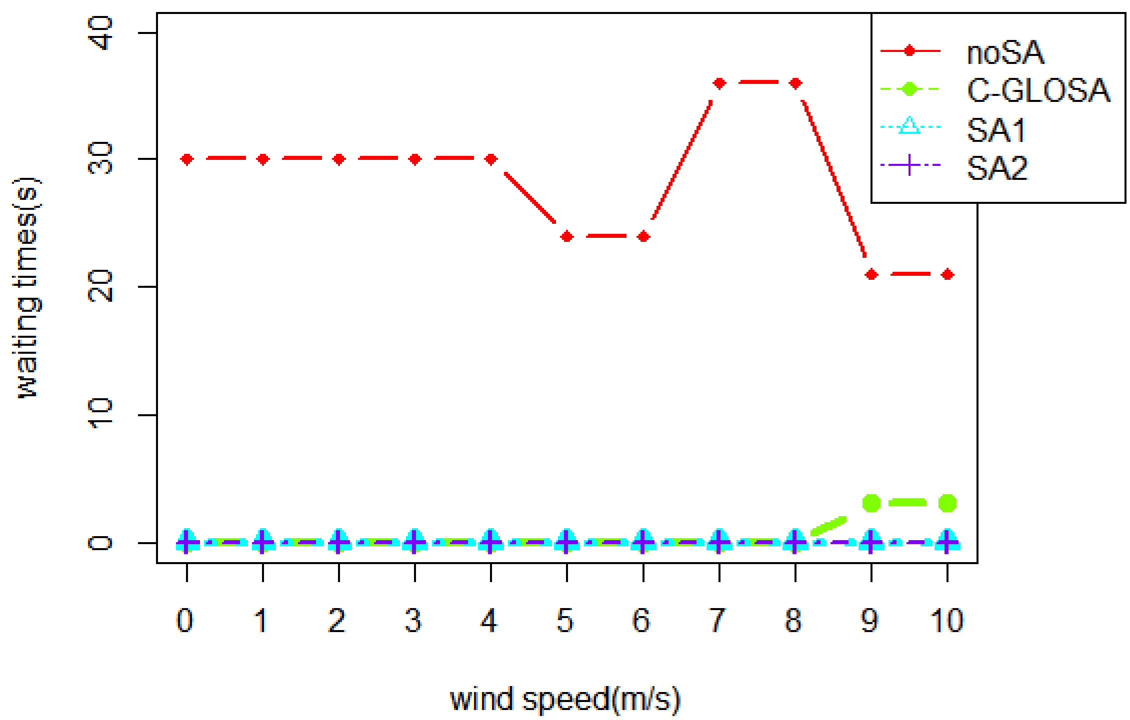


Figure 5.17. Waiting times – Route 3

The impact of the speed advisory systems on the total travel time is negligible as proven to be of minor extent in most of the cases illustrated in TABLE 5.4, TABLE 5.5 and TABLE 5.6. However, in few cases, SA2 causes some delays SA2 causes slight delays at some wind speeds which is acceptable as it recommends a decreased speed than the one that can be sustained by the bicycle in order to decrease the energy consumption. Some of the largest delays in the total travel time imposed by SA2 are for instance 132s for Route 2 (TABLE 5.5), and 87s for Route 3 (TABLE 5.6) when the wind speed is 9-10m/s. This means that SA2 adds approximately 2 minutes to a total travel time of 21 minutes in the first case, and 1.5 minutes to a total travel time of approximately 14 minutes in the second case.

TABLE 5.4. TOTAL TRAVEL TIME – ROUTE 1

Wind speed (m/s)	**Total travel time (s)**			
	noSA	**C-GLOSA**	**SA1**	**SA2**
0	469	467	467	467
1	469	467	467	474
2	469	467	467	474
3	469	467	467	474
4	469	467	467	474
5	505	505	505	572
6	505	505	505	572
7	664	664	663	669
8	664	664	663	669
9	722	720	722	788
10	722	720	722	788

TABLE 5.5. TOTAL TRAVEL TIME – ROUTE 2

Wind speed (m/s)	**Total travel time (s)**			
	noSA	**C-GLOSA**	**SA1**	**SA2**
0	851	850	850	850
1	851	850	850	848
2	851	850	850	848
3	851	850	850	848
4	851	850	850	848
5	945	940	938	1011
6	945	940	938	1011
7	1020	1022	1022	1105
8	1020	1022	1022	1105
9	1278	1278	1278	1410
10	1278	1278	1278	1410

B) Impact of the Cyclist Weight on the Energy Consumption

Among the variables that influence most the energy consumption of the bicycle is the cyclist weight. This was set to a predefined value in the previous tests. The other variables were set to realistic values and their influence was studied in the Simulation Settings section. Consequently, we investigated to see the influence of this parameter in the context of SAECy. In the previous tests, the cyclist weight was set to 80kg, same weight as that of the cyclist that helped perform the real tests. The same tests were performed varying the cyclist weight in the interval [60kg, 90kg]. Figure 5.18 – Figure 5.20 show the impact of the cyclist weight on the energy consumption reduction metric for SA2, while Figure 5.21 – Figure 5.23 show the impact for SA1. It can be seen that the solution is not sensitive to the cyclist weight, the benefits brought by SAECy are independent of this parameter.

TABLE 5.6. TOTAL TRAVEL TIME – ROUTE 3

Wind speed (m/s)	Total travel time (s)			
	noSA	C-GLOSA	SA1	SA2
0	554	553	553	553
1	554	553	553	569
2	554	553	553	569
3	554	553	553	569
4	554	553	553	569
5	654	653	653	669
6	654	653	653	669
7	754	754	753	776
8	754	754	753	776
9	857	857	856	944
10	857	857	856	944

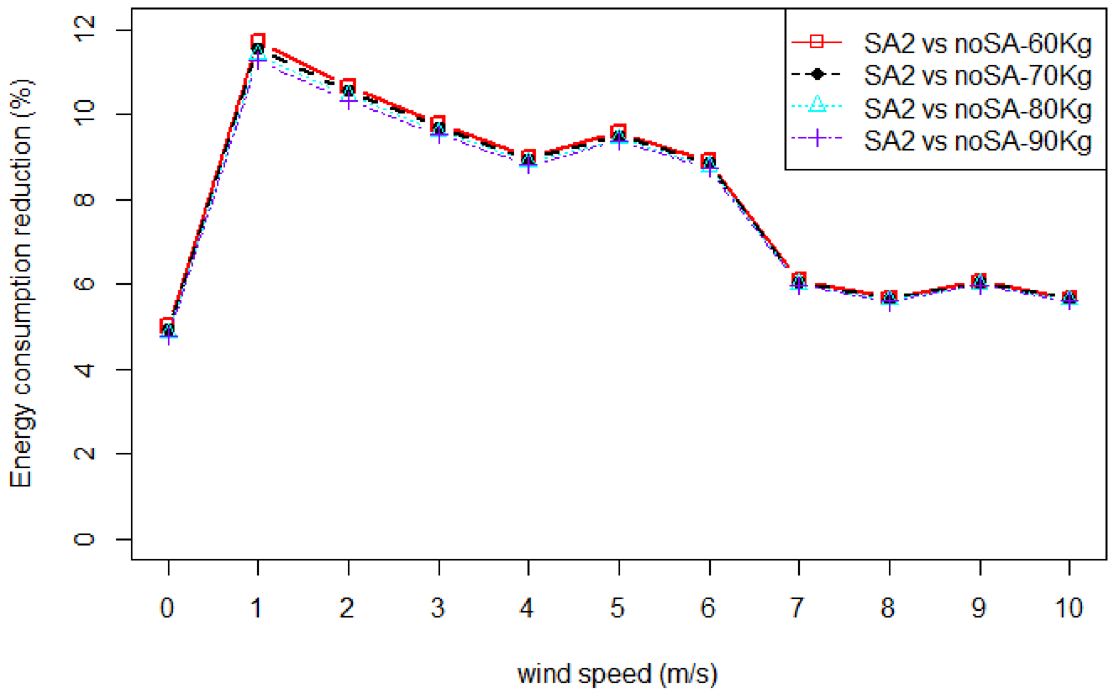


Figure 5.18. EC reduction for SA2 – Route 1

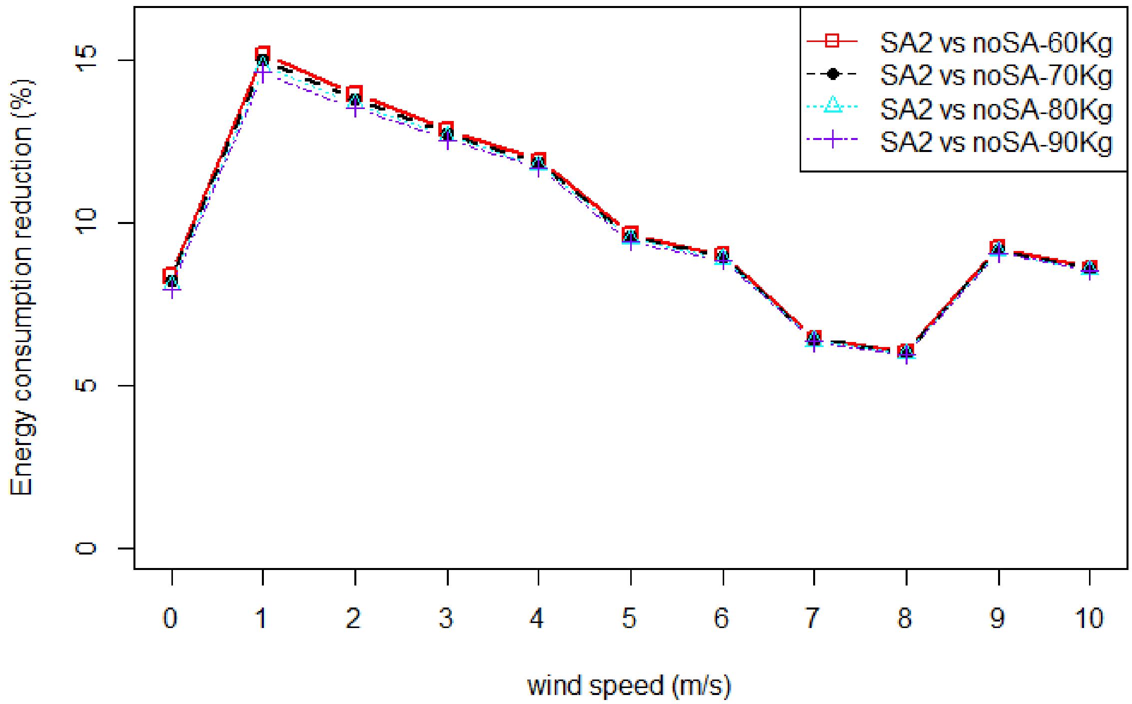


Figure 5.19. EC reduction for SA2 – Route 2

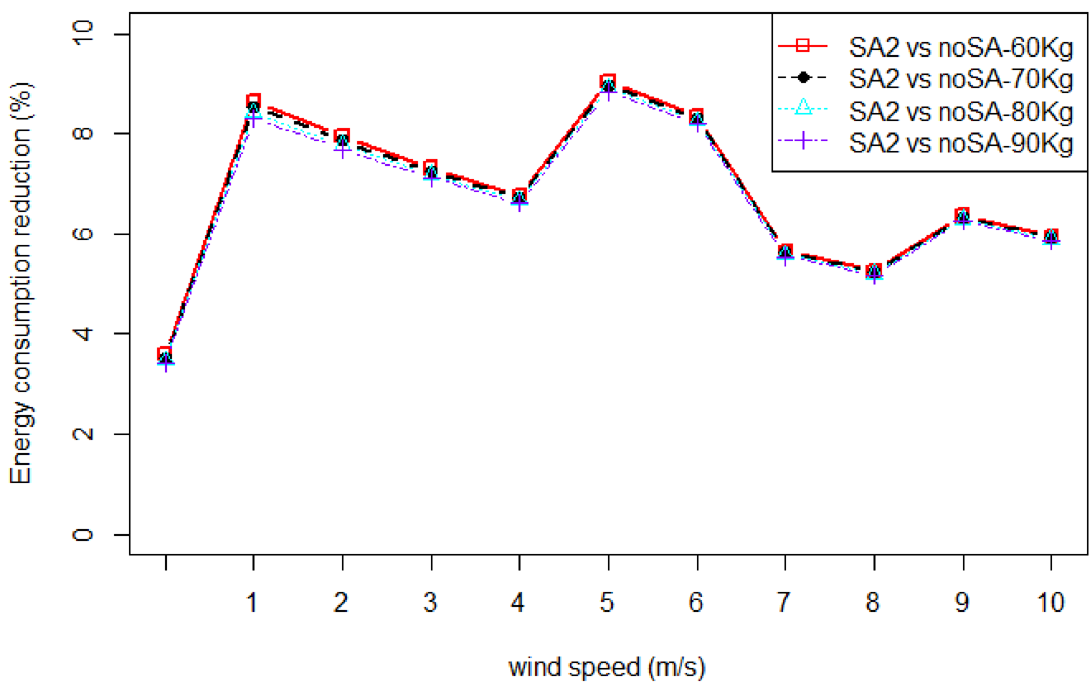


Figure 5.20. EC reduction for SA2 – Route 3

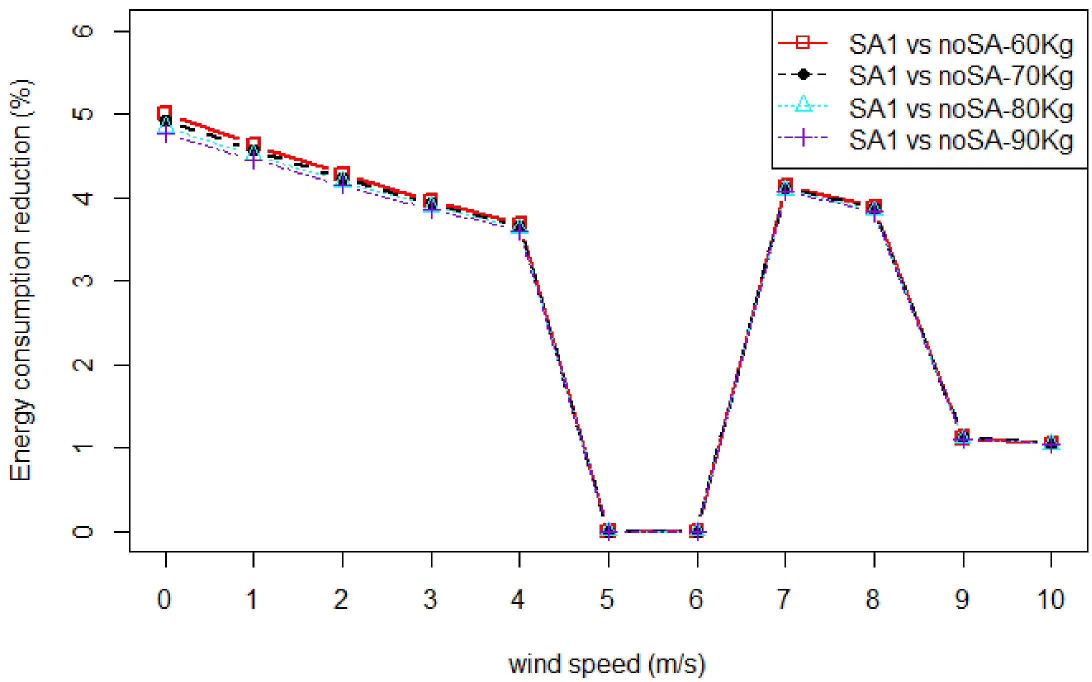


Figure 5.21. EC reduction for SA1 – Route 1

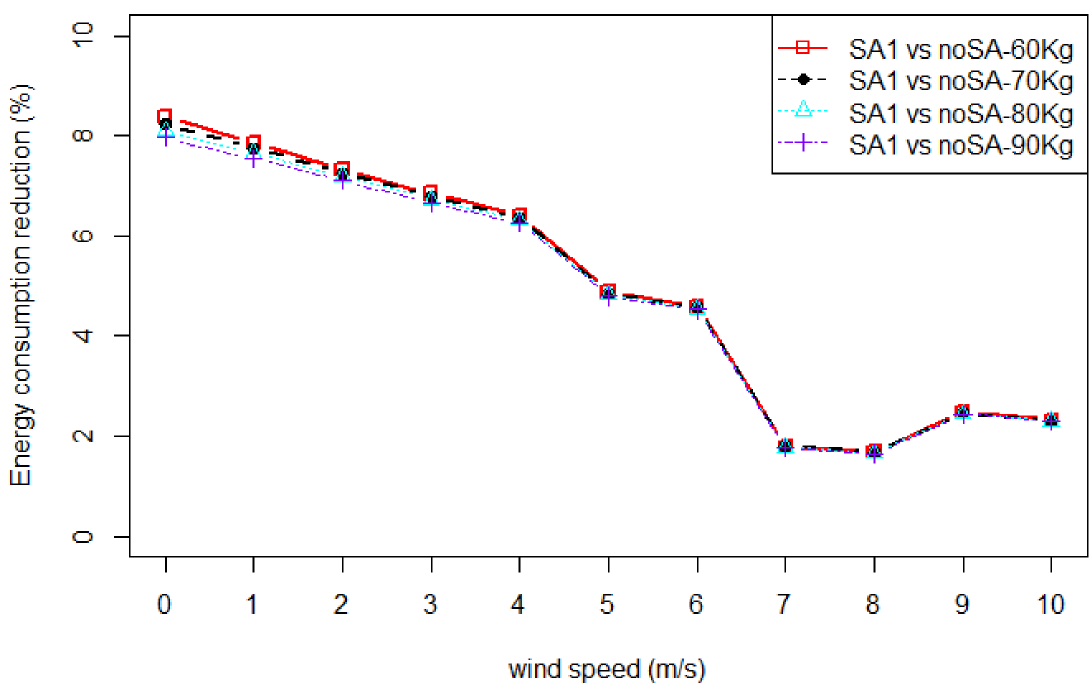


Figure 5.22. EC reduction for SA1 – Route 2

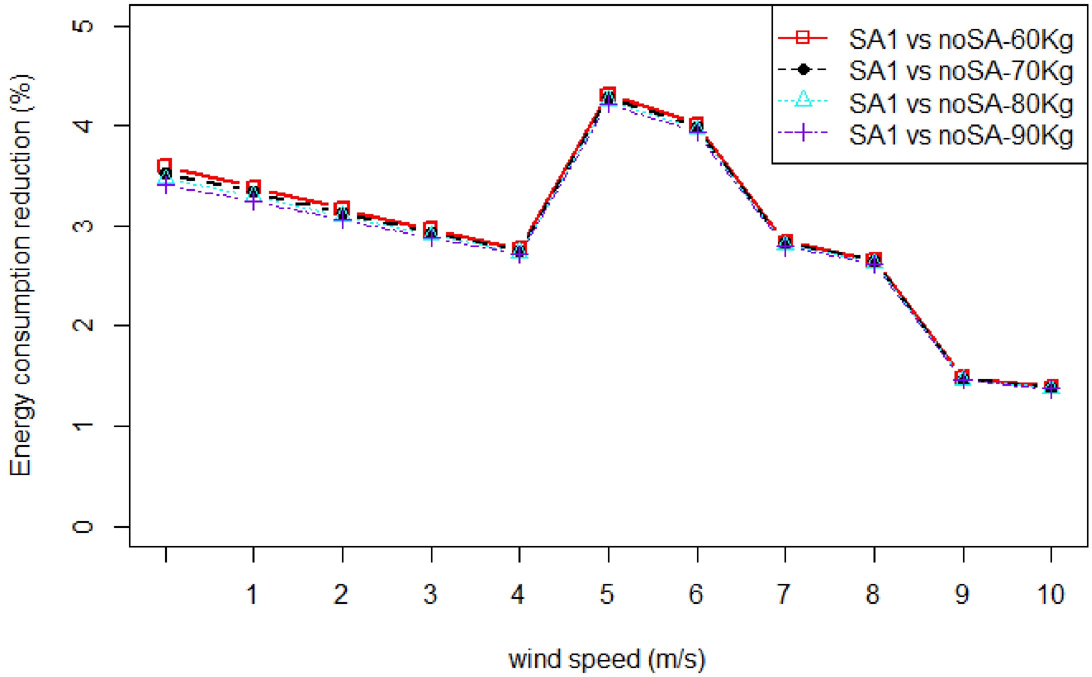


Figure 5.23. EC reduction for SA1 – Route 3

C) Power Consumption vs Time

This section presents a set of results for all three routes that show the power consumption vs. time and in which the modifications in terms of bicycle speed vs. time, number of stops, waiting times and total travel times can be clearly noticed. All the cases discussed were considered: rider with no advisory system, rider with C-GLOSA, rider with the proposed speed advisory system having GLOSA function only (SA1) and rider with the proposed speed advisory system having the complete functionality (SA2). These results are displayed in Figure 5.24 – Figure 5.26 and are meant to provide a better understanding on the previously presented results and a better comparison between the non-equipped bicycles and the bicycles equipped with the different speed advisory solutions.

The wind speed considered was 7m/s. This speed was chosen due to the fact that it reflects two special cases. The first special case is in the context of route 1 and reflects the use case of C-GLOSA that does not take into account the wind factor and consequently recommends maxSpeed = 6.95m/s which is impossible to be maintained due to bicycle's power limitations. Consequently, stopping at the traffic light is not avoided (Figure 5.24 – portion of C-GLOSA curve, after time step 79, where power equals 0). Moreover the energy consumption also increases on the portion of road where the unreachable recommended speed is forced (Figure 5.24 – portion of C-GLOSA curve, around time step 37, where power reaches 400W).

The second case is in the context of route 2, where the SA2 causes some delay to the total travel time, however due to the average lower power consumption over time, the energy consumption is still reduced. It can be seen in all the figures corresponding to all three routes (Figure 5.24 – Figure 5.26) that the Fuzzy Logic-based weather aware speed adaptation policy implemented in SA2 results in an energy consumption decrease on average, leading to a higher energy savings. This energy saving is more significant than that of SA1 or C-GLOSA which focus on avoiding stopping at the intersection only. The stops at the intersection for the non-equipped bicycle (noSA) can be easily identified in the graphs when the power consumption is 0. It can also be seen how the speed advisory systems avoid the stops by recommending for instance lower speeds. The lower speeds are marked by lower power consumption (e.g. Figure 5.26, the SA2 curve around time step 317 has the power value around 100W).

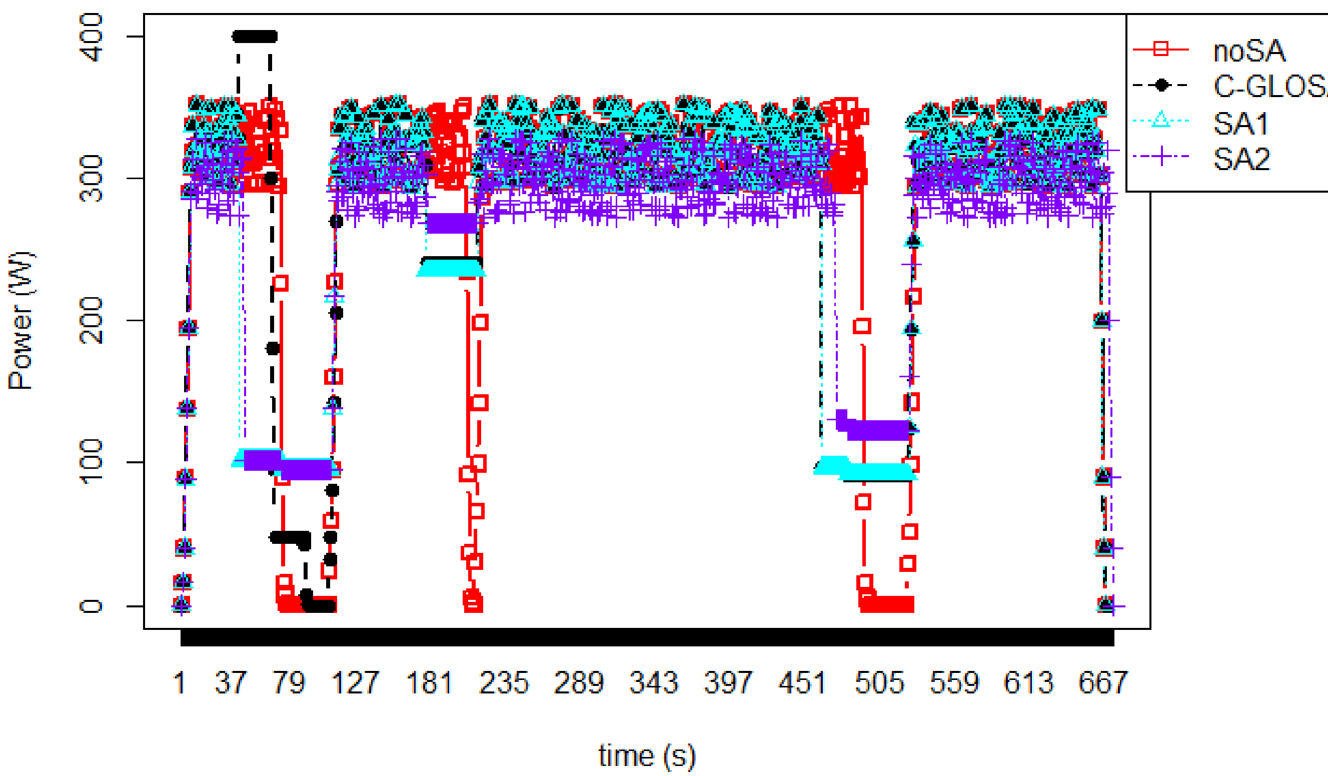


Figure 5.24. Power vs time – Route 1

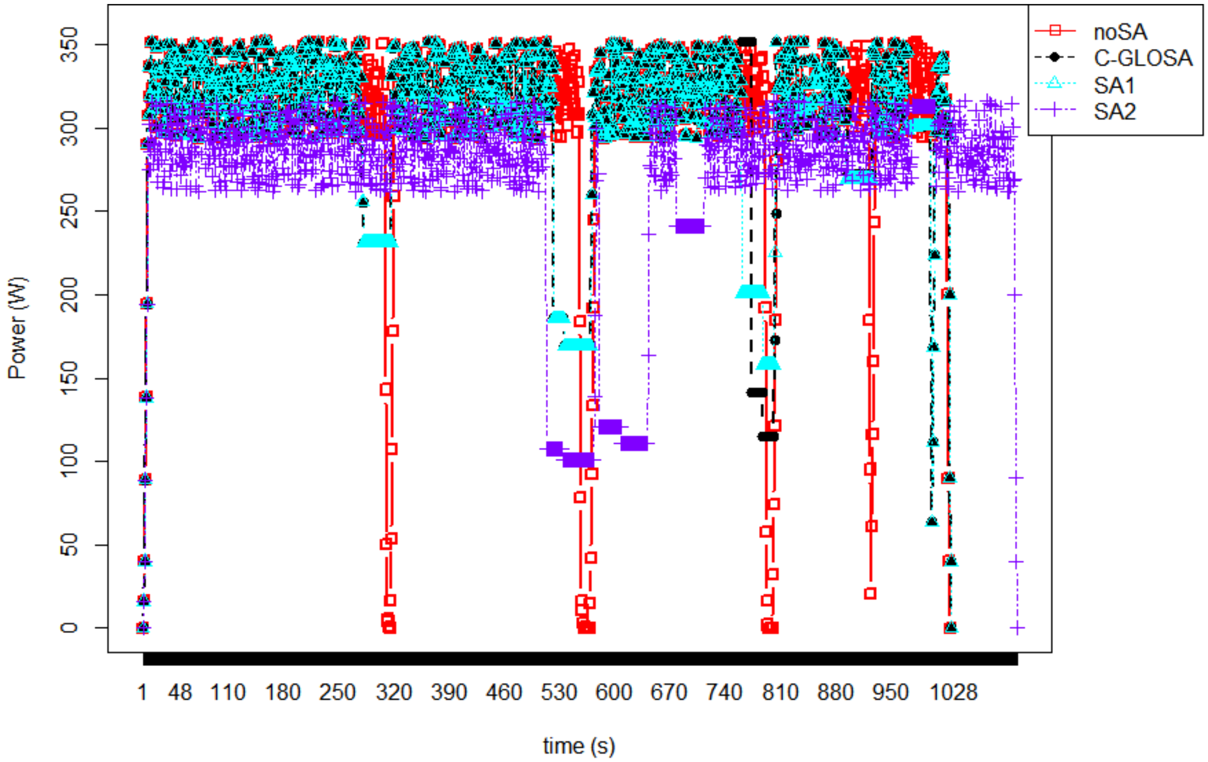


Figure 5.25. Power vs time – Route 2

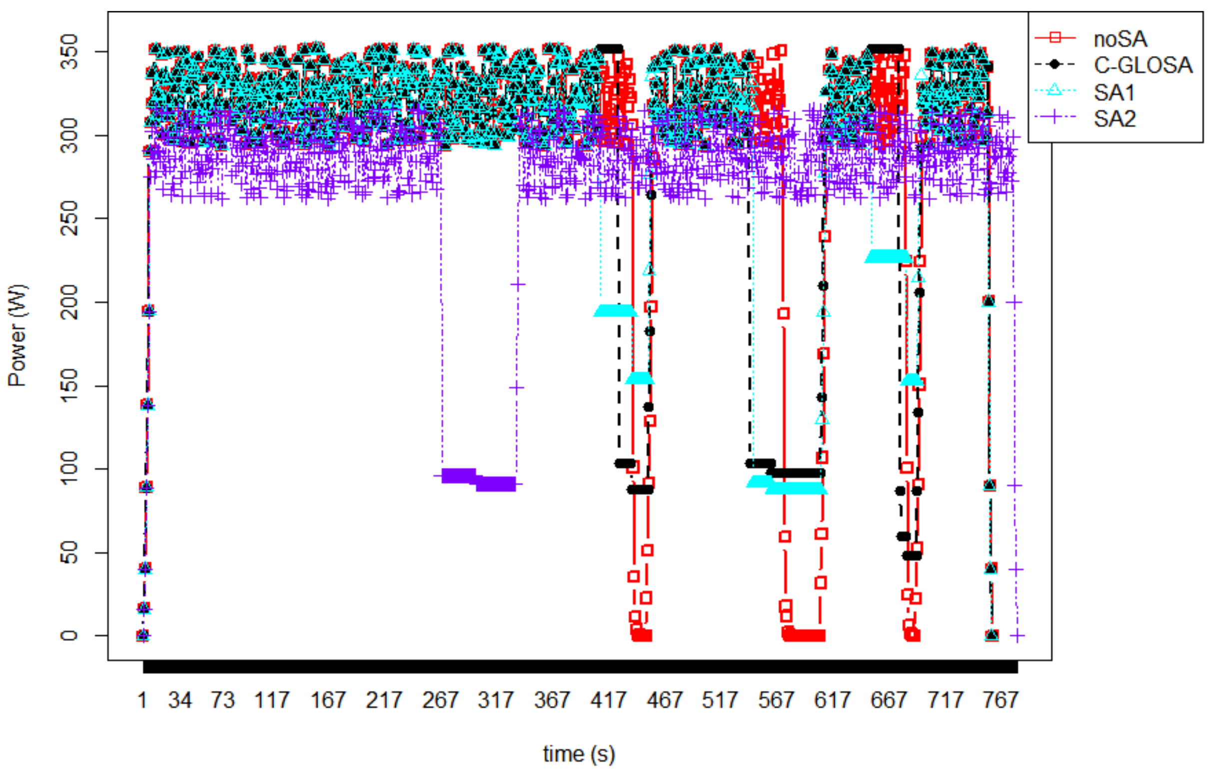


Figure 5.26. Power vs time – Route 3

5.2. eWARPE Performance Evaluation

This section presents the performance evaluation of eWARPE, the energy-efficient and weather-aware route planner dedicated to bicycles in general and electric bicycles in particular. Following the guidelines in [241], a survey study was conducted with the main goals listed below:

- to demonstrate the existence of the problem: poor weather conditions affect cyclists;
- to determine the dimensions of the problem (measure the impact of the poor weather conditions on the cyclists) and consequently to measure the need for the proposed solution;
- to validate the cyclists' interest in the proposed solution that aims at guiding them in addressing the aforementioned problem;
- to measure the benefits of the proposed solution: do cyclists think that the proposed solution will improve their cycling experience?

First, the survey was conducted as an interview, and then in order to obtain the opinions of a larger number of people, as an online survey using a questionnaire. Various online tools such as Impressity[29], Survey Monkey[30], LimeSurvey[31], SurveyShare[32], etc. meant to provide support in surveys are nowadays available. Impressity was chosen in order to conduct our survey. This is a free of charge web service which allows people to create marketing research on various areas of interest. The fact that Impressity does not impose any standard templates was of influence on our choice for this tool. Moreover, the decision was also based on the fact that there are no restrictions when it comes to number of surveys which could be completed by the audience and on the fact that it is highly secure. Regarding security and confidentiality of the survey created, Impressity allows for setting up to 10 confidential questions and ensures the detection for personal identifiable information if required.

5.2.1. Survey - Interview

The interview was run for a two week period, each interview contained 25 questions and took on average 20 minutes. The questions are listed at the end of the thesis in Appendix A. All the interviews were recorded and notes where taken during each interview in order to help with the results analysis. A number of 20 subjects (Male=11, Female=9) between 19 and 34 years old were interviewed. Subjects that presented no interest in cycling were not considered. For the purpose of this study, only subjects who are active cyclists or did cycle in the past and would still consider

---

[29] Impressity website, http://www.impressity.com/
[30] Survey Monkey website.https://www.surveymonkey.com/
[31] Lime Survey website, https://www.limesurvey.org/en/
[32] SurveyShare website, https://www.surveyshare.com/

cycling in the future were considered. The subjects were asked to select the class they belong to from the following classes [17]: regular cyclist (52 or more one-way trips per year), frequent cyclist (12-51 one-way trips per year), occasional cyclist (1-11 one-way trips per year) and potential cyclist (never in the past year). The distribution per classes of the subjects was: 35% regular cyclists, 15% frequent, 30% occasional and 20% potential. All subjects live in Dublin, Ireland where the climate is known to be mild, with no extreme temperatures[33]. This makes cycling possible throughout the year, but the problems for cycling are the frequent rain falls and strong winds.

After the completion of 20 interviews, with the help of the notes taken, as well as by reviewing the records, each interview was carefully analyzed. The analysis is presented below.

5.2.1.1. The problem and its dimensions

During the interview, the subjects were asked what they consider to be the main disadvantages of cycling. The answers in the order of their popularity were:

- negative impact of the poor weather, rain and wind mainly (identified by 50% of the subjects);
- safety issues (45% of the subjects);
- cycling is uncomfortable (10% of the subjects);
- reasons of maintenance (10% of the subjects);
- “not suitable for longer trips” (5% of the subjects);

The top answers to the question what are the main factors that negatively affect them when cycling were:

- poor weather conditions (55% of the subjects mentioned this negative factor);
- driver behavior (25%);
- not enough cycling facilities (“not enough space to cycle”, “roads not suitable for cycling”) (25%);
- lack of safe places for locking the bicycles (10%).

Note that before subjects answered these two aforementioned questions, no thought about weather conditions was induced. The questions were free with no suggested answers.

The negative effect of poor weather conditions on cycling was expressed by the subjects as follows: “more difficult to cycle”, “causes health problems”, “make cycling more dangerous”, “uncomfortable”, “not suitable for wearing fancy clothes”, “carried things get wet”. When they

[33] The Irish Meteorological Service Online, http://www.met.ie/

were asked to measure this effect, the subjects ranked in average with 4 the level of disturbance caused by the adverse weather conditions on a scale from 1 to 5 (not disturbed at all – very disturbed).

5.2.1.2. The interest in eWARPE and its benefits

The results presented above demonstrate the need for the proposed solution as the weather is an important issue for cycling. Moreover, only 15% of the subjects declared that they are usually tightly constrained by time so their departure time is not flexible, the rest of 85% said that their trips by bicycle are flexible so the departure time is flexible given a time interval. However, all the subjects stated that they would choose the departure time that corresponds to the best weather conditions (least chances of precipitation falls and less influence of the wind) in a certain time interval.

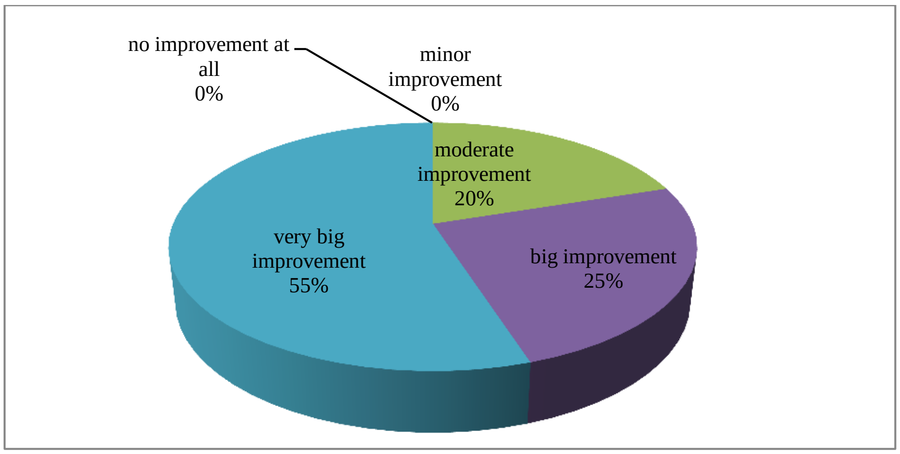


Figure 5.27. Proposed solution expected improvement on subjects' cycling experience

Thus the interest showed by the subjects in the proposed solution is not surprising: 19 out of 20 subjects stated that they are interested in such solution and that this solution would determine them to cycle more or to be more inclined to cycle. Only one subject said that he would be somehow interested in the solution, but this solution would not determine him to cycle more. In addition, the participants to the interview considered that the proposed solution would improve their cycling experience. On a scale of 1 to 5 (no improvement at all to very big improvement) the improvement was rated as follows: 5 – 55%, 4 – 25%; 3 – 20% (as illustrated in Figure 5.27), the average rate being 4.1 (big improvement).

Suggestions regarding additional adjustments which might be of interest for our subjects were collected during the interview. The decision of running the interview in advance of online survey proved to be inspired since we based the preparation of the questions for the online survey on the experience and suggestions given in the interviews.

5.2.2. Survey - Questionnaire

The online survey included 26 questions and was run to completion during a 2 months period. In total 183 people (Male=108, Female=75) responded to our questionnaire, mostly with ages between 19 and 65 years old. Similar to the interview, the participants were asked to select the class they belong to from the same classes: regular cyclist, frequent cyclist, occasional cyclist and potential cyclist. The distribution per classes of the participants was: 59% regular cyclists, 10% frequent, 18% occasional and 12% potential. The questions included in the questionnaire were intended to investigate the same topics as the interview, thus the findings are presented below in a similar way. The full questionnaire is listed at the end of the thesis in the Appendix B.

5.2.2.1. The problem and its dimensions

During the online survey the subjects were asked to choose from a list of factors, the ones that are of higher negative influence on their cycling motivation. The answers as retrieved from Impressity are presented in Figure 5.28. It can be seen that bad weather conditions factor is on top of the list of the negative factors influencing cycling motivation. Further, the full list of these factors in the order of their popularity as determined by the subjects is presented:

- Bad weather conditions, identified by 27.05% of the subjects;
- Poor or lack of cycling facilities (side-walks, cycle lanes, etc.), 23.64% of the subjects;
- No safe place to lock your bicycle, 22.78% of the subjects);
- Seasonality (22.50% of the subjects);
- Darkness (21.93% of the subjects);
- Drivers attitude (21.64% of the subjects);
- Traffic conditions (21.36% of the subjects);
- Safety (21.07% of the subjects);
- Distance (18.23% of the subjects);
- Hilliness (12.82% of the subjects);
- Effort (12.52% of the subjects);
- Pollution (10.82% of the subjects);
- Noise (5.58% of the subjects);

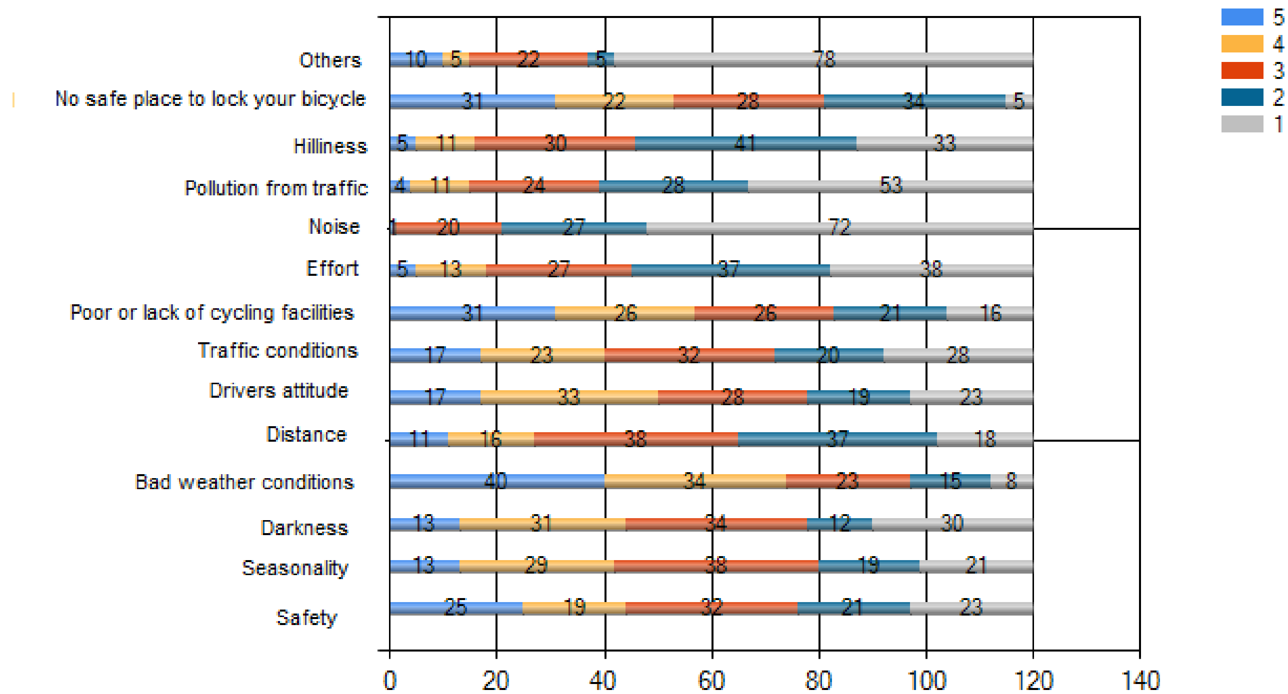


Figure 5.28. Factors negatively influencing cycling – answers

Figure 5.29 presents the results graphic as retrieved from Impressity that correspond to the question: “Do you agree that bad weather conditions, such as precipitation or wind, negatively influence your cycling experience?”. It can be seen that most of the participants at the online survey, 85%, agree or strongly agree that bad weather conditions have a bad influence on their cycling experience.

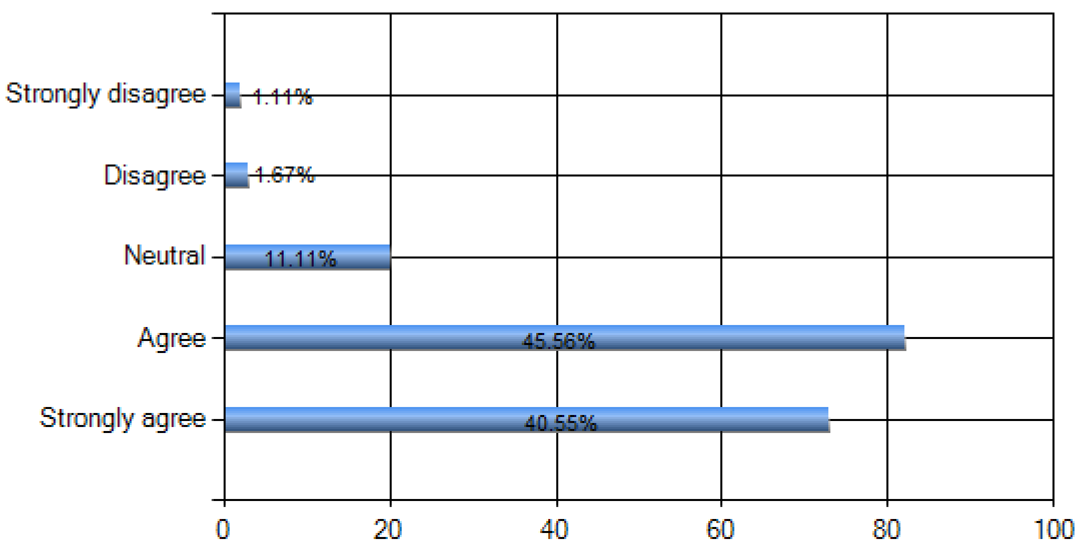


Figure 5.29. Bad weather conditions influence on cyclists

Through our next questions we identified that 60% of our survey respondents consider rain as being the first climate factor which directly influence cycling. In a relatively close percentage to rain, 51.51% of the people consider that wind has a huge influence, 17.96% of the audience specified that snow does also influence them and 16.16% considered ice is also important for cycling experience. 55% of the participants find it difficult and very difficult to cycle in windy conditions and 57% find the wind very disturbing or disturbing while cycling. The effects of assertive weather conditions were classified by our respondents to the survey as follows: “harder in terms of effort”; “negative influence on cyclist health and personal mood”; “less safety “; “longer time allocated to the trip”; “ruins make-up”; “decreased visibility”; “increased risk of accidents”; ”not suitable for wearing fancy cloths”; ”high risk to wet the carried items”.

5.2.2.2. The interest in eWARPE and its benefits

The results presented so far are a clear proof of the need for such a solution. Through the survey we were interested to investigate the start time flexibility of respondents cycling journeys, since we consider that this can be of influence on people’s decision of using the proposed solution. Only 28% of the subjects declared they are tightly constrained by time, 48% said their trips by bike are flexible so that the departure time is flexible given a time interval and 22% considered their trips by bike do not depend at all on time. However, 77% of the subjects confirmed that they will go for a departure time that corresponds with the best weather conditions for the trip in a certain time interval.

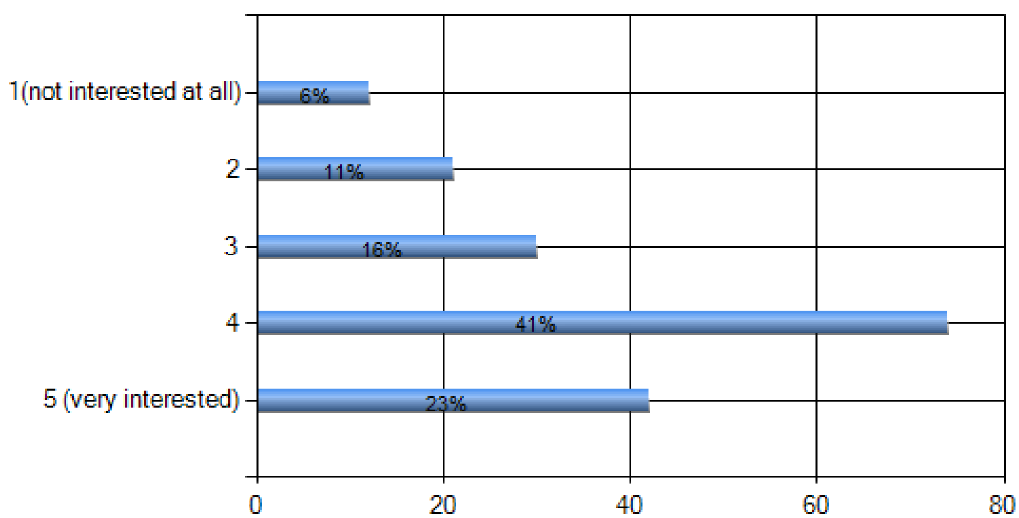


Figure 5.30. Interest in eWARPE

Given the above results, the people interest in the proposed solution is not surprising. As it can be seen in Figure 5.30, on a scale of 1 to 5 (not interested at all to very interested) the interest of the subject was rated as follows: 5 – 23%, 4 – 41%; 3 – 16%; 2-11%; 1-6%. Figure 5.31

presents the answers of the participants when asked how useful eWARPE is for improving the cycling experience. 60% of the online survey participants responded that eWARPE is useful or very useful. Moreover, eWARPE is proved to have a positive influence on cyclists' motivation, 46% of the participants declared that eWARPE will make them much more likely to cycle.

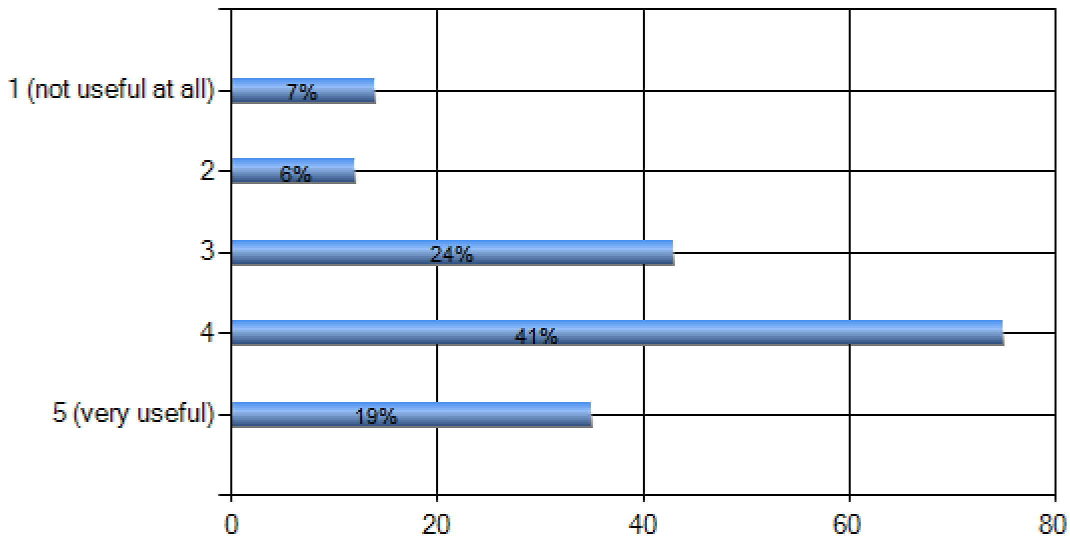


Figure 5.31. eWARPE estimated usefulness on improving cycling experience

5.2.3. Case Study

In order to validate the benefit brought by eWARPE in terms of energy savings, we considered a case-study performed on the route of one of the first participants to the interview that is cycling daily, Monday to Friday, from home to Dublin City University (DCU) and back. The aim was to show in terms of energy saving the benefits of using the proposed solution. As most of the participants in the survey, the subject declared that the departure time is flexible, in a certain interval of time, and that will definitely choose to go when there are fewer chances to rain and when the influence of the wind is less. The subject stated that the usual leaving time from home to DCU is in the interval [7:30 – 9:00] and from DCU to home is in the interval [17:30 – 19:00]. The cyclist also stated that traffic volume does not affect the time of the trip because on the segments of road with increased traffic there are cycle lanes.

The weather data was taken from Dublin City Airport meteorological station (EIDW) as this is the closest station to the subject's route. The weather data from January 2012 to June 2013 (18 months) was retrieved[34]. Saturdays, Sundays, public holidays and DCU closing days were eliminated from the weather data.

[34] METAR/Synop Weather Information for EIDW, http://weather.gladstonefamily.net/site/EIDW

Figure 5.32. Case-study route

The route was drawn in BBBike@Dublin (Figure 5.32), the BBBike web-based cycle route planner for Dublin used in eWARPE, and it was confirmed by the cyclist. In order to validate the benefits in terms of energy saving brought by eWARPE, the total energy consumption determined by air drag/route is computed. Note that only the power consumption needed to overcome the air drag is considered as this is where the wind influence is reflected and this is the weather component that has impact on the energy consumption. Moreover, the additive relationship between the powers in the power consumption model allows for this type of analysis: if the P_{drag} decreases/increases with a value the P_{total} decreases/increases with the same value. The route was segmented and the energy consumption was computed for each segment of the road and then cumulated in order to obtain the total energy consumption. The energy consumption/segment of road was computed based on the power consumption needed to overcome the drag and the time needed to travel the segment of road. The following parameter values were used in the computation:

- the bicycle speed, v_g, was set to the same value for all the segments of the road as it was considered the speed in average. Initially, the value of this parameter was set to 5.56m/s (20km/h). Then, this parameter was varied between 4.17m/s (15km/h) to 6.5m/s, in order to study its impact.
- the wind components, speed and direction: v_w, D_w are taken from the weather data
- the time in which a segment of road is travelled on was given by the cyclist and validated also in BBBike@Dublin

- D, A, C_d were set to the typical values that were also used in SAECy testing

TABLE 5.7. SIMULATION CASE-STUDY RESULTS

Month	Min energy saving (Wh)	Max energy saving (Wh)	Benefit in terms of full charges ≈ 300Wh
Jan-12	494.9033	1408.196	[1.65 –4.69]
Feb-12	211.5111	936.0425	[0.7 – 3.12]
Mar-12	183.4426	965.5881	[0.61 – 3.22]
Apr-12	240.7332	1062.507	[0.8 – 3.54]
May-12	277.381	1268.486	[0.92 – 4.23]
Jun-12	373.1165	1410.571	[1.24 – 4.7]
Jul-12	230.0784	1101.862	[0.77 – 3.67]
Aug-12	317.496	1530.028	[1.06 – 5.1]
Sep-12	329.0987	1396.062	[1.1 – 4.65]
Oct-12	197.262	907.5486	[0.66 – 3.03]
Nov-12	320.1918	1019.24	[1.07 – 3.4]
Dec-12	262.0158	728.8706	[0.87 – 2.43]
Jan-13	308.7842	1214.151	[1.03 – 4.05]
Feb-13	276.5966	1346.965	[0.92 – 4.49]
Mar-13	382.2239	1693.376	[1.27 – 5.65]
Apr-13	755.0655	3024.805	[2.52 – 10.08]
May-13	813.6198	3213.698	[2.71 – 10.71]
Jun-13	461.8528	1997.801	[1.54 – 6.65]

TABLE 5.7 summarizes the results of the case-study for v_g = 5.56m/s. The least energy consumption/month is taken as reference point in computing both *min* energy saving/month and *max* energy saving/month. The least energy consumption/month is computed by cumulating the least energy consumption values obtained for each day of the month. Whereas the least energy consumption in a day is obtained by choosing the optimum departure time from the aforementioned time intervals so as to have the lowest possible wind influence during cycling. The max energy saving/month is computed as the difference between the worst energy consumption/month and the least energy consumption/month. The worst energy consumption/month is obtained as the cumulative of each day of the month when the cyclist is leaving at the most inappropriate time so that the wind influence is worst possible as per the time intervals specified. Min energy savings are obtained as the difference between the next least energy consumption obtained and the least energy consumption/month. For a better understanding of the energy saving obtained, both min and max values are presented in terms of full charges of an electric bicycle (4^{th} column of TABLE 5.7). The

usual capacity of an electric bicycle is considered to be around 300Wh[35]. Figure 5.33 - Figure 5.38 present the results obtained for all the variations of v_g. As expected, the energy savings increase with the value of the ground speed. However, as it can be seen from the presented graphics, the savings on average are quite uniform. The medians in all cases show savings of 1 to approximately 4 full charges monthly.

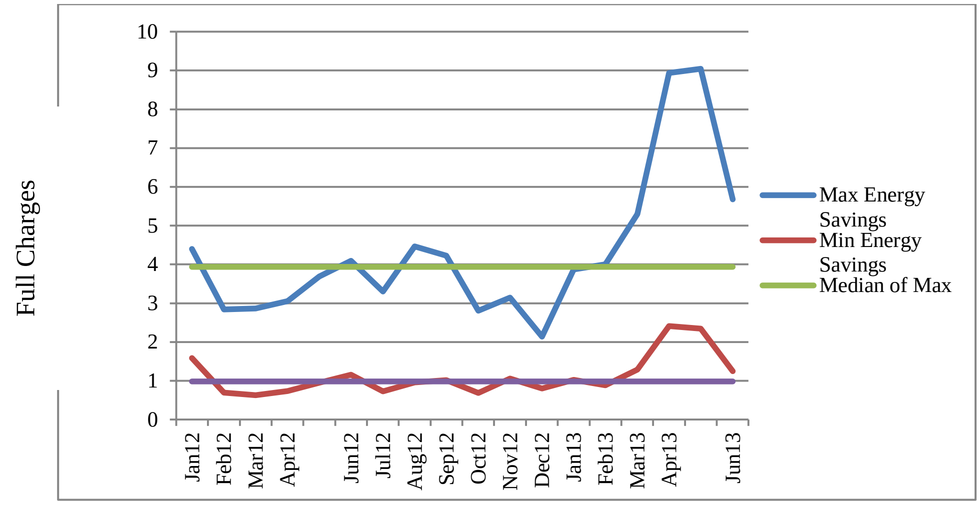


Figure 5.33. Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (v_g =4.17m/s)

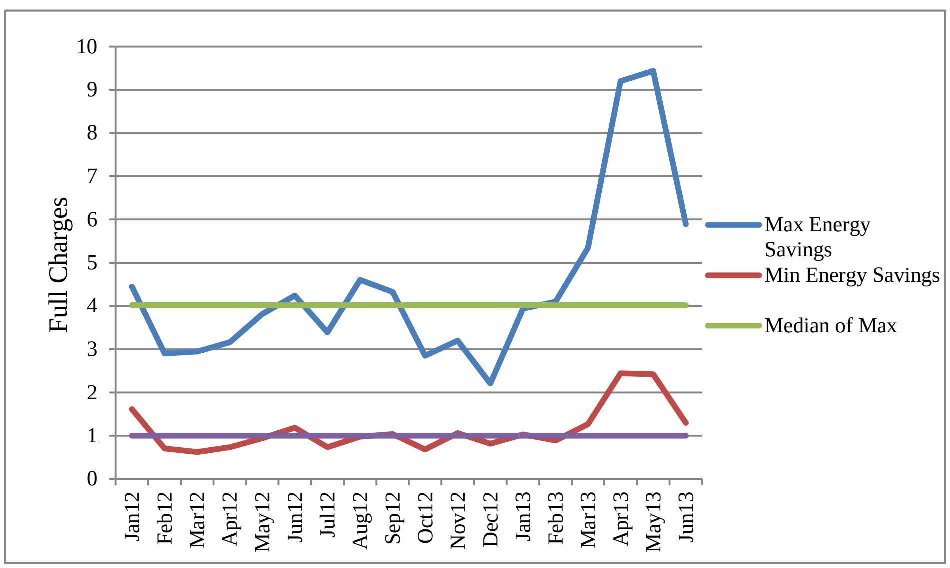


Figure 5.34. Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (v_g =4.5m/s)

[35]Electricbike website, http://www.electricbike.com/watt-hours/

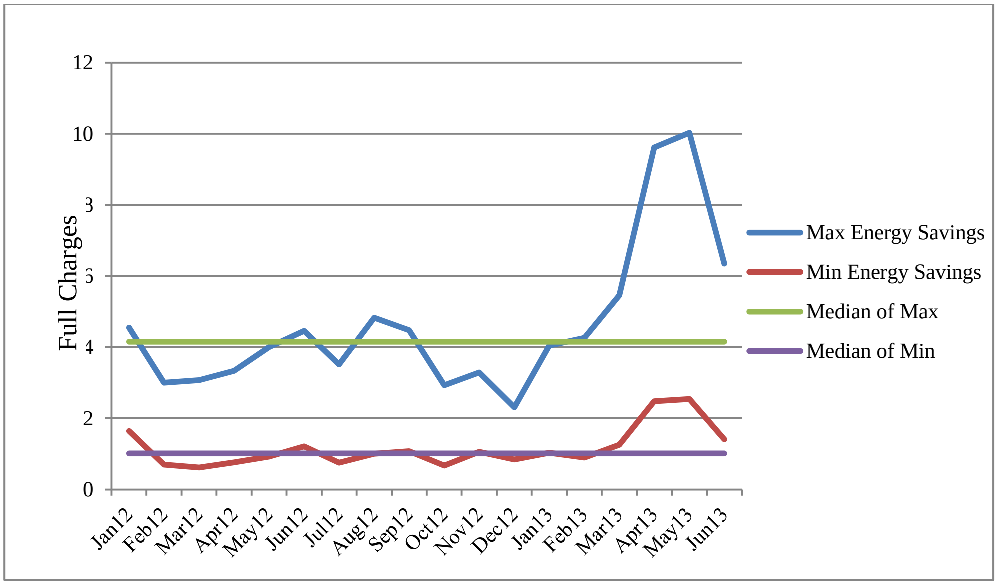


Figure 5.35. Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (v_g =5m/s)

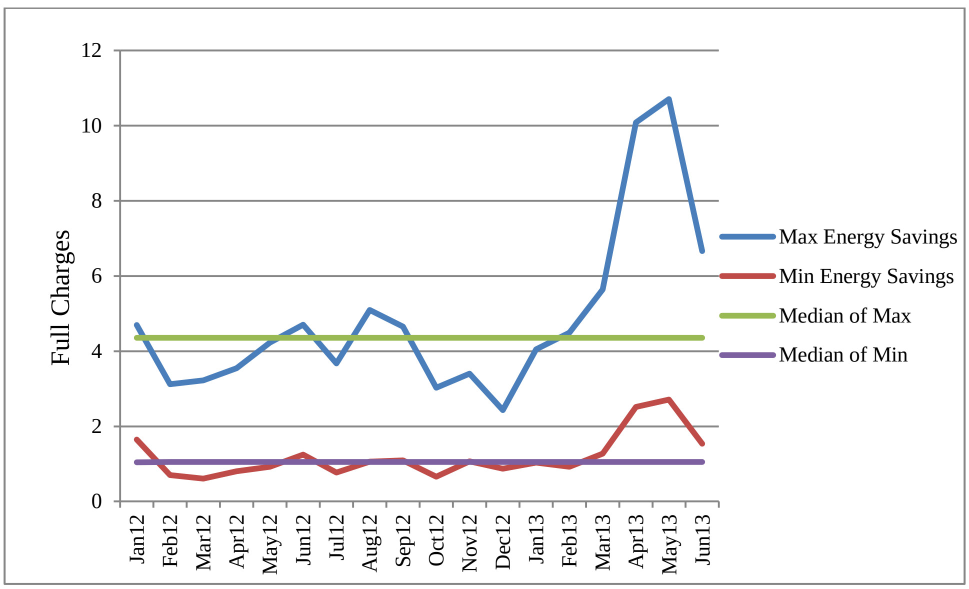


Figure 5.36. Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (v_g =5.56m/s)

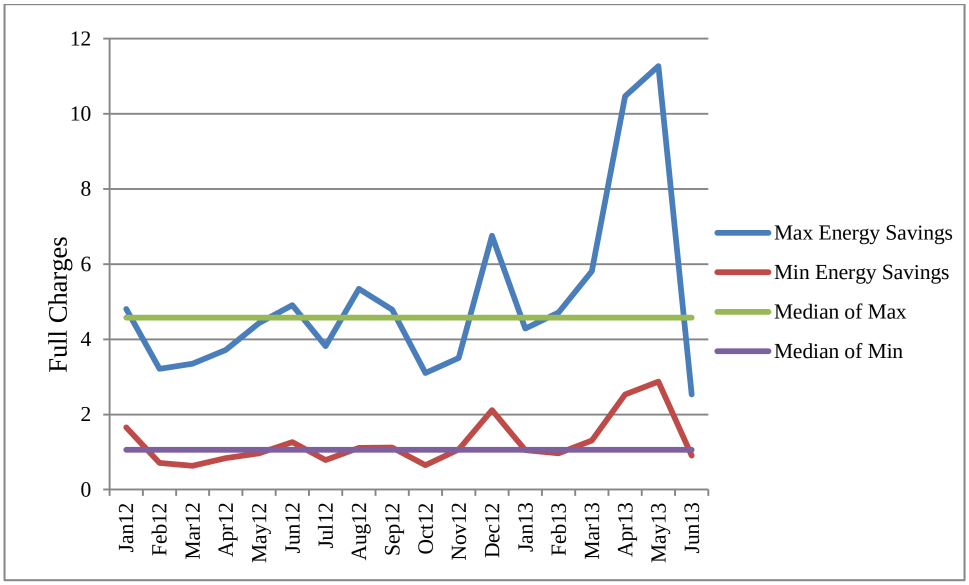


Figure 5.37. Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (v_g =6m/s)

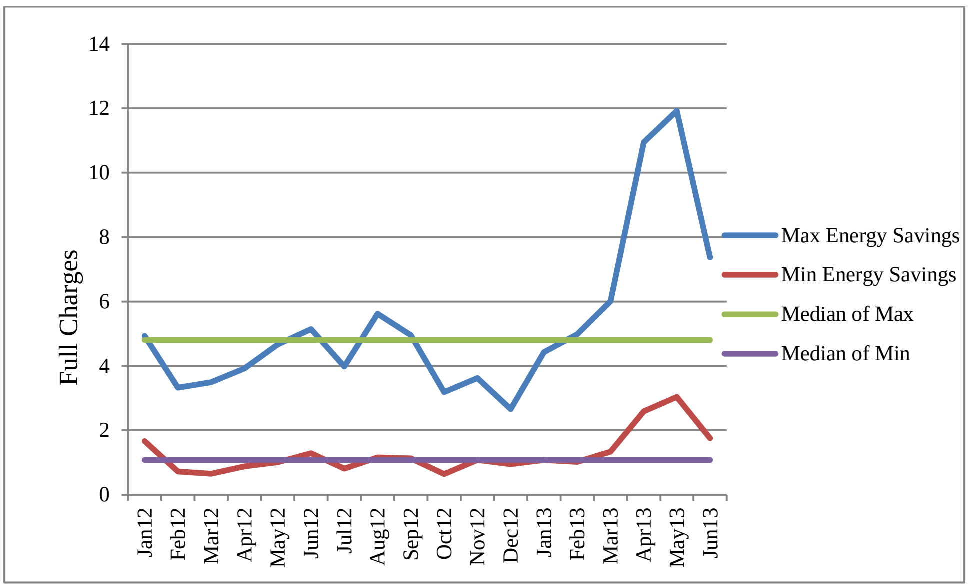


Figure 5.38. Monthly possible energy savings in terms of full battery charges for January 2012 – June 2013 (v_g =6.5m/s)

5.3. FuzzC-VANET Evaluation

This section presents the performance evaluation of the proposed FuzzC-VANET clustering algorithm. As previously described, iTETRIS simulation platform was also used for testing FuzzC-VANET performance. As it can be seen in Figure 5.4, the proposed algorithm has been implemented in the simulation platform together with the other two clustering algorithms that it was compared against: a state of the art algorithm in the area for mobile ad hoc networks, Lowest Id (LID) [217] and a recently proposed VANET-dedicated clustering algorithm. The latter algorithm chosen for comparison purposes is the algorithm – Affinity PROpagation for VEhicular Networks (APROVE) [121].

LID is probably the best known clustering algorithm in the area of ad-hoc networks. Its principle is very simple. The nodes have assigned a unique fixed id which is broadcasted periodically in the network. The clusters are formed around the node with the lowest id among them which is chosen as CH. Although this is a very simple algorithm it has been proven to be more efficient than others algorithm that use multiple clustering metrics [112].

APROVE is a generic VANET-dedicated clustering algorithm that exceeds the performances of other state of the art algorithms in the area of mobile-ad hoc networks and VANET area also: MOBIC [117], and the algorithms proposed in [119] and [216]. APROVE provides a good benchmark for our algorithm performance assessment. APROVE clusters similar behaving vehicles by analyzing multiple metrics: current and future predicted position, speed, and direction. Vehicles keep a list of their neighbours travelling in the same direction, and a cycling analysis of predicted future positions of these neighbours gives each vehicle a score which determines their suitability to being a cluster head. In score computation, Affinity Propagation technique [212] borrowed from the field of data clustering is employed.

5.3.1. Simulation Scenarios and Settings

Realistic simulation scenarios were used in testing the FuzzC-VANET algorithm. These were obtained from the publicly available dataset TapasCologne[36], part of a project that aims to reproduce "with the highest level of realism possible" the car traffic in the city of Cologne and which is highly used in the validation of VANET-related research.

Two simulation scenarios are chosen to allow a thorough testing of the algorithm in both dense and sparse traffic conditions. Scenario 1, represents the dense traffic scenario. This is a 3 km^2 crop from the Cologne city centre, coordinates (50.923896, 6.9221), while the traffic is the typical evening traffic (starting at 6pm) for the corresponding area. The simulation is run for 380 seconds and 715 vehicles are involved in traffic during this period.

Scenario 2 represents the sparse traffic scenario which is a a10 km^2 crop from a suburban area of Cologne city, coordinates (50.97029, 7.04784) , and the traffic is the typical evening traffic

[36] [241] Tapas Cologne project website: sumo-sim.org/userdoc/Data/Scenarios/TAPASCologne.html

(starting at 6pm) for the corresponding area. The simulation is run for 500 seconds and 214 vehicles are involved in traffic during this period.

TABLE 5.8. TRAFFIC SIMULATION SETTINGS

Setting	Value in Scenario 1	Value in Scenario 2	Explanation
minLat	50.923896	50.97029	The minimum latitude for the area of map that is used
minLong	6.9221	7.04784	The minimum latitude for the area of map that is used
maxLat	50.950888	51.080232	The minimum longitude for the area of map that is used
maxLong	6.96349	7.20403	The maximum longitude for the area of map that is used
Number of cars	715	214	The total number of cars during the simulation
Vehicles speed	30-50km/h	30-70km/h	The maximum speed on different segments of road for the map used
Simulation time	380s	500s	Total simulation time

For both scenarios, IEEE 802.11p was used to model the communication between the vehicles, the penetration rate of this technology is considered to be initially 100%. The transmission range between the vehicles-node was set to a typical value of 150m. TABLE 5.8 and TABLE 5.9 present more details about the simulation settings.

TABLE 5.9. NETWORK SIMULATION SETTINGS

Setting	Value	Explanation
MAC protocol	802.11p	The protocol used to exchange clustering messages and data messages between vehicles
Transmission range	150m	The communication transmission range between the vehicles nodes
RSU Transmission Range	250m	The RSU communication transmission range
Data packet size	1000bits	The size of the data messages sent by each CH to the CMs in its cluster
Data packet interval	1s	The time interval the CH sends data packet to its CMs
Penetration rate	100%	The penetration rate of the IEEE 80.11p among the vehicles used for all the tests (except for the tests where this parameter is varied)
TI	5s	The time interval that dictates when the CH re-election process is triggered
Propagation loss model	Log Distance Propagation Loss Model	The propagation loss model employed in ns3 having the following parameters: ReferenceLoss =41.8588 and Exponent=3

5.3.2. Results and Analysis

The proposed clustering scheme has been evaluated using highly known and used performance metrics in the context of VANET clustering, both topology-based metrics and network-specific metrics, as they were classified in chapter 2. The following topology-based metrics were employed: average CH lifetime (CHL), average cluster member lifetime (CML), average number of clusters (NoC), average percentage of vehicles clustered, and average cluster size (CS). The following network-specific metrics were employed: overhead imposed by the clustering messages, average throughput in the clustered network, and average throughput per CM. The last three network-specific metrics characterize the intra-cluster communications, and were measured considering that the CH of each cluster transmits data packets to its cluster members every second.

5.3.2.1. Impact of traffic density: sparse vs dense traffic

As presented above, two different real life scenarios were used in order to test the algorithm performance for different types of traffic density. Figure 5.39 – Figure 5.46 present the results in terms of the aforementioned metrics obtained for these scenarios. Next an analysis on these results is presented.

Figure 5.39 presents the average overhead imposed by the clustering messages. In terms of overhead, FuzzC-VANET imposes an increased overhead as compared to LID. This is expected as the size of clustering messages in LID is very low, the clustering being based only on the unique id assigned to each vehicle. However, FuzzC-VANET has less overhead as compared to APROVE, 40% less overhead for Scenario 1, and 59% for Scenario 2.

Average CML and CHL metrics are highly known as among the best measures of the clustered network stability. FuzzC-VANET far exceeds the performance of both algorithms, LID and APROVE, in both testing scenarios in terms of these two metrics. In Scenario 1, FuzzC-VANET increases the average CML by 65% and 45% as compared to LID and APROVE, respectively. It increases the average CHL by 50% and 34.5% as compared to LID and APROVE, respectively. In Scenario 2, FuzzC-VANET increases the average CML by 40% and 20% as compared to LID and APROVE, respectively. The average CHL is increased by 43% and 31% as compared to LID and APROVE, respectively.

FuzzC-VANET has a higher number of clusters when compared to APROVE and LID algorithms as it can be seen from Figure 5.43. If analyzing average NoC alone, it could be assumed that this is a negative result, as fewer clusters indicate a less fragmented and more stable network. However, for a correct interpretation of the results, this metric must be analyzed in conjunction with other two metrics: average percentage of vehicles clustered and the average CS. It can be seen in

Figure 5.42 that FuzzC-VANET outperforms APROVE and LID in terms of percentage of vehicles clustered. In Scenario 1, the proposed algorithm increases the average percentage of vehicles clustered with 10% and 41% as compared to LID and APROVE, respectively. In Scenario 2, FuzzC-VANET increases the average percentage of vehicles clustered with 23% and 33% as compared to LID and APROVE, respectively. Note that LID is an algorithm that clusters all the vehicles independent on their direction of travelling; this is how it achieves a high percentage of vehicles clustered. Thus, the proposed algorithm does not create a more fragmented network by creating more clusters, instead it is considerable more successful in clustering a higher number of vehicles. Moreover, the average CS is similar for all three algorithms in both testing scenarios. Considering this analysis and the results for average CHL, average CML, it can be concluded that FuzzC-VANET does not lead to a more fragmented network, on the contrary, the proposed algorithm achieves a better network stability than both APROVE and LID algorithms. The results for average percentage of vehicles clustered metric also indicates that FuzzC-VANET is more successful in dealing with the scalability issues of VANET than APROVE and LID.

The stability achieved by FuzzC-VANET has a direct impact in the throughput. In terms of throughput, FuzzC-VANET outperforms LID and APROVE in both testing scenarios as it can be seen in Figure 5.45 and Figure 5.46. In Scenario 1, FuzzC-VANET increases the throughput/CM with 62.5% and 56% as compared to LID and APROVE, respectively. In Scenario 2, the benefits are of 46% and 29% when compared to LID and APROVE, respectively. In terms of average throughput per clustered network, in Scenario 1, the benefits are of 9% and 78% when compared to LID and APROVE, respectively, while in Scenario 2 the benefits are of 50% and 60% when compared to LID and APROVE, respectively. The poor performance of APROVE when it comes to the average throughput per clustered network can be linked to the fact that APROVE clusters fewer vehicles, the percentage of vehicles clustered being the smallest among the three algorithms. However APROVE creates stable clusters with long average CML and CHL. The proposed algorithm, FuzzC-VANET not only creates stable clusters, but it also achieves a high percentage of vehicles clustered as it was outlined before. In consequence, the performance in terms of average throughput, both throughput per CM and throughput per clustered network are the best as compared to both LID and APROVE.

In conclusion, FuzzC-VANET outperforms the two clustering algorithms it was compared against, LID and APROVE, in both testing scenarios demonstrating high stability and scalability. FuzzC-VANET performs better in both sparse traffic scenario with smaller number of vehicles and dense traffic scenarios with a higher number of vehicles.

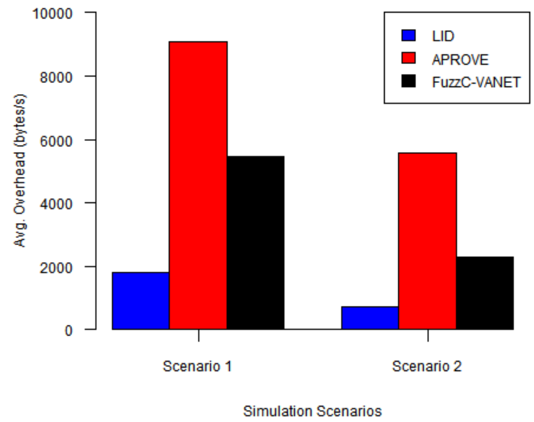


Figure 5.39. Avg. Overhead

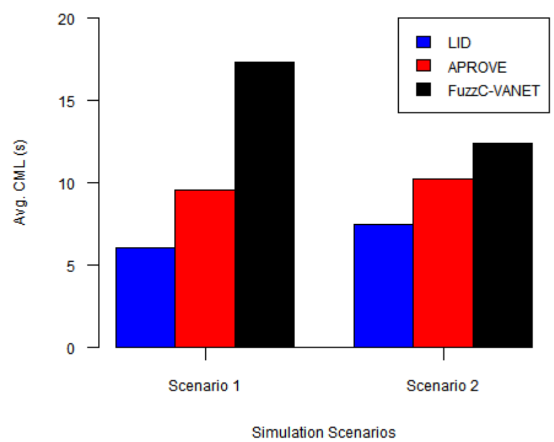


Figure 5.40. Avg. CML

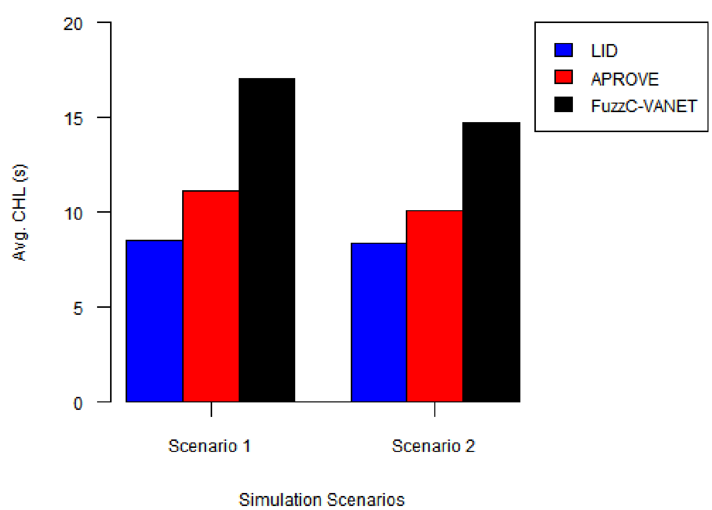


Figure 5.41. Avg. CHL

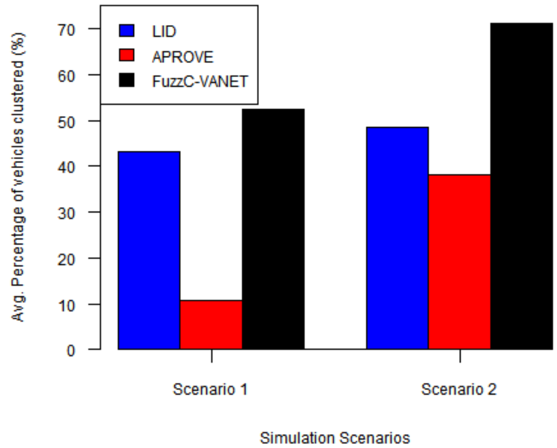


Figure 5.42. Avg. Percentage of vehicles clustered

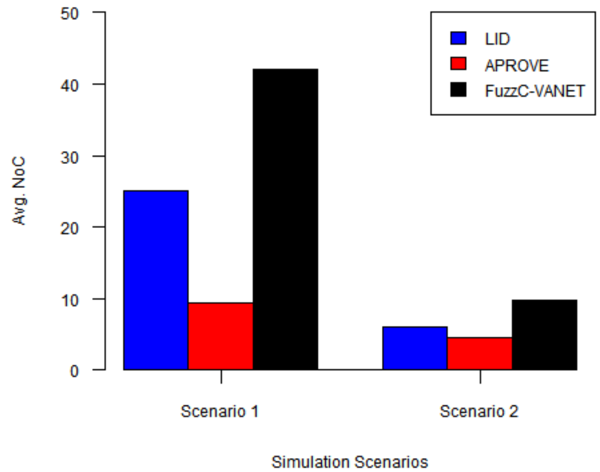


Figure 5.43. Avg. NoC

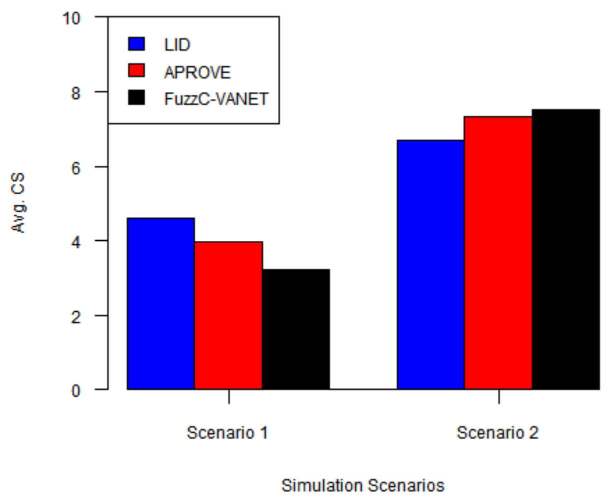


Figure 5.44. Avg. CS

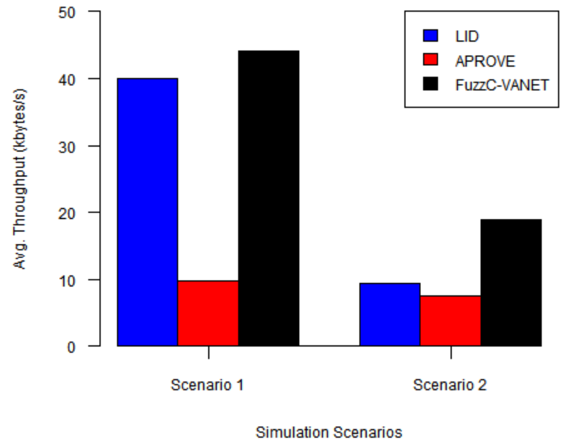


Figure 5.45. Avg. Throughput of the network

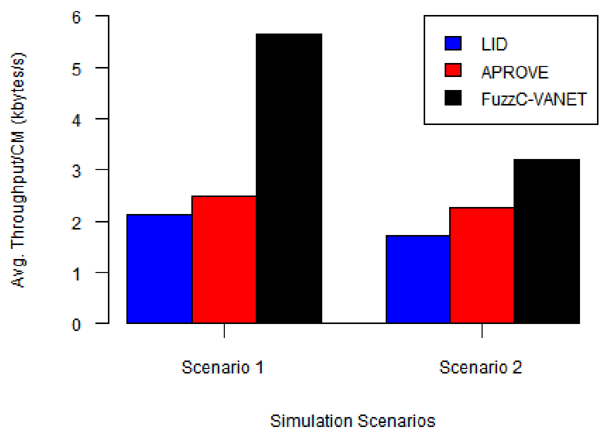


Figure 5.46. Avg. Throughput/CM

5.3.2.2. Impact of penetration rate

Although the acknowledged great potential of IEEE 802.11p technology in enhancing road safety and traffic efficiency attracted the interest of Governments (e.g. in US this technology is regulated), academia and industry, it was estimated that for instance in US alone, it will take 10 years for the penetration rate of the technology to reach 100% [59]. In this context, it is important the proposed solution to work well with penetration rates lower than 100%. Therefore, the testing has also considered different penetration levels. Figure 5.47 – Figure 5.62 display the results obtained when the penetration rate was varied from 100% to 25% in steps of 25%. FuzzC-VANET outperforms the two algorithms used for comparisons for all tested levels of the penetration rate, in both testing scenarios: FuzzC-VANET has higher average CML (Figure 5.48, Figure 5.56), CHL (Figure 5.49, Figure 5.57), percentage of vehicles clustered (Figure 5.50, Figure 5.58), throughput (Figure 5.53, Figure 5.54, Figure 5.61, Figure 5.62). The average NoC (Figure 5.51, Figure 5.59) is also higher for all the tested penetration rates, but considering the average percentage of vehicles clustered and CS shows that FuzzC-VANET creates a bigger clustered network, highly stable as it is demonstrated by the other metrics, average CHL and CML.

With few exceptions, FuzzC-VANET keeps the proportions of the benefits as compared to the other two clustering schemes for all the penetration rate levels and for all the metrics considered in both testing scenarios, sparse and dense. For instance in Scenario 1 (Figure 5.47), FuzzC-VANET increases the avg. CML by 34%, 36% and 45% as compared to APROVE, when penetration rate takes 50%, 75% and 100% value, respectively. In the same scenario, FuzzC-VANET increases the avg. CML by 45%, 59% and 65% as compared to LID, when penetration rate takes 50%, 75% and 100% value, respectively. The avg. CML when penetration rate is 25% is among the aforementioned exceptions, as the value of avg. CML for FuzzC-VANET is similar to the one for APROVE, but still greater than the value for LID by approximately 16%. Similar situation for avg. CHL metric in Scenario 1 represented in Figure 5.48. The improvement when the penetration rate is small, at 25%, is as follows: 6% improvement is achieved as compared to APROVE, but still significant improvement as compared to LID, 43%. For the other values of the penetration rate, the improvements are comparable: 16%, 45%, 34.5% when compared to APROVE and 72%, 60%, 50% when compared to LID, when the penetration rate takes 50%, 75%, 100% value, respectively. In Scenario 2, the improvements obtained by FuzzC-VANET for avg. CHL in comparison to APROVE and LID are quite similar when penetration rate changes its value: 58%, 27%, 41%, 43% when compared to LID and 23%, 24%, 25%, 31% when compared to APROVE when penetration rate takes 25%, 50%, 75%, 100% value, respectively. Similar situation is for avg. CHL metric (Figure 5.56), but also for the other metrics such as avg. percentage of vehicles clustered (Figure 5.57). For avg. CHL the improvements are of 62%, 27%, 34%, 40% when compared to APROVE and 50%, 34%, 28%, 18% when compared to LID when penetration rate takes the 25%, 50%, 75%, 100%, respectively. For avg. percentage of vehicles clustered the improvements are of 33%, 18%, 33%, 32% when compared to LID and 40%, 48%, 47%, 46% when compared to APROVE when penetration rate takes the 25%, 50%, 75%, 100%, respectively.

As a conclusion to the analysis performed above, it can be said that FuzzC-VANET demonstrates its performance consistency when the penetration rate is varied for both testing scenarios.

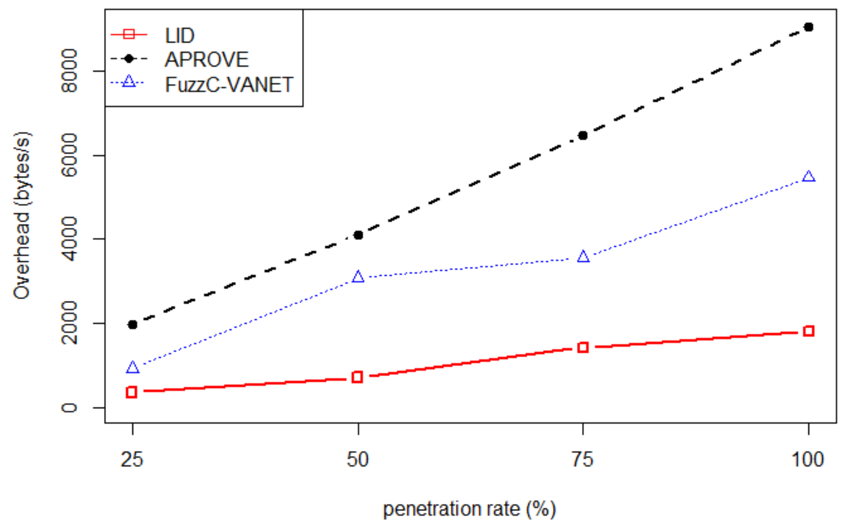


Figure 5.47. Avg. Overhead vs penetration rate – Scenario 1

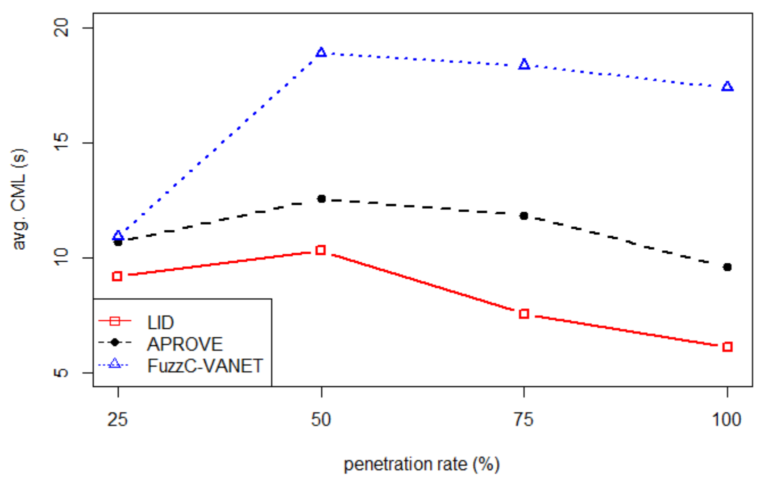


Figure 5.48. Avg. CML vs penetration rate – Scenario 1

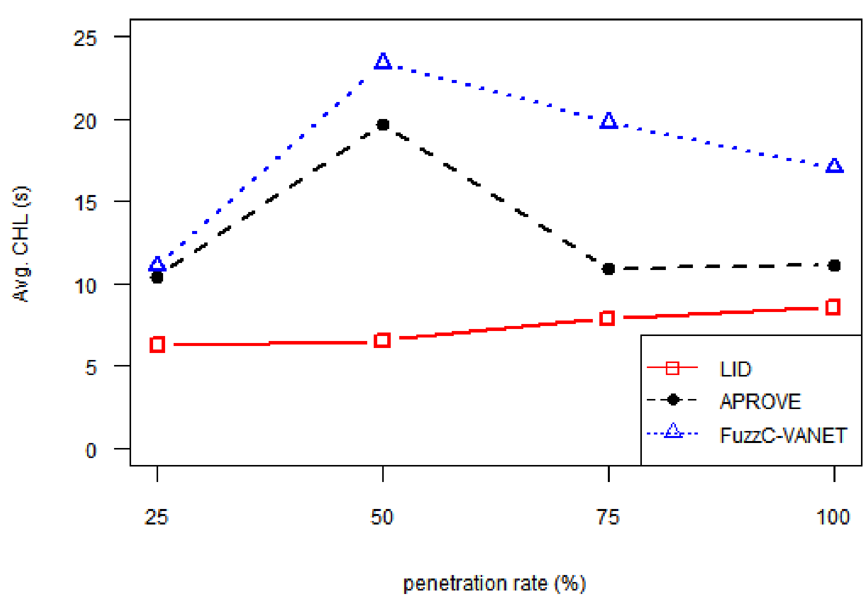


Figure 5.49. Avg. CHL vs. penetration rate – Scenario 1

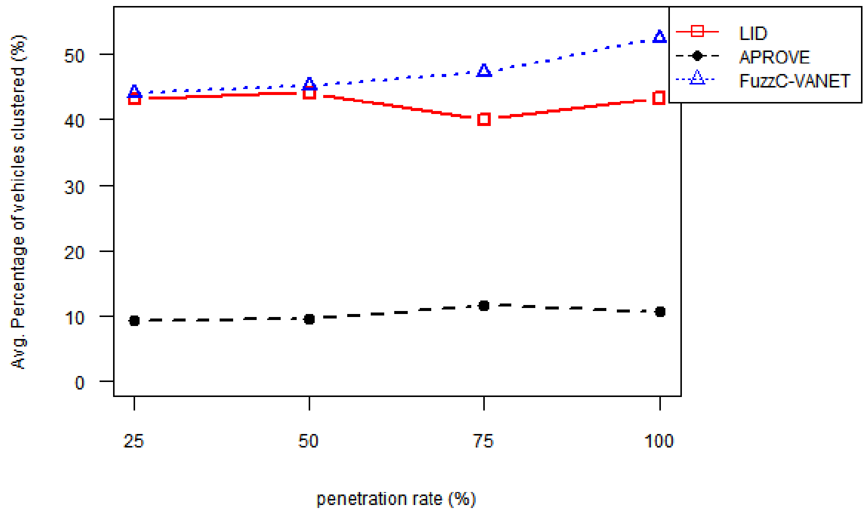


Figure 5.50. Avg. Percentage of vehicles clustered vs penetration rate – Scenario 1

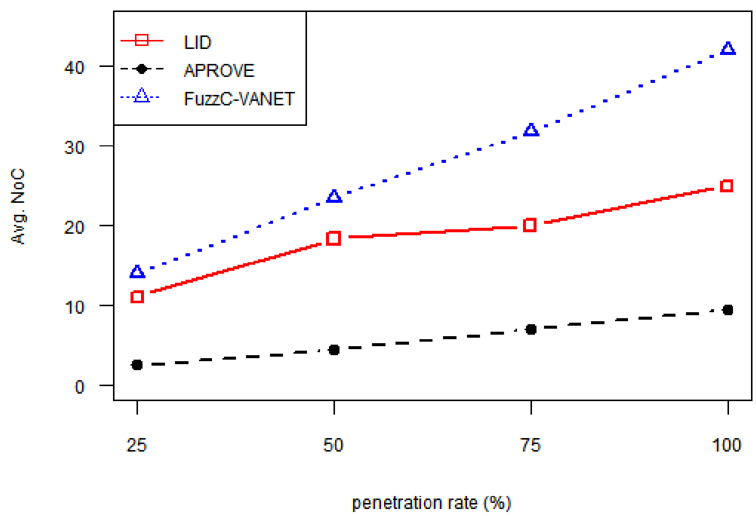


Figure 5.51. Avg. NoC vs penetration rate – Scenario 1

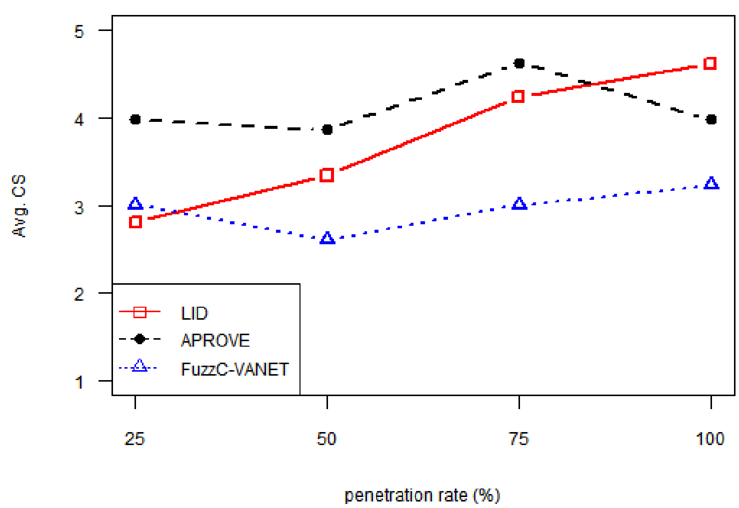


Figure 5.52. Avg. CS vs penetration rate – Scenario 1

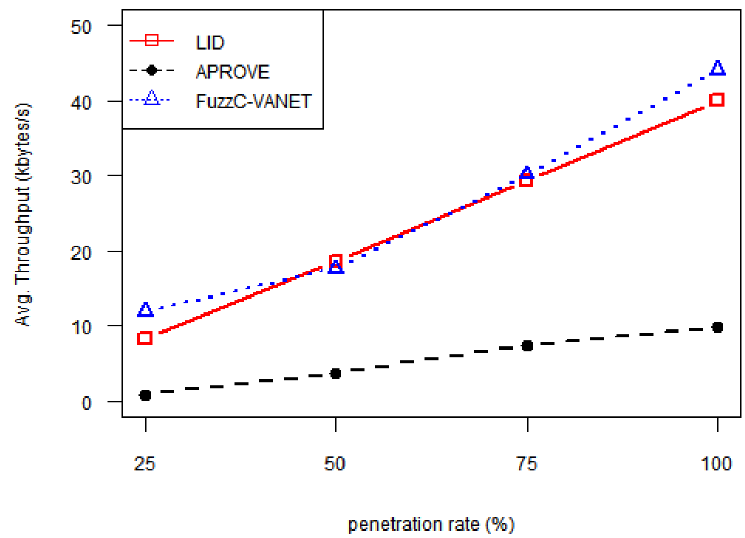


Figure 5.53. Avg. Throughput of the network vs penetration rate – Scenario 1

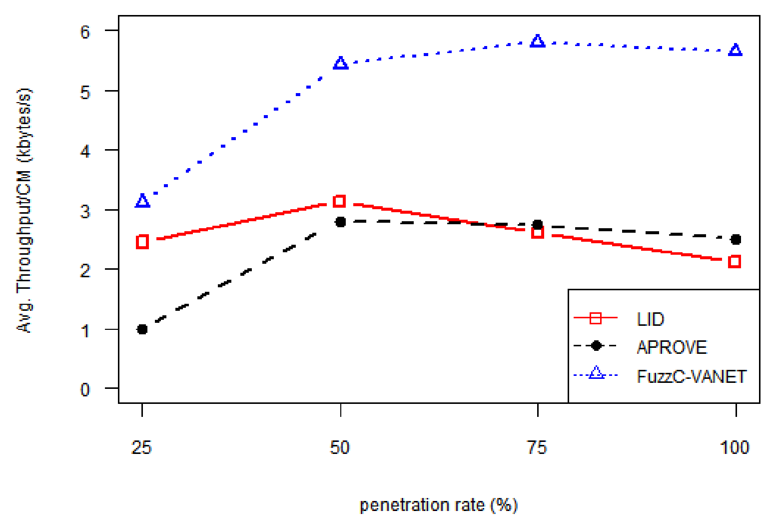


Figure 5.54. Avg. Throughput/CM vs penetration rate – Scenario 1

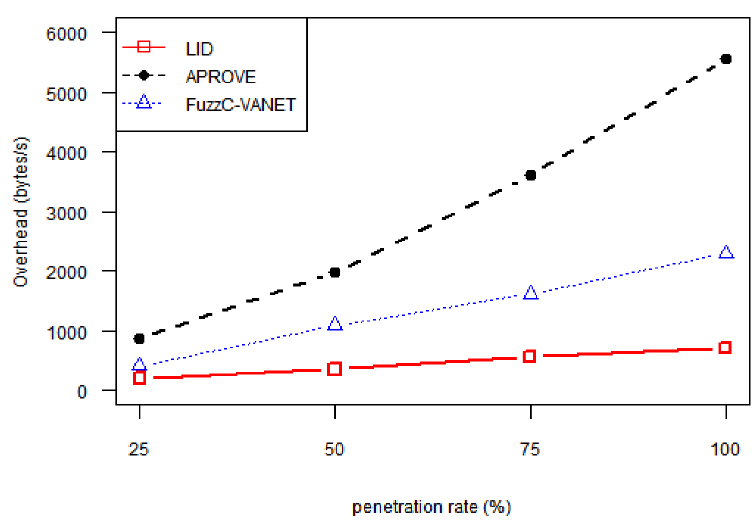


Figure 5.55. Avg. Overhead vs penetration rate – Scenario 2

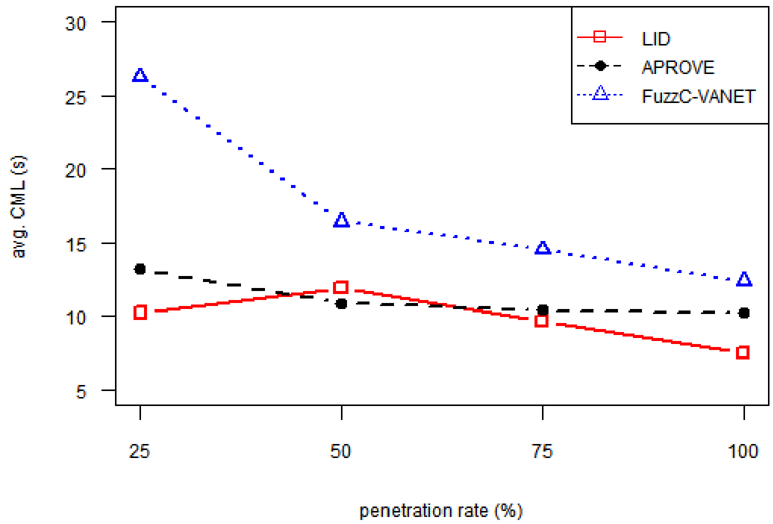


Figure 5.56. Avg. CML vs penetration rate – Scenario 2

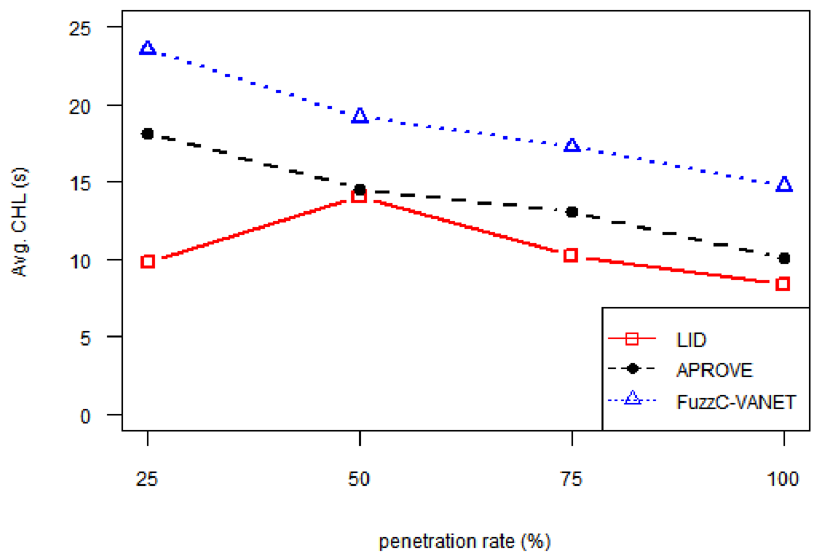


Figure 5.57. Avg. CHL vs penetration rate – Scenario 2

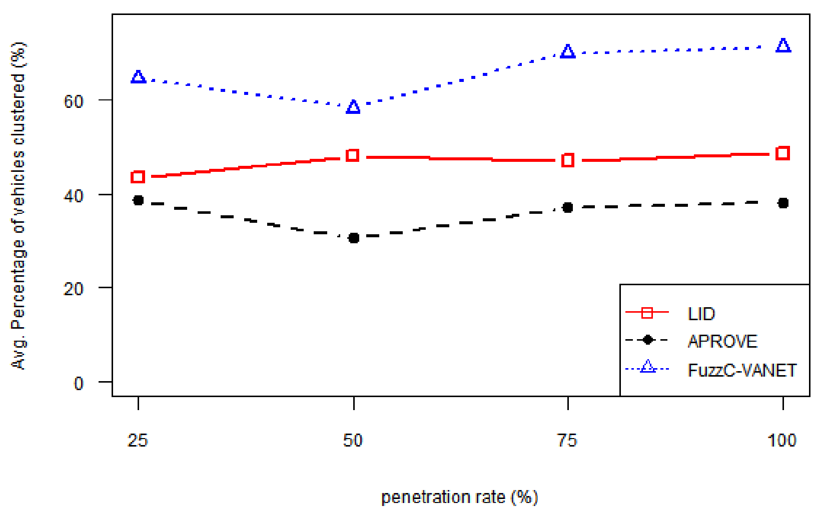


Figure 5.58. Avg. Percentage of vehicles clustered vs penetration rate – Scenario 2

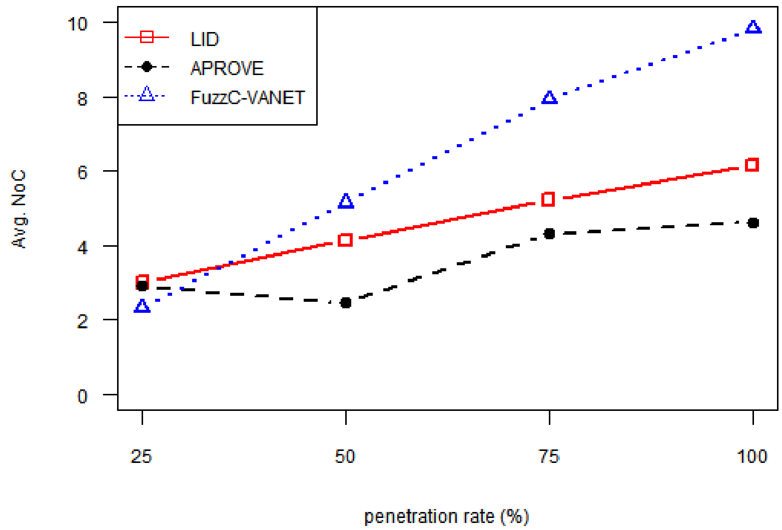


Figure 5.59. Avg. NoC vs penetration rate – Scenario 2

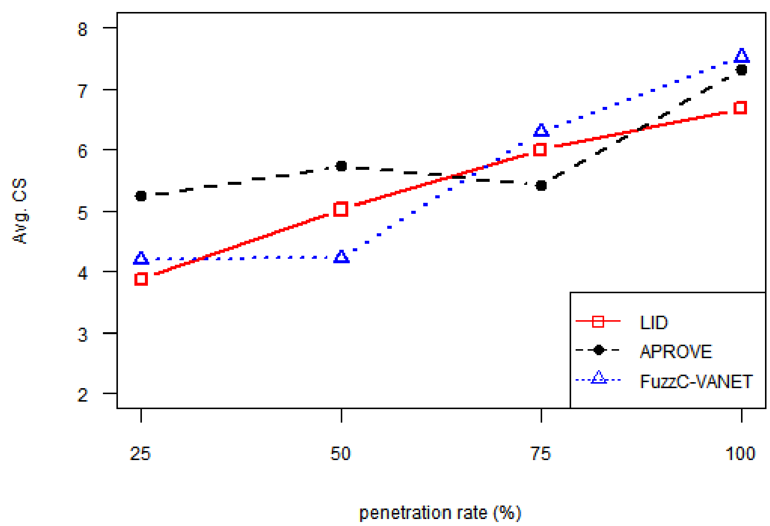


Figure 5.60. Avg. CS vs penetration rate – Scenario 2

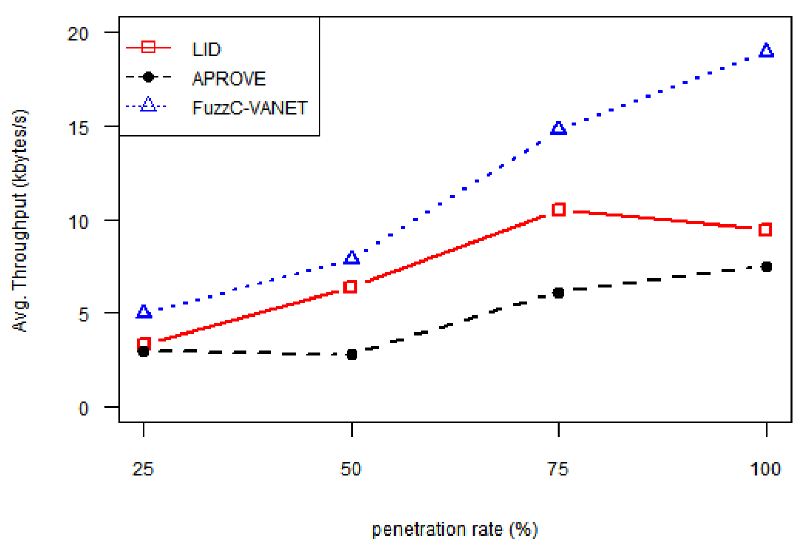


Figure 5.61. Avg. Throughput of the network vs penetration rate – Scenario 2

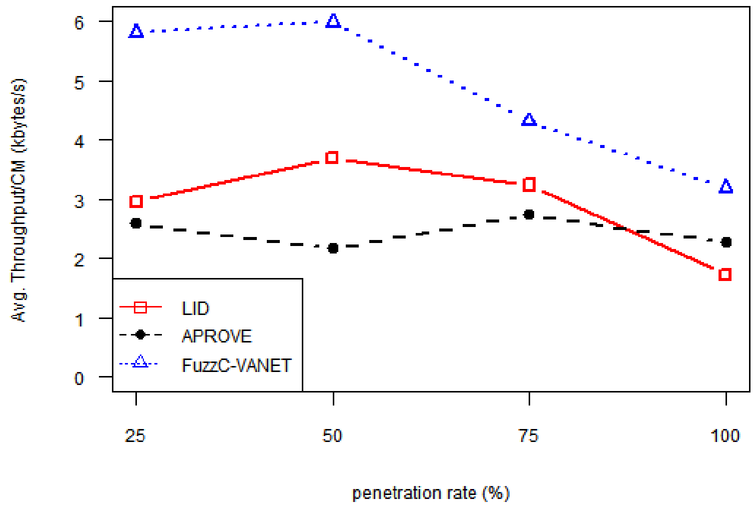


Figure 5.62. Avg. Throughput/CM vs penetration rate – Scenario 2

5.3.2.3. Impact of time interval controlling CH re-election process

In FuzzC-VANET algorithm description there is a time dependent variable which is the time interval (TI) that controls when the CH re-election process is triggered. In the previous tests TI was set to 5s. This section investigates the performances of FuzzC-VANET when TI is varied. Figure 5.63 – Figure 5.69 present the results obtained in terms of all metrics analyzed for both testing scenarios, Scenario 1 and Scenario 2, when TI is varied, taking the following values: 1s, 5s, 10s, 15s and 20s. The results show that the proposed algorithm is not very sensitive to this time interval, TI. For instance, the highest variation in case of avg. CML in Scenario 1 (Figure 5.63) is of 9%: the avg. CML is 9% less for TI = 20s than for TI=1s. The other variations for the same metric in Scenario1are of 6%, 4% and less than 1%. Similar for the other metrics, in both scenarios, there are very slight variations of their values when TI value is varied. For all the metrics, most of the variations are less than 5%, and all of them less than 10% with one exception: a 12% variation in the avg. CHL value, in Scenario 2 (Figure 5.64).

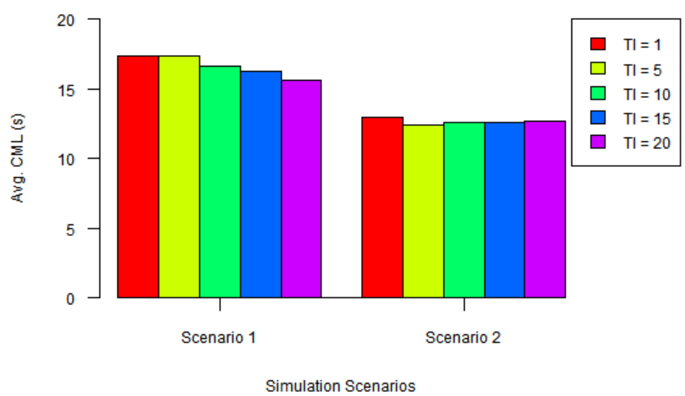


Figure 5.63. Avg. CML vs TI

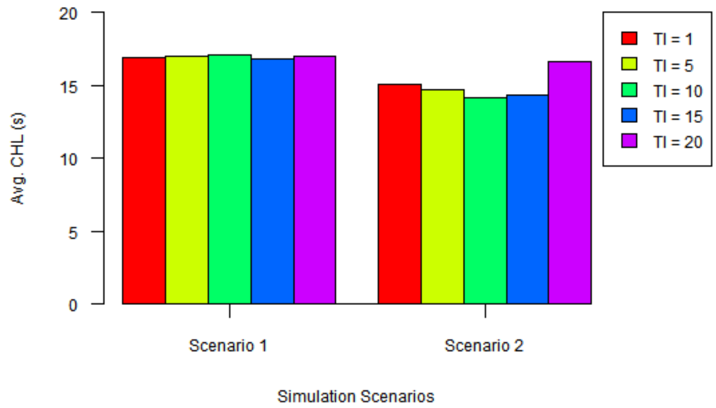


Figure 5.64. Avg. CHL vs TI

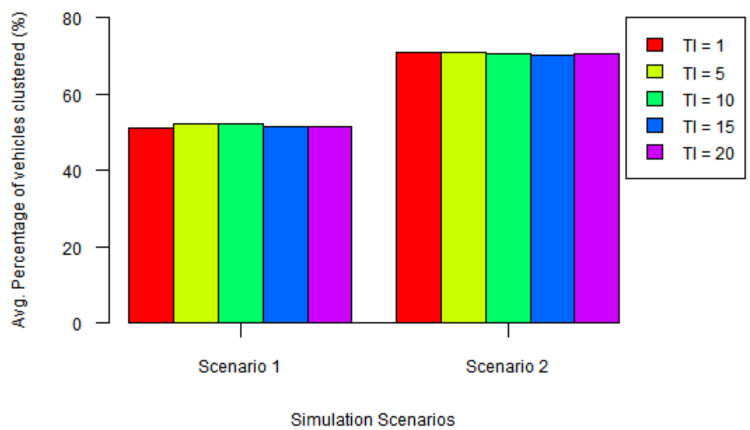


Figure 5.65. Avg. Percentage of vehicles clustered vs TI

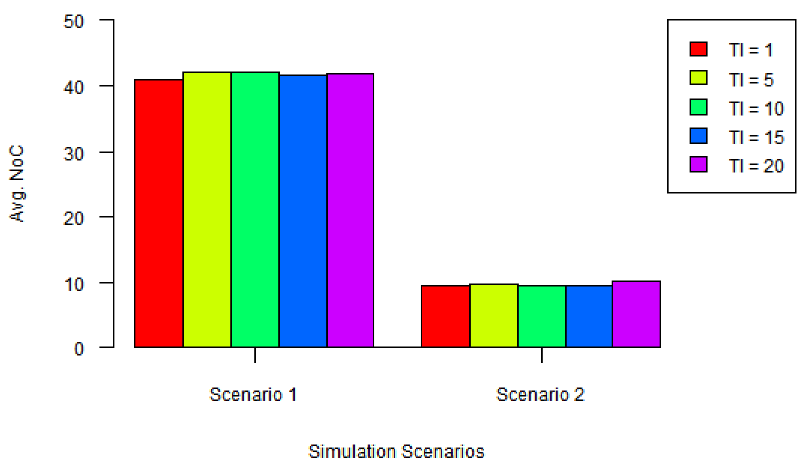


Figure 5.66. Avg. NoC vs TI

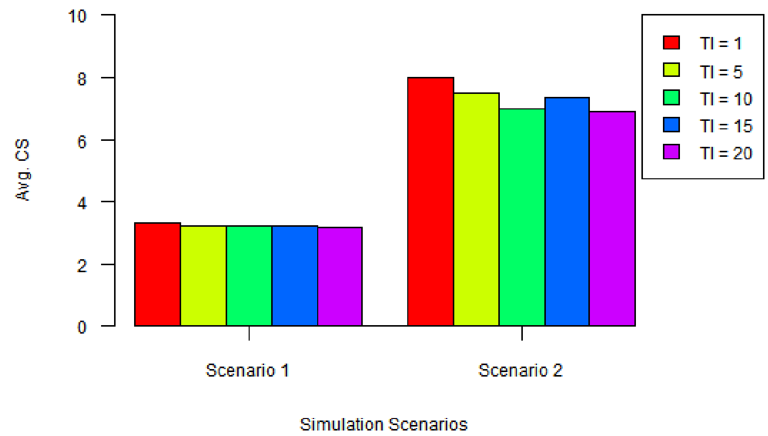


Figure 5.67. Avg. CS vs TI

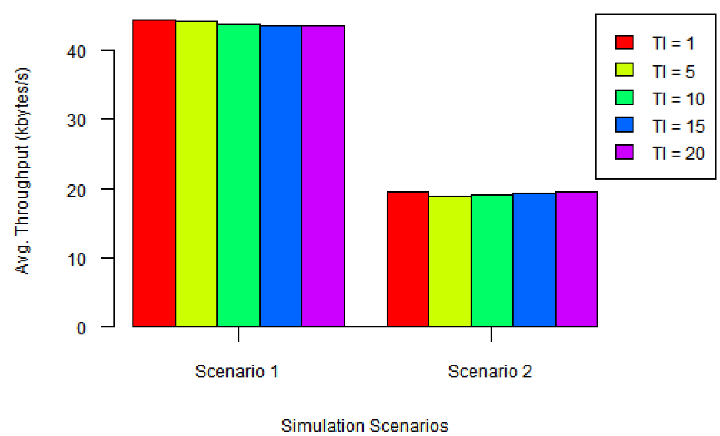


Figure 5.68. Avg. Throughput of the network vs TI

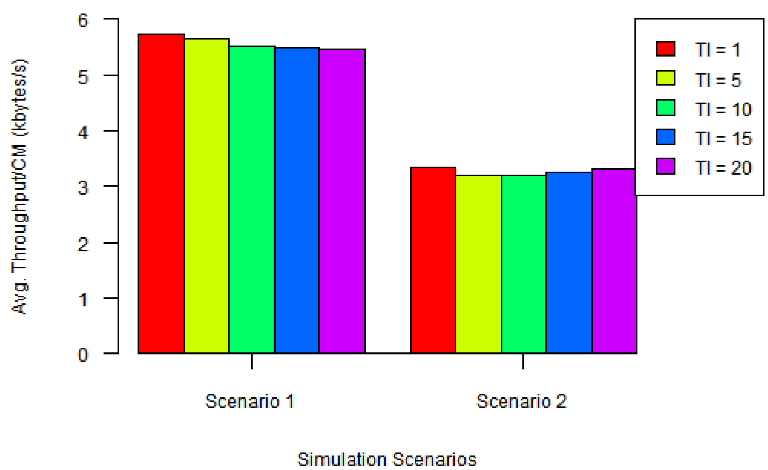


Figure 5.69. Avg. Throughput/CM vs TI

5.3.3. FuzzC-VANET Instantiation Testing

FuzzC-VANET is a generic VANET-dedicated clustering algorithm. An example of this algorithm instantiation on a hybrid VANET architecture and suitable for infotainment applications was given in chapter 4. In the clustering process a new metric is considered, the passengers interest in certain information. This section presents the evaluation of this instantiation named FuzzC-VANET*. First, FuzzC-VANET* is evaluated in the context of the two previously used testing scenarios, Scenario 1 and Scenario 2. The simulation settings are the same as presented in TABLE 5.8. Figure 5.70 – Figure 5.77 present the results obtained for all algorithms: LID, APROVE, FuzzC-VANET and include results for the instantiation FuzzC-VANET*. Note that the results for LID, APROVE, FuzzC-VANET are the same as discussed in section 5.3.2.1 and they are used in these figures for comparison purposes for FuzzC-VANET*. As these figures show, the latter has similar performance with its generalized version, FuzzC-VANET. The only metric that is drastically changed is the average overhead (Figure 5.70) that, for Scenario 1, is slightly higher than the APROVE overhead, the algorithm with the highest overhead so far among the three algorithms compared. In Scenario 2 the overhead is decreased as compared to APROVE. The increased overhead is a direct consequence of the extra metric used in clustering.

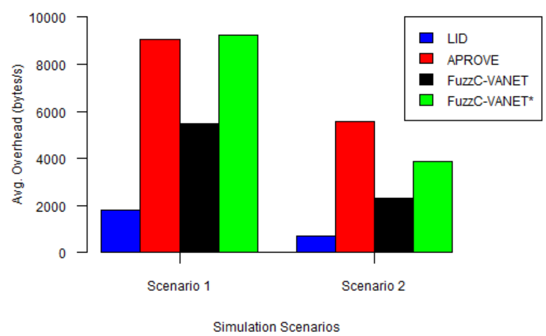


Figure 5.70. Avg. Overhead (including FuzzC-VANET*)

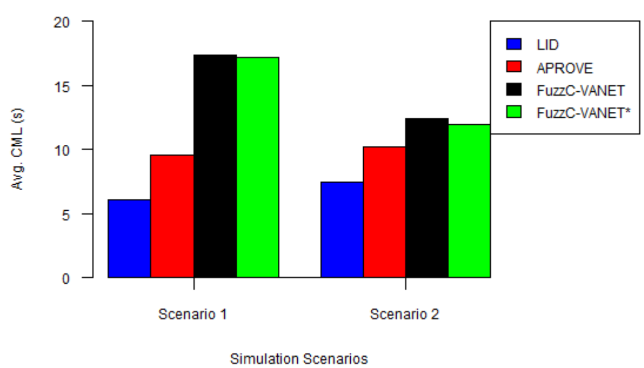


Figure 5.71. Avg. CML (including FuzzC-VANET*)

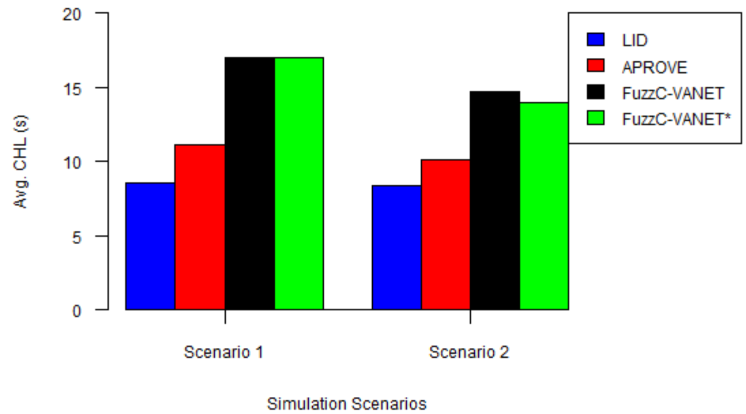


Figure 5.72. Avg. CHL (including FuzzC-VANET*)

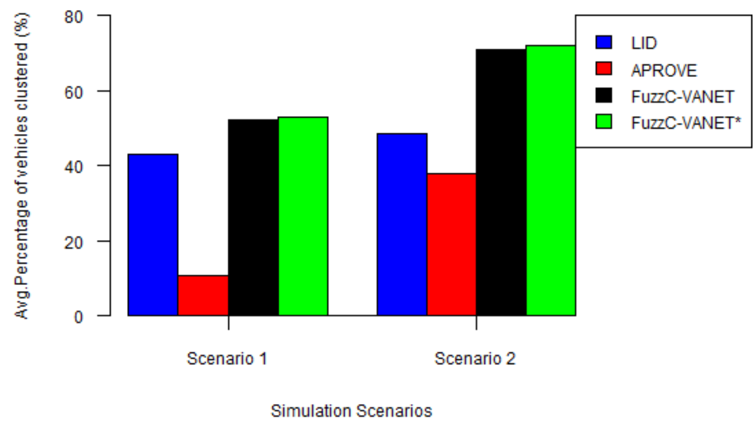


Figure 5.73. Avg. Percentage of vehicles clustered (including FuzzC-VANET*)

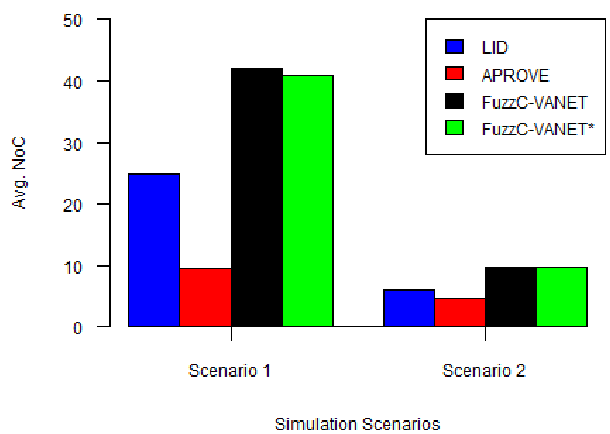


Figure 5.74. Avg. NoC (including FuzzC-VANET*)

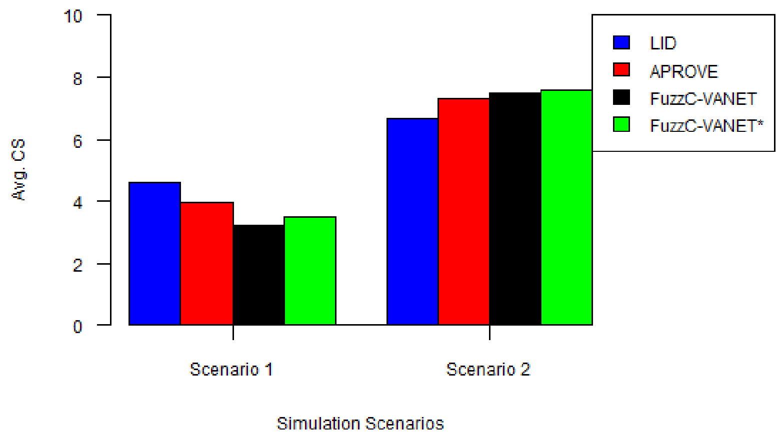


Figure 5.75. Avg. CS (including FuzzC-VANET*)

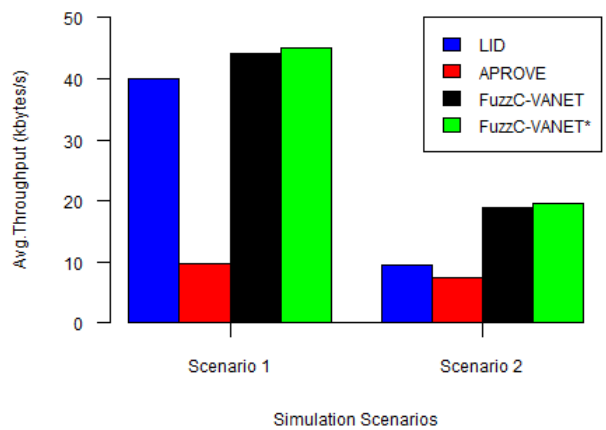


Figure 5.76. Avg. Throughput of the network (including FuzzC-VANET*)

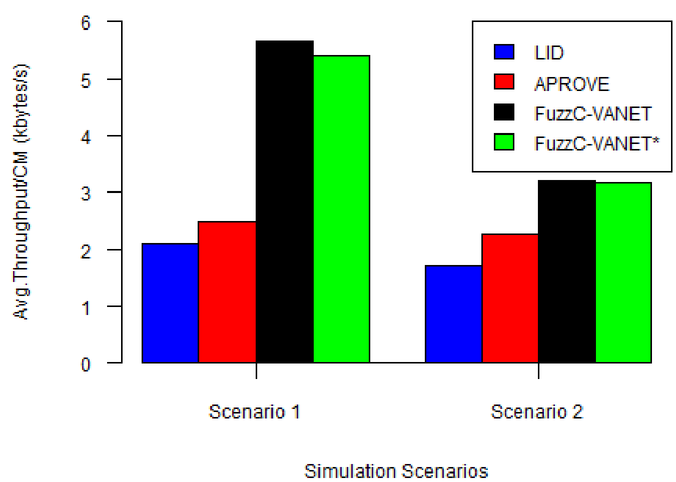


Figure 5.77. Avg. Throughput/CM (including FuzzC-VANET*

The results above show the performance of the FuzzC-VANET* compared to the others generic clustering algorithms and give a detailed overview to the intra-cluster performance. A third simulation scenario that considers the infrastructure as it is described in the architecture of FuzzC-VANET*, was chosen for a brief comparison between a non-clustered architecture and a clustered architecture based on the FuzzC-VANET* algorithm. Figure 5.78 presents this hybrid VANET scenario, named Scenario 3 that is made from two identical Manhattan grid like intersections. All the roads involved in this scenario have one lane for each way, excepting the last 100m before each intersection that has three lanes for one way. This latter aspect can be better noticed in Figure 5.79 that presents a detailed viewed on the intersections. At each of the 2 intersections in the scenario a RSU is placed. Tests have been performed considering a non-clustered architecture and a clustered network based on FuzzC-VANET* algorithm. In the first case, the two RSUs are broadcasting packets every second, while in the second, the packets are sent only to the CHs of the clusters. The transmission range of RSU is 250 m, while the transmission range between vehicles is 150m. The simulation was run for 10 minutes for both cases with a varied number of vehicles: 273, 293, and 327 vehicles. More detailed simulation settings can be found in TABLE 5.9. The results, summarized in TABLE 5.10 have shown a decrease of up to 13.3% in the number of packets being flooded in the network for the FuzzC-VANET-based clustered architecture as compared to the non-clustered architecture. Note that the clustering messages were also considered. On the other hand, an increase of up to 5% in the number of vehicles receiving the messages was obtained, meaning that the transmission range is enlarged by employing the proposed clustering scheme.

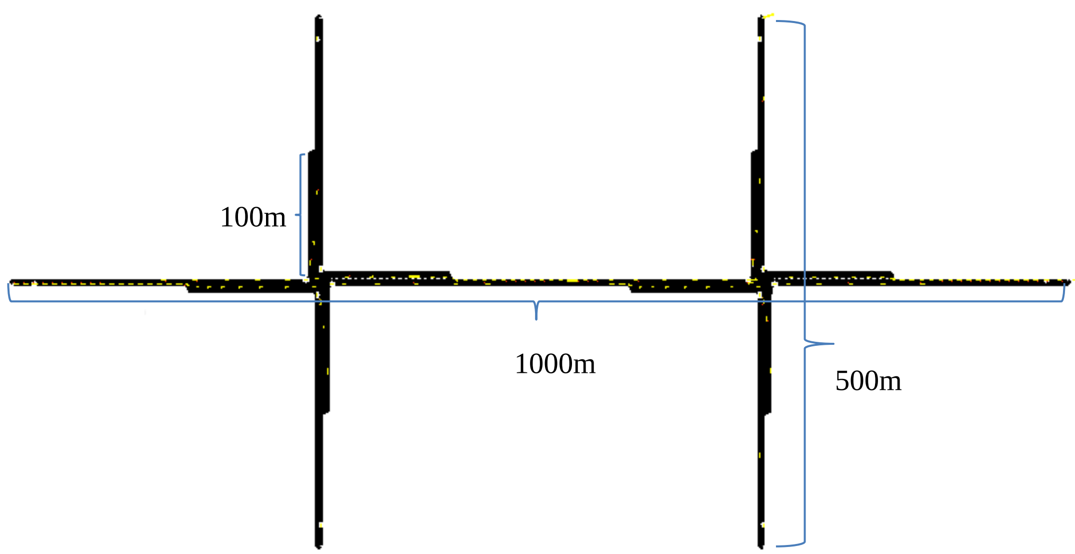


Figure 5.78. Scenario 3 – hybrid VANET scenario

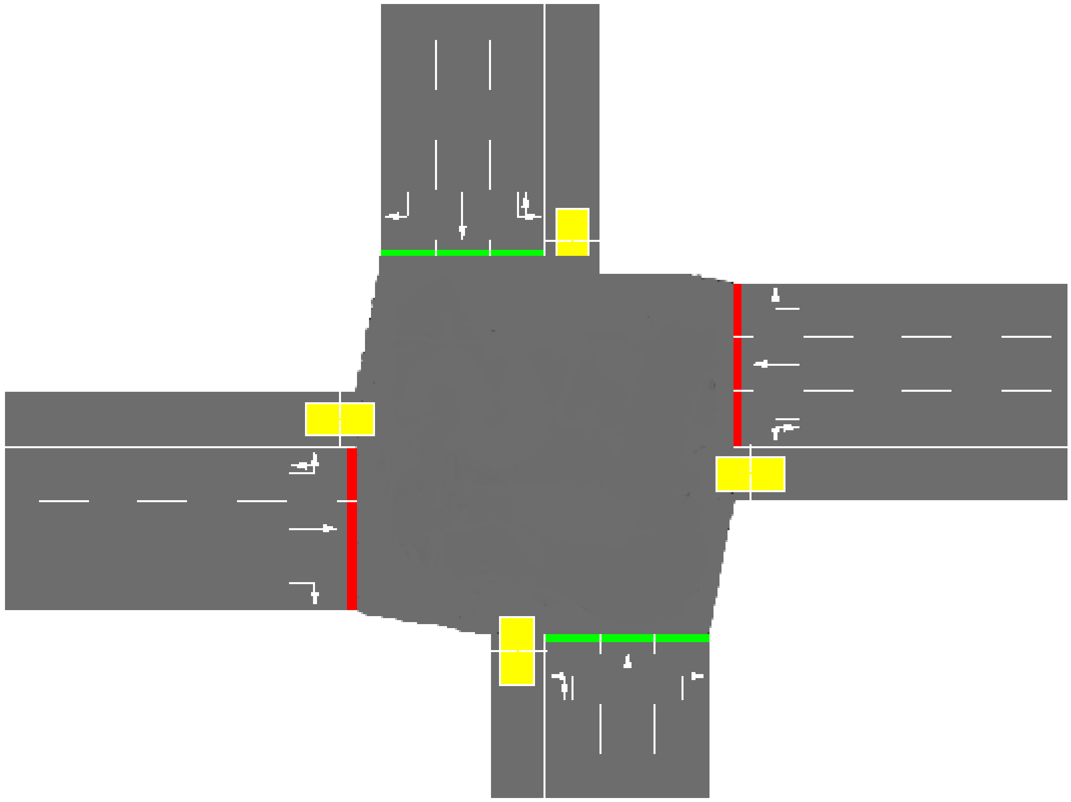


Figure 5.79. Zoom of Scenario 3 intersections

TABLE 5.10. RESULTS – CLUSTERED VS NON-CLUSTERED ARCHITECTURE

Number of vehicles	273	293	327
Reduction in number of packets flooded	5.5%	6.5%	13.3%
Increase in the number of vehicles receiving the messages	2%	5%	3%

5.4. Chapter Summary

This chapter has presented tests which evaluate the three proposed solutions in this thesis: SAECy, eWARPE and FuzzC-VANET.

Experimental results based on a real test-bed have shown that SAECy is leading to energy savings of up to 46% vs. the baseline (non-equipped bicycles) for the electric bicycles with assistance at start and up to 32% vs. the baseline for the electric bicycles without assistance at start. The test-bed was also used to validate the simulation model further employed for extensive testing on more complex scenarios. Simulations performed on the same scenarios used for the test-bed lead to comparable results. Due to logistics constraints, these scenarios include a single traffic light and follow the energy consumption on a relative short distance. Therefore more extensive testing was required to be performed using the validated simulation model. Considerably more complex scenarios were analyzed, on longer distances, with different number of traffic lights and different topologies. The solution was also compared against a classic GLOSA system proposed in the literature. Energy savings of up to 18% vs. the baseline have been obtained for the bicycles with assistance at start and up to 15% vs. the baseline for the bicycles without assistance at start. As compared to the classic GLOSA solution, our speed advisory system can increase the energy savings by up to 7%. In addition, an analysis on comfort-related metrics has shown that the proposed solution can also contribute to improving the cycling experience.

eWARPE also contributes to improving users experience while cycling as a survey performed both through interview and online questionnaire demonstrated. On a scale of 1 to 5 (no improvement at all to very big improvement), the 20 interviewed subjects rated in average the improvement brought by eWARPE with 4.1(big improvement). 60% of the 183 online survey participants responded that eWARPE is useful and very useful for improving their cycling experience. Moreover, eWARPE proved to have a positive influence on cyclists' motivation, 46% of the participants declared that eWARPE will make them much more likely to cycle. A case-study performed based on the input received from one of the interviewed subjects showed the benefits of eWARPE in terms of energy-efficiency.

Extensive simulation-based testing performed for highly realistic scenarios shows that FuzzC-VANET exceeds the performances of APROVE, a well-established VANET-dedicated clustering algorithm, and Lowest ID, a state-of-the-art clustering algorithm, in terms of highly known and used performance metrics that are employed in clustering assessment. When compared against a non-clustered architecture, the user-oriented FuzzC-VANET based clustered architecture increases the transmission range and decreases the number of packets flooded in the network up to 13.3% in the context of information dissemination.

Chapter 6

CONCLUSIONS AND FUTURE WORK

This chapter draws the conclusions of this thesis and indicates several directions that can be investigated in future research work.

6.1. Conclusions

6.1.1. Overview

Smart cities is a hot research topic for both academia and industry and focuses on making use of city facilities (buildings, infrastructure, transportation, energy, etc.) in order to improve people's quality of life and create a sustainable environment. Smart transportation, as a fundamental dimension of smart cities, relates to both intelligent and "green" transportation solutions. Cycling is one of the most sustainable and green forms of transportation. It can be the answer to many problems of the nowadays' society including large amounts of greenhouse gas emissions, traffic congestion, limited parking, etc. Therefore, it is not surprising that cycling occupies an important place among smart transportation initiatives in particular and smart cities initiatives in general. Lately, a modern form of cycling which uses electric bicycles has gained popularity. Electric bicycles improve the traditional riding experience, especially for the people who are not so fit, as it requires significantly less effort and can also lead to decreased travel time. In comparison with other green vehicles, electric bicycles have lower energy cost per distance travelled and avoid other additional costs (e.g. parking, insurance, registration, etc.). Consequently, it is not a surprising fact that electric bicycles are the most popular EVs and their popularity is increasing.

However, there are few disadvantages of bicycles in general and electric bicycles in particular. Weather conditions are affecting the cyclists the most among the traffic participants, bad weather conditions, such as wind and rainfall, are not pleasant for the rider and are identified as main de-motivators for cycling. Moreover, electric bicycles have a weak point related to the same aspect that makes them capable of providing some of the already mentioned advantages to the cyclists: the battery. The battery has a relative short autonomy, its range falling in the 16km-50km interval (range that is affected in time by the number of charges) and a long battery charging cycle of between 2 and 6 hours.

Vehicular ad hoc networks (VANET) or simply vehicular networks play a crucial role in supporting the creation of smarter cities. They are based on “smart” inter-vehicle communications and information exchange with the infrastructure via so called V2X communications (i.e. V2V – vehicle-to-vehicle and V2I/I2V – vehicle-to-infrastructure/infrastructure-to-vehicle). VANETs demonstrated their huge potential when designing not only intelligent transportation solutions and traffic management systems, but also green transportation solutions. The latter category was mostly focused on V2X communications-based solutions aiming to reduce fuel consumption and gas emissions. With the increased popularity of EVs, the focus has been recently moved on how V2X communications can help EVs save energy. So far, these solutions targeted exclusively the electric cars, but there is another category subscribing to the EVs class and this is represented by the electric bicycles. Moreover, most of the VANET research is dedicated to cars, although bicycles and even pedestrians are acknowledged in the context of vehicular networking research. The current trend is to integrate vehicles, bicycles and pedestrians in vehicular network research and ongoing efforts are been made in this direction.

In this context, the thesis proposes an Intelligent Advisory Solution for Bicycle Eco-riding and Eco-routing over VANET that provides on-route and off-route assistance for electric bicycles in particular and bicycles in general, energy savings being translated into reduced effort on behalf of the cyclist. Three main contributions are proposed in the context of this solution: **Speed Advisory System for Electric Bicycles (SAECy)**, an **Energy Efficient Weather-aware Route Planner solution for Electric Bicycles (eWARPE)** and a **Fuzzy Logic-based Clustering Scheme (FuzzC-VANET)** over VANETs. SAECy provides on-route assistance to the cyclists in order to improve their cycling experience and reduce the energy consumption in the particular case of electric bicycles. SAECy exploits mainly the I2V communication for obtaining traffic light related information, but also weather information. eWARPE provides off-route assistance to the cyclists aiming to support them in avoiding the adverse weather conditions, but also to save the battery in the case of electric bicycles. FuzzC-VANET is a generic clustering scheme dedicated for VANET that can be employed for information dissemination in SAECy's context in order to enhance its performances and to make better use of weather information. VANET is an emergent technology with huge potential in the context of smart transportation in general and in supporting cycling in particular. However, before the full adoption and deployment of this technology, there are several challenges that need to be addressed. Scalability and stability are numbered among these main challenges. FuzzC-VANET is also a response to these issues.

6.1.2. Contributions – Summary

SAECy represents novel vehicular communications-based speed advisory system dedicated to electric bicycles. The solution subscribes to the class of Green Light Optimal Speed Advisory (GLOSA) systems based on the I2V communications and is the first GLOSA system dedicated to electric bicycles. SAECy recommends strategic riding (i.e. the appropriate speed) when bicycles are approaching an intersection to avoid high power consumption scenarios. Moreover, the approach also includes an innovative Fuzzy Logic-based wind-aware speed adaptation policy as among all the other vehicles, bicycles are mostly affected by the wind. The benefits of the solution translate not only in energy-efficiency, but also in an increased user experience, as the waiting times at traffic lights are reduced or even avoided. Experimental results based on a real test-bed have shown that SAECy results in energy savings of up to 46% vs. the baseline (non-equipped bicycles) for the electric bicycles with assistance at start and up to 32% vs. the baseline for the electric bicycles without assistance at start. The test-bed was also used to validate the simulation model further employed for extensive testing on more complex scenarios. Simulations performed on the same scenarios used for the test-bed lead to comparable results. Due to logistics constraints, these scenarios include a single traffic light and they follow the energy consumption on a relative short distance. Therefore more extensive testing was required to be performed using the validated simulation model. Considerable more complex scenarios were analyzed, on long distances, with different number of traffic lights and different topology. The solution was also compared against a classic GLOSA system proposed in the literature. Energy savings of up to 18% vs. the baseline have been obtained for the bicycles with assistance at start and up to 15% vs. the baseline for the bicycles without assistance at start. As compared to the classic GLOSA solution, our speed advisory system can increase the energy savings by up to 7%. In addition, an analysis on comfort-related metrics has shown that the proposed solution can also contribute to improving the cycling quality of experience.

eWARPE represents a step forward for the cycling route planners, going beyond planning the route itself (how to get from point A to point B). eWARPE is planning the optimal departure time for the route: when to leave from point A towards point B on the previously planned route. The solution makes use of the weather information in order to recommend the departure time that allows the cyclist to avoid adverse weather conditions as much as possible and to increase the energy savings of the electric bicycle. eWARPE also contributes to improving users quality of experience while cycling as a survey performed both through interview and online questionnaire demonstrated. On a scale of 1 to 5 (no improvement at all to very big improvement), the 20 interviewed subjects rated in average the improvement brought by eWARPE with 4.1(big improvement). 60% of the 183 online survey participants responded that eWARPE is useful and very useful for improving their cycling experience. Moreover, eWARPE proved to have a positive influence on cyclists' motivation, 46% of the participants declared that eWARPE will make them much more likely to cycle. A case-study performed based on the input received from one of the interviewed subjects demonstrated the benefits of eWARPE in terms of energy-efficiency.

FuzzC-VANET is a general FL-based CH-based clustering scheme over VANETs, the first, to the best of our knowledge, to employ FL as the main decision tool in choosing the CH. Creating a clustered network model, this scheme solves some of the main issues of VANET, namely scalability and stability issues. As a consequence of this, the V2V communications in the network have an increased throughput. FuzzC-VANET clustering scheme creates a base for MAC, routing protocols and information dissemination. In the latter case, it is known that clusters contain local-based and highly up-to date information. In disseminating weather information, these two characteristics are highly important and in this context it is to be mentioned that there are quite a few applications that are based on VANETs capacity of providing weather information, one of this being SAECy. Extensive simulation-based testing performed for highly realistic scenarios shows that FuzzC-VANET exceeds the performances of APROVE [121], a well-established VANET-dedicated clustering algorithm, and Lowest ID [217], a state-of-the-art clustering algorithm, in terms of highly known and used performance metrics that are employed in clustering assessment. When compared against a non-clustered architecture, the user-oriented FuzzC-VANET based clustered architecture increases the transmission range and decreases the number of packets flooded in the network in the context of information dissemination.

6.2. Future Work

The section indicates possible new directions that can evolve from the work presented in this thesis into future research work.

In the context of improving mobility modelling and simulations, the creation of micro-simulation models for bicycles and cyclists and their integration with the existing vehicular traffic is a work in progress. Based on this work in progress, new steps can then be taken in the context of the proposed solution such as the integration of the bicycles in the clustered network and receiving the information via V2V communication. Future works also include the study of the proposed solution for multiple bicycles using cycling lanes, considering different factors such as: penetration rate of the technology, compliance rate and other characteristics that can be modelled in the context of the cyclist behaviour based on the future micro-simulation models for the cyclists. These micro-simulation models for bicycles should consider the wind for the bicycle's dynamic as its influence on the bicycles is greater than in the case of cars. Based on these advancements, the proposed solution could also enhance its own bicycle dynamics model. Further on, the proposed solution can be studied for the bicycles using the same lanes as other types of vehicles. Field-tests can also be extended considering other electric bicycles and more complex routes, scenarios and wind conditions.

An interesting research direction can derive then from the fact that the speed adapted to the traffic light conditions or weather conditions is currently changed by the cyclist at SAECy recommendation. Technical solutions can be further developed to automate this process of changing the speed and not to be left in the responsibility of the cyclist.

The on-line assistance provided to the cyclists by the proposed solution could be further extended in order to consider real-time routing depending on different factors such as traffic conditions, road conditions, emergency situations, but also weather conditions. All the information would be collected and delivered in real-time via V2X communications. This extension will allow the cyclist to avoid congested roads or to choose the less congested roads, to avoid for instance traffic incidents and the congestion involved by these or other types of incidents that would affect the road. Also being provided with real-time conditions about the roads, roads with problems could be avoided. Note that traffic density and traffic congestion for cyclists must be differently considered and understood as compared to the regular vehicles and this is one of the open issues that are expected to be solved by the models currently developed in order to integrate bicycles in the vehicular traffic. This is for instance caused by the fact that bicycles have dedicated facilities, such as cycling lanes, and the taking over model is quite different also. The off-line assistance could also be extended in order to take into consideration these factors, especially real-time traffic conditions and to provide an initial route based on these. For this, a solution should be proposed in order to manage the information gathered via V2X information and make it available via Internet.

In the context of the solution extensions mentioned in the above paragraph, another future work can be identified that is related to modelling user preferences. For instance, real-time routing and off-line assistance could be such designed to give the cyclists the possibility to choose and prioritize the factors that influence the routing decisions.

In the online survey performed, the participants had the option to leave some suggestions related to the features they would like to add to the proposed solution. Some of the suggestions were: “An option showing where the traffic is heavy”, “make it adapt to the user so that it takes into account user preferences in terms of cycling, include traffic avoidance”. It can be seen that these are in-line with the above indicated future works. Other suggestions related to the off-route assistance of the cyclist can open the perspectives on other future directions such as for instance the integration with other sustainable means of transportation. This idea is suggested by the following comment left by a participant: “if the weather conditions are too bad such as winds above a certain speed or flooding, there should be a warning built in to express the great risk with the journey so a bus timetable could be given”. Here choosing another mean of transportation is conditioned by the extreme weather conditions. Another participant conditioned travelling by bicycle on the dress code and suggests an interesting approach that would require taking the user modeling to another level: “if for instance a smart dressing is required then to the user integration with my agenda (calendar, meetings) and if no required dress code than message inviting me to cycle”.

REFERENCES

[1] S. Djahel, R. Doolan, G.-M. Muntean, J. Murphy, "A Communications-oriented Perspective on Traffic Management Systems for Smart Cities: Challenges and Innovative Approaches", *IEEE Communications Surveys & Tutorials*, 17, no. 1, pp: 125-151, 2015.

[2] The Organisation for Economic Co-operation and Development, OECD, "The cost of air pollution: Health Impact of Road Transport", OECD Publishing, 2014; online: http://www.oecd.org/env/the-cost-of-air-pollution-9789264210448-en.htm

[3] EPA United States Environmental Protection Agency, "US Greenhouse Gas Inventory Report: 1990 – 2013, [online] http://www3.epa.gov/climatechange/ghgemissions/usinventoryreport.html , National Greenhouse Emissions Data, 2014.

[4] European Innovation Partnership on Smart Cities and Communities, "Strategic Implementaion Plan", [online]: http://ec.europa.eu/eip/smartcities/files/sip_final_en.pdf, 2013.

[5] European Innovation Partnership on Smart Cities and Communities, "Operational Implementaion Plan: First Public Draft", [online]: http://ec.europa.eu/eip/smartcities/files/operational-implementation-plan-oip-v2_en.pdf, 2013.

[6] European Conference of Ministers of Transport, Implementing Sustainable Urban Travel Policies: Moving Ahead, "National Policies to Promote Cycling", ISBN 92-821-2325-1, 2004.

[7] K. Martens, "Promoting bike-and-ride: The Dutch experience", *Transportation Research Part A: Policy and Practice* 41, no. 4, pp: 326-338, 2004.

[8] J. Pucher, L. Dijkstra, "Promoting safe walking and cycling to improve public health: lessons from the Netherlands and Germany", *American journal of public health* 93, no. 9, pp: 1509-1516, 2003.

[9] E. Murphy, J. Usher, "The role of bicycle-sharing in the city: Analysis of the Irish experience", *International Journal of Sustainable* Transportation, vol. 9, no. 2, pp: 116-125, 2015.

[10] M. Ahillen, D. M.-Babiano, J.Corcoran, "The Dynamics of Bike-Sharing in Washington, DC and Brisbane, Australia: Implications for Policy and Planning", *International Journal of Sustainable Transportation*, in press (2015).

[11] "BYPAD Audit of Dublin City Council: Final Report [v2], 26.01.2012", http://cyclingindublin.com/wp-content/uploads/2012/08/BYPAD-Final-27-Jan-2012.pdf, 2012.

[12] M. Alsabaan, W. Alasmary, A. Albasir, K. Naik, "Vehicular Networks for a Greener Environment: A Survey", *IEEE Communications Surveys & Tutorials*, Vol. 15, no. 3, pp. 1372-1388, 2013.

[13] P.M. d'Orey, M. Ferreira, "ITS for Sustainable Mobility: A Survey on Applications and Impact Assessment Tools", *IEEE Trans. on Intelligent Transportation Systems*, Vol. 15, no. 2, pp. 477-493, 2014.

[14] C. Sommer, F. Dressler, "Vehicular Networking", *Cambridge University Press*, ISBN 978-1-107-04671-9, pp. 100, 2015.

[15] Connecting Mobility, M. van Sambeek, F. Ophelders, T. Bijlsma, B. van der Kluit, O. Türetken, R. Eshuis, K. Traganos, P. Grefen, "Towards an Architecture for Cooperative ITS Applications in Netherlands", working paper, 2015.

[16] J. González, F. Morte, T. Paadín, M. Alonso, O. Martín, S. Martín, K. Malone, E. Van Dam, M. Van Sambeek, K. Moerman, VRUITS project, "Deliverable 5.2, Trials Applications and infrastucture", 2015.

[17] M. Winters, G. Davidson, D. Kao, K. Teschke, "Motivators and deterrents of bicycling: comparing influences on decisions to ride" , *Transportation*, Vol.38, no. 1, pp. 153-168, 2011.

[18] Y. Ruan, C.C. Hang, Y. M. Wang, "Government′ s role in disruptive innovation and industry emergence: The case of the electric bike in China", *Technovation*, Vol. 34, no. 12, pp. 785-796, 2014.

[19] Navigant Research, Smart Transportation, Electric Bicycles, 2013, online: http://www.navigantresearch.com/tag/electric-bicycles?page=2

[20] S. Ji, C. R. Cherry, M. J. Bechle, Y. Wu, J. D. Marshall, "Electric vehicles in China: emissions and health impacts", *Environmental science & technology*, Vol. 46, no. 4, pp.: 2018-2024, 2012.

[21] A. Muetze, Y. C. Tan, "Electric Bicycles - a Performance Evaluation", *IEEE Industry Applications Magazine*, Vol 13, no. 4, pp. 12-21, 2007.

[22] H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164-171, 2008.

[23] H. Moustafa, S. M. Senouci, M. Jerbi, "Introduction to Vehicular Networks", *Vehicular Networks Techinques, Standards and Applications*, CRC Press, 2009.

[24] K.C. Lee, U. Lee, M. Gerla, "Survey of Routing Protocols in Vehicular Ad Hoc Networks", *Advances in Vehicular Ad-Hoc Networks: Developments and Challenges*, IGI Global 2010.

[25] H. Moustafa, S. M. Senouci, M. Jerbi, "Introduction to Vehicular Networks", *Vehicular Networks Techinques, Standards and Applications*, CRC Press, 2009.

[26] G. Araniti, C. Campolo, M. Condoluci, A. Iera, A. Molinaro, "LTE for vehicular networking: a survey", *IEEE Communications Magazine*, 51(5), pp. 148-157, 2013.

[27] A. Goldsmith, "Beyond 4G: What Lies Ahead for Cellular System Design", Keynote presentation at *IEEE Wireless Communications and Networking Conference (WCNC)*, 2012.

[28] R. A. Uzcategui, WAVE: A Tutorial, *IEEE Communications Magazine*, pp. 126-133, May 2009.

[29] IEEE 802.11p, “Draft Amendment to Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks- Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications-Amendment 6: Wireless Access in Vehicular Environment", 2010.

[30] IEEE P1609.1, "Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Resource Manager", 2006.

[31] IEEE P1609.2, "Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) ― Security Services for Applications and Management Messages", 2006.

[32] IEEE Std P1609.3, "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)-Networking Services", 2007.

[33] IEEE P1609.4, "Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) ― Multi-Channel Operation", 2006.

[34] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin,T. Weil, “Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions”, *IEEE Communications Surveys Tutorials*, Vol. 13, no. 4 pp. 584 –616, 2011.

[35] ITS JPO, “Vehicle safety applications,” US DOT IntelliDrive(sm) Project - ITS Joint Program Office, Tech. Rep., 2008.

[36] SAFESPOT D8.4.4, “Use cases, functional specifications and safety margin applications for the SAFESPOT Project,” IST Safespot Project, Tech. Rep. Safespot IST-4-026963-IP deliverable D8.4.4, pp. 1-54, 2008.

[37] PREDRIVE D4.1, “Detailed description of selected use cases and corresponding technical requirements,” IST PreDrive C2X project, Tech. Rep. PreDrive C2X deliverable D4.1, 2008.

[38] VSC, “Final Report,” US DOT, Vehicle Safety Communications Project DOT HS 810 591, April 2006.

[39] VSC-A, “Final Report,” US DOT, Vehicle Safety Communications Applications (VSC-A) Project DOT HS 810 073, 2009 January 2009.

[40] M. Fogue, P. Garrido, F. J. Martinez, J.-C. Cano, “An Adaptive System Based on Roadmap Profiling to Enhance Warning Message Dissemination in VANETs”, *IEEE/ACM Transactions on Networking*, 2012.

[41] F.J. Martinez, C.-K. Toh, J.-C. Cano, C.T. Calafate, P. Manzoni, “Emergency Services in Future Intelligent Transportation Systems Based on Vehicular Communication Networks.”, *IEEE Intelligent Transportation Systems Magazine* 2, pp. 6-20, 2010.

[42] S. Djahel, M. Salehie, I. Tal, P. Jamshidi, "Adaptive Traffic Management for Secure and Efficient Emergency Services in Smart Cities", *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops),* 2013.

[43] J. Ploeg, A.F.A. Serrarens, G.J. Heijenk, "Connect & Drive: design and evaluation of cooperative adaptive cruise control for congestion reduction", *Journal of Modern Transportation*, 19 (3). pp. 207-213, 2011.

[44] Z. Yang, M. Li, W. Lou, "CodePlay: Live Multimedia Streaming In VANETs using Symbol-Level Network Coding", *IEEE International conference on Network Protocols*, pp. 223-232, 2010.

[45] A. Razzaq, A. Mehaoua, "Video Transport over VANETs: Multi- Stream Coding with Multi-Path and Network Coding", *IEEE Conference on Local Computer Networks*, pp. 32-39, 2010.

[46] H. Hartenstein, B. Bochow, M. Lott, A. Ebner, M. Radimirsch, D. Vollmer, "Position-aware ad hoc wireless networks for inter-vehicle communications: the Fleetnet project", *ACM MobiHoc*, pp. 259–262, 2001.

[47] O.K. Tonguz, M. Boban, "Multiplayer games over vehicular ad hoc networks: A new application", *Ad Hoc Networks*, Vol. 8, pp. 531-543, 2010.

[48] H. Rakha, R. K. Kamalanathsharma, "Eco-driving at signalized intersections using V2I communication", *14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 341-346, 2011.

[49] M. Barth, S. Mandava, K. Boriboonsomsin, H. Xia, "Dynamic ECO-driving for arterial corridors", *IEEE Forum on Integrated and Sustainable Transportation System (FISTS)*, pp. 182-188, 2011.

[50] R. Doolan, G.-M. Muntean, "VANET-enabled Eco-friendly Road Characteristics-aware Routing for Vehicular Traffic", *IEEE Vehicular Technology Conference*, pp. 1-5, 2013

[51] K. Collins, G-M. Muntean, "A vehicle route management solution enabled by Wireless Vehicular Networks", *IEEE INFOCOM Workshops*, pp. 1-6, 2008.

[52] K. Collins, G-M. Muntean, "Route-based vehicular traffic management for wireless access in vehicular environments", *IEEE Vehicular Technology Conference*, pp. 1-5, 2008.

[53] K. Collins, Gabriel-Miro Muntean, "An Adaptive Vehicle Route Management Solution Enabled by Wireless Vehicular Networks", *IEEE Vehicular Technology Conference*, pp. 1-5, 2008.

[54] M. Caliskan, D. Graupner, M. Mauve, "Decentralized Discovery of Free Parking Places," *ACM International Workshop on Vehiular Ad Hoc Networks*, pp. 30–39, 2006.

[55] R. Lu, X. Lin, H. Zhu, X. Shen, "SPARK: a new VANET-based smart parking scheme for large parking lots", *IEEE INFOCOM*, pp. 1413–1421, 2009.

[56] G. Xu, "The research and application of RFID technologies in highway's electronic toll collection system", *IEEE Wireless Communications, Networking and Mobile Computing*, pp. 1-4, 2008.

[57] R. Zhengang, G. Yingbo, "Design of electronic toll collection system in expressway based on RFID", *IEEE Environmental Science and Information Application Technology,* vol. 3, pp. 779-782. IEEE, 2009.

[58] H. Hartenstein, K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164-171, 2008.

[59] P. Santi, “Mobility Models for Next Generation Wireless Networks ”, Wiley, ISBN 9781119992011, pp 143-151, 2012.

[60] S. Lancaster, “Fuzzy Logic Controllers”, Maseeh College of Engineering and Computer Science at PSU, Portland, 2008.

[61] J. M. Mendel, “Fuzzy logic systems for engineering: a tutorial”, *Proceedings of the IEEE*, vol. 83, no. 3, pp. 345-377, 1995.

[62] S. Ghosh, Q. Razouqi, H. J. Schumacher, A. Celmins, "A survey of recent advances in fuzzy logic in telecommunications networks and new challenges", *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 3, pp. 443-447, 1998.

[63] L. Wang, G. Kuo, "Mathematical Modeling for Network Selection in Heterogeneous Wireless Networks—A Tutorial", *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 271-292, 2013.

[64] R.V. Kulkarni, A. Forster, G. K. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey”, *IEEE Communications Surveys & Tutorials*, vol. 13, no. 1, pp. 68-96, 2011.

[65] Gavrilovska, V. Atanasovski, I. Macaluso, L. DaSilva, "Learning and Reasoning in Cognitive Radio Networks", *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1761-1777, 2013.

[66] M. Bkassiny, Y. Li, S. Jayaweera, "A Survey on Machine-Learning Techniques in Cognitive Radios", *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1136-1159, 2013.

[67] T. J. Ross, “Fuzzy Logic with Engineering Applications”, 3rd edition, ISBN 978-0-470-743746-8, 2010.

[68] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I.", *Information sciences*, vol. 8, no. 3, pp. 199-249, 1975.

[69] R. Bauza, J. Gozalvez, "Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications", *Journal of Network and Computer Applications*, Vol. 36, pp. 1295-1307, 2013.

[70] R. Bauza, J. Gozalvez, J. Sanchez-Soriano, "Road traffic congestion detection through cooperative vehicle-to-vehicle communications", *IEEE Local Computer Networks*, pp. 606-612, 2010.

[71] S. Dietzel, B. Bako, E. Schoch, F. Kargl, “A Fuzzy Logic based Approach for Structure-free Aggregation in Vehicular Ad-hoc Networks”, *ACM International Workshop on VehiculAr InterNETworking*, 2009.

[72] E.H.Mamdani, S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1-13, 1975.

[73] T. Takagi , M. Sugeno, “Fuzzy identification of systems and its application to modeling and control”, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, pp. 116–132, 1985.

[74] Y. Tsukamoto, “An approach to fuzzy reasoning method”, M.M. Gupta, R.K. Ragade, R.R. Yager (Eds.), *Advances in Fuzzy Set Theory and Application*, North-Holland, pp. 137–149, 1979.

[75] P.M. Larsen, “Industrial applications of fuzzy logic control”, International Journal of Man-Machine Studies, vol. 12, no.1, pp. 3-10, 1980.

[76] N. Siddique, H. Adeli, “Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing”, John Wiley and Sons, ISBN: 978-1-1183-3784-4, 2013.

[77] Major highway performance ratings and bottleneck inventory - State of Maryland – Spring 2008, Skycomp, Inc. in association with Whitney, Bailey, Cox and Magnani, LLC, November, 2009

[78] Y. He, G.-P. Liu, D. Rees, and M. Wu, "H∞ filtering for discrete-time systems with time-varying delay" *Signal Processing*, vol. 89, no.3, pp. 275-282, 2009.

[79] FuzzyPLC, available online: http://fuzzyplc.sourceforge.net/

[80] Matlab Fuzzy Logic Toolbox, official website: http://www.mathworks.co.uk/products/fuzzy-logic/index.html

[81] C. Wagner, S. Miller, J.M. Garibaldi, “A Fuzzy Toolbox for the R Programming Language.”, *IEEE International Conference on Fuzzy Systems*, pp. 1185 –1192, 2011.

[82] L. Markowsky, B. Segee, “The Octave Fuzzy Logic Toolkit”, *International Workshop on Open-Source Software for Scientific Computation*, pp. 118 –125, 2011.

[83] P. Cingolani, J. Alcala-Fdez, "jFuzzyLogic: a robust and flexible Fuzzy-Logic inference system language implementation", *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1-8, 2012.

[84] jFuzzyQT, available online: http://sourceforge.net/projects/jfuzzyqt/

[85] I. Baturone, F. J. Moreno-Velo, S. Sánchez-Solano, Á. Barriga, P. Brox, A. A. Gersnoviez, M. Brox, "Using xfuzzy environment for the whole design of fuzzy systems", *IEEE International Fuzzy Systems Conference (FUZZ-IEEE)*, pp. 1-6, 2007.

[86] R. Stanica, E. Chaput, A.-L. Beylot, "Simulation of vehicular ad-hoc networks: Challenges, review of tools and recommendations", *Computer Networks*, vol. 55, no. 14, pp. 3179-3188, 2011.

[87] L. Wanner,"Introduction to Clustering Techniques", International Union of Local Authorities, 2004.

[88] Yu, J.Y., P.H.J. Chong, “A Survey of Clustering Schemes for Mobile Ad Hoc Networks” *IEEE Communications Surveys Tutorials*, Vol. 7, pp. 32 –48, 2005.

[89] H. Su, X. Zhang, “Clustering-Based Multichannel MAC Protocols for QoS Provisionings Over Vehicualr Ad Hoc Networks”, *IEEE Transactions on Vehicular Technology*, Vol 56, No. 6, pp. 3309-3323, 2007.

[90] Y. Gunter,B. Wiegel, H.P. Grossmann, “Medium Access Concept for VANETs Based on Clustering”, *IEEEVehicular Technology Conference (VTC-2007 Fall)*, pp. 2189 –2193, 2007.

[91] Y-C. Lai, P. Lin, W. Liao, C-M. Chen, “A Region-Based Clustering Mechanism for Channel Access in Vehicular Ad Hoc Networks”, *IEEE Journal on Selected Areas in Communications*, Vol 29, pp. 83 –93, 2011.

[92] K. A. Hafeez, L. Zhao, Z. Liao, B. N. Ma, “Clustering and OFDMA-based MAC protocol (COMAC) for vehicular ad hoc networks”, *EURASIP Journal on Wireless Communications and Networking*, 2011.

[93] S. Teshima,T. Ohta, E. Kohno, Y. Kakuda, “A Data Transfer Scheme Using Autonomous Clustering in VANETs Environment”, *IEEE International Symposium on Autonomous Decentralized Systems (ISADS)*, pp. 477 –482, 2011.

[94] R. T. Goonewardene, F.H. Ali, E. Stipidis, “Robust Mobility Adaptive Clustering Scheme with Support for Geographic Routing for Vehicular Ad Hoc Networks”, *IET Intelligent Transport Systems*, Vol. 3, no. 2, pp. 148 –158, 2009.

[95] Y. Luo, W. Zhang, Y. Hu, "A new cluster based routing protocol for VANET", *IEEE Networks Security Wireless Communications and Trusted Computing (NSWCTC)*, vol. 1, pp. 176-180, 2010.

[96] D. Tian, Y. Wang, G. Lu, G. Yu, “A VANETs Routing Algorithm Based on Euclidean Distance Clustering”, *IEEE International Conference On Future Computer and Communication*, pp. 183-187, 2010.

[97] Q. Yang, A. Lim, P. Agrawal, “Connectivity Aware Routing in Vehicular Networks”,*IEEE Wireless Communications and Networking Conference, (WCNC)*, pp. 2218 –2223, 2008.

[98] D.Ranjini Girinath, S.Selvan, “ A Novel Cluster Based Routing Algorithm for Hydbrid Mobility in VANET”, *International Journal of Computer Applications*, Vol. 1, 2010.

[99] H. Trivedi, P. Veeraraghavan, S. Loke, A. Desai, J. Singh. “SmartVANET: The Case for a Cross-Layer Vehicular Network Architecture”, *IEEE Workshops of International Conference on Advanced Information Networking and Applications (WAINA)*, pp. 362 –368, 2011.

[100] M. Shoaib, W.-C. Song, K. H. Kim, “Cluster based data aggregation in vehicular adhoc network”, *ACM International conference on Communication Technologies for Vehicles (Nets4Cars/Nets4Trains'12)*, pp. 91-102, 2012.

[101] T. Gazdar, A. Belghith, A. Benslimane, “A Cluster Based Secure Architecture for Vehicular Ad Hoc Networks”, *IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, pp. 1 –8, 2010.

[102] T. Gazdar, A. Benslimane, A. Belghith, “Secure Clustering Scheme Based Keys Management in VANETs”, *IEEE Vehicular Technology Conference (VTC Spring), 2011*, pp. 1–5, 2011.

[103] B. Ramakrishnan, R. S. Rajesh, R. S. Shaji,“CBVANET: A Cluster Based Vehicular Adhoc Network Model for Simple Highway Communication”, *International Journal on Advanced Networking and Applications*, Vol. 2, pp. 744-76, 2011.

[104] N. Maslekar, J. Mouzna, H. Labiod, M. Devisetty, M. Pai, "Modified C-DRIVE: Clustering Based on Direction in Vehicular Environment",*IEEE Intelligent Vehicles Symposium,* pp. 845–850, 2011.

[105] O. Kayis, T. Acarman, "Clustering Formation for Inter-Vehicle Communication", *IEEE Intelligent Transportation Systems Conference (ITSC),* pp. 636 –641, 2007.

[106] C.-J. Huang, C.-F. Lin, C.-Y. Li, C.-Y. Lee, H.-M. Chen, H.-Y. Shen, I-F. Chen, "Service-Oriented Routing and Clustering Strategies for Vehicle Infotainment Dissemination", *International Journal of Innovative Computing, Information and Control,* Vol. 7, pp. 1467-1479, 2011.

[107] N. Maslekar, M. Boussedjra, J. Mouzna, L. Houda, "Direction Based Clustering Algorithm for Data Dissemination in Vehicular Networks", *IEEE Vehicular Networking Conference (VNC)*, pp. 1–6, 2009.

[108] Z. Y. Rawashdeh, S. M. Mahmud, "Toward Strongly Connected Clustering Structure in Vehicular Ad hoc Networks," *IEEE Vehicular Technology Conference (VTC2009-Fall),* 2009.

[109] C. Shea, B. Hassanabadi, S. Valaee, "Mobility-Based Clustering in VANETs using Afinity Propagation", *Global Communications Conference(GLOBECOM)*, pp. 1-6, 2009.

[110] E. Dror, C. Avin, Z. Lotker, "Fast Randomized Algorithm for Hierarchical Clustering in Vehicular Ad-Hoc Networks", *IEEE/IFIP Annual MediterraneanAd Hoc Networking Workshop (Med-Hoc-Net),* pp.1 –8, 2011.

[111] N. Maslekar, M. Boussedjra, J. Mouzna, H. Labiod, "A Stable Clustering Algorithm for Efficiency Applications in VANETs", *IEEE International Wireless Communications and Mobile Computing Conference (IWCMC),* pp. 1188–1193, 2011.

[112] P. Fan, J. G. Haran, J. Dillenburg, P. C. Nelson, "Cluster-Based framework in vehicular ad-hoc networks", *ACM International conference on Ad-Hoc, Mobile, and Wireless Networks (ADHOC-NOW'05),* pp. 32-42., 2005.

[113] S. Kuklinski, G. Wolny, "Density Based Clustering Algorithm for VANETs", *IEEE International Conference on Testbeds and Research Infrastructures for the Development of Networks Communities and Workshops,* 2009, pp. 1 –6, 2009.

[114] M.S. Almalag, M.C. Weigle. "Using Traffic Flow for Cluster Formation in Vehicular Ad-hoc Networks", *IEEE Conference on Local Computer Networks (LCN),* pp. 631–636, 2010.

[115] M.O. Cherif, S.M. Senouci, B. Ducourthial, "A New Framework of Self-organization of Vehicular Networks", *IEEE Information Infrastructure Symposium (GIIS '09),* pp. 1 –6, 2009.

[116] J. Blum, A. Eskandarian, L. Hoffman, "Mobility Management in IVC Networks", *IEEE Intelligent Vehicles Symposium,* pp. 150 – 155, 2003.

[117] P. Basu, N. Khan, T.D.C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks", *International Conference on Distributed Computing Systems Workshop*, pp. 413–418, 2001.

[118] G. Wolny, “Modified DMAC Clustering Algorithm for VANETs”, *IEEE International Conference on Systems and Networks Communications*, pp. 268 –273, 2008.

[119] Z. Wang, L. Liu, MC. Zhou, N. Ansari, “A Position-Based Clustering Technique for Ad Hoc Intervehicle Communication”, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 38, no. 2 pp. 201 –208, 2008.

[120] S. Basagni, "Distributed clustering for ad hoc networks", *IEEE InternationalSymposium on Parallel Architectures, Algorithms, and Networks*, pp. 310-315. IEEE, 1999.

[121] B. Hassanabadi, Behnam, C. Shea, L. Zhang, S. Valaee, "Clustering in vehicular ad hoc networks using affinity propagation", *Ad Hoc Networks*, Vol. 13, pp. 535-548, 2014.

[122] O. A. Wahab, H. Otrok, A. Mourad, “VANET QoS-OLSR: QoS-based clustering protocol for Vehicular Ad hoc Networks”, *Computer Communications*, 2013.

[123] R. S. Bali, N. Kumar, J. JPC Rodrigues, "Clustering in vehicular ad hoc networks: taxonomy, challenges and solutions", *Vehicular communications*, 1.3, pp. 134-152, 2014.

[124] L.-C. Tung, J. Mena, M. Gerla, C. Sommer, “A Cluster Based Architecture for Intersection Collision Avoidance Using Heterogeneous Networks”, *IFIP/IEEE Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2013),* 2013.

[125] S. Ucar, S. C. Ergen, O. Ozkasap, “VMaSC: Vehicular multi-hop algorithm for stable clustering in Vehicular Ad Hoc Networks”, *IEEE Wireless Communications and Networking Conference (WCNC*), pp. 2381-2386, 2013.

[126] E. Dror, C. Avin, Z. Lotker, “Fast randomized algorithm for 2-hops clustering in vehicular ad-hoc networks”, *Ad Hoc Networks*, 11(7), pp. 2002-2015, 2013.

[127] Y. Harikrishnan, J. He, “Clustering algorithm based on minimal path loss ratio for vehicular communication”, *IEEE International Conference on Computing, Networking and Communications (ICNC),* pp. 745-749, 2013.

[128] S. Ucar, S. C. Ergen, O. Ozkasap, "Multi-Hop Cluster based IEEE 802.11 p and LTE Hybrid Architecture for VANET Safety Message Dissemination", *IEEE Transactions on Vehicular Techonologyi*, 2015 (in press).

[129] Y. Chen, M. Fang, S. Shi, W. Guo, X. Zheng, "Distributed multi-hop clustering algorithm for VANETs based on neighborhood follow." *EURASIP Journal on Wireless Communications and Networking* 2015 (in press).

[130] H. R. Arkian, R. E. Atani, A. Pourkhalili, S. Kamali, "Cluster-based traffic information generalization in Vehicular Ad-hoc Networks", *Vehicular communications* 1, no. 4, pp. 197-207, 2014.

[131] H. Sedjelmaci, S. M. Senouci, "An accurate and efficient collaborative intrusion detection framework to secure vehicular networks", *Computers & Electrical Engineering* 43, pp. 33-47, 2015.

[132] Y. Qian, N. Moayeri, "Design of secure and application-oriented VANETs" *IEEE Vehicular Technology Conference*, pp. 2794-2799. IEEE, 2008.

[133] IEEE P1609.6, IEEE Project: "P1609.6 – Standard for Wireless Access in Vehicular Environments (WAVE) Remote Management Service", 2015 (active project).

[134] IEEE P1609.6, "P1609.11 – Standard for Wireless Access in Vehicular Environments (WAVE) Over-the-Air Exchange Protocol for ITS, D.4.1 ", 2010.

[135] IEEE P1609.0 "P1609.0 – 2013 – IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture", 2013.

[136] IEEE P1609.6, IEEE Project: "P1609.12 – IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE) – Identifier Allocations, 2015 (active project).

[137] M.D. Nathanson, "Vehicle communications via wireless access vehicle environment", US Patent App. 14/151,035, 2014.

[138] ETSI EN 302 663 V1.2.1, "Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5GHz", 2013.

[139] ETSI TS 102 636-3 V1.1.1, "Inteligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 3: Network Architecture", 2010.

[140] ETSI TS 102 637-2, "Inteligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: specification of cooperative awareness basic service".

[141] SAE Int. DSRC Committee, "Dedicated Short Range Communications (DSRC) Message Set Dictionary," SAE, Tech. Rep. J2735_200911, November 2009.

[142] European Telecommunications Standards Institute, "Intelligent Transport Systems (ITS); Harmonized Channel Specifications for Intelligent Transport Systems operating in the 5 GHz frequency band," ETSI, Tech. Rep. ETSI TS 102 724 V1.1.1, October 2012.

[143] H.-J.Fischer, "Standardization and Harmonization Activities Towards a Global C-ITS." In *Vehicular ad hoc Networks*, pp. 23-36. Springer International Publishing, 2015.

[144] K.C. Lee, U. Lee, M. Gerla, "Survey of Routing Protocols in Vehicular Ad Hoc Networks", *Advances in Vehicular Ad-Hoc Networks: Developments and Challenges*, IGI Global, 2010.

[145] Y.-W. Lin, Y.-S. Chen, S.-L. Lee, "Routing Protocols in Vehicular Ad Hoc Networks: A Survey and Future Perspectives", *Journal of Information Science and Engineering*, vol. 26, pp. 913-932, 2010.

[146] F. Li, Y. Wang, "Routing in Vehicular Ad Hoc Networks: A Survey", *IEEE Vehicular Technology Magazine*, no. 2, pp. 12 –22, 2007.

[147] B. Paul, Md. Ibrahim, A. N. Bikas, "VANET Routing Protocols: Pros and Cons.", *International Journal of Computer Applications*, vol. 20, pp. 28-34, 2011.

[148] W. Chen, R. K. Guha, T. J. Kwon, J. Lee, Y.-Y. Hsu, “A Survey and Challenges in Routing and Data Dissemination in Vehicular Ad Hoc Networks”, *Wireless Communications and Mobile Computing*, vol. 11, pp. 787–795, 2011.

[149] X.-b. Wang, Y.-l. Yang, J.-w. An, “Multi-metric Routing Decisions in VANET”, *IEEE International Conference on Dependable, Autonomic and Secure Computing*, pp. 551 –556, 2009.

[150] C.-J. Huang, I-. Che, K.-W. Hu, H.-Y. Shen, Y.-J. Chen, D.-X. Yang, “A Load Balancing and Congestion-Avoidance Routing Mechanism for Teal-Time Traffic over Vehicular Networks”, *Journal of Universal Computer Science*, vol.15, no. 13, pp. 2506-2527, 2009.

[151] K. Z. Ghafoor, K. A. Bakar, K. Lee, and H. AL-Hashimi, “A Novel Delay and Reliability- Aware Inter-Vehicle Routing Protocol”, *International Journal of Network Protocols and Algorithms*, vol. 2, No. 2, 2010.

[152] K. Z. Ghafoor, K. A. Bakar, S. Salleh, K. Lee, M. Mohamad, M. Kamat, and M. Arshad, “Fuzzy Logic-Assisted Geographical Routing over Vehicular Ad hoc Networks”, *International Journal of Innovative Computing Information and Control*, vol. 8, No.6, 2012.

[153] R.H. Khokhar, R.M. Noor, K. Z. Ghafoor, C.-H. Ke, M. A. Ngadi, “Fuzzy-assisted social-based routing for urban vehicular environments”, *EURASIP Journal on Wireless Communications and Networking*, no. 1, pp. 1-15, 2011.

[154] C. Wu, S. Ohzahata, T. Kato, “Fuzzy Logic for Multi-Hop Broadcast in Vehicular Ad Hoc Networks”, *Fuzzy Logic - Emerging Technologies and Applications*, InTech Publisher, 978-953-51-0337-0, 2012.

[155] C.-J. Huang, Y.-J. Chen, I-F. Chen, T.-H. Wu, “An intelligent infotainment dissemination scheme for heterogeneous vehicular networks”, *Expert Systems with Applications: An International Journal* vol. 36, pp. 12472-12479, 2009.

[156] C. Wu, S. Ohzahata, K. Toshihiko, "Routing in VANETs: A fuzzy constraint Q-Learning approach", *IEEE Global Communications Conference (GLOBECOM)*, pp. 195-200, 2012.

[157] C. Wu, S. Ohzahata, T. Kato, "Flexible, Portable and Practicable Solution for Routing in VANETs: A Fuzzy Constraint Q-Learning Approach", *IEEE Transactions on Vehicular Technology*, vol.62, no.9, pp.4251-4263, 2013.

[158] C. J. C. H. Watkins, "Learning from delayed rewards." PhD dissertation, University of Cambridge, 1989.

[159] C. Wu, K. Kumekawa, "Distributed reinforcement learning approach for vehicular ad hoc networks", *IEICE transactions on communications,* 93, no. 6 (2010): pp. 1431-1442.

[160] C.-J. Huang, Y.-T. Chuang, D.-X. Yang, I-F. Chen, Y.-J. Chen, K.-W. Hu, “A mobility-aware link enhancement mechanism for vehicular ad hoc networks”, *EURASIP Journal on Wireless Communications and Networking,* Article 24, 2008.

[161] H. Menouar, F. Filali, M. Lenardi, "A survey and qualitative analysis of MAC protocols for vehicular ad hoc networks", *IEEE Wireless Communications*, vol. 13, no. 5 pp. 30-35, 2006.

[162] F. Yu, S. Biswas, "Self-configuring TDMA protocols for enhancing vehicle safety with DSRC based vehicle-to-vehicle communications", *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1526-1537, 2007.

[163] T. Abdelkader, K. Naik, A. Nayak, F. Karray, “Adaptive Backoff Scheme for Contention-based Vehicular Networks Using Fuzzy Logic”, *IEEE International Conference on Fuzzy Systems*, pp. 1621–1626, 2009.

[164] Z. J. Haas, J. Deng, "On optimizing the backoff interval for random access schemes", *IEEE Transactions on Communications*, vol. 51, no. 12, pp. 2081-2090, 2003.

[165] T. Abdelkader, K. Naik, A. Nayak, "Using fuzzy logic to calculate the backoff interval for contention-based vehicular networks", *IEEE International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 783-788, 2011.

[166] A. Souza, A. L. Barros, A.S. Vieira, F. Roberto, J.C. Júnior, “An Adaptive Mechanism for Access Control in VANETs”, *International Conference on Networks*, pp. 183-188, 2011.

[167] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks", *IEEE Wireless Communications*, vol. 14, no. 6, pp. 84-94, 2007.

[168] Y. Wang, A. Ahmed, B. Krishnamachari, K. Psounis, "IEEE 802.11 p performance evaluation and protocol enhancement", *IEEE International Conference on Vehicular Electronics and Safety (ICVES*., pp. 317-322, 2008.

[169] C. Chrysostomou, C. Djouvas, L. Lambrinos, "Applying adaptive QoS-aware medium access control in priority-based vehicular ad hoc networks”, *IEEE Symposium on Computers and Communications*, pp.741-747, 2011.

[170] C. Chrysostomou, C. Djouvas, L. Lambrinos, "Dynamically adjusting the min-max contention window for providing quality of service in vehicular networks", *IEEE 11th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pp. 16-23, 2012.

[171] K. A. Hafeez, L. Zhao, Z. Liao, B. N. Ma, “Clustering and OFDMA-based MAC protocol (COMAC) for vehicular ad hoc networks”, *EURASIP Journal on Wireless Communications and Networking*, no. 1, pp. 1-16, 2011.

[172] S. Dietzel, F. Kargl, G. Heijenk, F. Schaub, "Modeling in-network aggregation in VANETs," *IEEE Communications Magazine*, vol.49, no.11, pp.142-148, 2011.

[173] L. Wischhof, A. Ebner, H. Rohling, M. Lott, R. Halfmann, “SOTIS – A Self-Organizing Traffic Information System,” *IEEE Vehicular Technology Conference*, pp. 2442–46, 2003.

[174] B. Yu, J. Gong, C.-Z. Xu, “Catch-up: a data aggregation scheme for vanets”, *ACM International Workshop on VehiculAr Inter-NETworking (VANET),* pp. 49-57, 2008.

[175] M. Caliskan, D. Graupner, M. Mauve, “Decentralized Discovery of Free Parking Places”, *ACM International Workshop on VehiculAr Inter-NETworking (VANET),* pp. 30–39, 2006.

[176] K. Ibrahim, M.C. Weigle, “CASCADE: Cluster-Based Accurate Syntactic Compression of Aggregated Data in VANETs”, *IEEE Global Telecommunications Conference Workshops, (GlobeCom Workshops),* pp. 1-10, 2008.

[177] F. Picconi, N. Ravi, M. Gruteser, L. Iftode, “Probabilistic Validation of Aggregated Data in Vehicular Ad-Hoc Networks,” *ACM International Workshop on Vehicular Ad Hoc Networks,* pp. 76–85, 2006.

[178] P. Caballero-Gil, J. Molina-Gil, C. Caballero-Gil, “Data Aggregation Based on Fuzzy Logic for VANETs”, *International Conference on Computational intelligence in security for information systems,* 2011.

[179] M. Raya, A. Aziz, J. Hubaux, “Efficient Secure Aggregation inVANETs,” *ACM International Workshop on Vehicular Ad Hoc Networks,* pp. 67–75, 2006.

[180] S. Buchegger, J. Mundinger, J. Le Boudec, “Reputation Systems for Self-Organized Networks: Lessons Learned,” *IEEE Technology and Society Magazine*, vol. 27, no. 1, pp. 41–47, 2008.

[181] B. Scheuermann, “A Fundamental Scalability Criterion for Data Aggregation in VANETs,” *ACM Int. Conf. on Mobile Computing and Networking*, New York, USA, 2009.

[182] ETSI Std., “Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications,” 2010.

[183] S. Dietzel F. Kargl, G. Heijenk, F. Schaub, “On the Potential of Generic Modeling for VANET Data Aggregation Protocols,” *IEEE Vehicular Networking Conference*, pp 78-85, 2010.

[184] S. Dietzel, E. Schoch, B. Bako, F. Kargl, "A structure-free aggregation framework for vehicular ad hoc networks*", International Workshop on Intelligent Transportation*, pp. 61-66. 2009.

[185] S. Dietzel, E. Schoch, B. Konings, M. Weber, F. Kargl, “Resilient Secure Aggregation for Vehicular Networks”, *IEEE Network*, pp. 26-31, 2010.

[186] M. Kassar, B. Kervella, G. Pujolle, “An overview of vertical handover decision strategies in heterogeneous wireless networks,” *Computer Communications*, vol. 31, no. 10, pp. 2607–2620, 2008.

[187] X. Yan, Y. A. Sekercioglu, S. Narayanan, “A survey of vertical handover decision algorithms in fourth generation heterogeneous wireless networks,” *Computer Networks*, vol. 54, no. 11, pp. 1848–1863, 2010.

[188] R. Trestian, O. Ormond, G.-M. Muntean, "Game Theory—Based Network Selection: Solutions and Challenges", *IEEE Communications Surveys Tutorials*, vol. 14, no. 4, pp. 1212-1231, 2012.

[189] J. Hou, D. C. O'Brien, "Vertical handover decision making algorithm using fuzzy logic for the integrated Radio-and-OW system," *IEEE Transactions on Wireless Communications*, vol. 5, no. 1, pp. 176–185, 2006.

[190] P. Chan, Y. Hu, R. Sheriff, "Implementation of fuzzy multiple objective decision making algorithm in a heterogeneous mobile environment," *IEEE Wireless Communications and Network Conference (WCNC)*, pp.332–336, 2002.

[191] S. Kher, A. K. Somani, R. Gupta, "Network selection using fuzzy logic", *International Conference Broadband Networks (Broadnets)*, pp. 876–885, 2005.

[192] Q. Guo, J. Zhu, X. Xu, "An adaptive multi-criteria vertical handoff decision algorithm for radio heterogeneous network," *IEEE International Conference on Communications (ICC)*, pp. 2769–2773, 2005.

[193] I. Chantaksinopas, P. Oothongsap, A. Prayote, "Framework for network selection transparency on vehicular networks", *IEEE International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON)*, pp. 593-597, 2010.

[194] M. Bechler, L. Wolf, "Fuzzy Logic Based Handovers in Vehicular Communication Environments", *Workshop on Intelligent Transportation (WIT)*, pp. 1-5, 2005.

[195] B. Ma, X. Liao, "Speed-adaptive vertical handoff algorithm based on fuzzy logic in vehicular heterogeneous networks", *IEEE International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pp. 371-375, 2012.

[196] S. Dhar, A. Ray, S. Chakravorty, R.N. Bera, "Intelligent vertical handover scheme for utopian transport scenario", *Trends in Applied Sciences Research*, vol.6, pp. 958-976, 2011.

[197] T.L. Saaty, "Decision making with the analytic hierarchy process", *International Journal of Services Sciences*, vol. 1, No. 1, pp.83–98, 2008.

[198] K.Z. Ghafoor, K. A. Bakar, M. van Eenennaam, R. H. Khokhar, A. J. Gonzalez, "A fuzzy logic approach to beaconing for vehicular ad hoc networks", *Telecommunication Systems*, pp. 1-11, 2013.

[199] B.S. Kerner, "The physiscs of traffic: empirical freeway pattern features, engineering applications, and theory", Berlin:Springer, 2004.

[200] V. Milanés, J. Pérez, E. Onieva, and C. González, "Controller for urban intersections based on wireless communications and fuzzy logic", *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 1, pp. 243-248, 2010.

[201] K. Z. Ghafoor, J. Alberto, R. Piney, A. Rios, J. Alcober, K. A. Bakar, "Fuzzy redundancy adaptation and joint source-network coding for VANET video streaming", *Wired/Wireless Internet Communications, Springer Berlin Heidelberg*, pp. 458-469, 2011.

[202] A. M. D. Souza, A. Boukerche, G. Maia, R. I. Meneguette, A. A. Loureiro, L. A. Villas, "Decreasing greenhouse emissions through an intelligent traffic information system based on inter-vehicle

communication”, *ACM International Symposium on Mobility Management and Wireless Access*, pp. 91-98, 2014.

[203] K. Demestichas, E. Adamopoulou, M. Masikos, T. Benz, W. Kipp, F. Cappadona, "Advanced Driver Assistance System Supporting Routing and Navigation for Fully Electric Vehicles", *Advanced Microsystems for Automotive Applications 2012*, Springer Berlin Heidelberg, pp. 197-206, 2012.

[204] N. Maslekar, J. Mouzna, M. Boussedjra, H. Labiod., “CATS: An adaptive traffic signal system based on car-to-car communication”, *Journal of Network and Computer Applications*, 36(5), pp. 1308-1315, 2013.

[205] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, L. Iftode, “Adaptive traffic lights using car-to-car communication”, *IEEE Vehicular Technology Conference*, pp. 21-25, 2007.

[206] T. Tielert, M. Killat, H. Hartenstein, R. Luz, S. Hausberger, T. Benz., "The impact of traffic-light-to-vehicle communication on fuel consumption and emissions", *IEEE Internet of Things (IOT)*, pp. 1-8, 2010.

[207] M. Alsabaan, K. Naik, and T. Khalifa, “Optimization of Fuel Cost and Emissions Using V2V Communications,” *IEEE Trans. on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1449–1461, 2013.

[208] D. Eckhoff, B. Halmos, and R. German,“Potentials and Limitations of Green Light Optimal Speed Advisory Systems,” *IEEE Vehicular Networking Conference (VNC)*, pp. 103-110, 2013.

[209] S. Hausberger, M. Rexeis, M. Zallinger, R. Luz, “Emission factors from the model PHEM for the HBEFA version 3,” University of Technology, Graz, Report Nr. I-20/2009 Haus-Em 33/08/679, 2009.

[210] K. Kraschl-Hirschmann, M. Zallinger, R. Luz, M. Fellendorf, S. Hausberger, “A method for emission estimation for microscopic traffic flow simulation,” *IEEE Forum on Integrated and Sustainable Transportation System (FISTS)*, 2011.

[211] T. Tielert, D. Rieger, H. Hartenstein, R. Luz, S. Hausberger, “Can V2X Communication Help Electric Vehicles Save Energy?”, *IEEE International Conference on ITS Telecommunications (ITST)*, pp. 232-237, 2012.

[212] B. J. Frey, D. Dueck, "Clustering by passing messages between data points" *Science* 315, no. 5814, pp.: 972-976, 2007.

[213] R. Sivaraj, A. K. Gopalakrishna, M.G. Chandra, P. Balamuralidhar, “QoS-enabled group communication in integrated VANET-LTE heterogeneous wireless networks”, *IEEE Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 17-24, 2011.

[214] T. Taleb, A. Benslimane, “Design guidelines for a network architecture integrating vanet with 3g & beyond networks”, *IEEE Global Telecommunications Conference*, pp. 1-5, 2010.

[215] A. Benslimane, T. Taleb, R. Sivaraj, Dynamic clustering-based adaptive mobile gateway management in integrated VANET—3G Heterogeneous Wireless Networks. *Selected Areas in Communications, IEEE Journal on*, *29*(3), pp. 559-570, 2011.

[216] E. Souza, I. N. Evandro, P. Gburzynski, "A new aggregate local mobility (ALM) clustering algorithm for VANETs", *IEEE International Conference on Communications*, pp. 1-5, 2010.

[217] M. Gerla, J. T. C. Tsai, “Multicluster, mobile, multimedia radio network”, *Wireless networks*, 1(3), pp. 255-265, 1995.

[218] H. J. Reumerman, M. Roggero, M. Ruffini, “The application-based clustering concept and requirements for intervehicle networks”, *IEEE Communications Magazine*, 43(4), pp. 108-113, 2005.

[219] A. Goldsmith, “Beyond 4G: What Lies Ahead for Cellular System Design”, Keynote presentation at IEEE Wireless Communications and Networking Conference (WCNC).

[220] Navigant Research, Smart Transportation, Electric Bicycles, 2013, online: http://www.navigantresearch.com/tag/electric-bicycles?page=2

[221] Bikemetro, 2009. Online Bicycle Route Planner for California, online: https://www.metro.net/bikes/bikes-metro/

[222] J. G. Su, M. Winters, M. Nunes, M.Brauer, "Designing a route planner to facilitate and promote cycling in Metro Vancouver, Canada", *Transportation research part A: policy and practice*, Vol. 44, no. 7, pp. 495-505, 2010.

[223] BBBike Cycling Route Planner, online: http://www.bbbike.org/

[224] M. Winters, G. Davidson, D. Kao, K. Teschke, "Motivators and deterrents of bicycling: comparing influences on decisions to ride" , *Transportation*, Vol.38, no. 1, pp. 153-168, 2011.

[225] M. Barbehenn, “A note on the complexity of Dijkstra’s algorithm for graphs with weighted vertices”, *IEEE Transactions on Computerss*, vol. 47, no. 2, p. 263, 1998

[226] M. R. Jabbarpour, R. M. Noor, R. H. Khokhar, "Green vehicle traffic routing system using ant-based algorithm." *Journal of Network and Computer Applications*, 2015 (in press).

[227] R. Bodenheimer, A. Brauer, D. Eckhoff, R. German, "Enabling GLOSA for adaptive traffic lights", *IEEE Vehicular Networking Conference,* pp. 167-174, 2014.

[228] K. Collins, G.-M. Muntean, "Route-based vehicular traffic management for wireless access in vehicular environments", *IEEE Vehicular Technology Conference*, pp. 1-5, 2008.

[229] X. Xiang, K. Zhou, W.-B. Zhang, W. Qin, Q. Mao, "A Closed-Loop Speed Advisory Model With Driver's Behavior Adaptability for Eco-Driving", *IEEE Transactions on Intelligent Transportation System*, 2015 (in press)

[230] S. Schellenberg, R. Berndt, D. Eckhoff, R. German, "A Computationally Inexpensive Battery Model for the Microscopic Simulation of Electric Vehicles." *IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1-6, 2014.

[231] E. Masalkina, D. Eckhoff, R. Berndt, R. German, "Towards the City-scale Simulation and Performance Assessment of Electric Vehicles", *Fachgespräch Inter-Vehicle Communication:*13, 2014.

[232] C. Wu, S. Ohzahata, Y. Ji, T. Kato, "Joint Fuzzy Relays and Network-Coding-Based Forwarding for Multihop Broadcasting in VANETs", *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no 3, pp.1415-1427, 2015.

[233] A. M. Vegni, M. Biagi, R. Cusani, "Smart Vehicles, Technologies and Main Applications in Vehicular Ad hoc Networks", *IntechOpen Books,* http://dx.doi.org/10.5772/55492, 2013.

[234] E. Cox, “Fuzzy Fundamentals.” *Spectrum, IEEE*, 29, pp. 58–61, 1992.

[235] H. Idjmayyel, W. Kumar, B. R. Qazi, J. M. H. Elmirghani, “Saving energy with QoS for vehicular communication”, *International Conference on Wireless and Optical Communications Networks (WOCN)*, pp. 1-5, 2011.

[236] K. Maenpaa, T. Sukuvaara, R. Ylitalo, P. Nurmi, E. Atlaskin, “Road weather station acting as a wireless service hotspot for vehicles”, *IEEE Intelligent Computer Communication and Processing*, pp. 159-162, 2013.

[237] G. Elert, The Physics Hypertextbook, Equations of Motion, http://physics.info/motion-equations/

[238] A. Moon, Y.-i. Choi, and B.-s. Lee, “Context-aware user model for personalized services,” *International Conference on Digital Information Management*, pp. 858-863, 2008.

[239] C. Sommer, R. German, F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved IVC analysis", *IEEE Transactions on Mobile Computing*, pp. 3-15, 2010.

[240] J.-M. Timmermans, J. Matheys, P. Lataire, J.Van Mierlo, J. Cappelle, "A comparative study of 12 electrically assisted bicycles, " *World Electric Vehicle Journal*, vol. 3, 2009.

[241] Survey Guidelines, Massachusetts Institute of Technology, Office of the Provost, Institutional Research, published September 2011.

[242] F. Hall, S. Wakefield, A. Al-Kaisy, “Freeway quality of service – what really matters to drivers and passengers?” *Transportation Research Record*, 1776, pp. 17–23, 2001.

[243] D. Papinski, D. M. Scott, S.T. Doherty, “Exploring the route choice decision-making process: A comparison of planned and observed routes obtained using person-based gps”, *Transportation Research Part F*, 12, pp. 347–358, 2009.

[244] D.G. Wilson, J. Papadopoulos, and F.R. Whitt, Bicycling science, Cambridge, Mass: MIT Press, 2004.

[245] J. C. Martin, D. L. Milliken, J. E. Cobb, K. L. McFadden, A. R. Coggan, "Validation of a mathematical model for road cycling power", *Journal of applied biomechanics,* Vol. 14, pp. 276-291, 1998.

APPENDIX A

Interview set of questions

1. Interview number:
2. What age are you?
    - Under 19;
    - 19-24;
    - 25-34;
    - 45-54;
    - 55-64;
    - 65 and older;
3. What gender are you?
    - Male;
    - Female;
4. What is your employment status?
    - Full-time;
    - Part-time;
    - Self-employed;
    - Student;
    - Retired;
    - Not-employed;
5. Do you live in Ireland?
    - Yes;
    - No;
6. Please select from the *following* options the one you describing you:
    - Regular cyclist(52 or more one-way trips per year);
    - Frequent cyclist(12-51 one-way trips per year);
    - Occasional cyclist(1-11 one-way trips per year);
    - Potential cyclist(never in past year);
7. Which is the main purpose you are using your bicycle for?
8. Are there any other reasons?

9. Did you ever consider cycling as main form of transportation? If not, why? What aspects make you think is inappropriate as form of transportation? Can you mention any factors?
10. What you consider to be the main advantages of cycling?
11. As form of transportation, what do you consider to be the main advantages of cycling as compared to other forms of transportation?
12. What do you consider to be the disadvantages of cycling?
13. Which are the main factors that negatively affect you when cycling?
14. Do you consider that bad weather conditions influence your cycling experience? If yes, what is their influence?
15. What do you consider to be the bad weather conditions for cycling?
16. Which do you consider to affect you the most?
17. Did you ever cycle in windy conditions?
18. Did you find it difficult in terms of effort?
19. On a scale of 1 to 5 how badly influenced you're cycling experience?
20. If given the possibility would you go for the least effort route where effort is defined as power needed to overcome the power of the wind?
21. Is your journey by bike constrained by time?
22. Let's assume the following scenarios:

    In a regular work day you start your journey to/from work by bike. At some point, it is starting to rain.

    - Would you feel disturbed by the rain? On a scale of 1 to 5, how disturbed would you feel?
    - Starting the journey at different time would represent a choice for you if the rain during cycling would be avoided?Note that different time it is in a time interval that is specified by you!
23. How interested would you be in a solution that can help you plan your journey by bike so that you have the best weather conditions for cycling .
    - Interested;
    - Somehow interested;
    - Not interested;
24. On a scale of 1 to 5 how useful do you find the application for improving you're cycling experience?
25. Would you be inclined to cycle even more being provided with this smart pre-planning solution?

- Yes;
- No;

APPENDIX B

Questionnaire set of questions

1. Age:
    - Under 19;
    - 19-24;
    - 25-34;
    - 45-54;
    - 55-64;
    - 65 and older;

2. Gender:
    - Male;
    - Female;
3. Employment status:
    - Full-time;
    - Part-time;
    - Self-employed;
    - Student;
    - Retired;
    - Not-employed;

4. What is your nationality?
5. Do you own a bicycle?
6. Is it an electrical bicycle?
7. Are you a client of a bicycle sharing system?
8. Which of the following cyclist types describe you?
    - Regular cyclist(52 or more one-way trips per year);
    - Frequent cyclist(12-51 one-way trips per year);
    - Occasional cyclist(1-11 one-way trips per year);
    - Potential cyclist(never in past year);
9. Please rank the following reasons for cycling as they apply to you:
    - Commuting to work/school;

- Destinations(errands/shopping);
- Exercise;
- Recreation;
- Others(Please specify in the comment);

10. What is on average the distance and the time of your one-way bicycle trips?
    - Time(Minutes) - ;
    - Distance(kilometres) - ;
11. Are your regular bicycle trips tightly constrained by time?
    - No, they are rather flexible;
    - Not really, they are flexible in a certain time interval;
    - Yes;
12. What do you consider to be the advantages of cycling ?(tick all that apply):
13. Please rank the potential negative influence of the following factors on your motivation to cycle (5-high influence to 1-no influence at all):
        - Safety;
        - Seasonality;
        - Darkness;
        - Bad weather conditions;
        - Distance;
        - Drivers attitude;
        - Traffic conditions;
        - Poor or lack of cycling facilities (side-walks, cycle lanes, etc.)
        - Effort;
        - Noise;
        - Pollution from traffic;
        - Hilliness;
        - No safe place to lock your bicycle;
        - Others(please specify in the comment);
14. What do you consider to be bad weather conditions when cycling? (please refer to Ireland's weather)
15. Do you agree that bad weather conditions, such as precipitation or wind, negatively influence your cycling experience?
    - Strongly disagree;
    - Disagree;

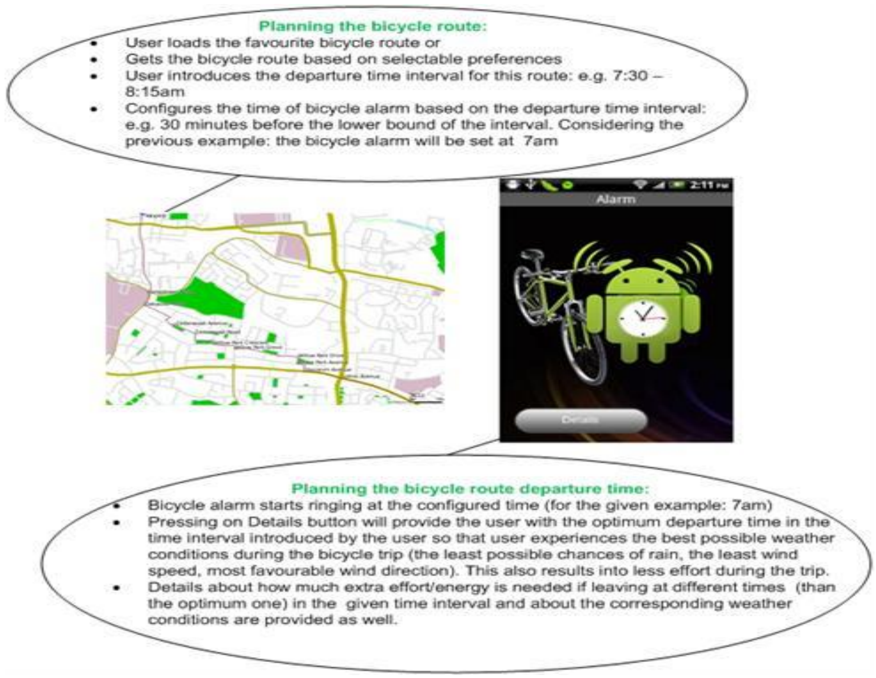
- Neutral;
- Agree;
- Strongly agree;

16. Do you find it difficult, in terms of effort, to cycle in windy conditions?
    - 1(not difficult at all);
    - 2
    - 3
    - 4
    - 5(very difficult).
17. Did you ever cycle in strong wind?(You can specify in the comment the minimum speed and measure of what you consider a strong wind(Optionally). Example: more than 50 km/h or more than 30 mph).
    - No(skip to question 19);
    - Yes;
18. How disturbing did you find the wind while cycling?
    - 1(not disturbing at all);
    - 2
    - 3
    - 4
    - 5(very disturbing).
19. Let's assume the following scenario: You are cycling on your usual route; at some point during the trip it starts raining. Would you feel disturbed by the rain?
    - 1(not difficult at all);
    - 2
    - 3
    - 4
    - 5(very difficult).
20. Would you start your journey at a different time if you could avoid the rain while cycling?
    - No;
    - Yes;
21. Describe how adverse weather conditions affect cycling/cyclist (Optional answer):

22. How useful is it for you to know in advance of starting a cycling journey the probability of bad weather conditions occurring?
    - 1(not useful at all);

- 2
- 3
- 4
- 5(very useful).

23. Would you be interested in an application that advises the departure time of your bicycle journey in order to provide you with the best/most favourable weather conditions for cycling?(Note that departure time is in a configurable time interval) .
    - 1(not interested at all);
    - 2
    - 3
    - 4
    - 5(very interested).
24. Before proceeding to the next questions please read the description of the mobile application provided in the below picture. This application is a weather aware cycling route planner and is referred in all the next questions!



How would the presented application influence your cycling motivation?

- Much less likely to cycle;
- No influence;
- Much more likely to cycle;

25. Rank your interest in the presented application:
    - 1(not interested at all);

- 2
- 3
- 4
- 5(very interested).

26. How useful do you find the presented application for improving your cycling experience?
    - 1(not useful at all);
    - 2
    - 3
    - 4
    - 5(very useful).
27. What would be your personal adjustment to this application?(Optional answer).