TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Hamed Noori

Modeling and Simulation of Vehicle to Vehicle and Infrastructure Communication in Realistic Large Scale Urban Area

TAMPERE UNIVERSITY OF TECHNOLOGY
MASTER OF SCIENCE THESIS

Examiners: Professor Yevgeni Koucheryavy
Professor Pasi Kallio
Dr. Alexey Vinel

The topic and examiners of Master of Science Thesis have been approved by the Council meeting of the Faculty of Engineering Sciences on 5th June 2013.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Noori, Hamed: Modeling and Simulation of Vehicle to Vehicle and Infrastructure Communication in Realistic Large Scale Urban Area
Master of Science Thesis, 86 pages, 8 Appendix pages
September 2013
Examiners: Professor Yevgeni Koucheryavy, Professor Pasi Kallio, Dr. Alexey Vinel
Keywords: Intelligent Transportation Systems; Vehicle to Vehicle Communication; Beaconing; Emergency Vehicles Response Time; Probability of Beacon Delivery; Dynamic Route Planning; OMNET++; SUMO; Veins; IEEE 802.11p.

During the last decades, Intelligent Transportation System (ITS) has progressed at a rapid rate, which aim to improve transportation activities in terms of safety and efficiency. Car to Car or Vehicle-to-Vehicle (V2V) communications and Car/Vehicle-to-Infrastructure (I2V or V2I) communications are important components of the ITS architecture. Communication between cars is often referred to Vehicular Ad-Hoc Networks (VANET) and it has many advantages such as: reducing cars accidents, minimizing the traffic jam, reducing fuel consumption and emissions and etc. VANET architectures have been standardized in the IEEE-802.11p specification. For a closer look on V2V and V2I studies, the necessity of simulations is obvious. Network simulators can simulate the ad-hoc network but they cannot simulate the huge traffic of cities. In order to solve this problem, this thesis studies the Veins framework which is used to run a traffic (SUMO) and a network (OMNET++) simulator in parallel and simulates the realistic traffics of the city of Cologne, Germany, as an ad-hoc network. Several different simulations and performance analyses have been done to investigate the ability of different VANET applications. In the simulations, cars move in the real map of the city of Cologne and communicate with each other and also with RoadSideUnits with using IEEE 802.11p standard. Then, Probability of Beacons Delivery (PBD) in different area of a real city are calculated and also are compared with the analytical model. This study is the first research performed on calculating PBD of IEEE 802.11p in realistic large urban area. Then, the thesis focuses on modelling and analysis of the applications of the V2I in real city. In these sections, two different simulations of application of the VANET are done by developing the Veins framework and also by developing two new programs written in Python which are connected to SUMO and control the real traffic simulation. One program simulates a real city with intelligent traffic lights for decreasing response time of emergency vehicles by using V2I. The results show that using V2I communication based on 802.11p between emergency cars and traffic lights can decrease the response time of emergency cars up to 70%. Another program, simulates dynamic route planning in real traffic simulation which is used V2I and V2V communication. The result of this simulation show the capability of V2V and V2I to decrease the traveling time, fuel consumptions and emissions of the cars in the city.

PREFACE

The research work carried out in this Master Thesis has been done at the Department of Electronic and Communication Engineering, Tampere University of Technology, Finland. This thesis is completed during the years 2012-2013.

Acknowledgements

First and foremost, I would like to express my heartfelt gratitude to my supervisor Prof. **Yevgeni Koucheryavy** for the continuous support of my M.Sc. study and research, for his patience, motivation, enthusiasm, and immense knowledge and for that all great forgiveness of my several mistakes. Your guidance helped me in all the times not only in research but also in real life. You gave me the chance to continue my studies in my interested field and you gave me the chance to release my potential in research and you taught me how to do research and what is important in real academic world. I could not have imagined having a better supervisor and mentor for my M.Sc. study.

I am also very grateful to Dr. Alexey Vinel who has been always there to listen and give advices. I am deeply grateful to him for the long discussions that helped me sort out the technical details of my work. Alexey is not only a mentor but also a dear friend. I could not have asked for better role models, each inspirational, supportive, and patient. I have been amazingly fortunate to have an advisor who gave me the freedom to explore on my own, and at the same time the guidance to recover when my steps faltered.

I would also like to thank my examiner, Professor Pasi Kallio, who provided encouraging and constructive feedback and supports. And for his great personality who gave me permission to continue my studies in my interested field.

I would like to thank my parents, who instilled within me a love of creative pursuits, science and language, all of which finds a place in this thesis. To my parents, thank you and my siblings, Faezeh and Maedeh, have also been the best of friends along this journey and sorry for being away.

To my dear friend, Mohammad Moosavi, thanks for being such a dear friend and an awesome brother, thanks for being there to listen. Thanks for thousands years friendship.

And, **Elham**, my dear wife, how can I say thank you for your supports, your Love, your feeling and for your eyes when you look at me. You are my whole life and whole my life is hollow without you. Thanks for being with me dear.

CONTENTS

List of Figures

List of Tables

List of Acronyms

AIFS	Arbitration Interframe Space
C2CC	Car to Car Communication
CAN	Controller Area Network
CCA	Clear Channel Assessment
CCH	Control Channel
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
CTT	Current Traveling Time
CW	Contention Window
DARPA	Defence Advanced Research Projects Agency
DCF	Distributed Coordination Function
DIFS	Distributed Interframe Space
DRP	Dynamic Route Planning
DSRC	Dedicated Short Range Communication
ECU	Electronic Control Unit
EDCA	Enhanced Distributed Channel Access Function
EIFS	Extended Interframe Space
EMS	Emergency Medical Services
ETC	Electronic Toll Collection
ETSI	European Telecommunications Standards Institute
FCC	Federal Communication Commission
FDM	Frequency Division Multiplex
FHWA	U.S. Department of Transportation and Federal HighWay Administration
GlomoSim	Global Mobile Information System Simulator
HBEFA	Handbook Emission Factors for Road Transport
IDM/IM	Driving Model with Intersection Management
IDM/LC	Intelligent Driving Model with Lane Changing
IEEE	Institute of Electrical and Electronics Engineers
IFS	Interframe Space
IP	Internet Protocol
ITS	Intelligent Transport System
ITS-DLR	Institute of Transportation Systems at the German Aerospace Center
ITT	Ideal Traveling Time
IVC	Inter Vehicle Communication
LAN	Local Area Networks
LIN	Local Interconnect Network
LLC	Logical Link Control
MAC	Media Access Control
MAN	Metropolitan Area Networks
MANET	Mobile Ad-Hoc Network

NCTUns	National Chiao Tung University Network Simulation
NED	Network Description
NHTSA	National Highway Traffic Safety Administration
OBU	On-board Unit
OD	Origin–Destination
OFDM	Orthogonal Frequency Division Multiplexing
OMNeT++	Objective Modular Network Tested in C++
OSM	Open Street Map
PBD	Probability of Beacons Delivery
PCF	Point Coordination Function
PIFS	Point Coordination Interframe Space
PoI	Points-of-Interests
QoS	Quality of Service
RDS	Radio Data System
RFID	Radio Frequency Identification
RSU	Road Side Unit
SCH	Service Channel
SIFS	Shortest Interframe Space
SMP	Shared-Memory symmetric processor
ST	Starting Time
TIC	Traffic Information Center
TMC	Traffic Messaging Channel
TraCI	Traffic Control Interface
TraNS	Traffic and Network Simulation Environment
TS	Time-Slot
UHF	Ultra High Frequency
V2I	Vehicle to Infrastructure
V2R	Vehicle to Roadside
V2V	Vehicle to Vehicle
VANET	Vehicular Ad-Hoc Network
VINT	Virtual Inter-Network Testbed
WAVE	Wireless Access in Vehicular Environment
WLAN	Wireless Local Area Network

Chapter 1

1. INTRODUCTION

The introduction chapter will present an overview to Vehicular Ad-Hoc Networks (VANET), and will give short explanation about what are the VANETs, characteristics of VANET, and the requirements of VANET. Furthermore, this chapter will introduce the outline of the thesis, the scope and objective of this work.

In the last decades, everything is change to become wireless. The flexibility, mobility and accessibility makes wireless technology dominant approach for sending and receiving all kind of information. Wireless internet, cellular phones and TV are famous applications of wireless technology. Wireless technologies are growing faster than any other field in communication. One promising application of the wireless technology is using communication between the vehicles.

Recently increasing the number of cars on city roads has created many problems, such as traffic congestion, the huge number of citizens getting killed in cars accidents, fuel consumption, emissions and etc. For example, according to the National Highway Traffic Safety Administration (NHTSA), there are about 43,000 people who are killed in fatal car accidents each year in the United States [1]. In order to solve these problems and make the roads convenient and safe, researcher have worked on many solutions, one of them is communication between cars [2][3]. Also, the increasing the number of cars on city roads of most countries is putting the cities at the risk for experimenting emissions, accidents and traffic congestions which is lead to very bad and low-quality of life. Very huge amount of fuels, resources, time and money have been wasted every single day, because of traffic congestions. For example, in the U.S. each driver spent an additional 38 hour in vehicle and wasted 19 gal of fuel because of traffic congestion in 2011 which leads to 818 $ congestion cost per auto commuter. This means, wasting 11 billion litters and 121 billion dollars for congestion cost just in U.S. in one year [9]. A comparison with 1982 (16 h, 30 litters) illustrates that the increasing congestions and traffic volume in cities, cause an increasing problem in urban life and transportation systems. Car to Car Communication (C2CC) or Vehicle to Vehicle (V2V) communication based on wireless technology have been designed to help cars to “talk” to each other. Communication between cars often referred to vehicular Ad-Hoc Networks (VANET) has many advantages such as: reducing cars accidents, minimizing the traffic jam, reducing fuel consumption and emissions and etc. The communication methods between the cars have various different approaches but the low cost solution is to use existing wireless networks such as 3G, Wi-Fi, WiMAX, etc. [4][5]. The term VANET was first introduced during the first ACM International Workshop on VANET (Vehicular Ad-Hoc Network) at Philadelphia in 2004.

1.1. What is VANET?

In the last years, an important domain in the computer and network science has been studied which is the communication between cars. Mentioned communication could be achieved with the using Vehicular Ad-Hoc Network or VANET which is similar to the Mobile Ad-Hoc Networks or MANET, used for providing transferring information between close cars and between cars and road side units. The focusing and also the main goal for VANETs are providing comfort and safety for vehicles' passengers. To achieve mentioned goal, an electronic device such as a wireless modem should be implemented inside cars for providing communication for the customers. Cars who has equipped with ad-hoc network devices, are considered as a node in VANET and they can send/receive information with other cars and road side units (RSUs).

VANET represents a research field that is a particular class of MANET (Mobile Ad-Hoc Networks). The basilar concept of the VANET is straight and simple: create a wide and cheap wireless technology to connect vehicles to each other and to road side units (RSU) for sending and receiving the information. The important features of VANETs include these items: Cars and RSU are nodes in VANET. The nodes can move very fast, and the considered network is highly dynamic which means that topology of the network is continuously changing with changing the position of the nodes and density [7] [8]. In summary, VANET is the technology for communication between vehicle to eachother and also with road side units.

Based on information from [6], nodes in VANET are in two types, as shown in Fig. 1-1; RSUs, Road Side Units or static nodes and OBUs, On Board Units or mobile nodes. In addition, Fig. 1-1, illustrates different types of communication in VANETs. Vehicle to Roadside/Infrastructure Communication (V2R or V2I) and second one is Inter Vehicle Communication (IVC). Furthermore, RSUs can communicate with each other which is called as Inter-Roadside Communication.

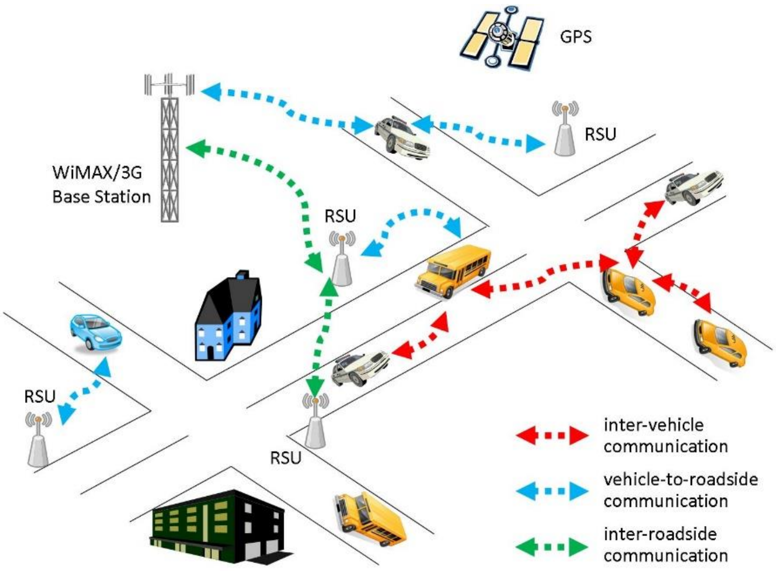


Figure 1-1. *Different Node and Communication types in VANETs[6].*

Another important issue which should be considered is that, in order to relax traffic related problems in future, several approaches are proposed, but it is clear that extending the road network is not suitable to solve the traffic related problems in future due to the limitation of resource, places, etc. Thus, novel scenarios are required. For example, the Federal Highway Administration in the U.S. (FHWA) proposed and defined three general tactics to decrease traffic related problems [10]:

- Work on current capacity of roads and extend them, e.g. increasing the size and number of streets and highways.
- Extension of alternative transportation which are required less resources. For example transportation of non-automotive.
- More efficient using of current capacities of cities and roads.

Wireless communication technology is one of the useful solutions in order to solve traffic problems which focuses on the third strategy. Car to Car Communication (C2CC) or Vehicle to Infrastructure (V2I) and Vehicle to Vehicle (V2V) Communication or V2X communication based on Wireless technologies is designed to help to solve these problems. V2V and V2I have several applications in Intelligent Transportation Systems (ITSs) to make transportation more efficient.

The most useful method for this wireless communication between cars is the VANET topology because of the unique characteristics of VANET. For example, some of unique characteristics are the rapidly changing network topology and mobility caused by the high travelling speed of the nodes, the constrained pattern due to the restricted roads.

1.2. VANET Characteristic:

As mentioned in previous section, VANETs have several unique characteristics. VANET is the extension of MANET, it means that, VANETs have most of MANET features and also several new features which make that greatly different from MANET. The characteristics of VANET are unique compared to the other ad-hoc networks. Mobile Ad-hoc Networks have no stationary infrastructure and rather they depend on ordinary nodes to perform network management and routing of messages. The human interaction, restriction on mobility and constantly changing vehicle speed generates an exclusive set of characteristics in vehicular networks. Moreover, a VANET can be modelled as continuous but fairly anticipated changing protocol, also the fragmentation in the network occurs quite often, additionally the communication range of VANET is smaller as compared to the MANET and VANET introduces significant number of networking challenges. In following, there are some important characteristic of VANETs (**data from [11]**):

a) **High Fragmentation of Network**: Because of the unavailability of VANET devices for all the vehicles, this will cause fragmentation in the network which will result in inaccessibility for the portion of the network. Fragmentation will occur because in the initial deployment of the network every vehicle does not have a transceiver installed in its OBU.
b) **Scalability**: In a VANET the effective diameter of the network is small and because of the rapidly changing movement of the vehicles the connectivity can be poor because many paths will be disconnected before they can be utilized so it is impractical for the vehicles to follow any global topology.
c) **Security and Privacy Constraints**: In VANET, every vehicle is basically a real person; and a data has been transmitted from the vehicle it somehow exposes that who is driving the vehicle. Therefore in order to make this technology publicly acceptable, this kind of security glitches should have to remove from VANET. Similarly there is no provision in any case that the user can temper the transmitted message. This will result in automatic collision and it is extremely dangerous for safety applications.
d) **Power**: The battery life is not a problem in OBU since they are normally powered by the vehicle battery and their power consumption is significantly lower than the vehicle own equipment. Therefore, there is no substantial design challenge when it comes to power consumption.
e) **Variable Network Density**: Due to the traffic movement, network density is always changing. In traffic jam, the network may be highly congested and during night time, it is possible that there would be a very sparse network.

f) **Mobility**: In VANET environment, because to the high movements of cars, the lifetime of a communication link is very short and it creates a worst case scenario if the vehicles are travelling in opposite direction.

g) **Low Communication Latency**: Latency is very critical for the broadcasted messages in an emergency situation.

1.3. Potential Applications of VANETs

The emerging VANET technologies are going to defines numerous applications for driver assistance, have efficient traffic, safety, etc. There are a number of potential applications that can be available in VANET (both V2V and V2I). The applications could considered based on safety related (non-safety related, safety related) or based on communication methods (Vehicle to RSU, Vehicle to Vehicle). Next sections, will describe the several potential applications of VANET and will give some examples for each types.

1.3.1. Safety:

VANET technologies could be applied to reduce injuries, save lives and to reduce accidents. Examples include accident warnings, collision warning, lane departure warning, work zone warnings, obstacle detection, vehicle breakdown, and so on.

In each application, data are disseminated after they have been processed to their destination, which can be another vehicle or infrastructure, depending on the reason and conditions to send such a message. Two different kind of messages could be used in safety related application: event-driven messages and periodic messages. When an actual accident has occurred, event driven messages are generated or when a danger has been detected. To inform neighboring cars, the periodic messages which are also known as beacons, are generated continuously. This message contains car‘s direction, speed, lane, position, etc. These beacons are used to predict a potentially hazardous situation.

- **Periodic Messages**: vehicle-related information such as direction, position and speed, need to be known from other surrounding vehicles to take it into consideration in order to make a decision in an attempt to prevent a hazardous situation from happening. Therefore, periodic messages are considered to be an important type of message that supports the decision that can be taken in safety applications; however, it may lead to wasted bandwidth consumption, especially in dense environments, in addition to increasing the probability of a storm problem occurring.
- **Event-Driven Messages**: this type of message is disseminated only when hazardous conditions are occurring, otherwise this type of messages is not sent. It occupies a high level of priority; the main difficulty in this type of message is the necessity to increase the guarantee of delivering this message to all vehicles that need to know about it.

Following part will present several applications of the VANET in safety related area:

1.3.2. Entertainment/ Information Applications

The information/entertainment applications, sometimes called non safety applications, focus on leveraging the efficiency of traffic and providing drivers with more comfortable journeys. For instance, informing the passenger where the nearest fuel station is, supplying information about the current weather condition in a specific area, illustrating restaurant menus and their prices, giving information about the availability of the parking spaces in the nearest car parks, providing the facilities to access internet services in a smooth way during the journey and enabling passengers to play games online, Coupling/Decoupling, Parking Lot Payment, Electronic Toll Collection (ETC), Inter-Vehicle Communications [12].

1.3.3. Traffic Efficiency

VANET technologies could be used for improving traffic efficiency and decrease traffic congestions. For example, electronic toll collection, highway traffic management cooperative adaptive cruise control, congestion related information for controlling traffic, etc.[3].

1.3.4. Driver Assistance

VANET could also used for providing good communications and accurate data to vehicles for improving security and safety. For example: navigation system, digital road maps downloading, automatic emergency call, various warning information, parking information, real-time traffic information, etc.[3].

1.4. Research Aims and Objectives

The objectives of this research are:

- Detailed survey on VANETs and different approaches and application of V2V and V2I communications in real world.
- To investigate different protocols and standards which are used in VANET and to select the appropriate standard and explain the challenges of the mentioned standard.
- Investigate the simulation of VANET and required tools in order to simulate V2V and V2I communication.
- Compare different simulators which are used for VANET simulation and explain all features and drawbacks and select the best choice in order to simulate a real VANET-based V2V and V2I communication.
- To find a real traffic demand data and realistic map in order to simulate realistic VANET and to obtain realistic and reliable result.
- Simulation of realistic V2I and Vehicle to Vehicle communication using real map and real traffic data in a large scale urban area.

- Performance analysis of the beaconing concept in IEEE 802.11p and calculate probabilities of beacon delivery and compare with analytical model.
- Study on safety application of the V2I communication in real simulation which consists of two main parts:
    - First, to study the effect of VANET-enabled traffic lights control on decreasing the response time of emergency vehicles in realistic large-scale urban area.
    - Second, to investigate the impact of VANET-based V2I communication using IEEE 802.11p on reducing vehicles traveling time, fuel consumption, emissions and traffic monitoring in realistic large scale urban.

1.5. Thesis Contribution and Outline

The contribution and tasks which are done in this Thesis is threefold:

A. First part, is a wide study on the VANET, application of VANET, protocols and standards in VANETs, simulation of VANET, comparing the simulators and selecting the best choice for simulating a realistic Vehicle to Vehicle and Vehicle to Infrastructure communication. **Chapter 1**, is a background and introduction for VANET characteristics and applications. **Chapter 2**, is a study on different standards and protocols which are used on VANET in details. **Chapter 3,** is related to simulation of VANET and presents different required tools for the VANET simulations.
B. Second part, includes simulation of the VANET-based V2V and V2I using IEEE 802.11p and performance analysis. In this part, realistic traffic information and realistic map are used in simulation. **Chapter 4**, presents realistic simulation of VANET and different specifications of simulation and all required data. This chapter, uses three different software in parallel in order to simulate realistic V2V and V2I communication in realistic large scale urban area. Furthermore, this chapter, presents a performance analysis of the VANET simulation and calculates Probability of Beacons Delivery as a one of the most important parameters in VANET and also compares the results with analytical model.
C. Third part, focuses on modelling and analysis of the application of the V2I in real city. In these sections, two different simulations of applications of the VANET are done. **Chapter 5**, focuses on modelling and simulation of the VANET-enabled traffic lights control on the response time of emergency vehicles in realistic large-scale urban area. **Chapter 6**, simulates another application of VANET which is related to decreasing the traveling time of the cars, decreasing fuel consumption, decreasing the emissions and traffic monitoring using VANET scenario.

Chapter 2

2. VANET PROTOCOLS AND STANDARDS

In this chapter, a wide study on the different standards and protocols which are used on VANET, is done in details. First, an introduction on different wireless technologies is presented. Then, the IEEE 802 family is introduced and related standards and task groups are presented. Then, related information about IEEE 802.11 MAC layer is explained and also WAVE system architectures are presented.

2.1. Introduction

VANETs used broadcasting for communication, it means that each vehicles will transmit several messages and these messages could be received by several neighbors' vehicles, and also neighbors RSUs. There is no communication coordinator in the VANET architecture and this is the main challenge of VANET. In the VANET one control channel (shared) is needed even if several channel would be available using several transmitter and receiver. Requirement for distributed control channel and also mentioned "One-Channel" are the key challenge of designs for VANETS. The design for MAC is the main and most important issue in the VANET

During the last decades, the control for vehicles has been changed to digital from analog systems. In the last years, novel systems such as (X-Wire) have been appeared and the research directions for automobiles industry have been changed totally. In order to understand the diverse functions (e.g. air bag deployment, engine management, intelligent brake system) the networked electric control units (ECU) are developed and implemented in the most of vehicles. As an example, in the Mercedes S Class car, more than 80 networked electric control units are used. Beside this technologies, emerging vehicular networks, vehicle to Infrastructure (V2I), vehicle to vehicle (V2V), and Intra-Vehicle (InV) communications are developed and become available very fast which is open numerous applications for driver assistant, efficiency of traffic and most importantly safety. Currently, there are several groups which are working on car communication standard to define the standards and protocols for V2V, V2I and InV communication. For example, FlexRay, Local Interconnected Network (LIN) and Control Area Network (CAN) is used for InV communication for communication between sensors, ECUs, etc.

Recently, many groups (automobile manufacturers, academia and standardization bodies) have been in the process of defining standards for vehicular applications and they have been worked together in for developing VANET-based communications systems. Dedicated Short Range Communication or DSRC are wireless protocols which is specifically designed for automotive application. These protocols (DSRC) offer high rate data communication between a vehicle and a RSU (V2I or I2V) or between two vehicles (V2V). For WAVE or Wireless Access in Vehicular Environments the DSRC standards include IEEE 1609.x family and IEEE 802.11p. DSRC groundwork is IEEE 802.11p, and higher layer standard is IEEE 1609. The physical and MAC (Medium Access Control) layer protocols are specified in IEEE 802.11p for single-channel operations. In addition, for multi-channel operations, a trial-use standard IEEE 1609.4 is specified on top of the IEEE 802.11p. Multi-channel operations are implemented by dividing the access time to two different channel with intervals of 50ms, Control and Service channel (CCH and SCH)

Numerous wireless technology have potential for wireless communication in InV, V2V and V2I communication with the rapid progress in information technology. Table 1 and Table 2 (data from [14]) and listed several technologies which are currently under used, developed or studied.

Table 2-1. *Wireless technologies for Intra Vehicle communications[14].*

	ZigBee	**UWB (ultra-wide band)**	**Bluetooth**	**Wireless USB (Universal Serial Port**	**Wireless CAN**
Standard/ Technology	Ratified in December 2004	Transmitting information Larger bandwidth (> 500 M Hz)	First launched (1998)	Short-range, high bandwidth based on the WiMedia Alliance's UWB	CANRF (CAN over RF)/ CAN Bridge
Coverage	10 to 100 meters	< 60 meters for a 500 MHz wide pulse, < 1.3 GHz bandwidth pulse 23 cm	1 meter, 10 meters, 100 meters	110 Mbits/s 480 Mbits/s distances	150 Meters
Bit Rate	20-250 Kbit/s per Channel	extremely high data rates 1000+ Mbps	3 Mbit/s (Version 2.0 + EDR) 53-480 Mbps (WiMedia Alliance (proposed)	110 Mbits/s 480 Mbits/s distances	20kbps/ 52.8kbps-164.4kbps

Table 2-2. *Wireless technologies for V2V and V2I communications[14].*

	GSM/3G	WiFi (Wi-Fi Alliance Version of 802.11n)	WiMax	DSRC
Standard/ Technology	Third generation cellular technology in 2001	New Wi-Fi technology with MIMO standard in 2009,802.11n standard in 2009	Broadband technology in 2007	A short to medium range communications
Coverage	kilometres	500m	5 km	1000m
Bit Rate	2-3 Mbps	600Mbps using MIMO	75Mbps	6 to 27 Mbps
Applications	Between Vehicle and mobile phone communication	V2V and V2I	Internet access, Email, VoIP (Voice over IP)	V2V and V2I

2.2. Wireless Technology

During the last decades, for the wireless communication, the novel technologies brought several advantages. Wireless technologies cause efficiently working, high mobility and extremely economical. Today, wireless technologies are divided into two main groups, Local area technologies and large area technologies. Wireless local area network (WLAN) is an example for local area technologies and UMTS, GPRS and GSM are the examples for large area technologies.

Apart from V2V, multiple access wireless technologies can be used for the information dissemination in V2I, e.g. IEEE 802.11p, IEEE 802.11a, IEEE 802.16, Infrared technology and cellular networks, e.g., UMTS, LTE. Besides, since the infrastructure part is fixed, V2I is more appropriate to be used for long distance communication. Furthermore, the V2I can easily have a larger communication range than the V2V communication range. Thus, for V2I, information can be disseminated at a certain area in one communication hop, while V2V will need to use multi-hop communication to disseminate the same information at the same area. However, the main drawback of V2I communication compared to V2V communication is the higher communication delay involved in disseminating information between two neighbouring vehicles.

2.3. The IEEE 802 family

The IEEE 802.11 standard [15] is included in 802 family. 802 standard is used for LAN, or Local Area Networks and MAN, or Metropolitan Area Networks. The 802 family is illustrated in the Fig. 2-1 and Fig. 2-2 [15].

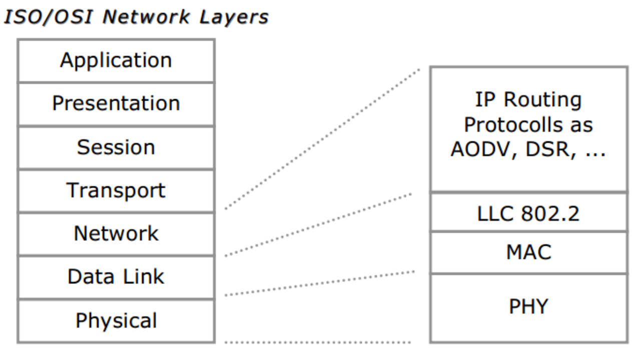


Figure 2-1. *ISO/OSI layer model.*

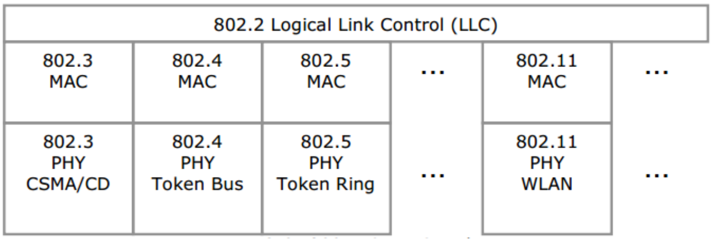


Figure 2-2. *802 PHY, MAC and LLC [15].*

2.4. The IEEE 802.11

PHY and MAC layer specification are standardized in the IEEE 802.11. There several extensions for 802.11 which are already been added to IEEE 802.11 either PHY or MAC Layer. Focusing part in the MAC extensions mostly were on improvement of quality of service (QoS) or security. The PHY extensions mainly changed the way in which the physical layer works. A structured overview of 802.11 is given in the Fig. 2-3 and a list of the current extensions could be seen on next parts.

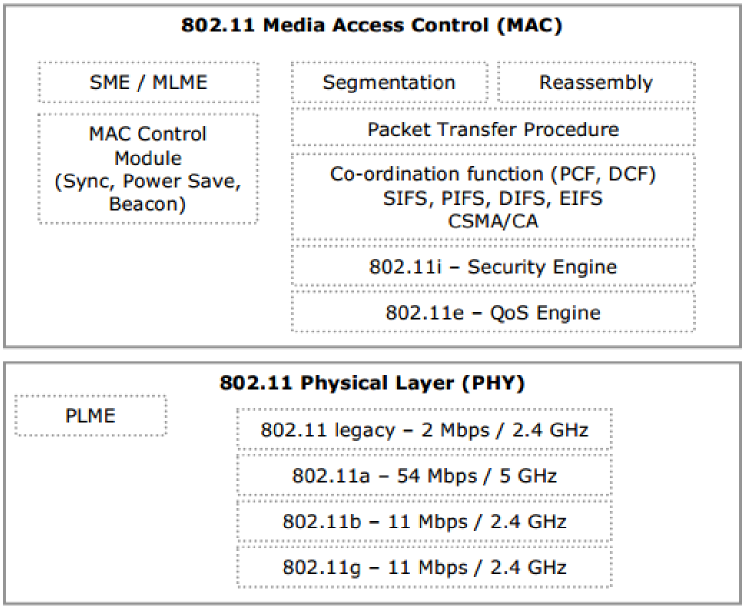


Figure 2-3. *PHY layer and MAC 802.11 [15].*

2.4.1. Standards and Task groups for the IEEE 802.11

Table 2-3 describe all extensions for 802.11.

Table 2-3. *Overview of the IEEE 802.11 standards[12].*

Standard	Description
802.11a	5GHz OFDM PHY – 54 Mbps
802.11b	802.11b 2.4GHz CCK PHY – 11 Mbps
802.11c	802.11 bridging
802.11d	International roaming
802.11e	QoS/efficiency enhancements
802.11f	Inter AP protocol
802.11g	2.4GHz OFDM PHY – 54 Mbps
802.11h	5GHz regulatory extensions
802.11i	Security enhancements
802.11j	Japan 5GHz band extensions
802.11k	Radio resource measurement
802.11m	Maintenance
802.11n	High throughput PHY
802.11p	V2V and V2I

2.5. IEEE-802.11 MAC Layer

This part are going to provide a brief explanation of some basic information for the IEEE 802.11 medium access control layer [16]. The IEEE-802.11 defines PHY and MAC layer specifications to use in the wireless communication of moving, portable or fixed stations in a local area network range. IEEE-802.11 explains several physical signaling and one set of MAC procedures for supporting the packet delivery services. To make the 802.11 more usable for some specific purposes, it is include a long list of amendments [17]. Every single one in this amendments, when defining some parameters for the PHY could share the common MAC. WAVE or Wireless-Access in the Vehicular Environments. [18].

2.5.1. Access Method

CSMA/CA or Carrier-Sense-Multiple-Access with Collision-Avoidance is the basic access method of IEEE 802.11. The basic operation of CSMA/CA is explained in the following:

Before transmitting a node has to sense the medium to find out if there are some other stations which are transmitting. If radio channel was busy with other transmissions, then the mentioned station defer its transmissions to another later times, but if the channel was free, the station is free to transmit.

When the medium is operating under normal load this kind of approach is very efficient, because it allows the terminals to transmit packets without any necessary delay. Even with this delay before the transmission, the collision is also possible since two stations can instantaneously senses the channel for being free to begin to transmits which will cause a collision. In IEEE 802.3 exponential random backoff algorithm, is used to retransmit the packet in case of a collision. The collision in Ethernet is recognized by the transmitting station after a set of message exchanges but this kind of collision detection cannot be used on a wireless channel mainly because of two reasons[11].

- In-order to deploy a collision detection mechanism in a wireless network would involve a full-duplex radio which will be able for receiving and transmitting at the same time, this method has economic constraint and will increase the price of such a system.
- One of the basic assumptions of a wireless channel is that stations cannot hear each other (on the contrary in a wired network this assumption is reversed) and also the waiting time that is incorporated before the sending the packet does not necessarily reveals that the channel is free near the intended receiver.

2.5.2. Functions for Channel Access

MAC for the IEEE 802.11 is explains 4 functions to channel access (as seen in the Table 2-4)

- Hybrid Controlled Channel Access or HCCA
- Point Coordination Function or PCF
- The Distributed Coordination Function or DCF
- The Enhanced Distributed Channel Access Function or EDCA

Table 2-4. *Functions for channel access in IEEE 802.11*

	Coordinator Point	**Ad-Hoc**
non-QoS	PCF	DCF
QoS	HCCA	EDCA

In the following a detailed description of the above mentioned channel access method is presented (data from [11]).

Distributed Coordination Function

CSMA algorithm is used in DCF. A station listens to the medium before it can transmit a MAC frame. As discussed earlier, the station can only transmit while the channle is idle. The stations have to wait til the current transmissions are complete before starting its own transmissions. There are no provisions for collisions detection, because on wireless radio channel, collisions detection is not realistic [19]. In order to make the transmission and reception function as fair as possible in CSMA, DCF provides with sets of delays which refers to a priority scheme. This delay is known as Interframe Space (IFS). Fig. 2-4 illustrate a flowchart for the DCF operation which is used IFS.

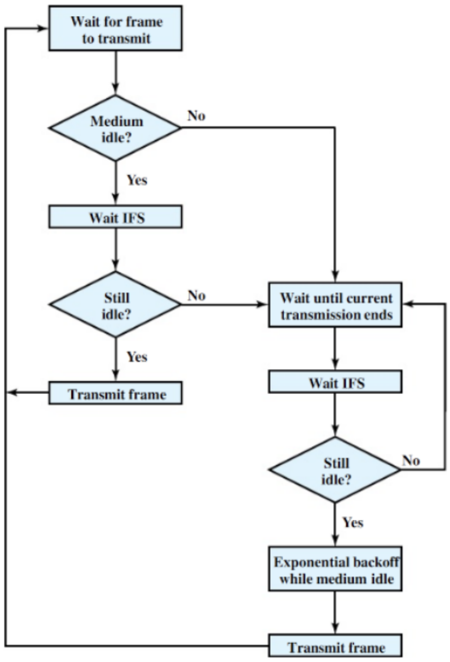


Figure 2-4. *IEEE 802.11 MAC Logic[19].*

The follow through of the flow chart goes like this.

- Initially, the stations with a packet for transmitting will sense channel. If channel is free, then station will wait for observing if the channel is idle for IFS. In that case the station immediately transmits.
- When the channels are busy; the stations will defers its transmissions and monitors the channel until the ongoing transmissions is over.
- The station will wait for another IFS when the current transmission is completed. If the mediums are idle for this amount of times, then the stations will used a random time (backoff) and again sense the channel. The station will transmit if it found that the channel is idle for that duration of time. The backoff timer will be stopped if the channel becomes busy and it will resume when the channel becomes idle again.

In order to explain the basic information of the protocols, the following parts are copied from [6] because the details explanation of the protocol is not part ot this thesis and explanation is repetitive work, So this part is copied from [6] to give some more details of DCF. “The DCF, follows the principle of carrier sense multiple access with collision avoidance (CSMA/CA), i.e. the channel is only accessed if the physical layer does not observe any ongoing activity on it. Collision avoidance is provided by several additional technologies on the MAC layer described in the following. To allow medium access strategies, the physical layer has to notify the channel status to the MAC layer, called clear channel assessment (CCA). It is not specified how exactly a wireless card should identify the status of the medium, instead, the medium should be indicated busy if the received power level is higher than a certain threshold in case a valid frame transmission is observed. It should also be indicated busy in the absence of a valid transmission if the received power level exceeds a second, higher threshold. An important mechanism are inter frame spaces (IFSs), which are time durations that the medium has to be indicated as idle before the station may transmit. IFSs of different length for different frame types allow prioritized access. For example, important control packets such as acknowledgments are sent after a short inter frame space (SIFS), whereas regular data packets are not transmitted before the medium was sensed idle for the duration of a distributed IFS (DIFS), that exceeds the length of SIFS by two so-called slot times.

In the case where the medium is determined busy, Fig. 2-5 illustrates the medium access strategy: the station selects a random number of backoff slots within a certain range, the contention window. The slots are counted down after the medium was sensed idle for the duration of a DIFS; the countdown is interrupted whenever the medium is determined busy. Whenever the countdown reaches zero the frame is transmitted. When there are unicast packets for which no acknowledgment is received, a retransmission is scheduled after a newly selected number of backoff slots under the use of an increased contention window (exponential backoff)" [6].

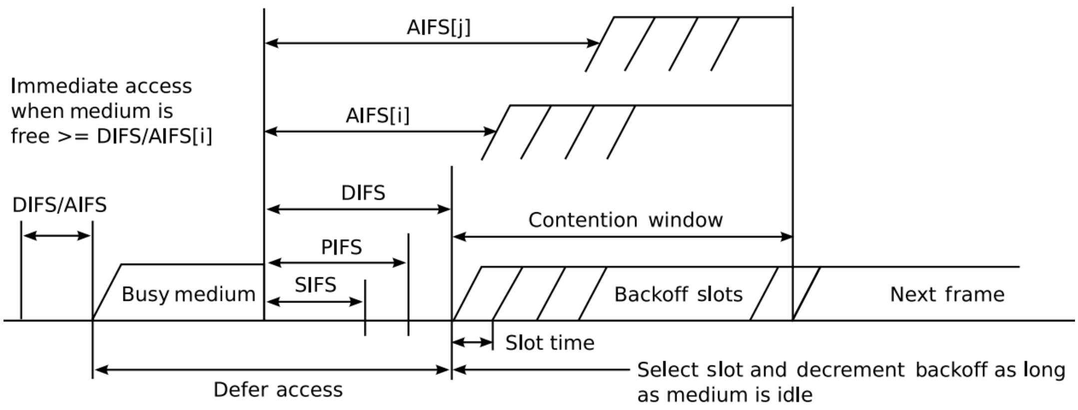


Figure 2-5. *IEEE 802.11 channel access Distributed coordination [6].*

Interframe Spaces (IFS)

This section provide some brief information about IFSs. "The IFS (Interframe space) are time interval between transmissions of two consecutive frames from different nodes, whether it was a new session or just a handshaking packet in the same session.[20]". Based on stations priority, each one shoulc wait for a different IFSs. There are 5 different IFS which are mentioned in following (Fig.2-5)

- Point Coordination Function (PCF) Interframe Space, PIFS
- Arbitration InterframeSpace (for using in the QoS facility), AIFS
- Short Interframe Space, SIFS
- Extended InterframeSpace, EIFS
- DistributedCoordinationFunction (DCF) Interframe Space, DIFS

Table 2-5. *Categories Access for QoS [20].*

Priority	AC	Designation
Lowest	AC_BK	Background
	AC_BE	Best Effort
	AC_VI	Video
Highest	AC_VO	Voice

Random Backoff Time

In the EDCA and DCF (contention based access functions), protocol for the channel access must be used in more efficient way when being distributed, that the nodes in the network must reach low collisions probabilities without helps of coordinator point.

In Summary, the CW must take a higher value if any collisions happened till achieving CW-max, then after transmissions that must be change to the CW-min. The value for the CW of each node deployed in DCF must be

$$CW = 2^{(i)} - 1$$

where the values for i are equal to 3-8 as illustrate in Fig. 2-6.

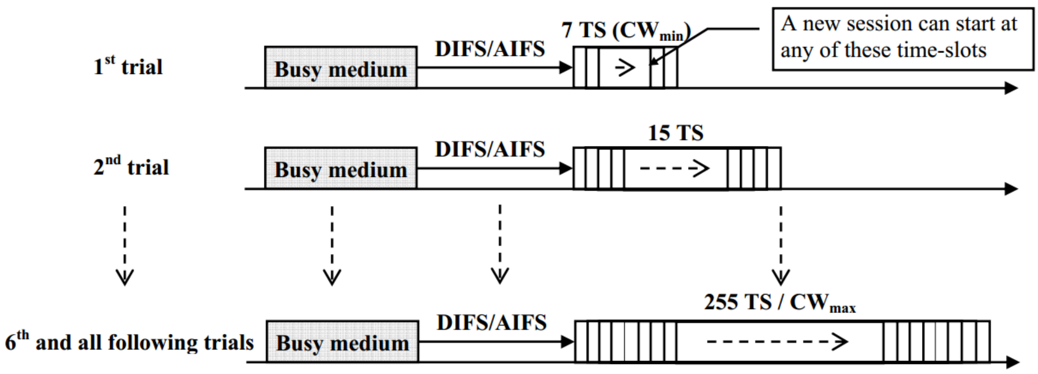


Figure 2-6. *Exponential increase of CW[20].*

2.6. WAVE System Architecture

Recently, many groups (academia, automobile manufacturers and standardization bodies) have been in the process of defining standards for vehicular applications and they have been worked together for developing VANET-based communication systems. Dedicated Short Range Communication or DSRC is a wireless protocol which is specifically designed for automotive application. These protocols (DSRC) offer high rate data communication between a vehicle and a RSU (V2I or I2V) or between two vehicles (V2V). For Wireless Access in Vehicular Environments (WAVE) the DSRC standards include IEEE 1609.x family and IEEE 802.11p. DSRC groundwork is IEEE 802.11p, and higher layer standard is IEEE 1609 [20][21][22].

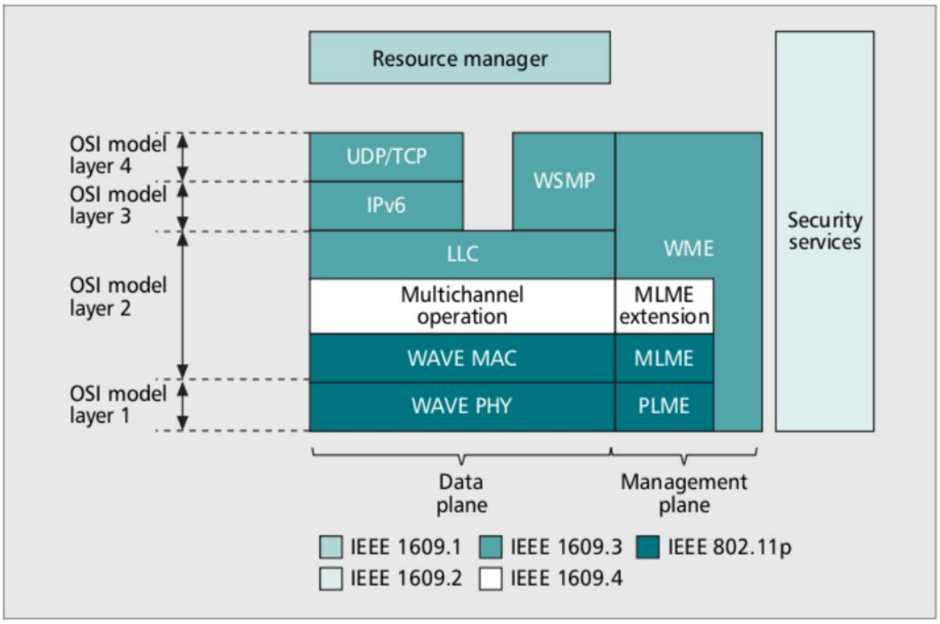


Figure 2-7. *WAVE architecture (copied from [23])*

WAVE protocols is includes of 5 complementary sections:

- IEEE-802.11p “Wireless Access in Vehicular Environments (WAVE)” [24], those are extension from IEEE-802.11 protocols and used for the PHY.
- IEEE-1609.1 “Resource Manager” [25] which is cover optional recommendations in App-Layer.
- IEEE-1609.2 “Security Services for Applications and Management Messages” [26] used for secure message processing, formatting.
- IEEE-1609.3 "Networking Services” [27] for the WAVE communications stacks.
- IEEE-1609.4 “Multi-Channel Operation” [28] for the how to use and arrangement the multiple channels .

The WAVE-protocol stacks illustrate in Fig. 2-7.

Chapter 3

3. VANET SIMULATION [1]

New protocols, scenarios and wireless technology schemes, because of complexity and high expenses cannot be accomplished in large testbed, especially due to high expenses and dangers of testing new technologies for transportation in the real world. Simulation plays an important role to find out the beneficial and effective technologies before implementation. This chapter is related to simulation of VANET and presents different required tools for the VANET simulations.

3.1. Introduction

Wireless technology has a huge effect on our life. As an example LANs, indoor wireless, to cellular mobiles, outdoor communication, and wireless technology has help billions of people in the world. For the VANET simulation, the software developers and the researchers together developed several programs in order to allows the studies and evaluations of numerous application such as emergency warning, routing and media access protocols.

MANET (Mobile Ad-Hoc networks) is a kind of MANET those are not need any fixed infrastructure. MANET is used for several situation when the communications are need to be mobile and there is no way to implementing fixed infrastructures. The VANET are a very fast growing research fields which is used for communication between vehicles and consider important protocols and standards for drivers and vehicles to help them to connect together and also for communication between the vehicles and Roadsideunit which are able to communicate between together and also with vehicles. The RSU could be any device which are implemented in the roads such as traffic lights. The characteristic of VANET could be illustrate as follow:

[1] The result of this chapter is published in following paper [30]: Noori, H "*Realistic Urban Traffic Simulation as Vehicular Ad-Hoc Network (VANET) via Veins Framework* ", 12th Conference of Open Innovations Framework Program. FRUCT, Nov.2012.

1- Traveling-based movement with ability to predict the location and time-based topology.
2- Several number of cars which are have correlated or independent speeds.
3- Very rapid time-varying medium. (for example when the transmission channel could be blocked with buildings)
4- Lane-based movement patterns.
5- Efficient power consumption is necessary for the VANET

Up to now, improvement of VANET is mostly have strong economic interests becuase of numerous application of the VANET such as V2V (vehicle to vehicle) communication which is make the vehicle to communicate and share the wireless channel for several application such as improving traffic safety (collision avoidance), efficent and advance traffic congestion control, Advance improved route planning, etc [31].

VANETs simulations are basically different from MANET simulations because in VANET simulation, vehicular environments have several new requirement and issues e.g. considering roadside obstacles and multi path fading, considering roads topologies, traffic lights, traffic congestion pattern, trip models, numbers different vehicles speed and mobility, traffic flow models, etc.

3.2. Required Tools

Simulation of VANETs networks are includes two different parts. Firstly, the issues which are should be considered about the vehicles communication. For this part, the network simulators are used and this network simulator are consider network protocols and focused on communication parts. Another aspect which should be considered is the pattern for movement of vehicles in the VANETs which is very important and essential. For this part, traffic simulators are mostly used to provide realistic traffic pattern and vehicles movement.

The important features of VANETs includes these items: Cars and RoadSideUnits (RSU) are assumed as nodes in VANET. The nodes can move very fast, and the considered network is highly dynamic which means that the topology of the network is continuously changing with changing the position of the nodes and density. VANET simulation requires two types of simulation components: Network and Mobility. In most cases the network and mobility simulator are separated. There are several simulators available that can be used for VANETs simulation. This study has classified existing VANET simulation software into three different categories: (a) Network generators, (b) Traffic simulators, (c) Software to integrate (a) and (b) or software which can simulate both mobility and network (VANETs simulator). Fig. 3-1 represents the classification of VANET simulators.

Traffic flow simulator generates required realistic vehicular mobility traces to be used in network simulator as an input. The network simulator calculates and creates the required components in a wireless network such as detailed structure of all nodes (cars), sending and receiving packets roles, data traffic transmission, channels, etc[32].

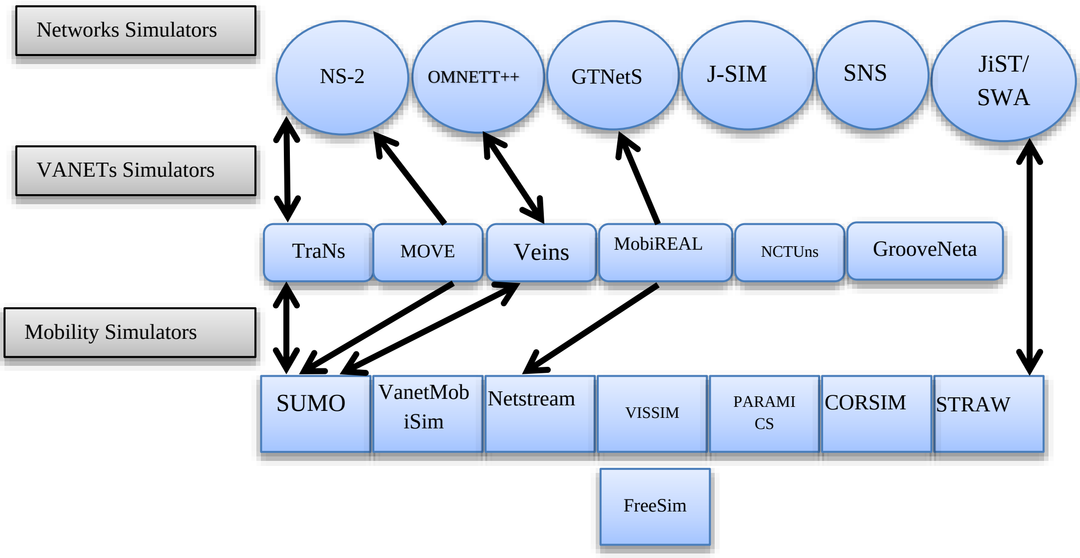


Figure 3-1. *Classification of VANET Simulators.*

3.2.1. VANET Mobility Generators (Traffic Simulation)

Creating a realistic mobility model for the simulation of VANETs is important. Vehicular traffic typically moves in relatively predictable ways along a set path. These movements are governed by how the road network is laid out. The placement of lanes and traffic features, e.g. traffic signs, turning lanes or traffic lights combined with both a source, destination and other vehicles decides how a vehicle will move in the real world. To get accurate results for how VANET technologies will work it is important to model these movements with a high degree of accuracy.

Initial work on MANETs often used random node movements. In essence nodes would choose random directions to move in and periodically change direction. This practice was initially carried over into VANET research. Of course it is nothing like vehicle traffic in the real world. Studies have shown that random node movements are a poor substitute for a mobility model and should not be used [34].

A second approach that was taken for a mobility model was the use of real world mobility traces. Obtained by tracking the location of real world vehicles using GPS or other technologies they mimic the real world exactly. Nodes within the simulation are then moved according to these traces exactly. While they do an excellent job of simulating mobility as it occurs in the real world they are of limited flexibility. Changing parameters, such as traffic density, is not feasible for large scale simulation. A better approach is the use of a dedicated traffic simulator. There are a wealth of traffic simulators available.

As explained before, in order to have realistic and acceptable simulation of VANET, mobility generator is required. Following sections will present several different mobility traffic models.

Traffic Model for VANET

In the in Civil engineering, traffic modeling and traffic flow pattern modeling creating are very important research area and those are vital to correct modeling of traffic flows for the designing phases of new streets and intersections [33]. Traffic models are categorized into 4 categories of traffic flow model based on their levels of detail of simulations, Macroscopic, Microscopic, Mesoscopic and Sub-microscopic.

The basic entity in the macroscopic model is the traffic flow and it is just modeled the traffic flow. The movement of every single cars in the roads are simulated in the Microscopic models. This model mainly assume that the behavior of each cars are depend on the physical ability of vehicles and also controlling behavior of drivers. Between Macroscopic and Microscopic simulation, there is mesoscopic model at the boundary. In the mesoscopic simulation, cars motilities are mainly modeled with use of queues approach, and each single car is moving through such queue. In the Sub-microscopic model, specification of the every cars is like the microscopic model, but in this model the mobility of vehicles is extended by dividing the characteristic of movement into more details. For example these details could explains the vehicles speed or gear switching in relation with rotation speed of engine. These details make this type of simulation more detailed compare to other simulation. But as it obvious, longer computation times is required in this sub-microscopic simulation.

In Fig. 3-2, the vehicular mobility models are in advance classified, where from left to right the following mobility model types are shown: sub-microscopic (within the circle: mesoscopic), microscopic and macroscopic.

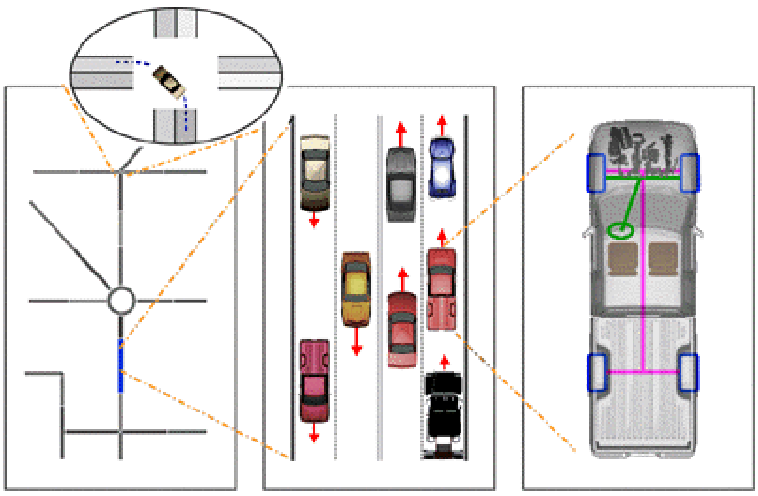


Figure 3-2. *Mobility Models: (From left to right) Macroscopic, Microscopic, Sub-Microscopic (in the circle: mesoscopic)[35].*

In the VANET simulation, the exact position of every single vehicle (nodes) are required and it is vital because of following reason: To have more accuracy for the modeling of communication and data transmission between nodes and also to understand the behavior of the medium and data lost based on the distance between cars. As explained before, the mesoscopic and macroscopic simulation could not provide these levels of the details, thus, the microscopic simulation model (which is able to simulate the movement of every single vehicle and also streets and intersections) are mostly used for the movement simulation for VANET.

Microscopic traffic simulation is based a microsimulation model of vehicle behaviour. The model must realistically mimic how a human driver would react to the world surrounding them. In addition, in microscopic model which is used in VANET simulation the required parameters for the mobility generator can be the roads map, scenario of cars traveling and some road and cars parameters like maximum cars speed, roads limitation, arrivals and departures times of each car, etc. Also the output can be the coordinate of each vehicle at every time and their mobility parameters like speed, acceleration, etc. Following are several examples for mobility simulator which will discuss later in next sections: SUMO, VanetMobiSim, CORSIM, CityMob, VISSIM, STRAW, PARAMICS, FreeSim and Netstream.

There are several distinctive aspect in mobility model in the VANET which are should be consider [12]:

- **Multi-description of motilities flows**:
  One of most important issues in a VANET simulators which are should be considered are that mobility and movement of each vehicles in the network should reflect the realistic behaviour of the traffic of roads, because this movement has significant effects on the network shape in the VANET. Movement of vehicles are typically considered at different layers:

  1- **Modeling of the Trip:**
  In this type of modeling, the movement of vehicles are define based on macroscopic model which is required two points (between Points-of-Interests (PoI)) based on the OD-Matrix (destination and origin matrix). This model is illustrated in the Fig. 3-3 (a). In this figure, the trip of the 2 cars is generated using OD-matrix (black and grey).

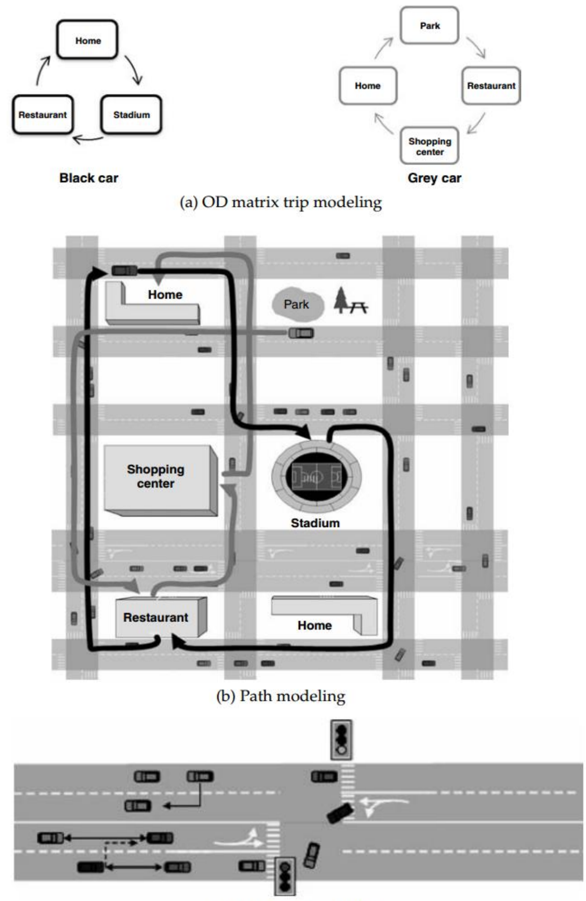
  2- **Path modeling,**
  Movement of vehicles in this type of modeling is defined based on the end to end paths which should the vehicles followed. In this models, every path might be use optimizing method based on preferences of the speed, traffic or drivers. Destination and origin points of the paths may generated according to trips model or randomly. For the black and grey cars, the path between points of interest are shown in the Fig. 3-3 (b).

3- **Flow modeling,**

Vehicles’ movement in this type are defined with more details levels. In the flow modeling, the interaction between cars are simulated in the flows. The sample precise interaction of a single intersections is illustrated in the Fig. 3-3 (c).

OD Matrix	Home	Restaurant	Stadium	Shopping center	Park
Home	-	30%	10%	50%	10%
Restaurant	40%	-	10%	40%	10%
Stadium	30%	30%	-	10%	30%
Shopping center	40%	20%	10%	-	30%
Park	30%	30%	10%	30%	-

Origin–destination (OD) matrix



(a) OD matrix trip modeling

(b) Path modeling

(c) Flow modeling

Figure 3-3. *The multy-layer models concepts for the paths, flows and trip [12].*

- **Communication between network and traffic simulation**

One of most important concepts in the VANET application is being benefited for exchanging messages and information between vehicles to change the traffics, to have efficient traffic for avoiding the traffic congestion and jam or for safety reason with using advance safety message. In the both mentioned situation, a very robust interaction between vehicles’ movement and networks protocols should be defined.

3.2.2. Network Simulation

Study on behavior of networks under several different condition are possible using the network simulators. Researchers are able to adjust the simulation to get results with required specification. The advance network simulator are relatively cheap and fast compare to time and cost which are required for set an test bed includes several computer in network, data links and routers. Therefore the network simulator helps user to simulate several scenario which are have difficulty to implement or has a high cost in real world specially for VANET. Network simulation is very useful in order to tests novel network standards or for proposing the novel modification of the existing protocols in a very reproducible and well-ordered manner.

Network simulator is usually used for simulation the computer networks. They are used for simulating the VANETs by evaluating the performance of network protocols for mobility of nodes and other required technique. Most currently used network simulators are developed for MANETs and hence require VANET extensions (such as using the vehicular mobility generators) before they can simulate vehicular networks. Examples are OMNET++, NS-2 and NS-3, GloMoSim, J-SIM, SNS, JiST/SWANS, and GTNetS.

3.2.3. VANET Simulation

As mentioned before, VANET simulators provide traffic and network simulation or can combine traffic and network simulator. Examples are Veins, TraNS, MOVE, NCTUns, GrooveNet and MobiREAL. There are several aspect which must be considered in the simulation model for V2X systems. One of the important one is that the response of driver to the novel applications of the V2X. The traffic efficiently could be affected based on the drivers' reactions, e.g. in the sample situation when collisions warning messages are received by a driver, he could either exit the street or he could hit brake, based on the distance between the driver and accidents and also if there is any exits or not. The VANET simulator must be able to changed the vehicles' behavior according to a defined applications contexts. This kind of simulation are knows as VANET simulator or an integrated framework.

3.3. Comparison of Simulators

A wide study on the different Traffic, Network and VANET simulators which are used currently and are popular is done in this section. This section, compares features and drawbacks and finally selects the current best choice for VANET simulation.

3.3.1. Traffic Simulator

Popular mobility simulators software are presented in this section. For each of the simulators, outlines how the roadway topology can be defined, what path generation algorithms are used, what mobility patterns are implemented, and what is the input/output, interface, and language, if that information is available (data are used from [36] and own test of simulator).

CORSIM

CORSIM (Corridor Simulation), is established by the "U.S. Department of Transportation and Federal Highway Administration (FHWA)". Microscopic simulation could be simulated using CORSIM which is very powerful software. CORSIM is specially designed for highways and surface streets. The software is being used in the transportations and civil research area and it is a featured tools by the US Federal Highway Administration. CORSIM is developed from two separate programs: NETSIM (an arterial simulator) and FRESIM (an expressway and interstate freeway simulator). CORSIM combines these two simulators to provide a complete simulator.

Inputs

CORSIM is a simulator with a large amount of inputs (information). A typical simulation usually is about 2,000 lines of input out of 3000 lines of code. The physical inputs of freeways include: Ramp Meter Timings, Node Locations and Link Lengths, Free Flow Speeds, Number of Lanes, Curvature, Acceleration Lane Lengths, Lane Alignment, Ramp Meter Locations. Moreover, the physical inputs in NETSIM include: Lane Utilization, Ramp Meter Timings, Signal Timings, Free Flow Speeds, Link Distances. Traffic Volume is the capacity of the traffic. The traffic volume data can be collected from the instrumented system (using some special devices) and uninstrumented system (without using devices). The Volume data can be manually input by using TRAFED (a graphical input editor) or TextEdit. It can also be input by database. Traffic Volume Inputs for Freeway are O-D matrix (Origin-Destination matrix). Traffics demand input: the movements information and the conditional turn movement data. An example of CORSIM GUI is illustrated in the Fig. 3-4.

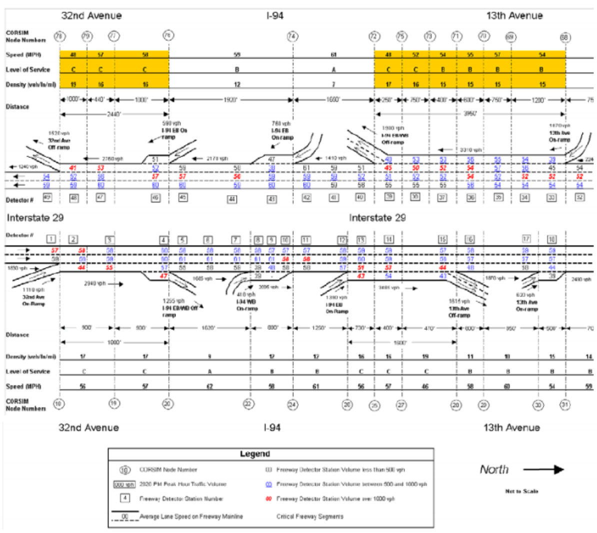


Figure 3-4. *CORSIM node evaluation output*

Interfaces, Languages and others

CORSIM supports graphic interface. The languages that are used to develop CORSIM are C/C++ and FORTRAN. FORTRAN is used because the original NETSIM and FRESIM are implemented by FORTRAN.

VisSim

VisSim [37] is a simulation software established “Planung Transport Verkehr AG in Karlsruhe, Germany,”. VisSim helps user to be able to make simulation of a large scale traffic flows and different scenarios, which are could be includes numerous different object such as cyclist, trucks, cars, pedestrian, etc. VisSim is a “microscopic multi-modal traffic flow simulation software”. It mean every entities of realistic simulations is considered in this simulator and it could be simulated.

Inputs

The desired simulation must be specified in a input file: vissim.ini. This file, specified by the graphic user interface, is the main input file and will import different sub-configuration file for different simulation purpose. There are mainly three types of sub-configuration files: network input files, test mode files, and dynamic assignment files. The network input files: *.FMA contains the OD-matrix, *.BEW current list of discovered, *.INP. These files specify the map of roads. Test mode files include three group files. *.FKT is trip chain file. *.SCH is input file for optional emission module.

Outputs

The presentation of VISSIM includes: animation, recording 3D video files. The output contains: windows output, file output, and database output. Besides, runtime errors as outputs include assertion error message and program warnings (in *.ERR files). VISSIM has twenty four output files with twenty four extension names. Each extension name represents a type of output. For example, *.KNA is for node evaluation (including number of vehicles, the average delay per vehicles). *.ANI is for animation records for playback. *.BEO is a files which is consist of cars acceleration, speed and position in every simulation steps. *.ROU records protocol of route choices for all vehicles.

Interfaces, Languages and others

Interfaces of VISSIM are 2d/3d graphic interface (see Fig. 3-5), command line and COM (component object model). Animation is provided too. The languages of VISSIM are C-like traffic control macro language and application of macro language VAP.

Figure 3-5. *VisSim example[37].*

PARAMICS

PARAMICS are a program established by “Quadstone Ltd”, which adopted the micro-simulation pattern and could be implemented just in the Windows. PARAMICS is able to make simulation of vehicles, trucks, urban zones, highways, streets. Similar to the VisiM, the PARAMICS has a 3-D tools. This software is includes numerous modules such as monitor, designer, programmer, processor, analyser, modeler, estimator.

Similar to other commercial simulators, PARAMICS has statistic output, mobile traces, error or debug outputs, and visualization and animation outputs. PARAMICS only provides GUI (Graphic User Interface), and API (for programming). C language is used in PARAMICS. Animation is also provided.

SUMO (Simulation of Urban Mobility)

SUMO or Simulation of Urban Mobility [38] is a very powerful traffic simulator (micoscorpic) which is open source as well. SUMO has been used in a wide variety of VANET projects [39]. The road network, vehicle types and vehicle routes are all highly configurable and allow for customized simulations. Furthermore, Traffic Control Interface (TraCI) allows SUMO to communicate bi-directionally with any network simulator implementing TraCI. This allows the results of traffic simulator to affect the network simulator and vice versa. By default, SUMO uses the Stefan Krau car following model to realistically model the acceleration and deceleration of each vehicle [40].

Simulation Processes

To set up a simulation for SUMO three steps should be followed.

1- The road networks which the vehicle traffic should be simulation on is needed. This goals could reach by following three methods:
   a. Generating an abstract road network using NETGEN,
   b. Setting up an own description in XML and importing it using the NETCONVERT tool,
   c. Importing an existing road network using also the NETCONVERT tool.
2- Each vehicle should find out its routes, which are a list of every edges that the vehicle must be passed and the edges should be defined. This can be achieved by:
   a. Describing explicit routes on the road network,
   b. Using predefined routes and activating only a percentage of them.
   c. Generating random routes.
   d. Importing OD-matrices,
   e. Importing existing routes. If necessary it will also be needed to compute the dynamic user assignment and to calibrate the simulation using the given measures.
3- The final step is to perform the simulation.

In more details, SUMO road networks are defined by a network file. In the network file lanes are defined as edges in a directed graph with vertices taking the form of connections between lanes. Individual lanes have attributes such as speed limits or turning restrictions. Connections between lanes can simply indicate a change in direction or can be complex multilane intersections with traffic lights or priority traffic direction. While quite complex there is a suit of included tools for generating SUMO road networks. Simple geometric road networks can be generated using the NETGEN utility. To model real life road networks map data from a variety of sources can be imported using the NETCONVERT utility. One such source is the Open Street Map (OSM) project. It provides a Google Maps like interface to viewing community generated map data. It is also possible to download the underlying map data to convert using NETCONVERT. These tools help to provide a way to generate realistic road networks. Vehicle traffic is defined by a route file. Again there is a suite of tools to generate routes. The simplest approach is random routes. The amount of traffic can be controlled by generating more or less routes in a given time period. For example, Fig. 3-6 illustrates Tampere map in SUMO.

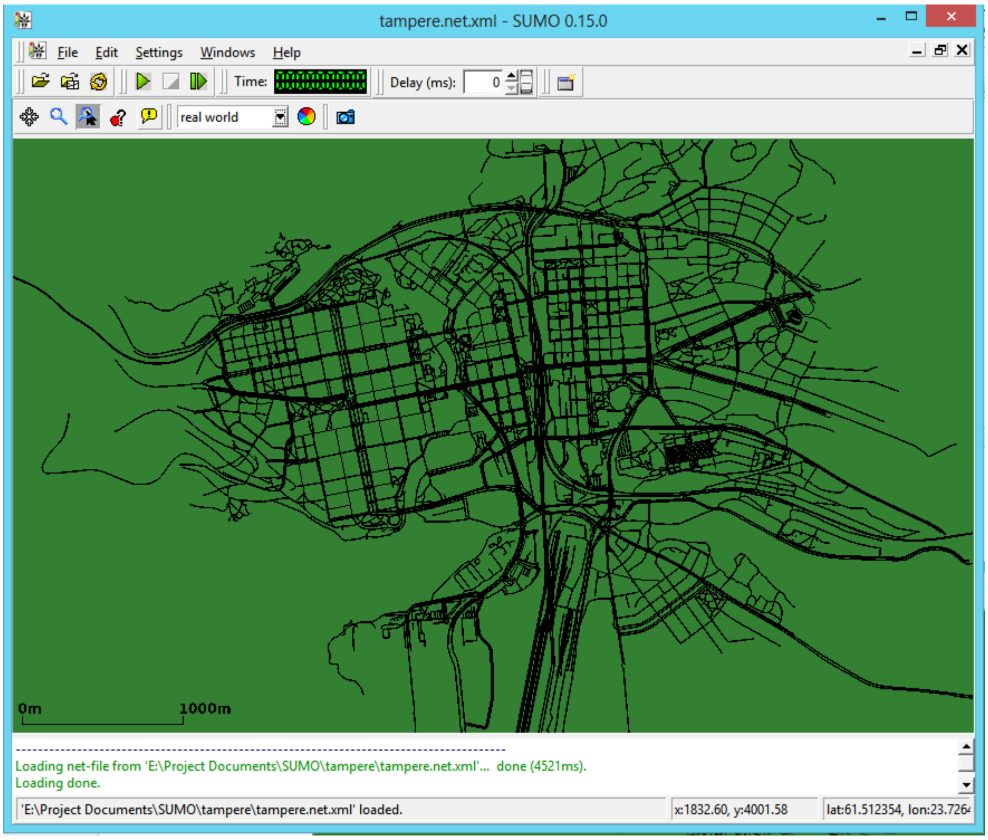


Figure 3-6. *Tampere Map in SUMO.*

3.3.2. Network Simulator

This section introduces some popular Network simulators, such as NS2, GloMoSim, OPNET, OMNET++, etc. Because the VANET is involved in the wireless communication, the explained network simulations could supports mobile node and wireless nodes. As explained before, a mobility simulator is used to provide nodes movements for the network simulator. Then, after the network simulator get the traces of the nodes movements (vehicles traveling), it could perform communication protocols and standards between the nodes and it could simulate the mobile node.

GloMoSim

"Global Mobile Information System Simulator", GlomoSim are important and popular network simulator after OMNET++ and NS-2. GlomoSim was established in USA-California, and the main goal was the simulation of wireless communication. The codes in GlomoSim are in Parsec [42] and all new extension should be coded in Parsec. Earlier GlomoSim have visualization, but currently it is include a java-based one. GlomoSim is able to run simulation is shared memory and this is helped of dividing the all networks in separated modules. Each modules in the simulation has ability to run as a distinct process. Therefore, this could significantly decrease the usage of CPU. This multi tasking features for the GlomSim, helps the simulator to be able to simulate tens of thousands node in a simulations.

OPNET

OPNET [43] is a sophisticated network simulator. The OPNET is include several implemented wireless communication technology (like satellite, Bluetooth, WiMAX, LAN, IEEE 802.11, etc.). It has a graphical and GUI package to present the results for simulated networks. In the OPNET simulation, random Drunken model is used for the mobility model. In this model directions are selected randomly from four directions in each steps. Also for the better mobility model, there are three other modeling, Path-loss-matrix is that a model which the movements are defined with a matrix, Trace, is that a model which is used the location replacement and Random Waypoint Model which is a model based on “Stop-Think-GO”.

NS-2 and NS-3

The NS-2 [44] simulator was originally created as part of the Defence Advanced Research Projects Agency (DARPA) sponsored Virtual Inter-Network Testbed (VINT) project at the University of California. It has since been extended and improved with a large community of users and developers. The core kernel is written in C++ but utilises a number of Tcl scripts for the particulars of wired and wireless networks (including some details of satellite and older technologies). The NS-2 simulation scenario scripts are written in TcL, which requires runtime compilation,that slows down the system and increases computation expense of running a complex large scale simulation. The use of TcL does simplify the creation of scenario scripts, and with deeper knowledge of the simulation system, direct C++ programs can be written.

The NS-2 system does not contain a model to simulate the IEEE 802.11p but with changes to the parameters of the simulation a single-channel operation is available, based on 802.11a, which is the basis for 802.11p. This change to parameters does not incorporate the OFDM subcarriers that 802.11p specifies, but by selecting the correct parameters a single BPSK or QAM channel can be created and set the network device to use this channel with the inter-frame spacing and bandwidth allocation, as would be seen in one of the OFDM subcarriers in a full 802.11p implementation. The lack of other subcarriers does reduce the realism of the simulation, as the signal channel is more tolerant to interference and signal attenuation.

The 802.11a MAC and physical layers in NS-2 contain full functionality, but the propagation models that have been used (two-ray ground) are only suitable for vehicular networks in a motorway situation. The use of ray-tracing and shadow propagation methods, where many routes between two nodes are calculated and the received signal strength incorporates multi-path and reflection effects, are more suitable for vehicular networks, as they more accurately represent an urban environment. The work done with NS-2 was mainly focused on motorway situations and investigating mobility patterns, so the lack of more complex propagation models did not have a negative influence on the research and results obtained [36].

OMNET++

OMNeT++ (Objective Modular Network Tested in C++), [45], is a framework and simulation library developed based on C++ that is running on different operating systems such as Linux, Mac OS X, other Unix-like systems and Windows. Primarily, OMNET++ is developed for building network simulators. OMNET++ could be use for protocols modeling, hardware architectures validating, evaluating performance aspects of complex software systems, traffic modelling of communication network, modeling of queuing networks, etc.

Component-based architectures are provided in OMNeT++ for models. These component programmed in C++ nested hierarchically and simpler components can assemble to compound components, see Fig. 3-7.

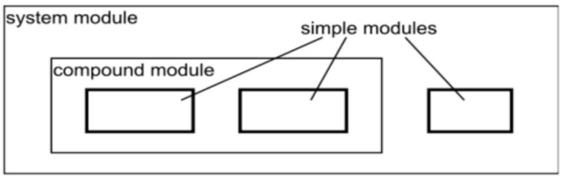


Figure 3-7. *Simple modules, compound module and system module.*

3.3.3. VANET Simulator (Integrated Simulators)

As explained in the previous section, in order to simulate a VANET application, two different simulation are required, Mobility generator and also network simulator. Up to now, these two issues in VANET simulation are decoupled. Both traffic simulation and also large scale network simulation have their own high quality simulation and their own high level modeling. However, the problems for VANET simulation is that how to integrate these two simulator. A simple solution to aim this goals is that to perform the movement model in the network simulation. This type of simulation is called one way communicating, which is just only have the network and mobility simulation separately and the network simulation cannot affect the traffic and mobility simulator. The use of these kind of simulation is for infotainment application in VANET (such as peer to peer appl, multi-media, internet connection, etc.) when the communications have no effects in the cars moving.

In the other side, VANET simulators are providing two-way communication, and mostly include two simulators (mobility and network) that could make a connection between mobility and network simulator. These type of simulation are more useful for traffic information and safety related applications those are have assumption of that feedbacks from the networks simulator should have effect on the cars mobility. For this kind of simulation, at first traffics are generated in traffic simulator and then the traffic are feeds into network simulation, and simulation are going to run. Network simulation can have effect on mobility of cars after simulation started.

This section is only considered two way communicating simulators which are include two simulator. These two sub-simulators can communicate with each other with an integrated simulator that is why they are called two-way communication simulators.

Veins

Veins [48] is one of the powerful framework for VANET simulation which is integrated network and traffic simulator together. SUMO has been selected to used as a the traffic simulator and while this time OMNeT++ has been selected to use as network simulator. Vehicles in Network Simulation or Veins is used these two simulator to generate the results. Veins make a TCP connection between SUMO and OMNET++ and it could simulate traffic and network. It works as follow, at first the traffic demand is generated by SUMO and veins send the traffic information to the OMNET++ a simulation is started. When any important event happened in the network simulator which should have effect in movement of vehicles, veins send the information to the SUMO, and the SUMO change the movement of vehicles.

3.3.4. Acceptable Simulator

According to the previous sections there are several candidates to simulate the VANETs. To select a proper group of network, traffic and VANET simulator, at the first step the network simulator should be considered.

Most popular network simulators are NS-2(3), GloMoSim and OMNET++. Due to provided technology and supported standards, NS-2 and OMNET++ are appropriate softwares to simulate the wireless networks [49]. Also according to [50] NS-3, OMNET++ and JiST are able to perform large-scale network simulations in an efficient way. There are several interfaces to couple these network simulators and traffic simulator. But all have drawbacks. For example MOVE could not create a reciprocal communication between network and traffic simulator or in TraNs (with NS-2) there is no analysis in 802.11a/b/p [51]. Also similar researches have been done in this area to select an acceptable interface, for example [52]-[59]. [52] presents the MOVE software that is based on NS-2 and SUMO. [53] works on TraNS which integrates SUMO and NS-2. VanetMobiSim simulator which is written in Java is studied in [54]. In [55] real maps are used to create mobility traces. The integration of SUMO and NS-2 is done in [56]. Recently a simulation framework with integration between NS-3 and the microscopic traffic simulator DIVERT [58] is presented in [57] and then is extended in [59]. One of The drawbacks of most of integrated simulators is that the network simulator cannot influence the cars behavior in the mobility simulator.

This study selects Veins framework which coupled the OMNET++ and SUMO due to these most important features: Online re-configuration and re-routing of cars in reaction to network simulator, Fully-detailed models of IEEE 802.11p and IEEE 1609.4 DSRC/WAVE network layers, Supporting the realistic map and realistic traffic. Veins enables running of two simulators in parallel, connected via a TCP socket. Veins framework developed based on MiXiM. OMNeT++ provides a powerful networking simulation tools but it has deficiency in the modelling of wireless communication. In [60] MiXiM, a framework for simulating wireless channels has presented which that provides detailed models of wireless channels, connectivity, mobility and MAC layer protocols for OMNeT++.

Veins connects SUMO and OMNET++. At First the cars are generated in SUMO and then exported to the network simulator. OMNET++ considers all cars as nodes and simulates the scenario. If any change occurs in the network, Veins can change the cars scenario in SUMO. Fig.3-8 depicts how Veins works in more details.

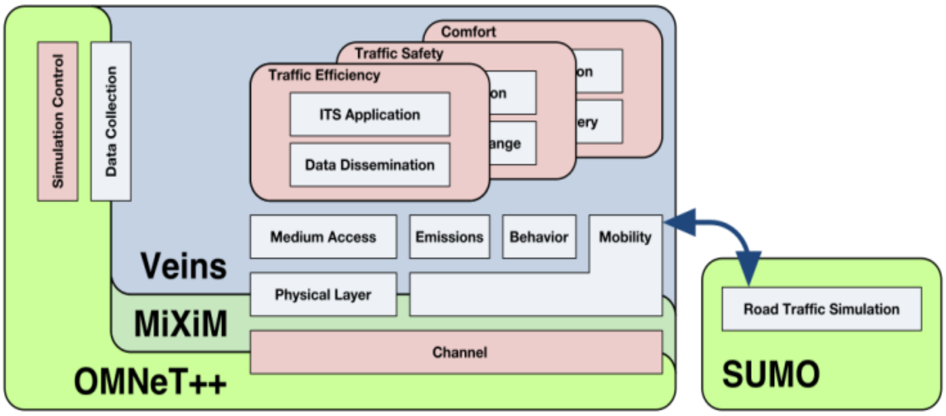


Figure 3-8. *Integration of SUMO and OMNET++ by Veins Framework[48].*

The important features of Venis framework are named as follows (copied from Veins website) [48]:

- “Based on 100% open source software offering unrestricted extensibility.
- Allows for online re-configuration and re-routing of vehicles in reaction to network packets.
- Relies on trusted vehicular mobility model and implementation done by Transportation and Traffic Science community.
- Relies on fully-detailed models of IEEE 802.11p and IEEE 1609.4 DSRC/WAVE network layers, including multi-channel operation, QoS channel access, noise and interference effects.
- Can simulate city block level simulations in real time on a single workstation.

- Can be deployed on compute clusters for simulation in MRIP distributed parallel fashion.
- Can import whole scenarios from OpenStreetMap, including buildings, speed limits, lane counts, traffic lights, access and turn restrictions.
- Can employ validated, computationally inexpensive models of shadowing effects caused by buildings as well as by vehicles.
- Supplies data sources for a wide range of metrics, including travel time and emissions.
- Supported by solid and diverse user base from five continents.”

Chapter 4

4. REALISTIC VANET SIMULATION AND PERFORMANCE ANALYSIS[2]

Beaconing in the V2V and V2I communication is played important role which is use to broadcast the data and information of vehicles and cars or RSUs to the neighboring cars or RSUs. This chapter deals with simulation of car to car communication in realistic large-scale urban area (real traffic and map) based on IEEE-802.11p standard and investigates the successful beacon delivery probability. This study calculates the Probability of Beacon Delivery (PBD) in realistic simulation and compares that with the mathematical model. PBD for cars in different situations are calculated (for example PBD for one particular car in the city, one group of cars in the street, cars in the crossroad, PBD in a large area with different traffic density, etc.). Studying the PBD of 802.11p is necessary to find out whether this technology is effective before implementing in the real world. This study is the first study performed on the PBD of 802.11p in realistic large urban area. The results show that the PBD of 802.11p in a large area with huge density of cars is quite low.

4.1. Introduction

The increasing the number of cars on city roads of most countries is putting the cities at the risk for experimenting emissions, accidents and traffic congestions which is lead to very bad and low-quality of life. In order to solve these problems researchers have worked on many solutions, one of them is the communication between cars. Car to Car Communication (C2CC) or Vehicle to Vehicle (V2V) Communication and also Vehicle to Infrastructure (V2I) Communication based on Wireless technology have been designed to help cars to “talk” to each other and also to the Road Side Units (RSU). V2V and V2I

---

[2] The result of this chapter is published in following paper [61]: Noori, H; Badihi Olyae, B; “A Novel Study on Beaconing for VANET-based Vehicle to Vehicle Communication: Probability of Beacon Delivery in Realistic Large-Scale Urban Area using 802.11p “,The 4th International Conference on Smart Communications in Network Technologies,(IEEE SaCoNet 2013) Paris, France, June 17-19, 2013.

enable several applications to make the roads convenient and safe. For safety application, V2X can be employed reduce injuries and save lives with using V2X to decrease accidents. Warning for lane departure, cars obstacle and breakdown detection, accidents and collisions are some examples of safety application. V2X can be applied to reduce the traffic density. Examples are electric toll, intelligent traffic light system, traffic congestion related application, cooperative adaptive cruise control, traffic management system, etc. beside the application for on-line traffic information, parking information, navigation system are some examples of application of V2X to help the drivers[8][62][63].

4.1.1. Beaconing

C2CC concept relies on continuous broadcast of information by all the cars, which allows each car to find out all the neighbor cars in real time. This information is broadcasted as periodic messages, which is also known as beacons. The beacons are generated and broadcasted consecutively to tell the neighbors about the car profile. Beacons can include the information about cars such as speed, coordination, the next coordination, acceleration, etc. The beaconing in the VANET must broadcasting frequently because the VANETs are dynamically changing. Thus for a VANET based system, probability of successful received beacons is an important parameter.

In addition, as mentioned earlier the IEEE 802.11p standard is used for communications in the VANET. IEEE 802.11p uses a Media Access Control (MAC) protocol that is based on a Carrier Sense Multiple Access protocol with Collision Avoidance (CSMA/CA). Beacons are sent on a fixed interval in VANETs. Most vehicular application used 10Hz for generation rate; also a beacon is about 400 bytes long (including security fields) [66].

Hence, numerous researchers have worked on calculating, measuring and modeling the beaconing concept. There are several studies which are focused on mathematical and analytical models for beaconing and several simulations have been done based on the basic concept of VANETs. Also a few studies have measured the beaconing statistics in real world. However, up to now, there is not any realistic large scale simulation about car to car communication in real city with real traffic and map. Due to the high expenses and dangers of testing new technologies for transportation in the real world, simulation plays an important role to find out the beneficial and effective technologies before implementation. Therefore, in this study the realistic large scale simulation is done. A real traffic demand and map are considered in a trusted mobility simulator (SUMO) and the real car to car communication in large city is simulated with using one of the well-known network simulators (OMNET++). This chapter focuses on beacon reception and calculates the Probability of Beacon Delivery (PBD) for the cars in different situations such as PBD for one particular car in the city, one group of cars in the street, cars in the crossroad and PBD in a large area with different traffic density. In addition, the results of PBD from simulation are compared with analytical model.

The remainder of this chapter is organized as follows: Section 4-2 presents the scenario of the realistic VANET simulation by using OMNET++, SUMO and Veins framework in large real city and in Section 4-3 the results are discussed. This chapter is concluded in Section 4-4 and the future work is proposed.

4.2. Simulation Scenario

In this section, realistic Traffic and Map are presented. Then the Network simulator configuration is defined and the simulation scenario will be presented. The results will be discussed in the next section.

4.2.1. Realistic Traffic and Map

Scenarios in SUMO simulator consist of two parts: Road Network (maps) including roads, streets, traffic light, junctions and etc. and Traffic demand, which mean details of the cars traveling like cars speed, some physical properties for each car, direction, departure and destination time and positions and etc. SUMO's road network can generated by an application named "netgen" that provided with SUMO package, or generate by importing a digital road map. The traffic demand can be define with different sources. For large-scale scenarios usually O/D matrices (origin/destination matrices) are used. O/D matrices describe the movement between traffic assignment zones in vehicle number per time.

A data set from [64] is used for road network and traffic demand simulation. The realistic traffic demand for city of Cologne, in Germany with realistic map for this city have been presented in [64]. Travel of the cars dataset is mainly based on data made by TAPAS-Cologne project. TAPASCologne, provided by the Institute of Transportation Systems at the German Aerospace Center (ITS-DLR), determines realistic car traffic in the city of Cologne. Also the street network of the city is imported from OpenStreetMap [65] (OSM) database. This data covers an area of approximately 400 km^2 around Cologne in 24 hours, is include around 700.000 vehicles traveling. These two dataset has been exported to SUMO. Fig. 4-1 shows the Cologne map in SUMO.

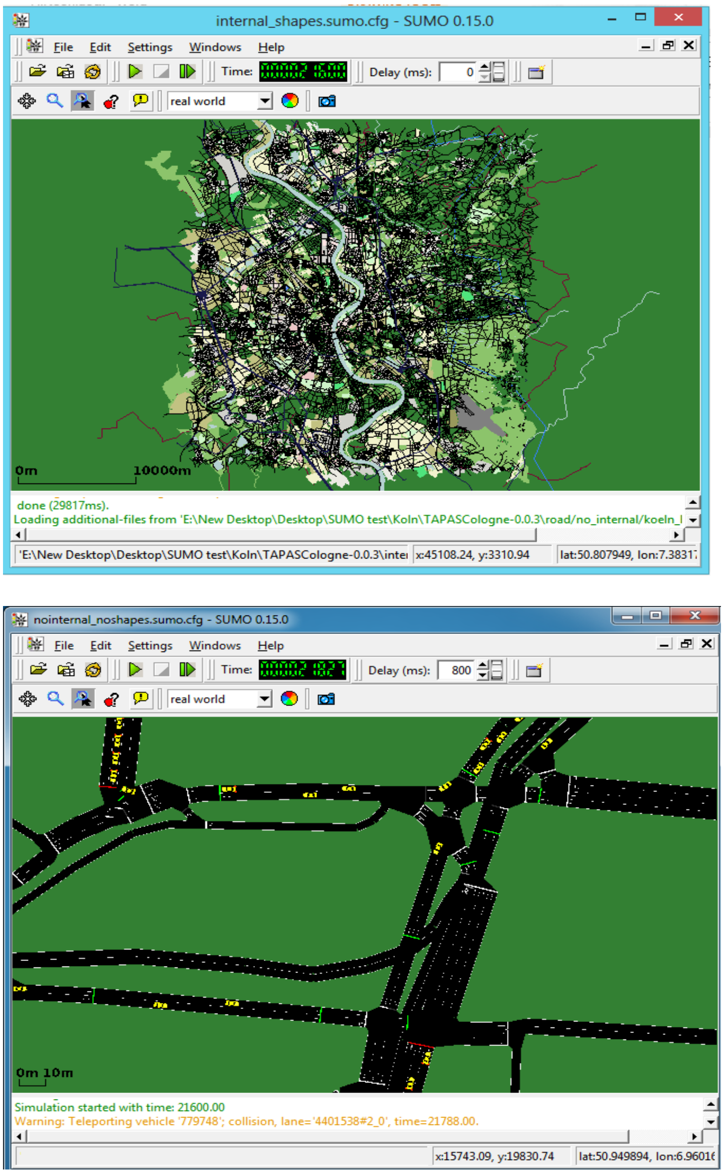


Figure 4-1. *Cologne map and traffic during the simulation in SUMO.*

4.2.2. Network Simulation

Previous section prepares the realistic traffic and map in SUMO. Next step is configuring the network simulator to run the VANET simulation. Veins framework implements the 802.11p in OMNET++ with all default parameters. As mentioned before, this study selects the Veins framework for simulation of VANET. Veins framework integrates SUMO and OMNET++. At first, traffic of the cars are generated in SUMO and then exported to the network simulator. OMNET++ considers all the cars as nodes and simulates the scenario. If any change occurs in the network, Veins can change the cars scenario in SUMO. The realistic traffic of Cologne is simulated as a VANET in OMNET++ where each car communicates by using IEEE 802.11p standard. The main parameters which are used in this study for the network simulations are summarized in Table 4-1. Simulation scenario is done using the configuration, dataset and parameters for 360 seconds (started from 7.am in realistic traffic of Cologne). One difficulty in large-scale simulation of VANET with OMNET++ is the time of simulation. For example for 360 seconds simulation, OMNET++ spends more than 30 hours in real time. Due to this problem, this chapter simulates the scenario till 360 seconds. About 3000 cars travel in Cologne map during this 360 seconds and broadcast beacons and communicate with each other.

Table 4-1. *Network Simulator Parameters*

Beaconing Rate	**10 Hz**
Beacon size	**400 Bytes**
Maximum Transmission Power	**20 mW**
Thermal Noise	**-110 dBm**
Header Length	**24bit**
Slot Duration	**16 µs**
Beacon delivery deadline	**100 ms**
EIFS	**188 µs**
AIFS	**64µs**
CWmin	**15**
CWmax	**1023**
CCH interval duration	**50 ms**
Data rate	**3 Mbit/s**

This scenario runs the simulation 12 times starting from zero. Each time 30 seconds is added to the simulation time up to 360 seconds and data is collected in each simulation. In this scenario, each car sends 10 beacons per second and all the neighbors can receive the beacons. Output of the simulation contains several statistics for each car such as sent packets, received broadcasted packets, lost packets, speed, position, etc.

4.2.3. Analytical Models

One of the goals in this study is to compare the realistic simulation results with the mathematical and analytical model. Several researches have been done on mathematical modeling of IEEE 802.11p/WAVE. Recently [67] has presented an analytical model for beaconing in VANETs. Also [68] studies the beaconing and presents the mathematical model for 802.11p and [69] has modeled the 3GPP LTE and IEEE 802.11p/WAVE. This thesis selects data from [69] for 802.11p beaconing and compares the simulation result and the mathematical one.

Another recent study which is model the beaconing in 802.11p is [69]. Based on data from [69], following formula are used in this study for calculating the PBD with analytical model:

$$X_{WAVE}(t,w,n) = P_0(w,n)X_{WAVE}(t-\sigma,w-1,n) + P_1(w,n)[1 + X_{WAVE}(t-T_s,w-1,n-1) + \sum_{k=2}^{n} P_k(w,n)\,X_{WAVE}(t-T_c,w-1,n-k)\,,$$

"where $X_{WAVE}(t,w,n)$ is the mean number of successful beacon transmissions during the Control Channel (CCH) interval of duration given that, at most,"w" contention slots are left at the vehicles' counters and "n" vehicles have not attempted to transmit yet. Other notations are as follows:"

$$P_i(w,n) = \binom{n}{i}(\frac{1}{w})^i(1-\frac{1}{w})^{n-i},$$

"$T_s = T_h + L/R + AIFS$ and $T_c = T_h + L/R + EIFS$, where L is beacon size and T_h is duration of the physical layer convergence protocol preamble and header. Finally, the target probability of beacon delivery is:"

$$P_{WAVE} = \frac{X_{WAVE}\left(T_{CCH} - T_g - \frac{L}{R}, W, N\right)}{N}.$$

4.3. Simulation Results

Simulation is done using OMNET++, SUMO and Veins. The cars travel in the city of Cologne and communicate with neighbor cars by broadcasting the beacons and the required data is collected. The main goal of this chapter is to investigate the Probability of Beacons Delivery (PBD) in real large area. Due to this, several analyses are done with collected data from OMNET++. Probability of Beacons Delivery are calculated by considering all the received broadcasted beacons (should be received in ideal form) and also lost beacons due to collision, sending while receiving, bit errors, etc.

At first, several sample cars are selected arbitrarily in different area and PBD are calculated for them and also are compared with analytical model. Fig.4-2 shows the result for two cars in different area versus number of all neighbor cars during the simulation. The result shows that the Probability of Beacon Delivery is different for each car. That can be change due to density of cars or number of received beacons in each period.

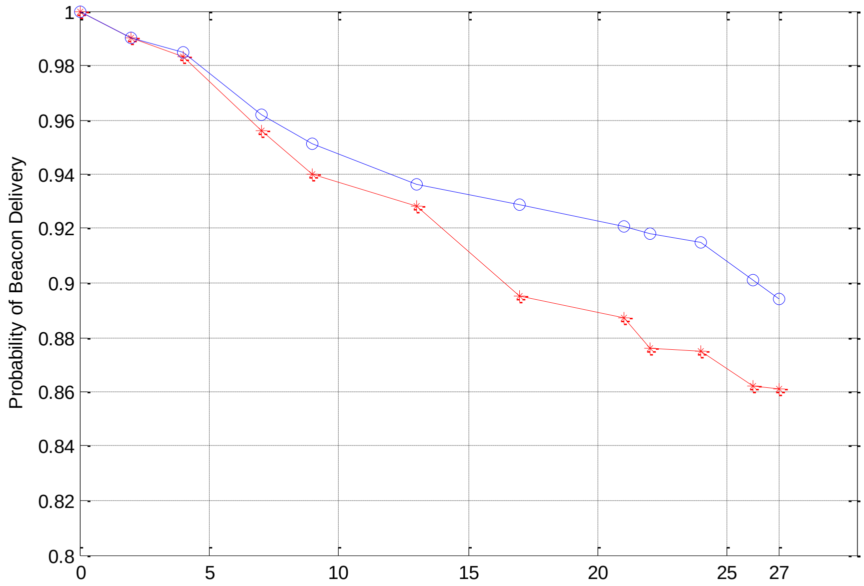


Figure 4-2. *Probability of Beacon Delivery for two different car in realistic simulation.*

Then, different streets in different area of city with dissimilar traffic density are selected (based on the data from [70], Fig.4-3) and PBD for all the cars in line are calculated (It means, all the sent beacons and all the lost beacons for all the cars are considered and PBD are calculated). In the next step, two 100 m^2 area with almost same number of cars are considered, one of them in a crossroad and another with straight streets and PBD are calculated for all the cars in this area and compared. In order to better analysis of PBD in the large scale area, the 1 Km^2 area in different traffic zones in Cologne are considered and PBD are calculated for the cars during the simulations.

It is obvious that PBD of the cars in the city are different from each other due to several parameters. To illustrate this fact, Fig.4-4 shows number of the lost packets for about 3000 cars in the simulation. This figure illustrates the fact that each particular car has a different lost packet and different PBD.

Fig. 4-5 shows PBD of one particular car which communicates with 30 other cars during 6 minutes simulation. In addition, this figure shows the analytical analyses for the same situation based on data from [69]. As it clear in Fig.4-5, the PBD in realistic simulation is better than mathematical model because mathematical model assumes that the car always has 30 neighbors and calculates the PBD based on this assumption. But in the actual situation, the car communicates with some of the neighbors and at the next step communicates with some other cars and skips the previous ones. The mentioned car communicates with 30 other cars during the 6 minutes simulation, but not at the same time. Due to this fact, the PBD in realistic simulation is better than analytical model.

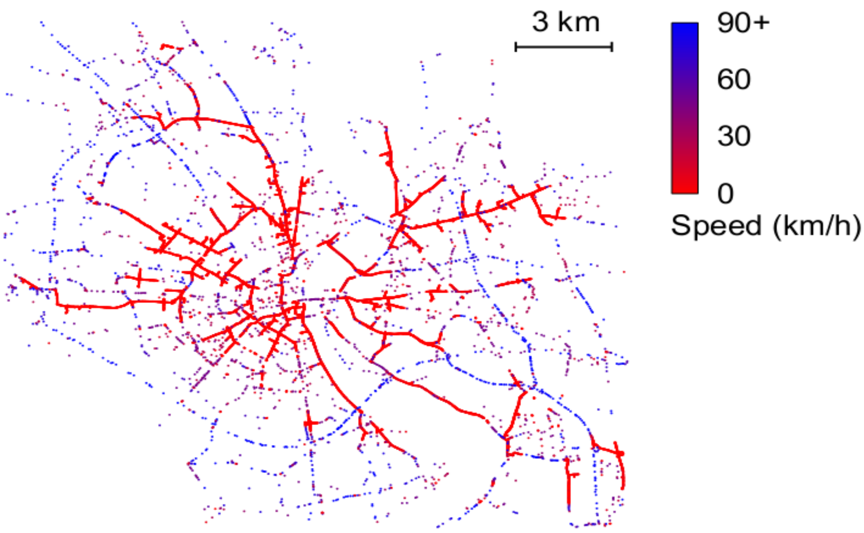


Figure 4-3. *Original-TAPAS Cologne project. The traffic situation exactly on 7:00 a.m., on the area with around 400 km2 regions in Cologne centered. Blue cars are traveling, and bright red were stopped [70].*

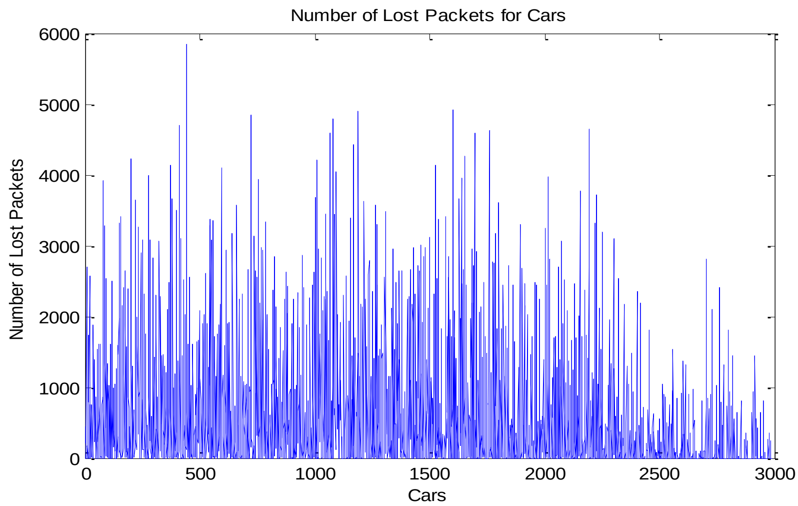


Figure 4-4. *Number of Lost Beacons for all cars in the city of Cologne during the simulation.*

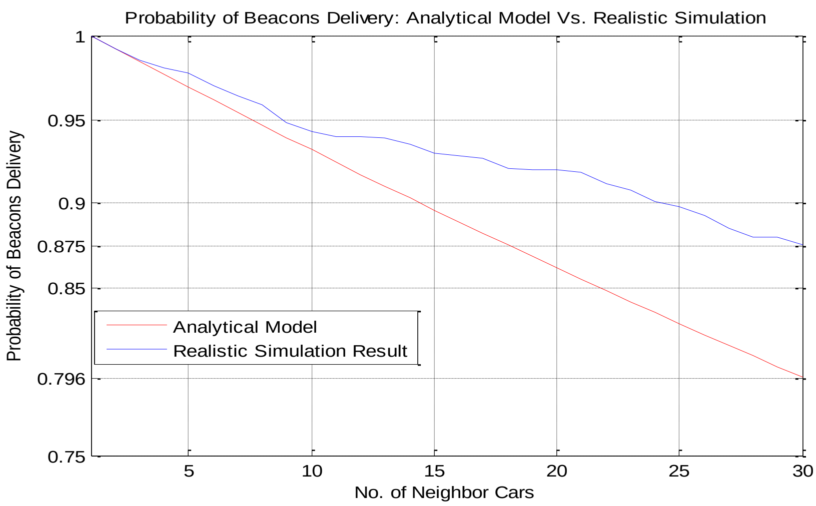


Figure 4-5. *Probability of Beacon Delivery for one car Vs. Analytical model.*

In the next step, four streets are selected in different area and the PBD for all the cars in the streets are calculated during the simulation. It should be noted that due to limitation of simulation time, this study simulates realistic VANET in 6 minutes, and during the first 5 minutes, the number of cars increases and at the end of the 300th second, reaches to the maximum No. of cars in the street, crossroad or area.

Fig.4-6 illustrates the PBD of four streets with different number of cars. With increasing the time and number of the cars in the streets, the PBDs are dropped. After the 5th minute, the numbers of cars is fixed and dropping the PBD reaches the minimum value. This means that increasing the simulation time, with fixed number of cars, does not affect the PBD. This figure illustrates the fact that with increasing the number of cars in the streets, the PBD decreases rapidly.

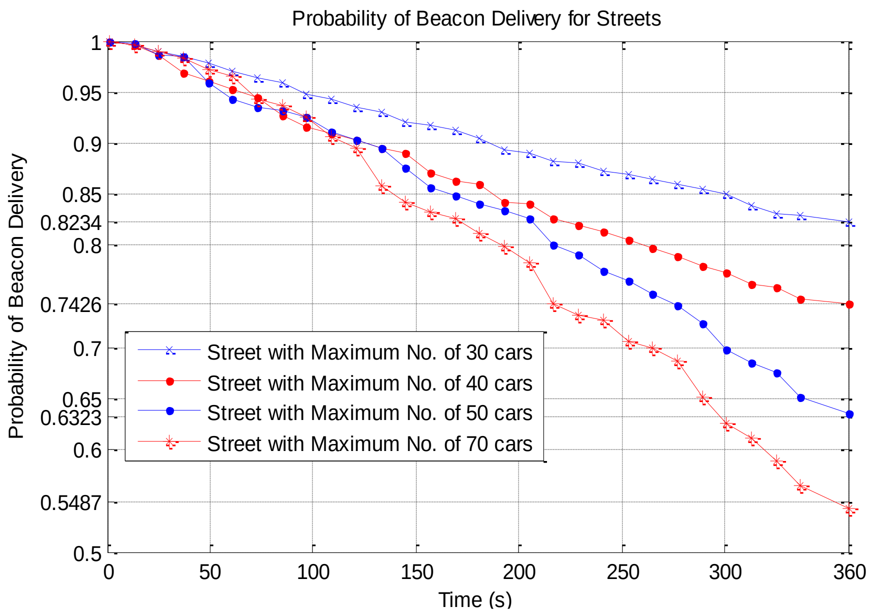


Figure 4-6. *Probability of Beacon Delivery for Different Streets with different Number of Cars.*

The next comparison is about streets and crossroads. Fig. 4-7 shows the PBD in two 100m^2 area, one of them includes the junctions and another one only considers straight streets. This figure demonstrates that the PBD in crossroads are less than the straight streets. The reason of this difference is due to distribution of the cars in the streets and crossroads. In the streets the connection between cars are less than the connection between cars in the crossroads, it means that in the crossroads, the cars communicate with more other cars and this leads to increase in the collision and decrease in the PBD. Duo to this fact the PBD in the streets is higher than the crossroads.

In order to investigate the PBD in a large area, several regions with 1 Km^2 area with different traffic densities are selected and illustrated in Fig. 4-8. As mentioned before, the number of cars increases till 300th seconds, after that the number of cars is fixed. This figure shows that, with increasing the number of cars in the area and reaching the maximum value, the PBD decreases and reaches to the minimum value. This is an interesting result for VANET which could illustrate that the PBD in VANET with the fixed number of cars (could be maximum value of the cars in the area) is higher than a minimum value, based on the density of traffic in the area. Fig. 4-8 proves that the probability of successful received beacons of 802.11p standards, in large area with high traffic density is low. Fig. 4-8 shows that the PBD in high density area, is less than 30%. But in medium and low traffic density, PBDs are acceptable

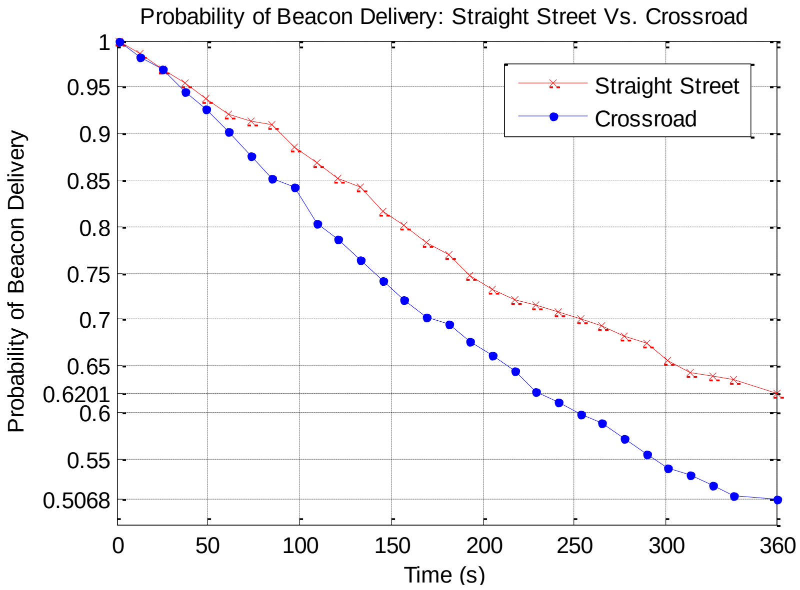


Figure 4-7. *Probability of Beacon Delivery in Straight Streets Vs. Crossroad. Two region with 100m2 area and about 50 cars.*

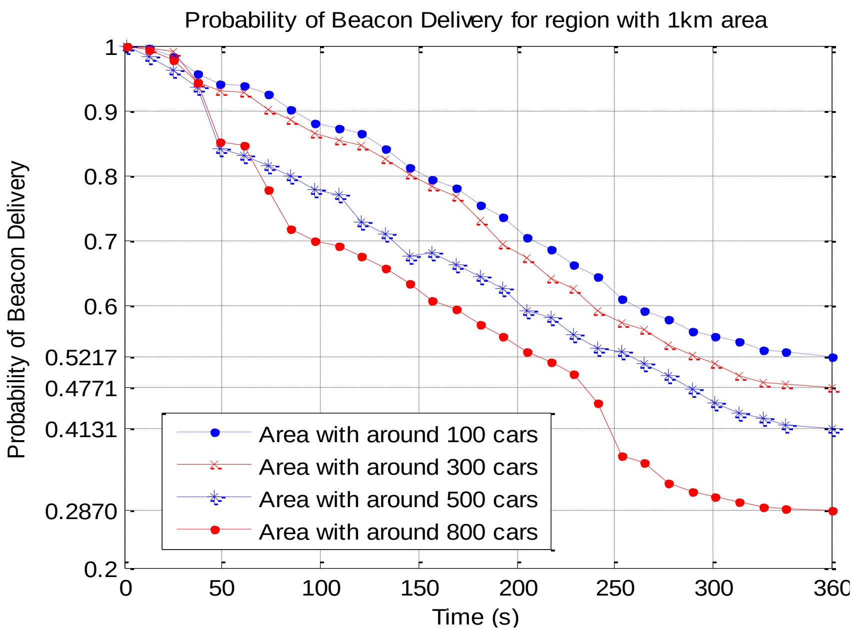


Figure 4-8. *Probability of Beacon Delivery in four different region with 1km^2 area with different No. of Cars. (Number of cars are calculated with SUMO).*

4.4. Chapter Conclusion

This chapter has investigated different VANET simulators and compared them and selects Veins framework which connects the SUMO and OMNET++. Then using the Veins, SUMO and OMNET++, a realistic large-scale simulation for VANET-based car to car communication in real urban area with real traffic is done. The cars are traveled in a real city and communicated by 802.11p standard and the results are collected. Then, a novel study is done on probability of successful beacons delivery. At first, the Probability of Beacon Delivery (PBD) for a particular car is calculated and compared with the analytical model. The result shows that, the PBDs in real simulation are better than analytical model. Then different area and situations with different traffic densities are selected (such as, streets, crossroad and region with $1km^2$ area) and PBDs are calculated for them and result are investigated. The results illustrated that using 802.11p standard in VANET, causes a low PBD in high traffic area. The future work includes simulation of the traffic of the city with other standards and technologies and comparing them with 802.11p.

Chapter 5

5. MODELING A CITY WITH VANET-ENABLED INTELLIGENT TRAFFIC LIGHTS[3]

Communication between cars and traffic lights is one of the important V2I applications which helps to have dynamic and automatic traffic lights that can create several benefits such as minimizing the traffic jam, reducing fuel consumption and emissions, etc. This chapter deals with decreasing the response time of the emergency cars by changing the traffic lights status with employing the communication technologies. The contribution of this chapter is twofold: First, the effect of the changing traffic lights status to green for emergency cars is investigated by using traffic simulator (SUMO). Second, this study uses OMNET++ (Network Simulator) in order to simulate the mentioned scenario as a VANET (with 802.11p standard) by using Veins framework to run SUMO and OMNET++ in parallel. This study has developed the Veins framework by adding a new module to OMNET++ to consider the traffic lights which are simulated in SUMO. Moreover this study has developed a new program written in Python which is connected to SUMO and controls the traffic simulation. This program uses SUMO to simulate a microscopic traffic (by considering every single vehicle movements) and also a city with intelligent traffic lights. Additionally, several statistics about traffic simulation is created for each car such as traveling time, waiting time, emissions, fuel consumption; or complete amount of car emissions in the street during the simulation, fuel consumption, number of vehicles and so on, for each street. This chapter uses Manhattan realistic map to describe the mentioned program, then uses realistic map and realistic traffic demand of Cologne, Germany, to obtain a realistic and reliable result.

---

[3] The result of this chapter is published in following paper [71]: Noori, H; “Modeling the Impact of VANET-Enabled Traffic Lights Control on the Response Time of Emergency Vehicles in Realistic Large-Scale Urban Area”; in Proceedings of IEEE ICC'13 - Workshop on Emerging Vehicular Networks: V2V/V2I and Railroad Communications, Budapest, Hungary 9-13 June 2013.

5.1. Introduction

Recently increasing the number of cars on city roads has created many problems, such as traffic congestion, the huge number of people who get killed in car accidents, fuel consumption, emissions, etc. For example, according to the National Highway Traffic Safety Administration (NHTSA) there are about 43,000 people who are killed in fatal car accidents each year in the United States [1]. One of the most important problems which is happening because of traffic congestion is the increasing emergency vehicle response time for an incident and this issue should be considered. (An Emergency vehicle is the explanation for police cars, fire trucks, ambulances, etc.). For example, to illustrate the importance of this issue, the response time for EMS (emergency medical services) are vital in decreasing disability and mortality rates.

Several research is done to show the important relationship between mortality rate and responses time [72][73]. An emergency vehicle responses time could be explained as follow: the time period between the called received to emergency car will arrived to the emergency scene[74][75]. Another example is that the mortality in the car accidents can be categorized as follows: A) Death occurs in first few seconds or minutes after each accident happened. This type include around 10% of all deaths. B) Then, another category is golden hour which is first hour after accident. This category of the death for accident have a very high mortality rate and it is about 75% of all deaths. C) Death occurs in the days or weeks after the accident [76].

Thus, it is clear that decreasing the response time for any public emergency car is one of the most important problems in new urban transportation systems that could save numerous people lives. To solve this problem the simplest approaches are building new infrastructures for roads or increase the number of emergency cars. But this approaches could not be implemented mostly because of lack of resources such as space, money and time. Thus, managing current infrastructure is the best ways. Wireless communication technology is one of the useful solutions in order to solve the traffic problems. Car to Car Communication (C2CC) or Vehicle to Vehicle (V2V) Communication and Vehicle to Infrastructure (V2I) Communication based on Wireless technology is designed to help cars to "talk" to each other and also to Road Side Units (RSU). Due to high expenses and dangers of testing new technologies for transportation in the real world, simulation plays an important role to find out the beneficial and effective technologies before implementation. In the last decade, several traffic simulators are developed so that they can simulate real traffic of cars easily, but simulation of emergency cars traveling in large real areas is difficult yet.

This study has developed a program which is connected to a microscopic traffic simulator, SUMO, and controls the cars and traffic lights. The main goal of this program is changing the traffic lights status automatically based on distance from a specific emergency vehicle. Also several statistics of traffic simulation is created. This program simulates traveling of emergency cars in a real city with real traffic in a straightforward manner. This chapter focuses on decreasing the response time of public emergency cars by using communication between an emergency car and traffic lights. This can be done by changing the status of traffic lights in the emergency car's route.

The remainder of this chapter is organized as follows: Next Section describes the program that is developed in this study by using Manhattan map. Then Section 5-3 presents the scenario of the realistic map and realistic traffic which is simulated by SUMO and the mentioned program and the result for this section are discussed. Section 5-4 explains VANET simulation by using OMNET++, SUMO and Veins framework and at the end of the section, the VANET simulation results are discussed. This chapter is concluded in Section 5-5 and the future work is proposed.

5.2. Developed Program

The mentioned program is able to simulate a city with intelligent traffic lights which means it is capable of changing traffic lights status according to the required demand. As an example changing the traffic lights status based on distance from an emergency car in Manhattan is illustrated. Fig. 5-1 shows the Manhattan map in SUMO. In addition 100,000 cars have been added to the map.

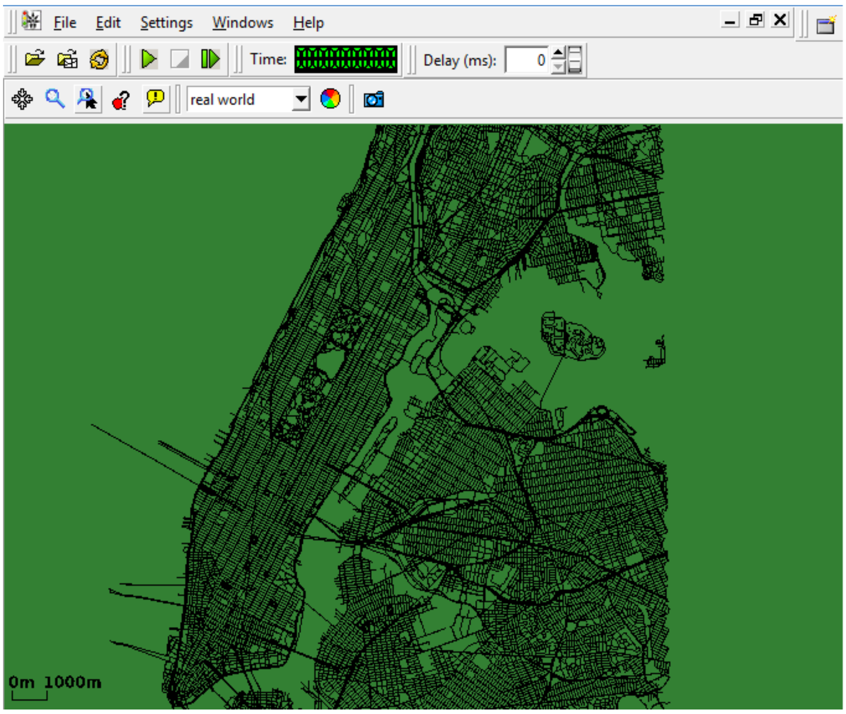


Figure 5-1. *Manhattan map in SUMO.*

The program is started by requesting the name of the emergency car and the route. Hence, the specified path from origin to destination is selected in Manhattan map. Then the emergency car which has higher priority than the other cars is added to the simulation and the simulation is started. All traffic lights in the route have been considered. At each step of the simulation, the distance between the emergency car and the traffic lights is calculated. If the distance is less than the specified value (which is defined at the beginning of the simulation), the traffic light status changes to green for the emergency vehicle and red for the other cars (Fig. 5-2). After the emergency vehicle passes the traffic light (with a distance of 20 meters) the traffic light status returns to its original state.

The program has generated several statistics which Table 5-1 shows the important ones. In addition the status of traffic lights at every instant of time could be observed. As clarified before, the main goal of this chapter is to decrease the response time of emergency vehicles by changing the status of traffic lights to decrease waiting time in junctions. In order to achieve a realistic and reliable result, real map and real traffic of cars are required. The next section describes the real map and traffic with simulation scenario to get a desirable result.

5.3. Traffic Simulation With SUMO

5.3.1. Real Map and Realistic Traffic

Same as previous chapter, this chapter are used Cologne map and traffic with data set from [64] for road network and traffic demand.

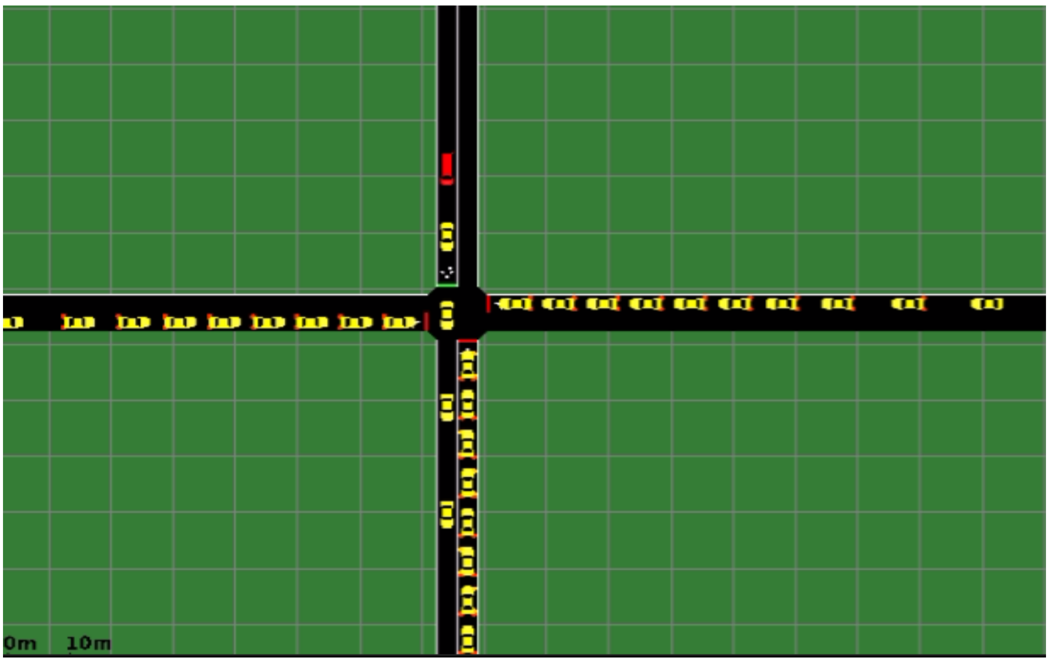


Figure 5-2. *Emergency Car in Crossroad.*

Table 5-1. *SUMO Simulation Statistics.*

Every Single Vehicle (Also Emergency Vehicle)	**Every Street**
Speed, acceleration, position at every instant.	Maximum and average speed of cars in the street
Length of the vehicle route and Traveling time.	Average traveling time on the street
Fuel Consumption	Complete amount of fuel consumption
Emitted gas (CO, CO2 ,HC, PMx, NOx)	Complete amount of gas emitted by vehicles on the lane (CO, CO2, HC, PMx, NOx)
All traveled streets and Junctions	Number of cars on the street
Emitted Noise	Emitted noise by cars on the street

5.3.2. Simulation Scenario

There is no official standard for response time of Public emergency services; but there are some definition for several emergency provider organization. For example in Montreal, QC, Canada, 90% of ambulance response must be arrived in 7 minutes [77], or based on standard rights in the urban areas in United States, 95% of called for emergency must be reach in 10 min, and for the rural areas, the time is 30 minutes [78]. In order to achieve desired response time, several components should be considered such as location of emergency vehicles station, number of vehicles in the station, population around the station, maximum coverage area of the station, etc. To investigate the response time, this study assumes that the maximum travel distance for emergency vehicles is 5 km with average speed of 60 km/h (based on data from[74]). Then simulation scenario is defined by using the Cologne map and traffic, the developed program, and SUMO as follows:

1) Initially, the map of Cologne is divided into several zones, and realistic traffic of cars between 6 am till 8 am is considered.
2) After that, 20 different zones with different traffic density are selected based on the real traffic status at 7:00 am (Fig. 4-3). Then, an emergency car is added at each zone and an identical traveling distance (5km) is defined for all the cars.
3) To find out the effect of changing traffic lights status in emergency vehicles response time based on distance between emergency car and traffic lights, five different simulations are done as follows:

a. Firstly, only 20 emergency cars travel in the Cologne map without any other car or any traffic light. Then traveling time (response time) and traveling distance for 20 emergency cars are obtained.
b. The second simulation is done with the realistic traffic of city of Cologne and also with 20 emergency cars. This means more than 250.000 individual cars moving in cologne in this two hours. Also in this simulation the traveling time for 20 emergency cars are obtained.
c. In the third simulation the status of some of traffic lights are changed as follows: the route of an emergency car is considered and if the distance between the emergency car and traffic light is less than 50 meters, the traffic light status changes to green for emergency car and red for others. After the emergency car passes the traffic lights (with the distance of 20 meters) the traffic status returns to their original state. This changing status of traffic lights is done for all 20 emergency cars and the traveling times are obtained.
d. The fourth simulation is done similar to the third one with using the 300 meters distance for changing traffic light status and the traveling time is obtained.
e. Last simulation is done like previous one with using a 5 Km distance between traffic light and an emergency car to change the status of the traffic lights.

5.3.3. Traffic Simulation Results

Simulations are done with the explained scenario. All data are collected and ordered based on traveling time for emergency cars (Response Time) from low to high in Fig. 5-3. It is obvious that changing the traffic lights status to green for emergency cars decreases the response time. In order to obtain a reasonable distance for changing traffic light status another factor is required. This study, selects cars delay time which occurs because of changing the traffic light status. It means that the route of each emergency car is considered and delay time for other cars that have stopped at traffic light due to the mentioned changing traffic light status is considered and maximum delay is obtained. Fig. 5-4 illustrates the maximum delay time for 20 cars that stopped in each emergency cars route. As it can be seen in Fig. 5-3 changing the traffic light status for each route has a different effect. For example, for changing distance equal to 50 meters, in low traffic area (car’s number 1-7) the average decrease of the response time is 36.39%, for medium traffic area (car’s No. 7-14) is 16.66% and for high traffic area (car’s No. 14-20) is 6.93%.

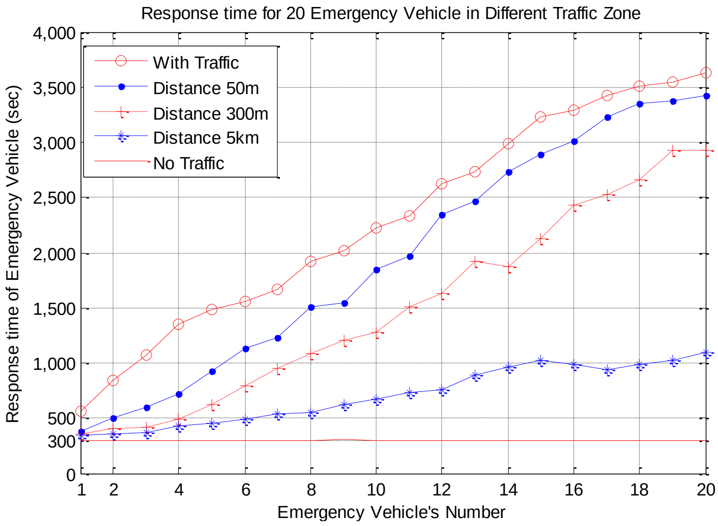


Figure 5-3. *Response Time for Emergency Cars with changing Traffic Lights Status based on distance between cars and traffic lights.*

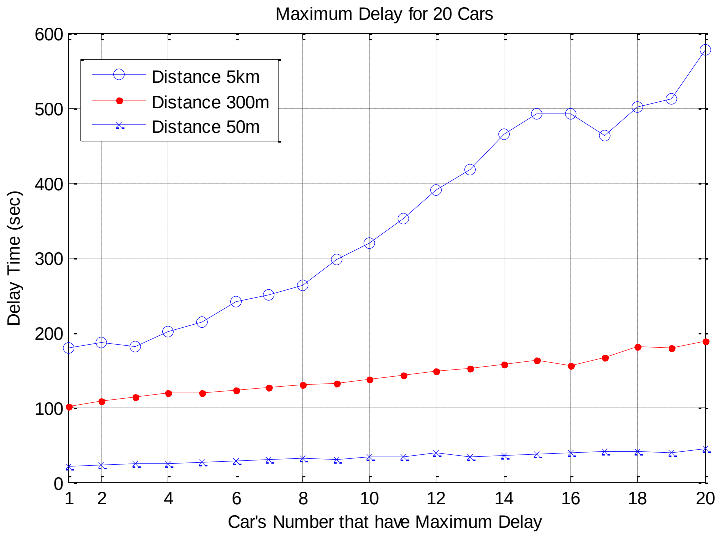


Figure 5-4. *Maximum Delay for 20 cars that have stopped in traffic lights because of Emergency Vehicles.*

5.4. VANET Simulation

5.4.1. VANET

The basic concept of the VANET (Vehicular Ad-Hoc Network) is straightforward and simple: create a widespread and cheap wireless technology to connect vehicles to each other and road side units (RSU) for sending and receiving the information. The important features of VANETs includes these items: Cars and RSU are considered as nodes in VANET, The nodes can move very fast, and the network is highly dynamic which means that topology of the network is changing continuously with changing the position of the nodes and density. Simulation of the VANET protocols, such as wireless communication, multi hop routing, etc, is implemented on the networks simulations such as OMNET++ or NS-3. However, network simulators cannot simulate the mobility of cars sufficiently.

Duo to this fact, to have a realistic and authentic result, the mobility of cars should be simulated by a trusted movement simulation such as SUMO. In order to run the network and traffic simulator in parallel, another software is required. There are several interfaces to couple the network simulators and traffic simulator. As mentioned before, this study selects Veins framework which has coupled OMNET++ and SUMO due to these most important features: Online re-configuration and re-routing of cars in reaction to network simulator, Fully-detailed models of IEEE 802.11p and IEEE 1609.4 DSRC/WAVE network layers, Supporting the realistic map and realistic traffic. Veins enables running of two simulators in parallel, connected via a TCP socket. Veins framework is developed based on MiXiM. OMNeT++ provides a powerful networking simulation tool but it has deficiency in modelling of wireless communication.

5.4.2. Simulation Scenario for VANET

As mentioned before, this study selects the Veins framework for simulation of VANET. Veins framework integrates SUMO and OMNET++. At first traffic of the cars are generated in SUMO and then exported to the network simulator. OMNET++ considers all cars as nodes and simulates the scenario. If any change occurs in the network, Veins can change the cars scenario in SUMO. Veins has a drawback which is that only cars are considered in OMNET++. This chapter has developed Veins framework by adding a new module in OMNET++ which considers the traffic lights as nodes as well.

By considering this new module and Veins, realistic traffic of Cologne (and also the mentioned emergency cars) are simulated as a VANET in OMNET++. Every node can communicate by using IEEE 802.11p standard. The main parameters which have been used in this study for the network simulation are summarized in Table 4-1.

Simulation is done as follows: All the cars and also every traffic light broadcast beacons periodically, if a traffic light receives a beacon from an emergency car, the traffic light status changes to green for the emergency vehicle and red for the other cars, After the emergency car passes the traffic lights (with the distance of 20 meters) the traffic status returns to their original state. The simulation is done for all 20 emergency cars and response times are obtained for them. Using the parameters of Table 4-1, yields a transmission range about 300 meters.

5.4.3. Simulation Result

The simulation has been done and the response times for all the emergency cars have been obtained. Fig. 5-5 shows the response times for 20 emergency cars which are obtained by using car to infrastructure communication to change the traffic lights status with OMNET++. Also Fig. 5-5 shows the data which were obtained in previous sections with using SUMO.

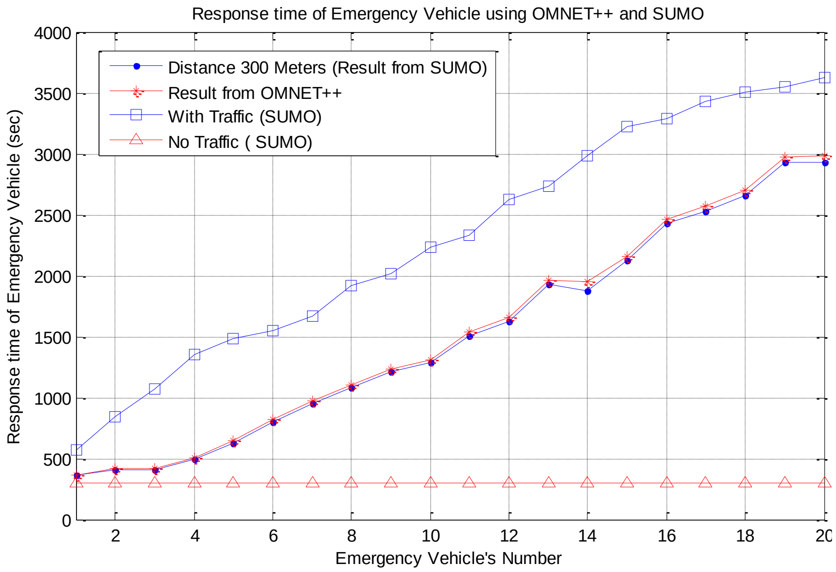


***Figure 5-5**. Response Time for Emergency Cars with changing Traffic Lights Status based on distance between cars and traffic lights in SUMO and using V2I communication to changing traffic lights status in OMNET++.*

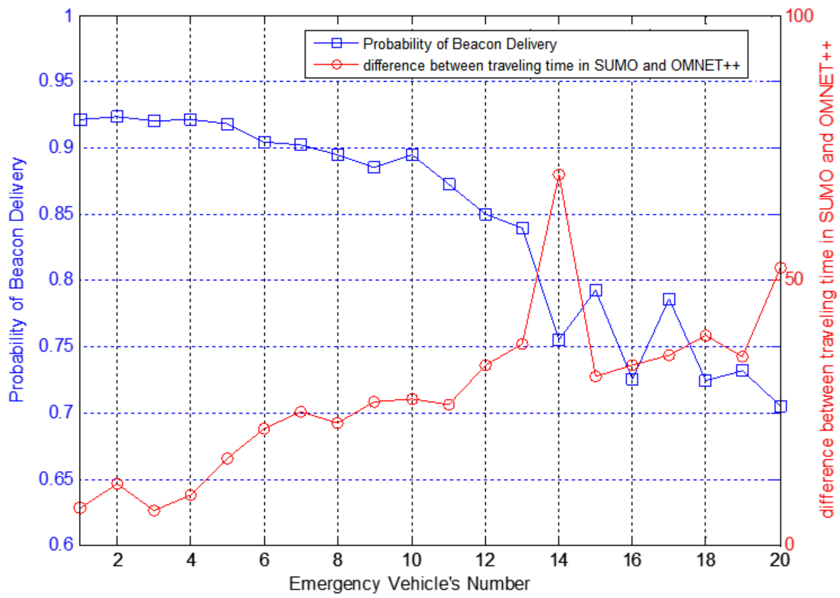


***Figure 5-6**. Probability of Beacon Delivery and the difference between response time in SUMO simulator (300 meters scenario) and OMNET++ simulator.*

Moreover, for analyzing the accuracy of VANET simulation, Probability of Beacon Delivery for 20 emergency cars is calculated by considering all the received and lost beacons during the travel from origin to destination. Also for each emergency car the difference between response time in SUMO simulator (300 meters scenario) and OMNET++ simulator are considered. Fig.5-6 shows probability of beacon delivery for 20 emergency cars and also corresponding time differences.

Fig. 5-5 and Fig. 5-6, show this fact that the difference between VANET simulation result (realistic) and traffic simulation result (ideal) is quite small and almost the same. Realistic result is obtained from simulation in OMNET++ by using the beaconing and 802.11p standard and the traffic light status is changed based on the received beacons from emergency cars; and ideal form is obtained from SUMO by using the developed program which considers the exact distances between emergency cars and traffic lights for changing the traffic lights status. One of the reasons for this accurate result is related to beaconing scenario with numerous sending messages from each car which helps to deliver the message as soon as possible. This fact illustrates that car to car communication and car to infrastructure communication based on IEEE 802.11p is able to decrease the response time of the emergency cars. In addition Fig. 5-6 shows that by increasing the traffic density (emergency car's number) the probability of beacon delivery has decreased and also the emergency cars response time has increased.

5.5. Chapter Conclusion and Future work

In this chapter, a new program has been developed which can be connected to SUMO and simulates a city with intelligent traffic lights that helps to decrease response time for emergency vehicles. This program can be used for any city and for any route for emergency vehicle to investigate effect of changing traffic lights status. Then, in order to obtain a realistic and reliable result, real map and realistic traffic of city of Cologne is selected and the developed program is simulated Cologne traffic with intelligent traffic lights. Response time for emergency cars in city of Cologne is investigated. Then this scenario is done by using network simulator OMNET++ and Veins framework. Probability of beacon delivery in real large area is obtained for emergency cars. VANET simulation and traffic simulation results have been compared. The results show that car to car communication can be implemented in real word by using 802.11p which helps to make roads safer and more comfortable.

Future work includes simulating traffic of the city with intelligent traffic lights with network simulators by using another wireless technologies such as WiMAX.

Chapter 6

6. REDUCING VEHICLES TRAVELING TIME USING V2I

One of the promising applications of ITS is calculating the estimated traveling time dynamically and showing drivers the fastest vehicular route to the destination, which has several benefits such as decreasing traffic congestion, fuel consumption and emissions, etc. This chapter proposes a new method to find the fastest route from origin to destination by using the V2I communication which provides real-time traffic information to drivers. This method, assigns a Current Traveling Time (CTT) for each street in a city which could help drivers to find the best route. The contribution of this chapter is threefold: First, the mentioned methods are proposed. Second, impact of the method is investigated by using traffic simulator (SUMO) and also dynamic route planning with employing CTT for each street which are calculated with the proposed method. Third, OMNET++ is used in order to simulate the mentioned scenario as a VANET (with IEEE 802.11p standard) by using Veins framework to run SUMO and OMNET++ in parallel. This study has developed the Veins framework by adding new modules to OMNET++ which aims to add several RoadSideUnits (RSU) in the realistic traffic simulation. Moreover this study has developed a new program written in Python which is connected to SUMO and controls the traffic simulation. This program uses SUMO to simulate a microscopic traffic (by considering every single vehicle movements) and also with calculating the Current Traveling Time for streets and dynamic route planning for the cars. This program dynamically calculates the fastest route for the specific car.

6.1. Introduction

Increasing number of accidents and rising greenhouse emissions stemming from high traffic volumes, combined with traffic congestions, decrease the living quality in urban city areas in most countries. A very huge amount of resources such as time and fuel are wasted because of traffic congestion. For example, in the U.S. each driver spent an additional 38 hour in vehicle and wasted 19 gal of fuel because of traffic congestion in 2011 which leads to 818 $ congestion cost per auto commuter. This means, wasting 2.9 billion gallons and 121 billion dollars for congestion related costs just in U.S. in one year [79]. A comparison with 1982 (16 h, 8 gal) illustrates that the increasing congestions and traffic volume in cities, cause an increasing problem in urban life and transportation systems.

In order to relax traffic related problems in future, several approaches are proposed, but it is clear that extending the road network is not suitable to solve the traffic related problems in future due to the limitation of resource, places, etc. Thus, novel scenarios are required. For example, the Federal Highway Administration in the U.S. (FHWA) proposed and defined three general tactics to decrease traffic related problems [80]:

- Work on current capacity of roads and extend them, e.g. increasing the size and number of streets and highways.
- Extension of alternative transportation that require less resources, e.g. non-automotive transportation.
- More efficient using of current capacities of cities and roads.

To solve traffic related problems, one of the useful solutions is wireless communication technology which focuses on the third strategy. Vehicle to Vehicle (V2V) or Car to Car Communication (C2CC) and Vehicle to Infrastructure (I2V or V2I) communication based on wireless technology is developed in order to makes vehicles capable to "converse" to Road Side Units (RSU) and each other. V2V and V2I have several applications in Intelligent Transportation Systems (ITSs) to make transportation more efficient. This paper focuses on the real-time monitoring of the traffic of the streets and calculates the best route for the drivers using the on-line traffic status of the roads. This efficient method for selection of the routes decreases traveling time for each journey which can decrease the traffic congestion and save the resources. Calculating the best route, requires two types of data, first the current traffic information of the cities and roads are required, then, by using this information, calculation of the routes becomes possible.

Several researches have been done in recent years to calculate the best route for drivers with using different methods and algorithms; also VANET-based traffic status monitor, has already been investigated earlier, and several methods have been proposed (see next Section), but in our knowledge, up to now, there is not any realistic large scale simulation in real urban area using the V2I communication in order to monitor the traffic and also calculating the fastest route for drivers.

The remainder of this chapter is organized as follows: Section 6-2 discusses the related work. Then Section 6-3 propose a new method to calculate the current traveling time in a street by using V2I communication. Efficacy of the proposed method is investigated by using traffic simulator (SUMO) in Section 6-4 and results are discussed. Section 6-5, uses OMNET++ in order to simulate the mentioned scenario as a VANET (with IEEE 802.11p standard) by using Veins framework to run SUMO and OMNET++ in parallel and simulating a realistic large scale V2I communication. This chapter is concluded in Section 6-6 and the future work is proposed.

6.2. Related Work

The main focusing area in this paper can be categorized in two parts: the first, using wireless communication in new traffic schemes in future world and the second, computing fastest paths for drivers from their origins to destinations. Thus, two different types of related works are studied. Study on ITS and use of new wireless technology in ITS are currently hotspots in mobile communication research fields and numerous research works have been done in this field, so in order to find new admissible approaches and applications and avoid repetitive research, study on related work is vital in this field. Thus, this paper has a wide survey on related works and technologies.

Recently, cameras, loop detectors and numerous sensors are implemented in most major cities. Centralized system structures are used in conventional transportation and transportation information systems which are defined as follows: information related to recent traffic volume has been transmitted to the Center of Traffic Information (TIC) by using sensors, loop detectors, etc. which are installed in or above the streets. Analysis about the current traffic situation is performed in the TIC, then the result information is transmitted via several possible approaches such as on-demand (via cellular systems, 3G, LTE or UMTS) or via Radio Data System (Traffic Messaging Channel) to the vehicles [83]. High delay and low details of traffic information are drawbacks of this method. Projects such as Mobile Millennium[84], Nericell[85], JamBayes [86], CarTel [87]and surface street estimation [88]are based on data collection from cars with using on-board GPS devices to monitor and analyse traffic situation.

On the other hand, in near future wireless communication technology will be soon sufficiently developed to enable V2V and V2I communication to collect real-time traffic related information. Thus, current and future approaches are likely to use V2V and V2I with conventional approaches in parallel to monitor the traffic and using this data in several applications such as finding the best routes for drivers, safety applications, etc. For example, in the IPERMOB project [89], IEEE 802.11p as a vehicular network is used in parallel with wireless sensor network and connected via IEEE 802.11h with 5 GHz wideband link to a centralized database. In Japan, VICS center[90], has provided real-time road traffic information about congestions. In this project, cars are connected to the RSUs and share streets and traffic information with them, and the RSUs analyse this information and broadcast them to surrounding cars. Some prior researches have been done (such as [91]) to monitor the roads traffic with investigating the number of cars on the streets and these numbers are used as measurements to identify congested streets. Calculating number of cars in the streets is not a simple task and in some case other tools (such as image processing) are compulsory. In some researches such as [92],[93] traveling times for cars in streets are used (which are measured by previous cars).

Up to now, projects and researches related to traffic monitoring have been shortly reviewed. In this part, related researches in finding the route for drivers will be explained. [83], has proposed a VANET-based Dynamic Route Planning (DRP) Guidance. Recently, Hara et al[94], have used two approaches in order to find the best route and DRP using real-time traffic data, one of them is extending the range of broadcasting to make traffic data available to farther cars and the other approach is duplicating the road segment data broadcasted with cars to increase the chance for others cars to get the information. More recently, [95] has presented five cost-effective and easy deployable traffic re-routing strategies for reducing traveling time for vehicular traffic guidance system. In [96], a DRP is presented based on future traveling time estimation and the route is periodically calculated from an origin to destination for cars based on online traffic information. [97], presented an approach based on DRP for cars with using latest recorded traveling time. Genetic algorithms are used for DRP in, and authors proposed an approach with estimating the future traveling time based on current collected traveling time. Sommer et al [98], used V2I communication to propose a new traffic information system for optimizing the route planning which provides information for cars about traffic congestions.[99], selected a limited urban network and investigated the impact of DRP on the traveling times. Several different levels of penetration rate are used for simulation. Simulations provided traveling time information to vehicles and the results show decreasing traffic congestion using this information. However, in this case the impact is worse when all the drivers use DRP than static route planning. Since the transportation network which is used in is too small, there is no conclusion for optimum strategy.

Recently, some systems are reporting capabilities to forecast traffic congestion and also its duration (such as Microsoft Bing and Google Map) by traffic pattern using statistical predictive analysis. In addition, infrastructure based traffic related information are being used by other companies (such as TomTom or Google) in order to calculate the shortest path. However, these solutions not try to prevent congestions clearly (i.e., reactive solutions) and also another problem is that they provide same guidance for all cars on the streets at a certain moment [95].

One of the most important issues in intelligent transportation systems is finding the real-time traffic information of the roads. Also one of the most important applications of the ITS is providing the best routes (dynamic or static) for drivers. Presented researches and projects, have several drawbacks behind their features. Most of them, have not used real-time traffic data, and some of them have a huge delay in order to provide the data. This paper, presents a novel approach to find the real-time traffic information of the roads using V2I communication technology and using IEEE 802.11p standards. Then, employing such online information for computing the dynamic route planning and decreasing the traveling time for vehicles are investigated. The important features of this study are presenting a novel method for real-time traffic monitoring, realistic communication simulation, using a realistic large scale urban map and also using realistic traffic data which make the results realistic and reliable.

6.3. Proposed Method

As explained earlier, recently, many groups have been in the process of defining standards for vehicular applications and they have been worked together for developing VANET-based communication systems [101]. In addition, V2I and V2V communication concepts rely upon broadcasting the information continuously by all RSUs and vehicles, which allow each vehicle to discover all the neighboring RSUs and vehicles in real time and also other information which are provided by RSUs. Information are broadcasted as periodic messages, known as beacons. In order to inform neighbors about the vehicle profile, beacons are broadcasted consecutively. Beacons contain information about vehicle profile such as position, acceleration, speed, next coordination, etc. VANETs structure are dynamically changing, thus beacon must to be broadcasting frequent. ITS, has defined numerous applications based on V2X and up to now, several different approaches are proposed in order to use V2X in real world. Almost all of them are similar in broadcasting beacons by vehicles, but there are several differences in location and application of RSUs. Communication approaches are also different.

This paper proposes a method using communication between RSU and vehicle via IEEE 802.11p to monitor the traffic information of the roads and also the traveling time in each street by employing the V2X communication. The proposed methods can be explained as follows:

- All road segments (streets, highways, etc.) have two main variables, Ideal Traveling Time (ITT) and Current Traveling Time (CTT). ITT is the required traveling time for a car to go through the street from the beginning point to the end point of road segments in ideal form, (when the road is empty and with the maximum allowed speed); and CTT is the same required time for a car to pass the road in real-time (with considering the other cars traffic, congestion, road construction etc.). Calculating the ITT is simple, by using the road segments lengths and maximum allowed speed on the road. This paper proposes using V2I in order to calculate CTT as follows:
- RSUs should be added in start and end of each street or each road segment (RSUs can be traffic lights or current RSU or new one). When a car comes to the street, at the start point of the street, corresponding RSU sends a message to the car. The message includes the current time and date (starting time for the car in this street, ST). The car holds this message and broadcasts it periodically till the car arrives to the end point of the road. In the end point, car sends the starting time (ST) to the corresponding RSU at the end point of street, when the car wants to leave the street. The RSU at the end point, calculates the traveling time for the mentioned car by subtracting the real time and the starting time for the mentioned car (ST).

- RSU at the end point of the street, assigns a CTT equal to the average traveling time for 5 recent cars in the street. The RSUs are broadcasting this CTT for the street and it could be available for all cars.

This simple approach, can be implemented in the future real world and calculate the real-time traffic data for all the streets in cities. Several aspects should be considered in this approach. This approach suggests expanding content of each beacon by two additional variables, ST and CTT and cars and RSUs broadcast the ST and CTT in addition to the other information. In addition, there are several methods in order to find real starting and ending times for the cars in the street, such as short range communication (for example using RFID tag), or calculating the distance from cars and RSU based on the received power etc. and many other methods.

Furthermore, to avoid broadcasting the wrong data, if the RSU does not receive data from cars or if time of the last received data is later than the ITT of the street, the RSU assumes that there is not any car on the road and assigns the CTT equal to ITT for that street. Also about the cars, the ST should be changed when a new ST message is received or the ST should be equal to zero when the cars start to travel. RSUs broadcast periodically the CTT of the street and also they are connected to each other and central station which can do analysis of the real traffic data and provide data to others. When a car decides to start a travel, it makes a connection to the nearest RSUs by sending and receiving beacons, and request the best route from origin to destination, based on current CTT for streets; (CTTs are calculated by RSUs and provided to traffic information center (TIC). RSUs can connect to the TIC, and collect the data from other RSUs including all streets' CTTs.). This approach, can be implemented in near future because nowadays most of the required technologies are available and the future ITS must be able to find the real-time and in detailed data and information of traffics in the cities. Also, RSUs can count the exact number of cars on each street, when all the cars broadcast the beacons.

In summary, the proposed method, assigns a CTT for each street which is used to provide real-time data of traffic on all streets. These data are used to find the best route for cars and manage the traffic. These CTT's are broadcasted periodically via RSUs and can be provided by other technologies such as cellular or via internet. CTTs can also be broadcast by cars by cars if it is necessary. For example if a traffic congestion happens on the specific street, then all cars can broadcast this CTT to inform others that such traffic congestion has happened by using multi-hop beaconing.

6.4. Traffic Simulation

In order to do detailed analysis and investigation on the proposed method, two different simulation scenarios are reported in this paper. First, SUMO (traffic simulator) is employed to simulate the dynamic route planning by using CTT and effect of dynamic route planning is investigated in the ideal form without any wireless communication technique. Second, OMNET++ (Network Simulator) is used in order to simulate mentioned scenario in realistic wireless communication scheme.

6.4.1. Dynamic Route Planning

As mentioned before related work, there are several approaches in order to find the best route and calculate the dynamic route for cars. Searching for a route is in essence a classic shortest path problem. Often classic algorithms are used to solve this problem, like Dijkstra or A* [102]. These are commonly deployed. GNM consist of several links and vertices. Road segment (or streets) shown by links and intersections are defined by vertices. In GNM, each link has one direction, meaning that for example if J and K are intersections which are connected to each other via a street, there are two different links to define the mentioned street, link JK which defines traffic flows from intersection J to intersection K direction and another one, link KJ which defines traffic flows from intersection K to intersection J direction. Also, each link has a weight (cost) which can be the traveling time, fuel consumption, etc.

This study, uses and improves the A* search algorithm which is implemented in [81] for calculating and searching the best route. This A* search algorithms are combinations of uniform cost searches and Greedy and it is an optimal and complete search method [103]. A* algorithm considers sum of two functions, g(n), the cost (value) to reach from starting node to current node (node n), and h(n), cost (value) to reach from node n to goal node. Therefore, f(n), the costs function for node n is estimated least-cost path from a given initial node to the goal node through n. The process of A* algorithm can be explained as follows: The algorithm defines two different sets, Open and Close set. Starting with the initial node, the algorithm maintains openset as priority queue of nodes which should be traversed through.

The higher priority belongs to lower f(x) for node x. At each step, the node which has lowest f(x) is removed from the openset (priority queue) and is added to closeset, then, neighbors of node x are added to the openset and the h(n), g(n) and f(n) values of these neighbors are updated accordingly. The final step of algorithm is reached when algorithm finds the goal node in the queue with lowest f in openset (or till openset gets empty). (It should also be noticed that, algorithm may pass the goal node several times if algorithm find that other nodes have lower f). The length of the shortest path is equal to f value of the goal node and the shortest route can be found by revising the traversed path

In [81], several cost functions such as traveling time, Class of road segment, Expiration time, Delay in intersections, Fuel Consumption and pollution are considered and assigned to the roads. This paper, improves the traveling time cost function using real traveling time (CTT). The following pseudocode summarizes the overall algorithm:

A* Algorithm Pseudo Code:

Define origin and destination intersection with position
Define *closeset*. (Set of street intersections already passed.)
Define *openset*. (Set of intersections in the queue to be passed, initially containing the start intersection)
Define *g* Value. (Traveling Time to reach from origin intersection to current intersection)
Define *h* Value. (Lowest estimated traveling time from current intersection to the destination)
Define *f* Value. (g+h)
While *openset* Not Empty
Add neighbor intersections of current one to *openset*
Calculate *f* Value for all neighbors
Sort intersections in *openset* by *f* value in ascending order
Get first array from *openset* call intersection X
Compare X and goal, if true exit loop
Else
Remove current intersection from *openset* and add to *closeset*
Expand all reachable intersections from X call Y
If Y is not included in *closeset*, call "Next X"
f(Next X) = g(Next X) + h(Next X)
Loop

6.4.2. Simulation in SUMO

In order to obtain realistic and reliable results, this paper uses the realistic traffic and map. For the traffic demand and network of roads and streets, a data set from [64] has been used. (Any traffic data for the cities can be used in this simulation by employing the proposed method and future work could focus on the effect of this method in different traffic pattern in major cities and countries such as US, Japan, etc. which are having different regulations and policies affecting the traffic). The realistic map of the city of Cologne, Germany, with real traffic demand have been presented in[64]. TAPAS-Cologne project has provided information related to travel of the vehicles and traffic behaviour in this dataset. German Aerospace Center, Institute of Transportation Systems (ITS-DLR), provide TAPAS-Cologne project which determines realistic vehicle traffic and traffic related information in the city of Cologne. In addition, the street network and realistic map of the Cologne is imported from OpenStreetMap(OSM).

To controls the traffic simulation, this study has developed a new program written in Python capable to connect to SUMO. This program can calculate the CTT for the streets and find the fastest route for the cars. The program weights the street based on the presented methods and calculates the best route using presented A* algorithm. SUMO is able to find the exact traveling time for the last car which has passed the street and this program uses this feature to calculate 5 recent cars' traveling time on each street and calculates the CTT for each street. If there is not any car on the route, the program assigns the CTT equal to ITT which is calculated considering the maximum allowed speed and street length. Moreover, several statistics related to traffic simulation are created such as fuel consumption, waiting time, traveling time, emissions for each car; or number of vehicles during simulation, total amount of vehicles emissions in streets during the simulation, fuel consumption, etc. for each street.

The developed program is started with requesting name of the car, origin and destination. First, by using mentioned route planning, and using CTT for streets, the best route (fastest route) from origin to destination is calculated and the car starts to move through the route. To have dynamic route planning, in each simulation step, the program calculates the current CTT for streets. Then, when a car arrives to an intersection, the program calculates "the best route" again, based on current CTT and if the "current best route" is different with the "current route" for the car, the program changes the car's route. These changes in routes happen until the car arrives to the destination.

6.4.3. Simulation Scenario

In order to investigate the effect of the route changing, initially, map of Cologne is divided into several zones, and realistic traffic of cars between 6 am till 8 am is considered. After that, 20 different zones with different traffic densities are selected based on the real traffic status at 7:00 am. Then, a car is added at each zone (with the origin and destination) and an identical traveling distance (5km) is defined for all the cars.

To find out the effect of dynamic route planning in the vehicles traveling time with using CTT, three different simulations are done as follows:

a. Firstly, only 20 cars travel in the Cologne map without any other car or any traffic light (Ideal Form). Then traveling time for 20 cars are obtained.
b. The second simulation is done with the realistic traffic of city of Cologne and also with 20 cars. This means more than 250.000 individual cars moving in cologne in this two hours. Also in this simulation the traveling time for 20 cars are obtained.
c. In the third simulation, the mentioned dynamic route planning using the developed program are employed in order to find the best route for the cars. This changing route for cars are done for all 20 cars and the traveling times are obtained.

6.4.4. Traffic Simulation Results

Simulations are done with the explained scenario. All data is collected and ordered based on traveling time for cars from low to high in Fig. 6-1. Also, in order to obtain a better view of this method and illustrate the efficiency of using dynamic route planning, the fuel consumption and emitted CO_2 for cars are obtained (based on provided data from “Handbook Emission Factors for Road Transport (HBEFA)[104]” for typical personal car) and illustrated in Fig. 6-2 and Fig. 6-3.

As it can be seen in Fig. 6-1 changing the route for each car has a different effect in decreasing the traveling time. For example, in low traffic area (car’s number 1-7) the average decrease of the traveling time is 41.12%, for medium traffic area (car’s No. 7-14) is 52.84% and for high traffic area (car’s No. 14-20) is 60.79%.

In addition, about decreasing CO_2 and Fuel consumption, in almost all the cars, decreasing the traveling time reduces fuel consumption and CO_2 emissions. Fig. 6-2 shows Fuel consumption diagram and illustrates the important fact that using DRP in low traffic area increases fuel consumption. But in the average, using DRP decreases the fuel consumption. The average decrease in the fuel consumption in this simulation is 48.27%. It means that using this approach can save huge amount of money and resources.

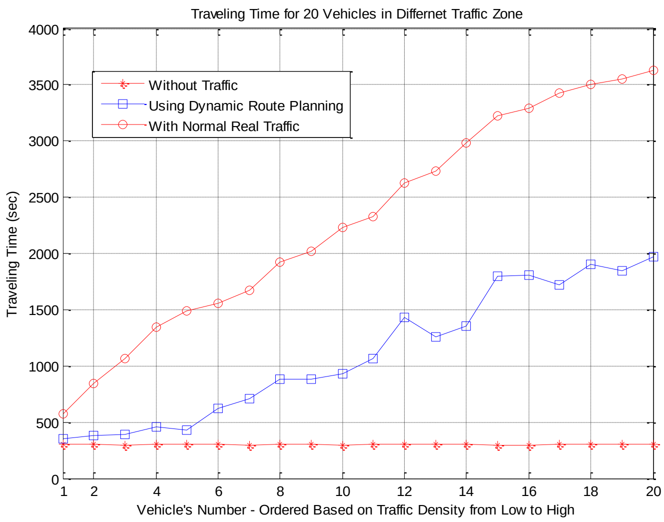


Figure 6-1. *Traveling Time for 20 Vehicles, in three traffic simulation in SUMO: with and without traffic and Using Dynamic Route Planning.*

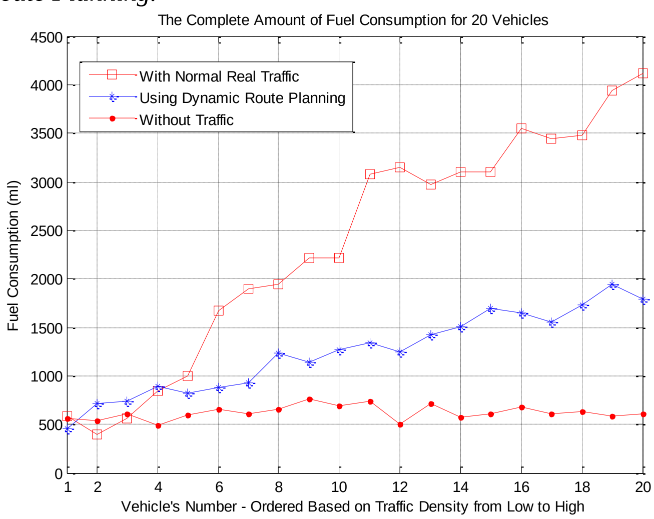


Figure 6-2. *Total amount of Fuel Consumption of 20 Vehicles during three different Simulation: with and without traffic and Using Dynamic Route Planning..*

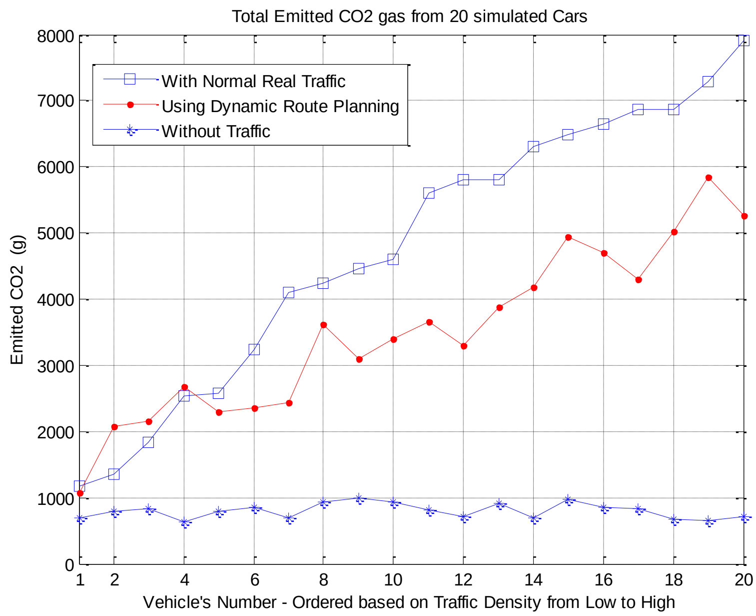


Figure 6-3. *Total Emitted CO2 gas from each 20 simulated cars during travel from origin to destination, in three different simulation situations.*

6.5. VANET simulation

As mentioned before, this study selects the Veins framework for simulation of VANET because of several features. Veins framework integrates SUMO and OMNET++. At first traffic of the cars are generated in SUMO and then exported to the network simulator. OMNET++ considers all cars as nodes and simulates the scenario. If any change occurs in the network, Veins can change the cars scenario in SUMO.

This study has developed Veins framework by adding a new module in OMNET++ which adds several RSUs in each intersection which are defined in SUMO and consider them as nodes as well.

By considering this new module and Veins, realistic traffic of Cologne (and also the mentioned 20 cars) is simulated as a VANET in OMNET++. Every node is able to communicate using IEEE 802.11p standard. The main parameters which have been used in this study for the network simulation are summarized in Table 4-1. Simulation is done as follows: All the cars and also every RSUs broadcast beacons periodically.

CTTs for each street are calculated as follows: if a RSU in the starting point of the street, receives a beacons from a car, then, if the distance between the RSU and vehicle is less than 5 meters (the car has entered to the street), the RSU saves the simulation time and the car ID. Then, if the RSU which is located at the end point of the street, receives a beacon from the same car, and the distance between the car and the RSU is less than 5 meters, the current simulation time is saved as ending time for the car in the mentioned street (the car leaves the street); and by using the saved starting and ending times, the traveling time for the cars in the street are calculated and CTT is equal to average of five recent cars which have traveled on the street. (In order to reduce the computing process and time, the streets and the RSUs which are located along the route of the cars are considered. One difficulty in large-scale simulation of VANET is “the time of simulation”. For example for one hour simulation, OMNeT++ spends more than 100 hours in real time. Also the distances are calculated based on the position of the cars and RSUs which are provided by OMNET++ and SUMO.) Up to now, the simulation calculated and assigned the CTT for streets. All RSUs are broadcasting the CTT for streets which they are related to.

Dynamic route planning for mentioned 20 cars is implemented as follows: at the start time of journey, the best routes are calculated based on current CTTs which are provided with OMNET++ simulation and the mentioned algorithm in previous section; and the best routes are sent to the SUMO to assign as cars routes. Similar to SUMO simulation, in OMNET++, in every simulation step, current CTTs are calculated. When a car gets close to the end point of the street, (RSU at the end of the street receives a beacon from the cars) a new best route is calculated based on current CTT and is sent to SUMO as new route. The simulations are done for 20 cars and traveling times are obtained for them.

6.5.1. VANET simulation Results

The simulation has been done and the traveling times for all 20 cars have been obtained. Fig. 6-4 shows the traveling times for 20 cars which are obtained by using car to infrastructure communication to calculate the CTT and best route with OMNET++. Also Fig. 6-4 shows the data which was obtained in the previous sections using SUMO. Moreover, in order to analyze the accuracy of VANET simulation, Probability of Beacon Delivery for 20 cars is calculated by considering all the received and lost beacons during the travel from origin to destination. Also for each of the mentioned 20 cars, the difference between traveling time in SUMO and OMNET++ simulator are considered. Fig. 6-5 shows probability of beacon delivery for 20 cars and also corresponding time differences.

Fig. 6-4 and Fig. 6-5, show the fact that the difference between VANET simulation result (realistic) and traffic simulation result (ideal) is quite small. Realistic result is obtained from simulation in OMNET++ by using the beaconing and 802.11p standard and the routes are changed based on the received beacons from cars and calculating CTT; and ideal form is obtained from SUMO by using the developed program which considers the exact traveling time (CTT) and the exact position of the cars and RSUs for changing the routes. One of the reasons for this accurate result is related to beaconing scenario with numerous sending messages from each car which helps to deliver the message as soon as possible. This fact illustrates that car to car communication and car to infrastructure communication based on IEEE 802.11p is able to decrease the traveling time of the cars. In addition Fig. 6-5 shows that by increasing the traffic density (car’s number) the probability of beacon delivery is decreased and also the cars traveling time is increased.

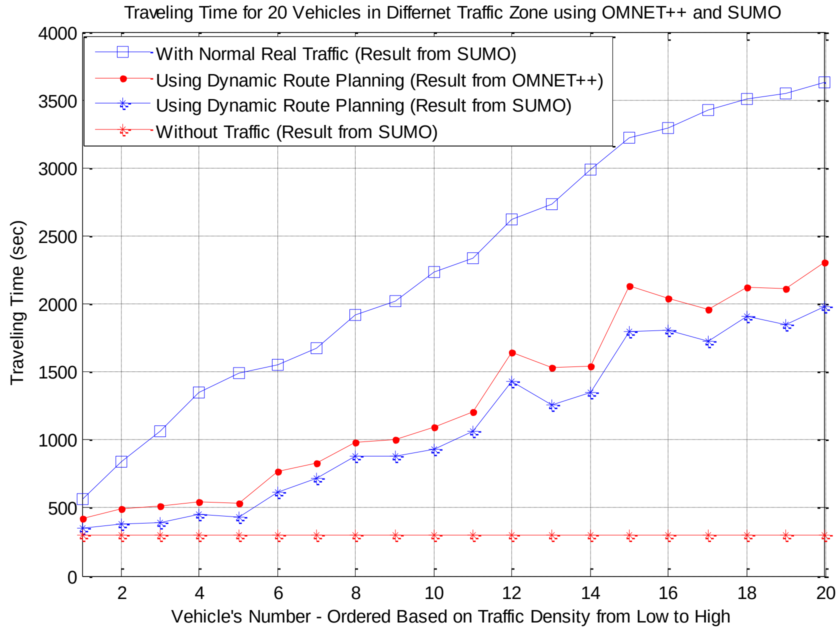


Figure 6-4. *Traveling Time for 20 Vehicles, in three traffic simulation in SUMO: with and without traffic and Using Dynamic Route Planning in SUMO and using V2I communication in OMNET++.*

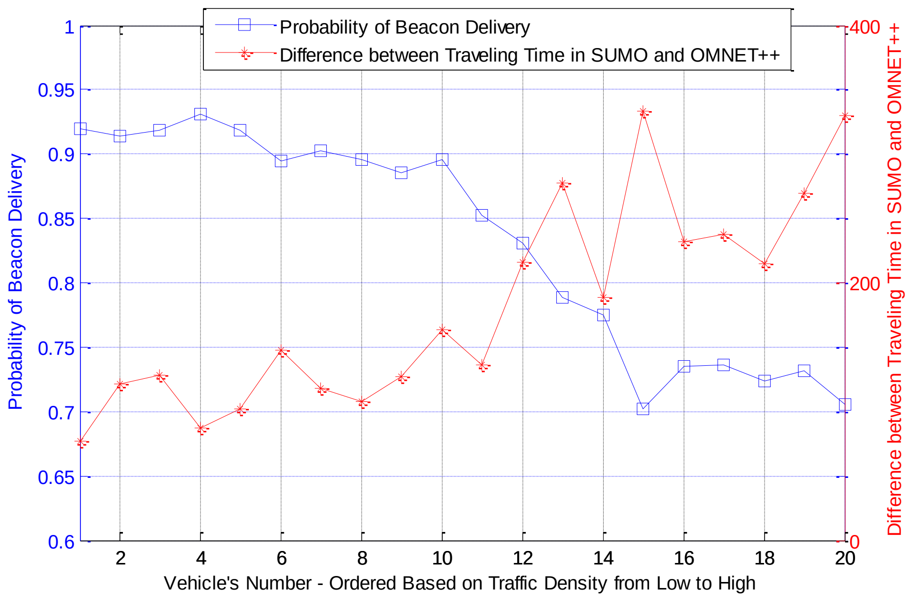


Figure 6-5. *Probability of Beacon Delivery and the Difference between Traveling Times obtained from SUMO and OMNET++ simulator using Dynamic Route Planning.*

6.6. Chapter Conclusion and Future Work

This chapter proposes a new method in order to calculate the fastest route for vehicles from origin to destination using V2I communication which provides real-time traffic information to drivers. This method, assigns a Current Traveling Time (CTT) for each street in a city which could help drivers to find the best route and also provides the on-line traffic data of the street (number of cars in roads segment). Then, effect of the method is investigated by using traffic simulator (SUMO) and also dynamic route planning employing CTT for each street which are calculated with the proposed method. In addition, OMNET++ is used in order to simulate the mentioned scenario as a VANET (with IEEE 802.11p standard) using Veins framework to run SUMO and OMNET++ in parallel. This study has developed the Veins framework by adding new modules to OMNET++ which aims to add several RoadSideUnits (RSU) in the realistic traffic simulation. Moreover this study has developed a new program written in Python which is connected to SUMO and controls the traffic simulation. This program uses SUMO to simulate a realistic traffic and also with calculating the Current Traveling Time for streets and dynamic route planning for the cars. This program dynamically calculates the fastest route for the specific car. The dynamic route planning helps to decrease the traveling time of the cars. Also VANET simulation results show the high accuracy for IEEE 802.11p. VANET simulation and traffic simulation results have been compared. The results show that car to car communication can be implemented in real world using 802.11p which helps to make roads safer and more comfortable. Future work includes simulating traffic of the city with other Dynamic Route Planning approaches and also with network simulators by using another wireless technologies such as WiMAX.

Chapter 7

7. CONCLUSION

Intelligent Transportation Systems (ITS) have progressed at a rapid rate, which aim to improve transportation activities in terms of safety and efficiency. Vehicle to Vehicle (V2V) communications and Vehicle to Infrastructure (V2I) communications are important components of the ITS architecture. Communication between cars often referred to Vehicular Ad-Hoc Networks (VANET) and it has many advantages such as: reducing cars accidents, minimizing the traffic jam, reducing fuel consumption and emissions and etc. VANET architecture has been standardized in the IEEE 802.11p specification. Network simulators can simulate the ad-hoc network but they cannot simulate the huge traffic of cities. In order to solve this problem, this thesis has studied on the Veins framework which is used to run a traffic (SUMO) and a network (OMNET++) simulator in parallel and simulates the realistic traffics of the city of Cologne, Germany, as an ad-hoc network.

In this thesis, a detailed study on VANETs and different approaches and application of V2V and V2I communication in real world are done. Then, different protocols and standards which are used in VANET and required tools in order to simulate V2V and V2I communication, are investigated. Then, different simulators which are used for VANET simulation are compared and almost all features and drawbacks are explained and the best choice in order to simulate a real VANET-based V2V and V2I communication are selected.

Then, realistic Vehicle to Vehicle and Vehicle to Infrastructure communication using real map and real traffic data in a large scale urban area are simulated, and performance analysis of the beaconing concept in IEEE 802.11p standard are done and probability of beacon delivery is calculated and the results are compared with analytical model.

Next, study on applications of the V2I communication in real world, are done which consists of two main parts: First, the effect of VANET-enabled traffic lights control on decreasing the response time of emergency vehicles in realistic large-scale urban area, are done. Second, the impact of VANET-based V2I communication using IEEE 802.11p on reducing vehicles traveling time, fuel consumption, emissions and traffic monitoring in realistic large scale urban, are investigate.

REFERENCES

[1] http://www.nhtsa.gov.

[2] VD Khairnar, SN Pradhan, " Comparative Study of Simulation for Vehicular Ad-Hoc Network ",International Journal of Computer Applications (0975 – 8887) Volume 4– No.10, August 2010.

[3] Collier, W.C.; Weiland, R.J.; , "Smart cars, smart highways," Spectrum, IEEE , vol.31, no.4, pp.27-33, April 1994 doi: 10.1109/6.272224.

[4] Widhiasi, A.; Mohanan, V.; Pasha, M.F.; Budiarto, R.; , "Vertical Handover Scheme for Car-to-Car Communication Based on IEEE 802.21 Standard," Computer Engineering and Applications (ICCEA), 2010 Second International Conference on , vol.1, no., pp.143-147, 19-21 March 2010 doi: 10.1109/ICCEA.2010.36.

[5] IEEE 802.11p Wireless Access for Vehicular Environments, Draft Standard.

[6] Hannes Hartenstein (Editor), Kenneth Laberteaux (Editor), VANET Vehicular Applications and Inter-Networking Technologies, John Wiley & Sons ISBN: 978-0-470-74056-9.January 2010.

[7] M. Fiore, J. Haerri, F. Filali, and C. Bonnet, \Vehicular mobility simulation for VANETS," in Proceedings of the 40th Annual Simulation Symposium (ANSS 2007), Norfolk, Virginia, March 2007.

[8] Vaishali D. Khairnar, S. N. Pradhan, "Comparative Study of Simulation for Vehicular Ad-Hoc Network", International Journal of Computer Applications (0975 – 8887) Volume 4– No.10, August 2010.

[9] D. Schrank, B. Eisele, T. Lomax, "Urban Mobility Report 2012," Texas Transp. Inst., Texas A&M Univ. Syst., College Station, TX, Dec 2012.

[10] F. H. Administration, "Traffic congestion and reliability - trends and advanced strategies for congestion mitigation," Federal Highway Administration, Tech. Rep., 2005.

[11] Ali Bin Tariq, "Simulation of Beaconing in Vehicular Ad-Hoc Networks", Master of Science Thesis, December 2011, Tampere University of Technology.

[12] H. Hartenstein and K. Laberteaux. VANET: Vehicular Applications and Inter-Networking Technologies. John Wiley and Sons, 2009.

[13] "Assessment of the applicability of cooperative vehicle-highway automation systems to bus transit and intermodal freight: case study, "California Partners for Advanced Transit and Highways (PATH), 2004.
http://repositories.cdlib.org/cgi/viewcontent.cgi?article=1629&context=its/path.

[14] Huaqun Guo; "Automotive Informatics and Communicative Systems: Principles in Vehicular Networks and Data Exchange"; British Cataloguing in Publication Data ; 2009; ISBN 978-1-60566-338-8.

[15] LAN MAN Standards Committee of the IEEE Computer Society.Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard 802.11. Technical report, 1999.

[16] IEEE P802.11-REVma™/D7.0, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,” Revision of 802.11-1999, June 2006.

[17] Official IEEE 802.11 working group project timelines. http://grouper.ieee.org/groups/802/11/Reports/802.11_Timelines.htm.

[18] Status of Project IEEE 802.11 Task Group p. http://grouper.ieee.org/groups/802/11/Reports/tgp_update.htm.

[19] William Stallings Data and Computer Communications 6th edition Publisher: Prentice Hall Copyright: 2007.

[20] "IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operation," IEEE Pl609.41D9, August 2010, pp. 1 -61, 2010.

[21] Chi-Sheng Lin; Jia-Chin Lin; , "Novel Channel Estimation Techniques in IEEE 802.11p Environments," Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st , vol., no., pp.1-5, 16-19 May 2010.

[22] van Eenennaam, M.; van de Venis, A.; Karagiannis, G.; , "Impact of IEEE 1609.4 channel switching on the IEEE 802.11p beaconing performance," Wireless Days (WD), 2012 IFIP , vol., no., pp.1-8, 21-23 Nov. 2012.

[23] R. A. Uzcategui and G. Acosta-Marum, "Wave: A Tutorial," IEEE Communications Magazine, vol. 47, pp. 126-133, May, 2009.

[24] IEEE P802.11p, “Amendment 3: wireless access in vehicular environments (WAVE),” Draft D0.26, January 2006.

[25] IEEE P1609.1, “Trial-use standard for wireless access in vehicular environments (WAVE) - resource manager,” Draft D17, July 2006.

[26] IEEE P1609.2, “Trial-use standard for wireless access in vehicular environments (WAVE) - security services for applicationsand management messages,” Draft D7, April 2006.

[27] IEEE P1609.3, “Trial-use standard for wireless access in vehicular environments (WAVE) - networking services,” Draft D22, January 2007.

[28] IEEE P1609.4, “Trial-use standard for wireless access in vehicular environments (WAVE) - multi-channel operation,” Draft D08, July 2006.

[29] “Fcc allocates spectrum in 5.9 ghz range for intelligent transportation systems uses,” News, Federal Communications Commission, 1999.

[30] Noori, H “Realistic Urban Traffic Simulation as Vehicular Ad-Hoc Network (VANET) via Veins Framework “, 12th Conference of Open Innovations Framework Prgramm. FRUCT, Nov.2012.

[31] Bechler M, Franz WJ, Wolf L. Mobile Internet access in FleetNet. InVerteilten Systemen KiVS 2003, 2003.

[32] Gongjun Yan, Khaled Ibrahim and Michele C. Weigle, "Vehicular Network Simulators," In Vehicular Networks: From Theory to Practice, Stephan Olariu and Michele C. Weigle, Eds. Chapman & Hall/CRC, 2009.

[33] Choffnes DR, Bustamante FE. Modelling vehicular traffic and mobility for vehicular wireless networks.Technichal Report, Department of Computer Science, Northwestern University,NWU-CS-05-03, July 2005.

[34] J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, 2003.

[35] SUMO introduction in the sourceforge website. Available: http://sourceforge.net/apps/mediawiki/sumo/

[36] Gongjun Yan, Khaled Ibrahim and Michele C. Weigle, "Vehicular Network Simulators," In Vehicular Networks: From Theory to Practice, Stephan Olariu and Michele C. Weigle, Eds. Chapman & Hall/CRC, 2009.

[37] http://www.vissim.com/

[38] Michael Behrisch, Laura Bieker, Jakob Erdmann and Daniel Krajzewicz. SUMO - Simulation of Urban MObility: An Overview In: SIMUL 2011, The Third International Conference on Advances in System Simulation, 2011.

[39] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," IEEE Transactions on Mobile Computing, vol. 10, pp. 3–15, January 2011.

[40] S. Krau, "Microscopic traffic simulation: Robustness of a simple approach," Tech. Rep. 306, Deutsches Zentrum f ¨ ur Luft- und Raumfahrt, 1997.

[41] TraNS official website. Available: http://lca.epfl.ch/proje cts/trans

[42] Rajive Bagrodia, Richard Meyer, Mineo Takai, Yu an Chen, Xiang Zeng, Jay Martin, and Ha Yoon Song. "Parsec: A parallel simulation environment for complex systems. Computer, 31(10):77–85, 1998.

[43] OPNET Technologies, 2008. Available at: http://www.opnet.com

[44] Kevin Fall and Kannan Varadhan. The ns Manual (formerly ns Notes and Documentation). VINT, 2006.

[45] OMNeT++ official website. Available: http://www.omnetpp.org

[46] OMNeT++ User Manual. Available: http://www.omnetpp.org/doc/omnetpp41/manual/usman.htm

[47] S. Y. Wang, C. L. Chou, Y. H. Chiu, Y. S. Tzeng, M. S. Hsu, Y. W. Cheng, W. L. Liu, and T. W.Ho. NCTUns 4.0: An Integrated Simulation Platform for Vehicular Traffic. In Communication, and Network Researches," WiVec'07, 2007.

[48] Sommer, C.; German, R.; Dressler, F.; , "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," Mobile Computing, IEEE Transactions on , vol.10, no.1, pp.3-15, Jan. 2011.Veins - Vehicles in Network Simulation, http://veins.car2x.org/

[49] Aamir Hassan, Tony Larsson "On the requirements on models and simulator design for integrated VANET Simulation"8th International Workshop on Intelligent transportation System, March 2011 Hamburg Germany.

[50] Weingartner, E.; vom Lehn, H.; Wehrle, K.; , "A Performance Comparison of Recent Network Simulators," Communications, 2009. ICC '09. IEEE International Conference on , vol., no., pp.1-5, 14-18 June 2009 doi: 10.1109/ICC.2009.5198657.

[51] Harri, J.; Filali, F.; Bonnet, C., "Mobility models for vehicular Ad-Hoc networks: a survey and taxonomy," Communications Surveys & Tutorials, IEEE , vol.11, no.4, pp.19-41, Fourth Quarter 2009 doi: 10.1109/SURV. 2009.0904039.

[52] MOVE (MObility model generator for VEhicular networks): Rapid Generation of Realistic Simulation for VANET, 2007.Available at: http://lens1.csie.ncku.edu.tw/MOVE/index.htm.

[53] Michal Piorkowski, Maxim Raya, Ada Lezama Lugo, Panos Papadimitratos, Matthias Grossglauser and Jean Pierre Hubaux, “TraNS: Realistic Joint Traffic and Network Simulator for VANETs”, Poster at the 13th ACM Annual International Conference on Mobile Computing and Networking MobiCom'07, Montreal, Canada, September 2007.

[54] Fiore, M.; Harri, J.; Filali, F.; Bonnet, C.; , "Vehicular Mobility Simulation for VANETs," Simulation Symposium, 2007. ANSS '07. 40th Annual , vol., no., pp.301-309, 26-28 March 2007 doi: 10.1109/ANSS.2007.44.

[55] David R. Choffnes, Fabián E. Bustamante, “An integrated mobility and traffic model for vehicular wireless networks”, VANET '05 Proceedings of the 2nd ACM international workshop on Vehicular Ad-Hoc networks Pages 69 – 78.

[56] Karnadi, F.K.; Zhi Hai Mo; Kun-chan Lan; , "Rapid Generation of Realistic Mobility Models for VANET," Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE , vol., no., pp.2506-2511, 11-15 March 2007 doi: 10.1109/WCNC.2007.467.

[57] R. Fernandes, P. d’Orey, and M. Ferreira, “Divert for realistic simulation of heterogeneous vehicular networks,” in Mobile Ad-Hoc and SensorSystems (MASS), 2010 IEEE 7th International Conference on. IEEE, 2010, pp. 721–726.

[58] H.Conceição, L. Damas, M. Ferreira, and J. Barros, “Large-scale simulation of V2V environments,” in Proceedings of the 2008 ACM symposium on Applied computing. ACM, 2008, pp. 28–33.

[59] Fernandes, R.; Ferreira, M.; , "Scalable VANET Simulations with NS-3," Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th , vol., no., pp.1-5, 6-9 May 2012.

[60] A. Kopke, M. Swigulski, K. Wessel, D. Willkomm, PT Haneveld, TEV Parker, OWVisser, HS Lichte, and S. Valentin. Simulating wireless and mobile networks in OMNeT++ the MiXiM vision. In Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, pages 1-8. ICST, 2008.

[61] Noori, H; Badihi Olyae, B; “A Novel Study on Beaconing for VANET-basedVehicle to Vehicle Communication: Probability of Beacon Delivery in Realistic Large-Scale Urban Area using 802.11p “,The 4th International Conference on Smart Communications in Network Technologies,(IEEE SaCoNet 2013) Paris, France, June 17-19, 2013.

[62] B.Zhou, J. Cao, H. Wu; , “Adaptive traffic light control of multiple intersections in WSN-based ITS”, 978-1-4244-8331, IEEE, 2011.

[63] Huaqun Guo; Shen Tat Goh; Foo, N.C.S.; Qian Zhang; Wai-Choong Wong; , "Performance evaluation of 802.11p device for secure vehicular communication," Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International , vol., no., pp.1170-1175, 4-8 July 2011.

[64] Uppoor, S.; Fiore, M.; , "Large-scale urban vehicular mobility for networking research," Vehicular Networking Conference (VNC), 2011 IEEE , vol., no., pp.62-69, 14-16 Nov. 2011.

[65] OpenStreetMap, http://www.openstreetmap.org

[66] Reinders, R.; van Eenennaam, M.; Karagiannis, G.; Heijenk, G.; , "Contention window analysis for beaconing in VANETs," Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International , vol., no., pp.1481-1487, 4-8 July 2011 doi: 10.1109/IWCMC.2011.5982757.

[67] van Eenennaam, M.; Remke, A.; Heijenk, G.; , "An analytical model for beaconing in VANETs," Vehicular Networking Conference (VNC), 2012 IEEE , vol., no., pp.9-16, 14-16 Nov. 2012.

[68] Vinel, A.; Staehle, D.; Turlikov, A.; , "Study of Beaconing for Car-to-Car Communication in Vehicular Ad-Hoc Networks," Communications Workshops, 2009. ICC Workshops 2009. IEEE International Conference on , vol., no., pp.1-5, 14-18 June 2009.

[69] Vinel, A.; , "3GPP LTE Versus IEEE 802.11p/WAVE: Which Technology is Able to Support Cooperative Vehicular Safety Applications?," Wireless Communications Letters, IEEE , vol.1, no.2, pp.125-128, April 2012.

[70] Sandesh Uppoor, Marco Fiore, "Vehicular mobility in large-scale urban environments", Mobile Computing and Communications Review (MC2R), Special Issue on Mobicom 2011 Student Research Competition posters.

[71] Noori, H; "Modeling the Impact of VANET-Enabled Traffic Lights Control on the Response Time of Emergency Vehicles in Realistic Large-Scale Urban Area"; in Proceedings of IEEE ICC'13 - Workshop on Emerging Vehicular Networks: V2V/V2I and Railroad Communications, Budapest, Hungary 9-13 June 2013.

[72] R. Sánchez-Mangas, A. García-Ferrrer, A. de Juan, and A. M. Arroyo,"The probability of death in road traffic accidents. How important is a quick medical response?"Accid. Anal. Prev., vol. 42, no. 4, pp. 1048–1056, Jul. 2010.

[73] R. P. Gonzalez, G. R. Cummings, H. A. Phelan, M. S. Mulekar, and C. B. Rodning, "Does increased emergency medical services prehospital time affect patient mortality in rural motor vehicle crashes? A statewide analysis,"Amer.J.Surg., vol. 197, no. 1, pp. 30–34, Jan. 2009.

[74] Cheng Siong Lim; Mamat, R.; Braunl, T.; , "Impact of Ambulance Dispatch Policies on Performance of Emergency Medical Services," Intelligent Transportation Systems, IEEE Transactions on , vol.12, no.2, pp.624-632, June 2011.

[75] M. Castrén, R. Karlsten, F. Lippert, E. F. Christensen, E. Bovim, A. M. Kvam, I. Robertson-Steel, J. Overton, T. Kraft, L. Engerstrom, and L. Garcia-Castrill Riego, "Recommended guidelines for reporting on emergency medical dispatch when

conducting research in emergency medicine: The Utstein style,"Resuscitation, vol. 79, no. 2, pp. 193–197, Nov. 2008.

[76] Fogue, M.; Garrido, P.; Martinez, F.J.; Cano, J.; Calafate, C.T.; Manzoni, P.; , "Automatic Accident Detection: Assistance Through Communication Technologies and Vehicles," Vehicular Technology Magazine, IEEE , vol.7, no.3, pp.90-100, Sept. 2012.

[77] M. Gendreau, G. Laporte, and F. Semet, "A dynamic model and par-allel tabu search heuristic for real-time ambulance relocation,"Parallel Comput., vol. 27, no. 12, pp. 1641–1653, Nov. 2001.

[78] M. O. Ball and L. F. Lin, "A reliability model applied to emergency service vehicle location,"Oper. Res., vol. 41, no. 1, pp. 18–36, Jan./Feb. 1993.

[79] D. Schrank, B. Eisele, T. Lomax, "Urban Mobility Report 2012," Texas Transp. Inst., Texas A&M Univ. Syst., College Station, TX, Dec 2012.

[80] F. H. Administration, "Traffic congestion and reliability - trends and advanced strategies for congestion mitigation," Federal Highway Administration, Tech. Rep., 2005.

[81] Khosroshahi, A.H.; Keshavarzi, P.; Koozehkanani, Z.D.; Sobhi, J., "Acquiring real time traffic information using VANET and dynamic route guidance," Computing, Control and Industrial Engineering (CCIE), 2011 IEEE 2nd International Conference on , vol.1, no., pp.9,13, 20-21 Aug. 2011.

[82] T. Hunter, R. Herring, P. Abbeel, and A. Bayen. Path and travel time inference from gps probe vehicle data. NIPS Analyzing Networks and Learning with Graphs, 2009.

[83] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan. The pothole patrol: using a mobile sensor network for road surface monitoring. In Proceeding of the 6th international conference on Mobile systems, applications, and services (MobiSys 2008), pages 29–39. ACM, 2008.

[84] E. Horvitz, J. Apacible, R. Sarin, and L. Liao. Prediction, expectation, and surprise: Methods, designs, and study of a deployed traffic forecasting service. In Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence(UAI-2005), pages 244–257, 2005.

[85] P. Mohan, V.N. Padmanabhan, and R. Ramjee. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In Proceedings of the 6th ACM conference on Embedded network sensor systems, pages 323–336. ACM, 2008.

[86] M. Haklay and P. Weber. Openstreetmap: User-generated street maps. IEEE Pervasive Computing, 7(4):12–18, 2008.

[87] http://www.ipermob.org/.

[88] http://www.vics.or.jp/english/vics/index.html.

[89] Ghandeharizadeh, S.; Krishnamachari, B., "C2P2: a peer-to-peer network for on-demand automobile information services," Database and Expert Systems Applications, 2004. Proceedings. 15th International Workshop.

[90] Lars Wischhof, André Ebner, Hermann Rohling. "Self-Organizing Traffic Information System based on Car-to-Car Communication: Prototype

Implementation.”Hamburg, Germany : s.n., March 2004. 1st Intl.Workshop on Intelligent Transportation.

[91] Jiang Zhang; Ziliaskopoulos, A.K.; Ning Wen; Berry, R.A., "Design and implementation of a vehicle-to-vehicle based traffic information system," Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE , vol.

[92] Hara, K.; Suzuki, M.; Tokuda, T.; Shigeno, H., "Comparative evaluation of information dissemination method for Dynamic Vehicular Route Planning," ITS Telecommunications (ITST), 2012 12th International Conference on , vol., no., pp.119,123, 5-8 Nov. 2012.

[93] Pan, J.; Sandu Popa, I.; Zeitouni, K.; Borcea, C., "Proactive Vehicular Traffic Re-routing for Lower Travel Time," Vehicular Technology, IEEE Transactions on , vol.PP, no.99, pp.1,1, 0.

[94] Fontanelli, S.; Bini, E.; Santi, P., "Dynamic route planning in vehicular networks based on future travel estimation," Vehicular Networking Conference (VNC), 2010 IEEE , vol., no., pp.126,133, 13-15 Dec. 2010.

[95] Eiichi Taniguchi and Hiroshi Shimamoto. Intelligent transportation systems based dynamic vehicle routing and scheduling with variable travel times. In Transportation Research, part C, pages 235–250, 2004.

[96] H. Kanoh. Dynamic route planning for car navigation systems using genetic algorithms. InInt. Journal of Knowledge-based and Intelligent Engineering Systems, pages 65–78, 2007.

[97] Jedrzej Rybicki, Bjrn Scheuermann, Markus Koegel, and Martin Mauve. Peertis: A peer-to-peer traffic information system. InProc. ACM VANET, pages 23–32, 2009.

[98] Yin Yang and Rajive Bagrodia. Evaluation of a vanet-based advanced intelligent transportation system. In Proc. ACM VANET, pages 3–12, 2009.

[99] Christoph Sommer, Armin Schmidt, Yi Chen, Reinhard German, Wolfgang Koch, and Falko Dressler. On the feasibility of umts-based traffic information systems. Ad-Hoc Networks, 8(5):506–517, 2010.

[100] Irina Matschke. The impact of dynamic navigation on the travel times in urban networks. Advanced OR and AI Methods in Transportation, pages 506–517.

[101] "IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operation," IEEE Pl609.41D9, August 2010, pp. 1 -61, 2010.

[102] Miller, Shaw.” GIS-T data Models,Geographic Information Systems for Transportation:Principals and Applications.” Oxford University Press, 2004.

[103] Stuart J. Russell, Peter Norvig. “artificial intelligence a modern approach”. New Jersey : Prentice-Hall, 1995. ISBN 0-13-103805-2.

[104] http://www.hbefa.net

[105] P. E. Engelstad and O. N. Osterbo, “Non-saturation and saturation analysis of IEEE 802.11e EDCA with starvation prediction,” in MSWiM ’05: Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems. New York, NY, USA: ACM, 2005, pp. 224–233.

[106] G. Bianchi, ”Performance Analysis of the IEEE 802.11 Distributed Coordination Function,” IEEE JSAC, 18(3), pp. 535-547, Mar. 2000.

[107] F. Cali, M. Conti, E. Gregory “IEEE 802.11 Wireless LAN: Capacity Analysis and protocol enhancement,” Proc. INFOCOM’98, San Francisco, 1998. P. 142-149.

[108] Sommer, C.; Joerer, S.; Dressler, F., "On the applicability of Two-Ray path loss models for vehicular network simulation," *Vehicular Networking Conference (VNC), 2012 IEEE* , vol., no., pp.64,69, 14-16 Nov. 2012.

[109] S. Ahmed, G. C. Karmakar, and J. Kamruzzaman, “An EnvironmentAware Mobility Model for Wireless Ad Hoc Network,”Elsevier Computer Networks, vol. 54, no. 9, pp. 1470–1489, May 2010.

[110] R. Nagel and S. Eichler, “Efficient and Realistic Mobility and Channel Modeling for VANET Scenarios Using OMNeT++ and INET-Framework,” in 1st ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2008). Marseille, France: ICST, March 2008.

[111] T. S. Rappaport,Wireless Communications: Principles and Practice, 2nd ed. Upper Saddle River, New Jersey: Prentice Hall PTR, 2009.

APPENDIX A: THE PYTHON SCRIPT DEVELOPED FOR THE EMERGENCY VEHICLE APPLICATION.

```
import os, subprocess, sys, random
sys.path.append(os.path.join(os.path.dirname(__file__), '..', '..', '..', '..', 'tools'))
sys.path.append(os.path.join(os.environ.get("SUMO_HOME",
os.path.join(os.path.dirname(__file__), '..', '..', '..')), 'tools'))
from sumolib import checkBinary
import traci

PORT = 8813

NSGREEN = "GrGr"
NSYELLOW = "yryr"
WEGREEN = "rGrG"
WEYELLOW = "ryry"

PROGRAM =
[WEYELLOW,WEYELLOW,WEYELLOW,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSGRE
EN,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSYELLOW,NSYELLOW,WEGREEN]

sumoBinary = checkBinary('sumo-gui')
sumoConfig = "data/Cologene.sumocfg"
if len(sys.argv) > 1:
    retCode = subprocess.call("%s -c %s --python-script %s" % (sumoBinary, sumoConfig,
__file__), shell=True, stdout=sys.stdout)
    sys.exit(retCode)
else:
    sumoProcess = subprocess.Popen("%s -c %s" % (sumoBinary, sumoConfig), shell=True,
stdout=sys.stdout)
    traci.init(PORT)
traci.simulationStep()
programPointer = len(PROGRAM)-1
traf_light_changed = 0
i=0
j=0
s=0
t=0
k=0
step = 0
FrTraff = 0
ToTraff = 0
ListAll = []
Find=0
Find2=0
FromEdge=[]
TraffX=[]
```

```
TraffY=[]
emergency_car = raw_input("Enter Emergency Car Name: ")
print "Emergency Car Name is :", emergency_car
route= traci.vehicle.getRoute(emergency_car)
RN= len(route)
A = traci.trafficlights.getIDList()    # Get the all Traffic Light List
N=len(A)
for i in range(0,N):                   # check all Traffic Light is in Route or not
    B=traci.trafficlights.getControlledLinks(A[i]) # Get the Link of A[i] Control
    NL=len(B)            #Number of Links A[i] is connected
    #print B , "00000000"
    for j in range(0,NL):  #splite each link of A[i] to 2 fromedge and toedge
        if Find2 == 0:
            DS=B[j][0]
            #print "J AND 0", DS
            DS[0]=DS[0].split('_')[0]
            DS[1]=DS[1].split('_')[0]
            FrEdge=DS[0]
            ToEdge=DS[1]
            print "Traffic Light From edge is.....", FrEdge
            print "Traffic Light TO EDGE is ...  ", ToEdge
            print "ROUTE", route
            for s in range (0,RN):  # compare the each fromedge and Toedge with the arrays
of Emergency car Route
                if  Find == 1: Find=0;Find2=1;break
                if  FrEdge == route[s]:    # if FromEdge is in route then check the ToEdge
else go to another link
                    #print "From Edge is Route"
                    for t in range (0,RN):  # check the ToEdge
                        if ToEdge == route[t]:
                            #print "YES Traffic light is on route"
                            #print "From edge is", FrEdge , "and To Edge is", ToEdge
                            LTJ = B[0][0]
                            TJun = LTJ[2].split('_')[0]
                            Traff_Jun = TJun.translate(None, ':')
                            Traff_x , Traff_y = traci.junction.getPosition(Traff_Jun)
                            TraffX.append(Traff_x)
                            TraffY.append(Traff_y)
                            FromEdge.append(FrEdge)
                            #print "FROM EDGE IS" , FromEdge
                            #print "i is ", i, "J is ", j
                            #print "t is ", t , "s is" ,s
                            ListAll.append(A[i]) # add this traffic light to list
                            Find = 1
                            break
        else: Find2=0;break
NT=len(ListAll)
print "Number of Traffic Lights in Emergency car's route is ", NT
```

```
print ListAll
Havechange=[0]*NT
NearTraff=[0]*NT
PassTraf=[0]*NT
#print "XXXXXXXXXXX", TraffX ,"YYYYYYYYYYYYYYYY", TraffY
distance = raw_input("Enter Desirable Distance between Emergency Car and Traffic Light: ")
distance = int(distance)
FE=len(FromEdge)
print "Distance is : " , distance
while step == 0 or traci.simulation.getMinExpectedNumber() > 0:
   #print (traci.simulation.getCurrentTime())
   traci.simulationStep()
   programPointer = min(programPointer+1, len(PROGRAM)-1)
   TrafProg=[0]*NT
   for k in range (0,NT):
         #TJ = traci.trafficlights.getControlledLinks(ListAll[k])
         #LTJ = TJ[0][0]
         #print "Firssssstttttt" , LTJ
         #TJun = LTJ[2].split('_')[0]
         #print "splittteeedddddddddddd", TJun
         #Traff_Jun = TJun.translate(None, ':')
         #print "ss",Traff_Jun;
         #Traff_x , Traff_y = traci.junction.getPosition(Traff_Jun)
      veh_x,veh_y=traci.vehicle.getPosition(emergency_car)
      print " Current Position of Emergency Car is : [" , veh_x , veh_y, "]"
      print " Current Speed is " , traci.vehicle.getSpeedWithoutTraCI(emergency_car)
      print " Position of Traffic Light", ListAll[k] , " isssss", Traff_x , Traff_y
      Distance_Emer_Traff = traci.simulation.getDistance2D(veh_x, veh_y, TraffX[k], TraffY[k],
isGeo=False, isDriving=False)
      print "Distance from Traffic Light to Emergency cars issss" ,Distance_Emer_Traff
      F= Distance_Emer_Traff
      print "Distance is" , Distance_Emer_Traff
      if PassTraf[k]==0:
            if NearTraff[k]==0:
                  if Havechange[k] == 0 :
                        if Distance_Emer_Traff < distance:
                              TrafProg[k]=traci.trafficlights.getProgram(ListAll[k])
                              print "PROGRAMM ISS", TrafProg[k]
                              print "DISTANCE IS LESS THAN ", distance
                              B=traci.trafficlights.getControlledLinks(ListAll[k])
                              CNL=len(B)
                              for p in range(0,CNL):    #splite each link of A[i] to 2 fromedge and
toedge
                                    print "traffic lighte name is " , ListAll[k]
                                    #print "B IS " , B
                                    DS=B[p][0]
                                    #print "J AND 0", DS
                                    DS[0]=DS[0].split('_')[0]
```

```
                        FrEdge=DS[0]
                        #print "FROM EDGE IS " , FrEdge
                        for z in range (0,FE):
                            #print "P is " , p
                            print "FE IS" , FE
                            print "Z is" , z
                            print "FROM EDGE IS" , FromEdge
                            print "Traffic Light From Edge is" , FrEdge
                            if FrEdge == FromEdge[z]:P=p;print "P isss", P;break
                                #print "traffiic light is " ,
traci.trafficlights.getRedYellowGreenState(ListAll[k])
                    Sta=['r']*CNL
                    print "CNL IS",CNL
                    Sta[p]='G'
                    STATUS= ''.join(Sta)
                    print "STATUSSSS IS", STATUS
                    traci.trafficlights.setRedYellowGreenState(ListAll[k], STATUS)
                    print "PPPP kashhf shhodd"
                    Havechange[k]=1
                    #print "XX IS " , x
                    print "distance is less than ", distance
            elif Distance_Emer_Traff < 10:
                NearTraff[k]=1
                print "Vehicle is near that 10 meter"
        elif Distance_Emer_Traff > 20:
                print "cars is pass the traffic light and 20"
                traci.trafficlights.setProgram(ListAll[k],str(TrafProg[k]))
                PassTraf[k]=1



no = traci.inductionloop.getLastStepVehicleNumber("0")
if no > 0:
    programPointer = (0 if programPointer == len(PROGRAM)-1 else 3)
step += 1
traci.close()
sys.stdout.flush()
```

APPENDIX B: THE PYTHON SCRIPT DEVELOPED FOR DYNAMIC ROUTE PLANNING.

```
import os, subprocess, sys, random
sys.path.append(os.path.join(os.path.dirname(__file__), '..', '..', '..', '..', 'tools'))
sys.path.append(os.path.join(os.environ.get("SUMO_HOME",
os.path.join(os.path.dirname(__file__), '..', '..', '..')), 'tools'))
from sumolib import checkBinary
import traci

PORT = 8813

NSGREEN = "GrGr"
NSYELLOW = "yryr"
WEGREEN = "rGrG"
WEYELLOW = "ryry"

PROGRAM =
[WEYELLOW,WEYELLOW,WEYELLOW,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSGREEN,NSYELLOW,NSYELLOW,WEGREEN]

sumoBinary = checkBinary('sumo')
sumoConfig = "data/Cologne.sumocfg"
if len(sys.argv) > 1:
    retCode = subprocess.call("%s -c %s --python-script %s" % (sumoBinary, sumoConfig,
__file__), shell=True, stdout=sys.stdout)
    sys.exit(retCode)
else:
    sumoProcess = subprocess.Popen("%s -c %s" % (sumoBinary, sumoConfig), shell=True,
stdout=sys.stdout)
    traci.init(PORT)
traci.simulationStep()
programPointer = len(PROGRAM)-1
i=0
s=0
t=0
k=0
CTT5cars=[0,0,0,0,0]
step = 0
VEH=[]
EDG=[]
TIM=[]
VehCTT=[]
ED=[]
ITT=[]
```

```
V=[]
CTT=[]
Jun=[]
L_Veh=[]
Trav_tim=[]
traci.edge.setEffort=[]
VehiD=["0","1"]
Curr_Trav=[]
Trav_tim_sys=[]
A=[]

sample_vehicle = raw_input("Enter sample vehicle Name: ")
#print "sample vehicle Name is :", sample_vehicle

# Get the all Lane in Network

All = traci.lane.getIDList()

#print A
NA=len(All)
print "NA iss", NA
for s in range(0,NA):
        LA=traci.lane.getLength(All[s])
        if LA>100:
                A.append(All[s])
N=len(A)
print "N is", N
output = open ("output.txt", "w")
output.write ("\n\n\n")
output.write ("-----------------------------------------------------------------------------------------\n")
output.write ("Simulation Time\t\t| Fuel Consumption\t\t | CO2 emissions \t\t | CO emssions\t\t | 
Noise emission \t\t | Current Speed \t\t | Current Lane\n")

# Calculate ITT for all Lanes

for i in range(0,N):
                L=traci.lane.getLength(A[i])          # Get the Length of all Lane
                print " Lane name is", A[i],"\t"
                print " is equal", L[i],"\n"
                V=traci.lane.getMaxSpeed(A[i])       # Get Maximum Allowed speed
                print "Maximum Allowed Speed for Lane", A[i],"\t"
                print " is", V[i],"\n"
                ITT= L/V
                print "ITT for Lane", A[i],"\t"
                print " is equal to", ITT[i],"\n"

                # Set ITT for Lanes
```

```
            ED= traci.lane.getEdgeID(A[i])
            traci.edge.setEffort = [A,ITT]
            CTT5cars=[ITT,ITT,ITT,ITT,ITT]
            CTTcheck=0

# Simulation starts here:

while step == 0 or traci.simulation.getMinExpectedNumber() > 0:
    print "Simulation Time is", (traci.simulation.getCurrentTime())
    traci.simulationStep()
    programPointer = min(programPointer+1, len(PROGRAM)-1)


    # Calculate CTT for all Lanes

    for j in range(0,N):

        VEH=traci.lane.getLastStepVehicleIDs(A[j])
        EDG= traci.lane.getEdgeID(A[j])
        TIM= traci.simulation.getCurrentTime()
        VehiD[0]=VEH
        if VehiD[0]== '[0]':
            VehCTT=traci.vehicle.getAdaptedTraveltime(VehiD[0], TIM[j], EDG[j])
        VehCTT=traci.lane.getTraveltime(A[j])
        if CTTcheck == 0:
            CTT5cars[0]=VehCTT
            CTTcheck=1
        elif CTTcheck == 1:
            CTT5cars[1]=VehCTT
            CTTcheck=2
        elif CTTcheck == 2:
            CTT5cars[2]=VehCTT
            CTTcheck=3
        elif CTTcheck == 3:
            CTT5cars[3]=VehCTT
            CTTcheck=4
        else:
            CTT5cars[4]=VehCTT
            CTTcheck=0
        CTT = sum(CTT5cars)/5
        print "Current Traveling Time (CTT) for Lane", A[j], "is equal to", CTT
        traci.edge.setEffort=[A[i],CTT]


        CO=str(traci.vehicle.getCOEmission(sample_vehicle))
        Fue=str(traci.vehicle.getFuelConsumption(sample_vehicle))
        LAV=str(traci.vehicle.getLaneID(sample_vehicle))
        Noise=str(traci.vehicle.getNoiseEmission(sample_vehicle))
```

```
        curr_spe=str(traci.vehicle.getSpeed(sample_vehicle))
        speed_noTra=str(traci.vehicle.getSpeedWithoutTraCI(sample_vehicle))
        CO2=str(traci.vehicle.getCO2Emission(sample_vehicle))
        current_time = str(traci.simulation.getCurrentTime())
        output.write (current_time), output.write ("\t\t\t")
        output.write (Fue), output.write ("\t\t\t")
        output.write (CO2), output.write ("\t\t\t")
        output.write (CO), output.write ("\t\t\t")
        output.write (Noise), output.write ("\t\t\t")
        output.write (curr_spe), output.write ("\t\t\t")
        output.write (LAV), output.write ("\t\t\t")
        output.write (speed_noTra), output.write ("\t\t\t\n")

        # for Re-routing

    route = traci.vehicle.getRouteID(sample_vehicle)
    print "current route of sample vehicle is", route
    RoutEDG = traci.route.getEdges(route)
    print "RoutEDG", RoutEDG
    EDGN= len(RoutEDG)
    print EDGN
    veh_x,veh_y=traci.vehicle.getPosition(sample_vehicle)
    print "Vehicle position is", veh_x, veh_y
    Jun=traci.junction.getIDList()
    JunS= len(Jun)
    VEH_on_Inter = 0
    PN=0
    # check if Vehicle near the Intersection (less than 1 meter) for Re-routing
    if JunS <1:
        for k in range(0,JunS):
            jun_x,jun_y=traci.junction.getPosition(Jun[k])
            Distance_Veh_Edg = traci.simulation.getDistance2D( jun_x, jun_y, veh_x, veh_y
, isGeo=False, isDriving=False)
        if Distance_Veh_Edg < 1:
            VEH_on_Inter = 1
        if VEH_on_Inter == 1:
            P_routes=traci.route.getSubscriptionResults()
            PN= len(P_routes)
            TIME= traci.simulation.getCurrentTime()
            VEH_on_Inter = 0
        for d in range(0,PN):
            P_routes_EDG=traci.lane.lanegetEdgeID(P_routes[d])
            Trav_tim = traci.vehicle.getEffort(sample_vehicle, TIME, P_routes_EDG[d])
            Curr_Trav = traci.vehicle.getAdaptedTraveltime(sample_vehicle, TIME,
P_routes_EDG[d])
            Trav_tim_sys = traci.vehicle.rerouteTraveltime(sample_vehicle, TIME,
P_routes_EDG[d])
```

```
            if Trav_tim < Curr_Trav:
                traci.vehicle.rerouteEffort(sample_vehicle)
            elif Trav_tim_sys < Curr_Trav:
                traci.vehicle.rerouteTraveltime(sample_vehicle)

        traci.vehicle.rerouteTraveltime(sample_vehicle)
    no = traci.inductionloop.getLastStepVehicleNumber("0")
    output.close()
    if no > 0:
        programPointer = (0 if programPointer == len(PROGRAM)-1 else 3)
    step += 1
    traci.close()
    sys.stdout.flush()
```