Integrated V2V Wireless Network and Vehicular Traffic Simulator Design

A Thesis

Presented in Partial Fulfillment of the Requirements for

the Degree Master of Science in the

Graduate School of The Ohio State University

By

Yalcin Balcioglu, B.S.

* * * * *

The Ohio State University

2009

Master's Examination Committee:

Prof.Fusun Ozguner, Adviser

Prof.Umit Ozguner

Approved by

Adviser
Graduate Program
Electrical and Computer
Engineering

ABSTRACT

Intelligent Transportation Systems (ITS) is an extensive area of ongoing research and development which has started to revolutionize the driving experience; initially with traffic management applications such as electronic toll collection and automated parking sites, but also with promises of radical changes. The ITS landscape is full of potential for innovative applications in areas such as safety, traffic management, and comfort, although certainly not limited to these. In addition, the evolution and general availability of inexpensive WLAN equipment and worldwide progress towards acceptance of a DSRC standard has brought these technologies front and center in the design and implementation of IVC/RVC applications such as collision warning systems.

This study introduces three integrated wireless and vehicular simulators, each focusing on different aspects of the VANET environment. While, the two vehicle one intersection simulator from Chapter 3 concentrates on a intersection access safety system testbed's real-time simulation to determine the unsafe maneuvers for a driver, in Chapter 4 a multiple intersection multiple vehicle simulator with real-time V2V communication and collision warning system is implemented. Moreover, Chapter 5 concentrates on a single intersection with multiple vehicles by incorporating the effects of the buildings in the propagation model and selectable MAC protocols again with a collision warning system.

This is dedicated to the one I love ... Bahar Yumrutasli and my family ...

ACKNOWLEDGMENTS

Many individuals contributed in different ways to this effort. I am very grateful to my adviser, Dr. Fusun Ozguner, for her support and guidance throughout the entire process, and to Dr. Umit Ozguner for his insight and for being part of the review committee. In addition, I am deeply indebted to Boangoat Jarupan for her valuable assistance and for dedicating a significant amount of time, effort, and energy teaching me. Also, I am thankful to Robert Murawski for our frequent discussions and for sharing his educational experience. Finally, I acknowledge the invaluable support of Dr.Keith Redmill, Kevin Streib, Arda Kurt and Scott Biddlestone for their work contributing to my studies. This research was partially supported by Honda Research America.

VITA

April 13, 1984 .... Born - Goslar, Germany

2007 .... B.S. Electrical and Electronics Engineering

2007-present .... Graduate Teaching or Research Associate,
The Ohio State University.

PUBLICATIONS

Research Publications

B. Jarupan, Y. Balcioglu, F. Ozguner, U. Ozguner, E. Ekici "An Integrated Wireless Intersection Simulator for Collision Warning Systems [19]". *ICVES*, 00(2):277–287, Sep. 2008.

FIELDS OF STUDY

Major Field: Electrical and Computer Engineering

Studies in:

Vehicular Communication Networks	Prof. Fusun Ozguner
Collision Warning Systems	Prof. Umit Ozguner

TABLE OF CONTENTS

Appendices:

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Intelligent Transportation Systems (ITS) is an extensive area of ongoing research and development which has started to revolutionize the driving experience. It has started with traffic management applications such as electronic toll collection and automated parking sites; and now ITS are promising radical changes. The ITS landscape is full of potential for innovative applications in areas such as safety, traffic management, and comfort, although certainly not limited to these. Possible applications range from dynamic vehicle routing, to downloading on-demand video, or even automatic collision avoidance. In fact, the notion of completely automated driving is being actively explored and may become a reality before long. Within this area, the concepts of inter-vehicle communication (IVC) and roadside-to-vehicle communication (RVC) have surfaced to categorize a variety of applications within ITS. Common to these variations is the need for vehicles to communicate, either to each other or to less mobile or completely static elements. Consequently, any time that communication between entities is required, there is a variety of options and possibilities to enable it. The evolution and general availability of inexpensive broadcast networking equipment (IEEE 802.11p) and worldwide progress towards acceptance of a dedicated short-range communications (DSRC) standard has brought these technologies front and center in the study and implementation of IVC/RVC applications. However, even though the physical layer specification may be converging, much effort and energy is being invested in developing or evolving a workable medium access protocol (MAC) for such applications since the current wireless local area network (WLAN) specifications are not explicitly designed for environments with highly mobile elements. Along these lines, development of the IEEE 802.11p standard [4] defining the physical layer and the medium access layer for vehicular environments has started and reached final phases of the standard release. This standard defines a variant of the CSMA based IEEE 802.11a MAC protocol with variable transmission priorities for the 5.9 GHz DSRC wireless channel to support a wide range of ITS applications, including collision warning systems with its seven 10 MHz channels.

This study includes design and performance analysis of three different integrated wireless and vehicular simulators developed at the Ohio State University. All of these three simulators investigate the vehicle-to-vehicle communication research area. While conceptually sharing some design considerations they all concentrate on different aspects of the vehicular communication environment. The simulators mainly differ in the number of intersections in the topology or by the maximum amount of vehicles that are simultaneously simulated.

The first one of these simulators is a two vehicle and single intersection simulator presented in Chapter 3. It is developed as part of a research project at the Ohio State University. Main purpose of the simulator is to replicate the behavior of the V2V safety system hardware testbed developed in this research project. A vehicular simulator and a wireless network simulator are integrated by modifying the "Two Vehicle Crash Simulator" developed at OSU and the NS-2 wireless network simulator.

The wireless safety units with IEEE 802.11p radios are tested for their RF range and delay performances. Moreover, the wireless network simulator's related parameters are empirically set to reflect the performance of these radios. Finally, the intersection access safety system application developed for the hardware testbed is successfully implemented on the software testbed, and its operation is demonstrated.

The second integrated wireless and vehicular simulator is presented in Chapter 4. It is a multiple vehicle and multiple intersection simulator developed for analyzing micro-traffic networks as a research project within Center for Automotive Research Vehicle-to-Vehicle Communication Consortium. The main purpose of this simulator is to analyze the performance of the IEEE 802.11p standard performance for dense networks and implement a collision warning system utilizing V2V communication. A vehicular network simulator developed at The Ohio State University is modified and integrated with the modified NS-2 wireless network simulator. The design of the collision warning system, driver model and the performance of the IEEE 802.11p MAC layer is carefully presented with results verifying the design goals.

The third and final integrated wireless and vehicular simulator is presented in Chapter 5. It is a single intersection and multiple vehicle simulator developed as a research project within Center for Automotive Research Vehicle-to-Vehicle Communication Consortium. The main target of this simulator is investigating the effects of buildings and a dedicated repeater on V2V communication performance and comparing three different MAC protocols. Also, a collision warning system application is implemented along with a driver model to analyze the vehicle traffic.

This thesis gives background information and presents related work in Chapter 2 and finally summarizes and discusses the whole study in the final chapter.

CHAPTER 2

BACKGROUND AND RELEVANT WORK

2.1 Background on Intelligent Transportation Systems

Intelligent Transportation Systems (ITS) continues to receive significant worldwide attention and investment from academic circles, research communities, public agencies, and private corporations. ITS encompasses a wide range of applications and technologies whose goal is to improve traffic monitoring and management capabilities, as well as enabling vehicle and occupant safety, comfort, productivity, and efficiency. [12] ITS is a vast area of research and development which includes the following application categories: advanced traveler information systems, commercial vehicle operations, advanced vehicle control systems, electronic toll and traffic management systems, advanced public transportation systems, and advanced transportation management systems. [3] Advanced traveler information systems seek to distribute pertinent information to vehicle occupants. The intent is to provide travelers access to weather and road condition data, accident, route, and congestion information, as well as comfort-related information, such as service station location or restaurant directions. Another important goal of such systems is dissemination of vehicle emergency communications.

Advanced vehicle control systems integrate vehicle communication and automatic controls to enable collision warning systems and responses in which the driver is notified of dangerous conditions. Beyond notifications, some applications intend to enable vehicles to react appropriately to hazardous conditions through the use of devices such as intelligent cruise control and automated braking or steering systems. The primary focus of this area is to increase vehicle safety and thereby reduce congestion due to accidents.

Advanced public transportation systems have the objective of improving the efficiency and utilization of such facilities. Primary efforts investigate improving scheduling and communication, particularly for intra-modal and inter-modal systems. Another goal of this group is minimizing the response to emergency situations or vehicle outages.

Advanced transportation management systems aim to dynamically monitor and control traffic systems, including real-time data collection, to predict and respond to congestion and accidents. Ultimately, the goal is to increase and maintain traffic efficiency on highways and roads. Substantial efforts in these categories have yielded mature and commercially available applications; for example, electronic toll collection (ETC) solutions exist and continue to be deployed around the world. [1]

2.1.1 Inter-Vehicle Communication(IVC) and Road-to-Vehicle Communication(RVC)

Communication is an essential requirement of any ITS application. This area is continuously evolving as new and improved physical and medium access control layer specifications are developed and standards are approved. Given the wide range of applications, communication requirements among and within categories vary significantly; however, it is important to understand how these communication channels are used within ITS.

ITS applications rely on wired and wireless technologies to establish communication paths between data sources, sensors, vehicles, and other application elements. These communication mechanisms are generally divided into two groups: inter-vehicle communication (IVC) and road-to-vehicle communication (RVC). As their names suggest, IVC entails the communication between two or more vehicles, whereas RVC involves the transfer of information between a vehicle and a fixed entity.

RVC efforts are generally focused on traffic monitoring and management services. In addition, other services such as establishing mobile voice communication links, enabling satellite broadcasts, and supporting large-scale data downloading are being explored. [28]

Likewise, IVC has a large set of ongoing efforts and possible applications. These are generally divided into three categories: communication based longitudinal control, co-operative assistance systems, and information and warning functions. [17] Communication based longitudinal control applications leverage IVC to improve road utilization; for example, to enable convoy driving or platooning. Such control applications could eventually enable automated driving. Generally these applications do not impose very high bandwidth requirements; however low latency, increased reliability, and security are key elements for applications in this category. Co-operative assistance systems aim to improve vehicle safety by communicating time-critical information to nearby vehicles in specific environments, such as urban intersections without traffic signals and highway on-ramps. While some notifications may be strictly informational, such as security distance warnings or passing assistance, when coupled with intelligent vehicle controls, these applications could actively engage the driver and the vehicle; for example, by performing emergency braking.

Information and warning functions encompass the ability to transmit road, vehicle, and traffic conditions among vehicles. Beyond the safety aspect of such communication, this category also includes applications to improve passenger comfort or entertainment, such as live chat and video, or enabling online gaming. The latter purposes have typically less stringent latency or delivery requirements; however, they generally require significantly higher bandwidth.

2.1.2 Dedicated Short Range Communications

Infrared and radio waves have been studied as medium possibilities for IVC and RVC. Infrared and millimeter radio waves are generally limited to directional line-of-sight communication; however, this limitation has not prevented the use of infrared waves for some IVC applications, specifically co-operative driving. On the other hand, VHF and micrometer radio waves perform broadcast propagation. [17]

Broadcast propagation is an important communication property for IVC and RVC, where moving nodes are constantly changing location and direction and where obstacles such as other vehicles and buildings create the possibility of undetectable nodes. Broadcast media allow communication between elements through direct links, provided there is a line-of-sight path or multi-path propagation paths with little delay dispersion. The communication range issue can be addressed by leveraging a multi-hopping protocol, making it possible to establish a connection between elements that are outside of their respective radio ranges. [17]

Within the microwave spectrum, the dedicated short range communications (DSRC) service has been at the heart of a number of research efforts and projects. DSRC is defined as a short to medium range communication service operating within a 75 MHz spectrum at 5.9 GHz. DSRC complements cellular technology by providing very high data transfer rates as well as minimizing latency and isolating relatively small communication zones. By leveraging a multi-channel operation model, DSRC allows multiple users simultaneously on limited frequencies. In the United States, the frequency range has been divided into six service channels and one control channel.

The DSRC standard has been progressing toward worldwide acceptance, including processes in Japan, European Union, and the United States and has become a critical, enabling technology for IVC and RVC.

2.1.3 Medium Access Control(MAC) Protocols

Medium access control (MAC) protocols are another critical development area for RVC and IVC. While the wireless medium allows for simultaneous transmissions among multiple users on different channels, MAC protocols specify how elements on the medium interact with other elements leveraging the same medium, essentially defining how wireless traffic is managed. Two distinct protocol families offer advantages and disadvantages for this medium. These are carrier-sense multiple-access (CSMA) and code division multiple-access (CDMA) protocols. The first family includes a variety of wireless local area network protocols (WLAN), including the IEEE

802.11 group of specifications, whereas the CDMA family includes the third generation (3G) extensions to cellular communication standards.

3G CDMA offers dedicated channels for transmission and flexible radio resource allocation; however, this is provided through a centralized coordinating entity, an assumption that cannot be made about IVC environments. In the absence of such centralized infrastructure, CDMA protocols suffer from resource allocation management problems, inadequate power level controls, and time synchronization management. [17] WLAN protocols offer an inherently ad-hoc, distributed communication capabilities ideal for loosely coupled environments. However, it is not well suited for highly dynamic mobile environments since it lacks the ability to robustly handle multi-path effects or synchronizing time between mobile nodes. Further, it cannot guarantee resource allocation fairness in a constantly changing topology.

These challenges are significant and have led some researchers to propose combining features of both protocols to leverage their advantages. Others have proposed systems in which a single vehicle possesses the ability to transmit with both protocols and performed handoffs as needed.

Within the WLAN realm, it has been demonstrated that carrier-sensing protocols are far superior in reducing channel collisions than non-carrier sensing ones. While synchronization techniques also yield similar reductions in channel collisions, they are harder to implement without a coordinating entity. It has also been demonstrated that retransmitting packets is an effective technique to increase information delivery rates over standard 802.11 results.

In addition, much attention has been given to optimizing broadcast and multi-hop heuristics within MAC protocols to improve channel utilization by reducing unnecessary transmissions but also extending the range of a given transmitter.

2.2 Relevant Work on Integrated Wireless Network and Vehicular Simulators

An intersection collision warning system uses ITS technologies to prevent collisions and reduce their severity. A warning system deployed on a vehicle monitors the movement of neighboring vehicles to send out alert signals to drivers on possible collisions. To investigate the performance of such systems in a cost efficient manner, development of integrated simulators is necessary. An integrated simulator is a complete software architecture consisting of both a traffic simulator and a wireless network simulator. In this thesis, the complete architecture of the integrated wireless and vehicular simulator (*Integrated Wireless Intersection Simulator*(IWIS)) is presented; and it has been developed since 2003 [5, 7].

Recently many researchers have proposed integrated simulators such as VanetMobiSim [10], ISP [11], HiTSim [29], and the simulator in [13]. VanetMobiSim uses the Advanced Intelligent Driver Model (AIDM) for traffic generation and can use different network simulators such as ns-2 [18], GloMoSim [9], or QualNet [22]. This simulator studies different type of traffic behaviors. However, it does not focus on a collision warning system model. Similar work is being reported by Toyota Information Technologies [11] aiming to develop a simulator platform(ISP) with the ability to integrate vehicular traffic simulations and wireless network simulations. ISP studies V2V and V2I communications, and it can simulate multiple streets and intersections. ISP only models 802.11 in its MAC layer whereas IWIS has the capability to study a number of different MAC protocols. Young and Chang introduced HiTsim [29] which incorporates a collision avoidance system. It is similar to the work done in this thesis in the sense that the traffic simulator is not collision free and the way it incorporates the levels for generated collision avoidance warning messages. In addition, it uses the well-known ns-2 network simulator in a similar manner. HiTSim does not only generate random traffic but can also load input traffic scenario files. However, HiTSim can simulate only highways and it has a simple propagation model which does not model communication channel related issues such as shadowing, blocking and buildings. Kerner, Klenov, and Brakemeier [13] also try to overcome the simulation time scale problem by doing offline simulations. However this new test-bed only implements a simple two-ray ground radio propagation model.

CHAPTER 3

IEEE 802.11P BASED VEHICULAR SAFETY SYSTEM HARDWARE TESTBED AND SIMULATOR

3.1 Overview of the Vehicle-to-Vehicle Communication Safety System

3.1.1 Introduction

The Ohio State University is distinguished from the other universities by the amount of research done in automotive industry. Intelligent Transportation Systems Laboratory is one of the several laboratories in the Center for Automotive Research. This laboratory is mainly focused on the implementation of intelligent transportation applications such as autonomous vehicles competing in the DARPA Challenges, active safety systems using sensors like Radars and LIDARs, vehicle-to-vehicle or vehicle-to-infrastructure communication, development of vehicular simulators, and intersection collision avoidance systems. Since 2008, a research project about an active safety system, based on vehicle-to-vehicle communication, has been ongoing in the Intelligent Transportation Systems Laboratory in the Center for Automotive Research Center. This project is funded HONDA and the outputs of the project are going to be the main source and framework for the future vehicles of this company that are equipped with the vehicle-to-vehicle active safety systems. The project has been realized by the efforts of a group of students. While problem statement and the definition of the project will be comprehensively explained for completeness, the modules of the project done by the other students are not going to be explained in detail in this chapter. Starting from section 3.2 my individual contribution to the project is written.

3.1.2 The Problem Definition

Today's world, is unpredictable; there are many unpleasant events such as traffic accidents that could happen any minute and change our lives forever. They are unexpected and unfortunately they have sudden vital and monetary implications. Although the drivers are trained to be aware, human beings always make mistakes. Moreover, for instance, human perception of the environment can be easily degraded with the weather conditions. Therefore, assuming that the traffic accidents are unavoidable, it is reasonable to utilize technological advancements for the well being of the mankind thereby making the roads as safe as possible. One of the safety system categories adopted by the automotive companies is the active safety systems category. This category includes applications utilizing the vehicle-to-vehicle communication and/or the various types of sensors along with the CANBUS and the GPS modules of the vehicles.

The major fatal crash scenarios at intersections in the USA stem from the five different scenarios which are illustrated in Figure 3.1. This project aims to find a solution which would help reduce both the number of collisions by warning the driver and the cost of the crashes that will still occur in the presence of the collision warning system. Achieving this ambitious goal is not easy as it is inherently based on the interactions between the drivers and personality of the drivers.

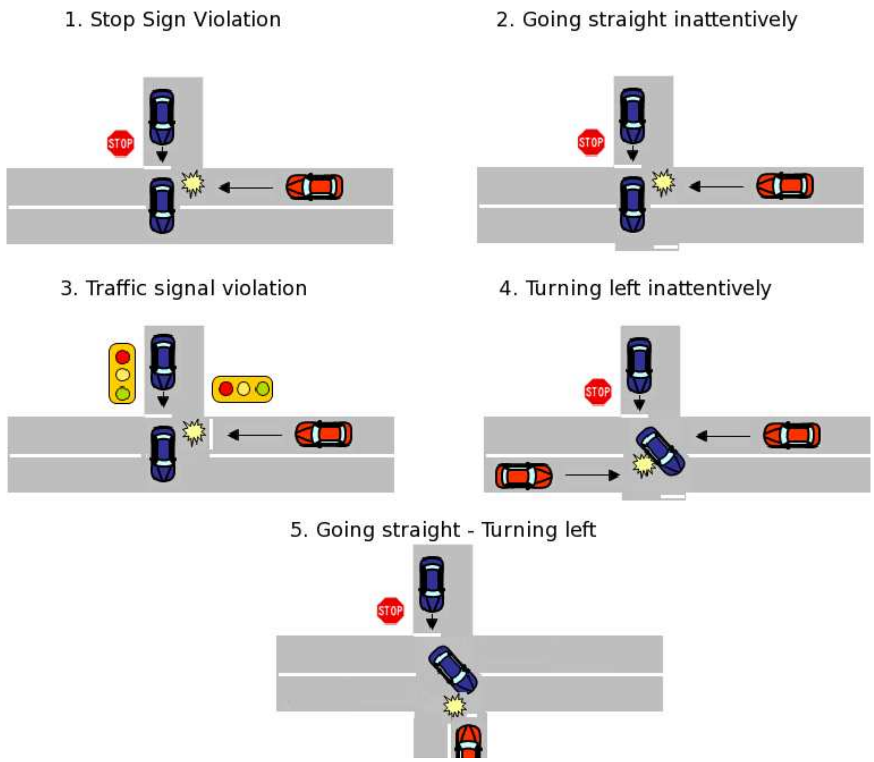

Figure 3.1: Major fatal crash scenarios in USA

The project done at The Ohio State University mainly focuses on the accident scenarios 2, 4 and 5. The host vehicle(HV) is defined as the observed vehicle and the target vehicle is the one interacting with the observed vehicle. The research is focused on two topics. The first one is the relation between time-to-intersection and the estimation accuracy (EA) of driver state to evaluate estimator performance [26]. This first topic investigates the possibility of estimating more accurate time-to-intersection(TTI) by using the target vehicle driver state estimator. The second topic is to research the crossing path probability in case of using a normal GPS based on a relative position estimation.

The outline of the hardware framework consists of vehicles equipped with IEEE 802.11p prototype radios along with a high precision GPS module, a low precision GPS module, and connection to CANBUS of the vehicles. This hardware framework connects the hardware units to each other and allows the applications to run by providing critical information about the host vehicle and the target vehicle.

The application layer of the framework requires the development of a driver state estimator. This estimator is supposed to keep track of the host vehicle's and the target vehicle's driver behavior. The development of the driver state estimator is based on the improvement of a previous research study done at The Ohio State University [15]. As one of the project milestones, the target vehicle and the host vehicle driver state estimator model is developed based on the ideas from this preliminary work on the driver's decision model for intersections.

The driver state model, developed for the collision warning system, was tested for various intersection scenarios based on the requirements defined by the research project. These test scenarios are composed of three main components as shown in the following list:

- *Test Scenario 1 - Target vehicle driven on the main road and host vehicle on the secondary road at a right angle.*
- *Test Scenario 2 - Target vehicle and host vehicle in opposite directions on a road and the target vehicle intended for a left turn.*
- *Test Scenario 3 - Target vehicle driven on the main road and the host vehicle merging to the main road from a secondary road.*

The test procedure for these three intersection scenarios was developed by spanning the possible moves that the vehicles possibly would be able to do. This was achieved by generating comprehensive test cases for the test scenarios. The main test scenarios are given in the Figure 3.2.

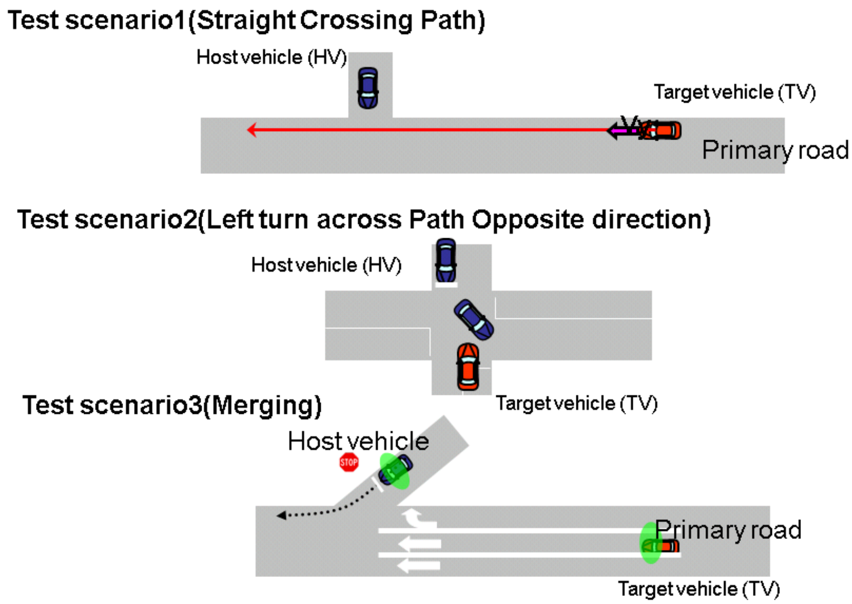

Figure 3.2: Test Scenarios in the intersection

The test scenarios one and two are divided into test cases to span the possible scenarios as tabulated in Table 3.1

Driver state estimator model is developed to predict the behavior of the target vehicle and warn the host vehicle driver about which possible maneuvers would be dangerous depending on the predicted behavior of the target vehicle. Arrows with changing colors and shapes on the dashboard supported with audio aid constitute the driver level user interface of the V2V Warning System as shown in Figure 3.3.

Warning Status Indicators for visual warning alerts are as shown below:

- *Green- all maneuvers are OK (i.e. approaching vehicle is stopping)*

Case	Vehicle Orientation	TV Maneuver	HV Maneuver
1.1	perpendicular	45mph to stop	stationary
1.2	perpendicular	45mph to go straight	stationary
2.1	parallel opposite direction	35mph to stop	stationary
2.2	parallel opposite direction	35mph to go straight	stationary
2.3	parallel opposite direction	35mph to 10 mph right turn	stationary
2.4	parallel opposite direction	35mph to stop	25mph to stop
2.5	parallel opposite direction	35mph to go straight	25mph to stop

Table 3.1: Test Cases selected to test the Vehicular Safety System.

- *Yellow- OK to proceed ahead (i.e. oncoming vehicle is turning right)*
- *Red- Stop (i.e. approaching vehicle is going through intersection)*

Driver state estimator model is the one which generates the predicted states of the target vehicle based on the information it takes from the hardware framework. Critical data such as latitude, longitude, heading, yaw rate, velocity, turn signal indicator, brake pedal indicator, gas pedal indicator, and steering wheel position enable the algorithm of the driver state estimator model to predict the next move of the target vehicle. These variables are conveyed using WSM messages through the V2V communication framework and will be discussed in more detail in the upcoming sections of this chapter. Driver state estimator model is developed by OSU and will be presented in summary here for completeness.

Driver state estimator model is developed as a finite state machine driver model which has evolving probabilities to make state transitions over time. This model is generated using Viterbi Algorithms and Trellis graphs. The state machine uses certain threshold values and events to capture state transitions. The finite state machine level user interface of the warning system user interface is illustrated in Figure 3.4. The

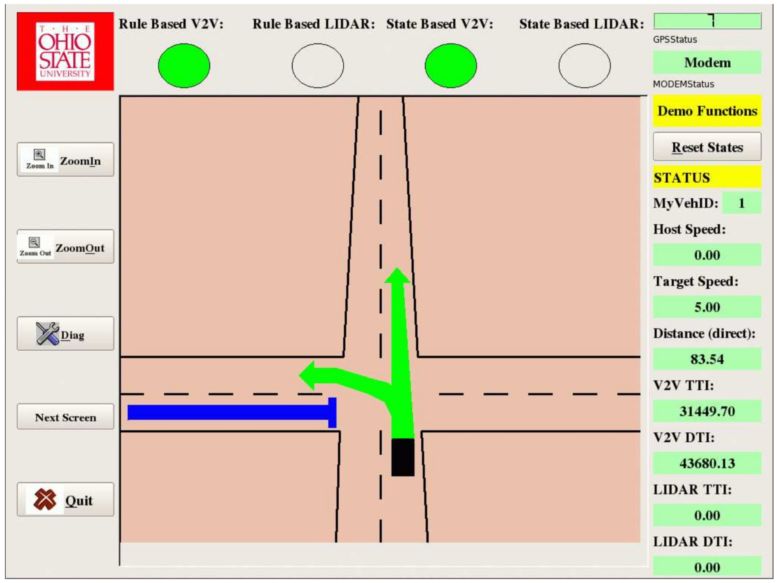

Figure 3.3: Driver Level User Interface of the V2V Warning System.

colors in the states indicate the following; the purple is the actual state for the target vehicle while the light green is the predicted state of the target vehicle. The green or possibly red circles on top notify the safe and unsafe straight driving behavior respectively. The threshold values and the events that create the state transitions are as follows:

- *S0:Car within range*
  S0->S1: Distance to TV <120m
- *S1:Go through*
  S1->S4: Distance to TV is increasing
  S1->S3: TV speed <20 mph
- *S2:Accelerate*
  S2->S1: TV keeps accelerating for n (2-3) updates
  S2->S7: TV speed <15 mph AND TV long. acc. <0.2 m/s2
- *S3:Slow I*
  S3->S2: TV long. acc. >0.2 m/s2
  S3->S7: TV speed <15 mph
- *S4:Exit*
  No transitions
- *S5:Stop*
  No transitions
- *S6:Turn*
  S6->S4: Distance to TV is increasing (TV is getting away)
- *S7:Slow II*
  S7->S2: TV long. acc. >0.2 m/s2
  S7->S6: TV yaw rate >x for y updates OR turn signal on OR steering wheel angle >z

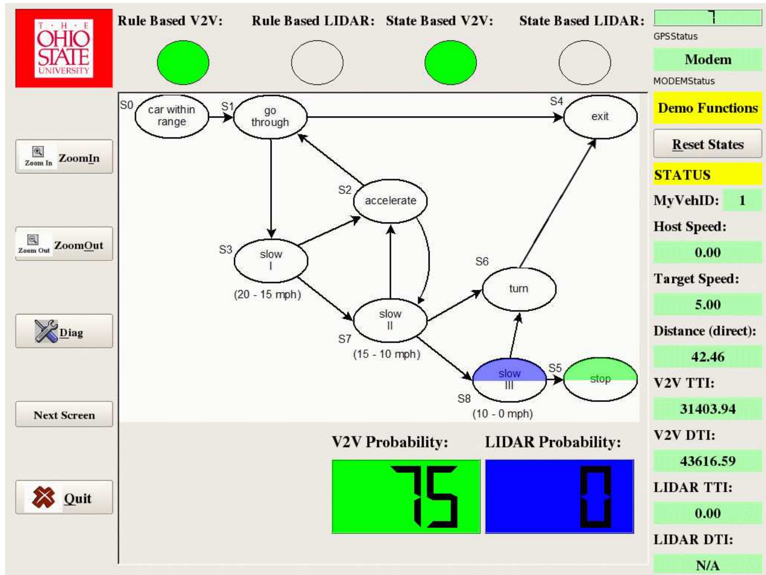

Figure 3.4: FSM Level User Interface.

S7->S8: TV speed <5 mph

- *S8:Slow III*

  S8->S6: TV yaw rate >x for y updates OR turn signal on OR steering wheel angle >z

  S8->S5: TV speed <1 mph

The research project requires the performance investigation of the driver state estimator model. In order to achieve this goal, the figure of merit is defined as the ability of the estimator to predict the target vehicle's stop intention from further distances. Results from the test runs done in real life are presented by the PhD student Arda Kurt in OSU. Moreover, the replication of these real life runs are done with the simulator designed in this thesis and presented in the upcoming sections of this chapter.

The project includes the investigation of the error sources emerging from both the V2V Communication system, the measurements done by the vehicle and the GPS module. Although, the radio communication does not introduce error to the content of the Wireless Safety Messages(WSMs), it acts as an error source in the sense that it might delay the information or even completely fail the communication which could be due to several different reasons. While speed, heading, latitude and longitude errors are the errors emerging from the measurements and degrading accuracy, the communication delay (milliseconds), communication range (100s of meters), frame error rate as a function of SINR and fast fading are the factors that can abruptly affect the performance of the vehicular safety system. In order to quantify the effects of the radio performance of the vehicular safety system, several radio performance measurements are required to observe the range and delay performance of the 802.11p wireless radios. The basic radio performance analysis of the prototype radios is done with parameters such as TX power, packet size, and the data rate adjusted as they are defined in the IEEE 802.11p standard.

Figure 3.5 shows the map level user interface of the V2V safety system, where one can see the host vehicle and the target vehicle moving along with details such as distance to intersection, time to intersection, and target vehicle intention. The upcoming sections of this chapter continue with the detailed explanation of the hardware testbed, radio performance measurements, and replication of test case runs in the simulator.

3.2 IEEE 802.11p Prototype Radio Performance Measurements

US government and automotive industry in general have been supporting the idea of incorporating the broadband wireless devices to establish active vehicle safety systems. After the 5.9GHz DSRC band has been introduced by FCC in 1999, the pace for

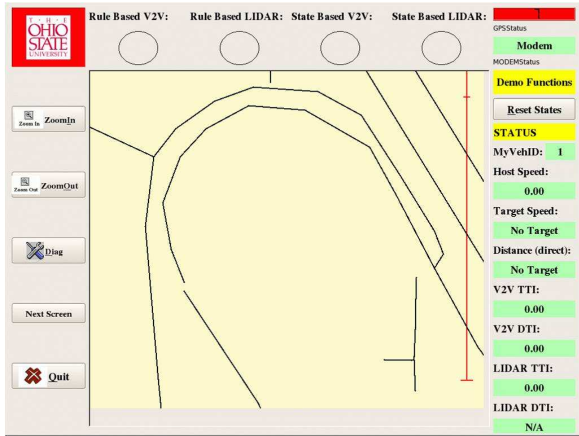

Figure 3.5: Map Level User Interface of the V2V Safety System.

the development of the V2V safety applications increased and the standards defining the layers of the Wireless Access for Vehicular Environment(WAVE) stack started to get shaped and became clearer. While WAVE includes sophisticated capabilities, the basic idea is to provide wireless links up to 1000m at high mobility environments. The dedicated band provides seven channels in a 75 Mhz band, defining the designated applications for each channel, specifically one being the safety channel reserved for vehicle to vehicle communications.

Wireless links between the vehicles establish the foundation for a broad range of Intelligent Transportation System(ITS) applications. The framework has already given rise to the new safety application ideas such as V2V collision warning and avoidance systems, emergency braking in high speed environments, and intersection collision avoidance.

The WAVE protocol stack has its lower layers such as MAC and Physical Layer governed by the IEEE 802.11p standard. This standard is similar to the IEEE

DSRC Specification	DSRC available values	Tested values
DSRC channel	Ch.172,174,176,178,180,182,184	Channel 178(V2V safety)
TX output power level	0,10,20,29 dBm	20 dBm(DSRC Default)
Packet Size	64B-1000B	100B
Packet Generator	Periodic, Distance or Event based	Periodic (100ms)
Data rate	3,4.5,6,9,12,18,24,27 Mbps	6 Mbps(Default)
Tx Priorities	level 0-to-3	priority 2

Table 3.2: DSRC Specifications for radio performance tests.

802.11a in physical layer while it incorporates the Enhanced Distributed Channel Access(EDCA) algorithms from IEEE 802.11e in the MAC layer to differentiate the priorities of the Wireless Safety Messages(WSMs). Appendix A gives information about the structure of WSM messages in detail. Most of the academic work on this topic have exploited the similarities of the 802.11p to 802.11a and researched the topic for a fixed priority and single channel for hardware testbeds and network simulators. In this thesis, I was fortunate enough to investigate the WAVE environment using a prototype 802.11p radio and do simulations for the 802.11p environment.

The prototype radio used by OSU V2V Research group is a 5.9 Ghz DSRC radio module. It works on the dedicated RF spectrum for ITS applications and has low latency for vehicle to vehicle communication, thereby successfully providing wireless link service to high speed vehicles. DSRC specifications are defined in the ASTM E2213-03 and tabulated in Table 3.2 along with the parameter values set for the field tests done in OSU.

3.2.1 Field Test Setup and Measurement Results

In this section field test setup, hardware testbed architecture, field test goals are discussed. As required by the research project, the field test goals are set to be the determination of the average delay, transmission range and the packet error rate(PER) for the basic safety message applications. The packet error rate is acceptable as long as it is smaller than 10% as it is defined in the standard. Therefore the Tx-Rx range is defined by the breakdown point when the packet error rate goes above the threshold defined in the standard. The field tests include only the line of sight conditions.

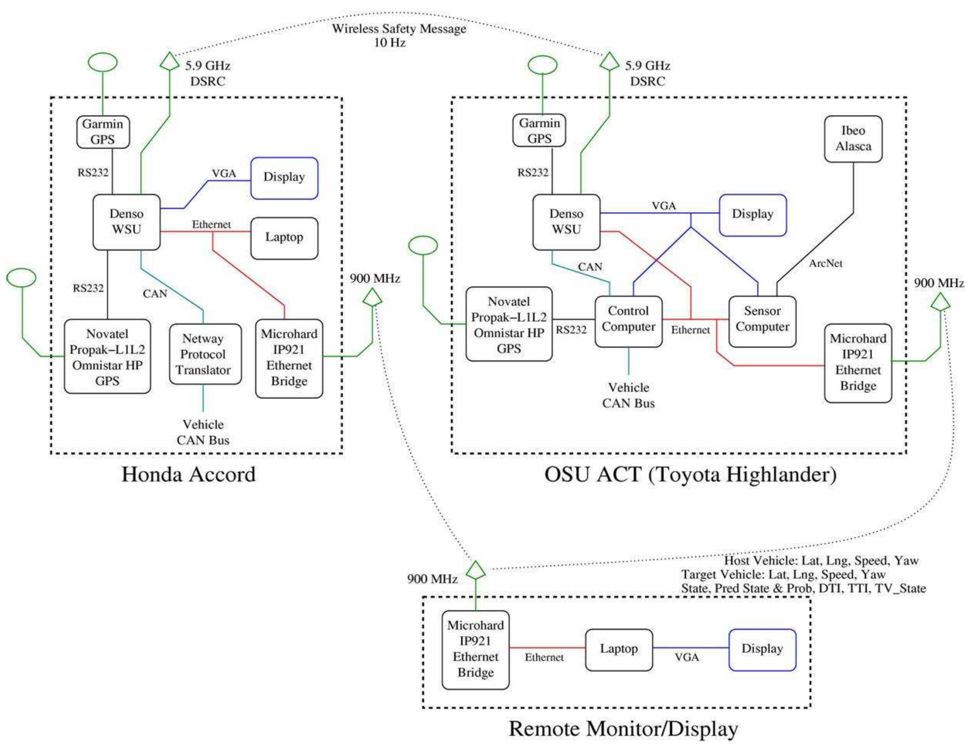

Figure 3.6: Hardware testbed system diagram.

Hardware testbed framework consists of 3 main elements being GPS module, DSRC radio module, and CANBUS connection. Two test vehicles are equipped with necessary equipment to power and connect the components of the framework and laptops are placed to control the system. High precision GPS modules with WASS correction and off the shelf cheap GPS modules are connected for comparison. The hardware testbed has its data center as the DSRC radio module which is actually a minicomputer. It collects the incoming data from GPS and CANBUS so that the applications running on the DSRC safety unit can access the real-time information about the host vehicle. Omnidirectional single DSRC antenna is utilized with the wireless safety units. Figure 3.6 shows the system diagram for the hardware testbed.

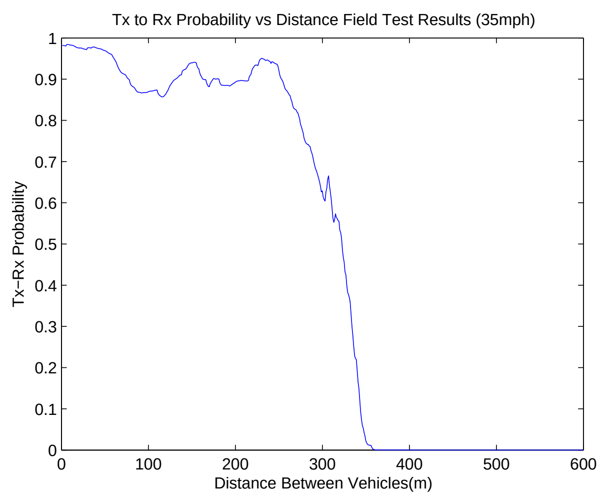

Figure 3.7: Packet transmission probability vs. vehicle separation(35mph).

Field test area is chosen to be Transportation Research Center(TRC) in Ohio. This research center has extended LOS tracks longer than 1000m which allows the execution of successful field tests. The vehicles are set up to broadcast periodically

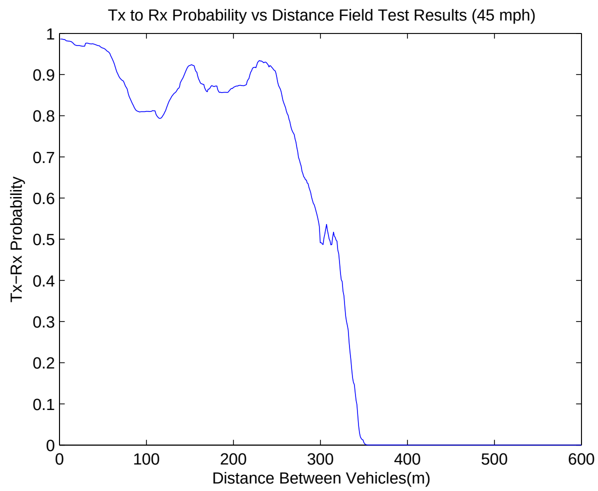

Figure 3.8: Packet transmission probability vs. vehicle separation(45mph).

with 10Hz frequency and no routing is done. In other words, every vehicle receiving a packet logs the data and drops the packet without forwarding it. The field tests are done in two different speeds which are 35mph and 45 mph and with both vehicles moving towards each other. The measurements are avaraged using a 5m moving window for smoother results and to account for the possible GPS errors. Figure 3.7 and Figure 3.8 show the LOS field test results by illustrating the packet transmission probability vs. vehicle separation for vehicles moving at speeds 35mph and 45mph respectively. From the figures, it can be deducted that at higher speed, 45mph, the packet error rate is slightly higher compared to the slower speed case as expected. One can see that the packet transmission suffers slightly more for vehicle separations of approximately 100m and 230m. This can be explained with the propagation characteristics of the channel and possible data collection mistakes such as insufficient samples. As the project funding allows a limited usage time for the tracks in Transportation Research Center, it is considered as future work to gain a better understanding of these performance reductions at specific vehicle separations.

Speed	TX-RX Range	PER in range	Mean delay	Delay variance
45mph	approx. 300m	10%	21.3msec	0.45msec
35mph	approx. 300m	10%	21.5msec	0.44msec

Table 3.3: Results for radio performance tests.

Determination of the radio range at 20dBm output power level is achieved by selecting the shoulder of the graph where the packet error ratio starts increasing rapidly. In the field tests, range is seen to be approximately 300m after which the packet error ratio goes above 10% forever. Table 3.3 shows the end-to-end mean delay and delay variance for different speeds including the processing time of the incoming packets. One can observe that the delay and range performance do not vary considerably for the selected speeds. The basic radio performance field tests are useful to have an idea about the capabilities of the V2V safety system while the cars are moving on the urban roads. Results of the field tests are going to provide a basis for tweaking the transmission performance envelope of the wireless simulator.

3.2.2 Modification of NS-2.33 Wireless Network Simulator

The Ohio State University ITS research group has been developing vehicular simulators and integrating wireless network simulators with the vehicular simulators in the last decade. [5, 7] Simultaneously with the studies done towards hardware testbed development for cooperative active safety systems in vehicular environment, simulators integrating vehicular simulators and wireless network simulators have been developed. One of these integrated wireless intersection simulators is the one developed for the reproduction of the test scenarios that have been demonstrated to HONDA. While the details of this simulator and the results of the test runs will be explained in the next sections, the modification done to the wireless simulator NS-2.33 [18] to match the propagation characteristics of the DSRC channel and the wireless safety unit to the propagation channel of the wireless simulator will be explained in this section.

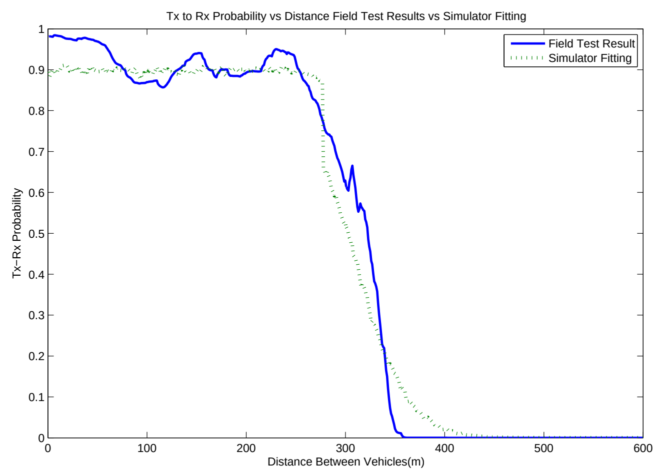

Figure 3.9: Packet transmission probability vs. vehicle separation comparing field test results and simulator fitting(35mph).

The NS-2.33 network simulator has the Nakagami propagation model available including options that let you adjust the parameters depending on the type of the road on which the vehicle is cruising. Moreover, Nakagami propagation model implemented in NS-2.33 allows the user to add power fluctuations to the calculated received power in order to account for the fast fading characteristics of the environment. During the process of modifications on the wireless network simulator, the parameters of the model are selected for a fairly high speed municipal road with a speed limit of 45mph and a gaussian distributed power noise is added to the calculated received power of the Nakagami propagation model. The mean and the variance of the power fluctuations are tweaked empirically until the Packet Error Rate vs. Vehicle separation envelopes of the wireless simulator and the field tests are matched. Figure 3.9 illustrates the empirical fitting done to match the performance of the simulator to the field tests.

Results of the V2V transmission delay measurements of the field tests are not incorporated into the simulator modifications as the measurements include processing delay which is prone to change in every system. Transmission delay of the simulator is a basic transmission delay resulting from the chosen header length, packet length, and the data rate. As these parameters are chosen equal to the values used in the field tests, basic communication delay is supposed to be similar to the actual communication delay of the field tests.

The bit error ratio and the frame error ratio algorithms are added to the simulator in pursuit of generating more realistic simulation results when interference from third party nodes are present. Currently, the field tests are done using only two wireless safety units(WSU). Therefore, it is considered a future work to make field test measurements in the presence of interference and modify the parameters of the simulator to reflect the DSRC environment. Currently, BER and FER algorithms are executed based on theoretical knowledge and will be explained in greater detail in Chapter 4.

3.3 Simulation of the Test Scenarios

Simulation studies in the vehicular network area are crucial. In order to do field tests with more than a few number of vehicles, one needs to spend a considerable amount of funds. Field test area rental cost, driver cost, and the hardware cost are significantly high. Therefore, it is desirable to simulate the active safety system designed for the research project. Simulator developed for this project targets to reproduce the field test demonstrations done to HONDA. As there are only two vehicles in the test scenario demonstrations, the simulator also has only two vehicles. The upcoming phases of the project will incorporate field tests with several vehicles. Therefore, the simulator developed in Chapter 4 is targeted for several vehicles. In this section, the architecture of the simulator designed for scenarios with two vehicles and the results of the test cases will be explained in detail.

3.3.1 Simulator Architecture and Design

Field tests are done with vehicles which have the CANBUS, GPS modules and 802.11p wireless radios interconnected. On top of this hardware testbed, the intersection access safety system application runs using the critical information about the host and target vehicle.

Desired simulator is supposed to execute the same intersection access safety system application by simulating both the vehicles with a vehicular simulator and the 802.11p radios with a network simulator. This approach results in executing three processes: the safety application, vehicular simulator, and the network simulator using a shared memory structure. The proposed simulator architecture diagram is shown in Figure 3.10. In order to achieve this goal, the vehicles are simulated by a

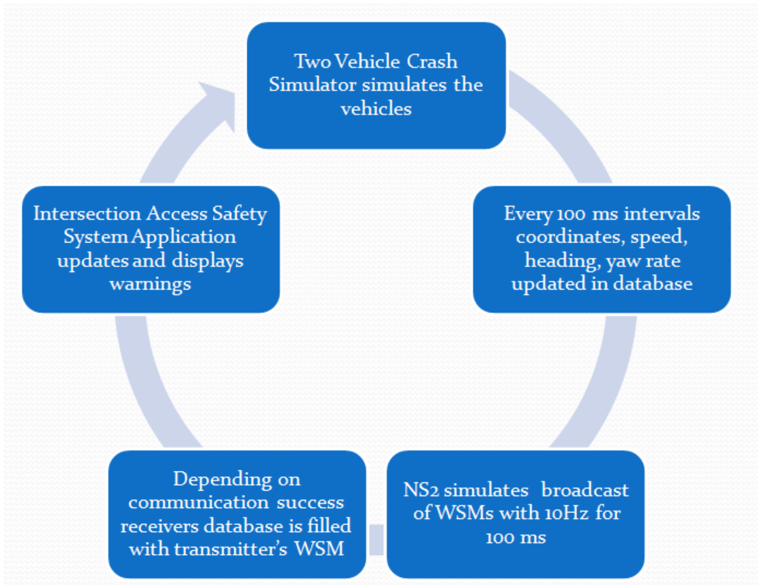

Figure 3.10: Simulator Architecture Diagram

simulator developed in Ohio State University named "two vehicle crash simulator". This simulator is capable of simulating two vehicles approaching an intersection in many different configurations. It is very well suited to replicate the behavior of the vehicles' maneuvers during the test scenarios. The simulator is modified to provide information such as latitude, longitude, speed, heading, and yaw rate to the shared memory structures which act as a real-time database for the simulator architecture. Figure 3.11 illustrates the vehicular simulator process.

802.11p radios are behaviorally simulated using the modified NS-2.33 which is explained in the previous sections. The wireless network simulator also runs as a process of the simulation environment and broadcasts the safety messages(WSMs) with 10Hz frequency. Upon receiving a WSM from the other vehicle, it fills the appropriate structures in the real-time shared memory database which enables the

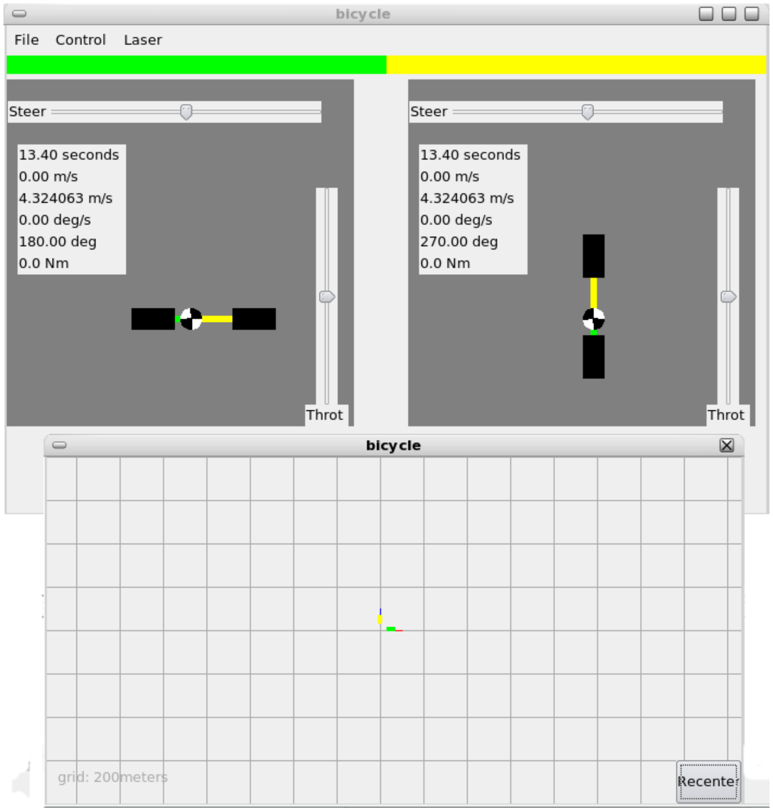

Figure 3.11: Two vehicle crash simulator

safety system application access to the crucial information about the target vehicle to run the estimators and the state machines. The vehicular simulator makes a location update of the vehicles in every 100ms, and the network simulator is triggered to run for 100ms every time a location update is done. When the communication simulation is finished, network simulator sleeps until the vehicular simulator triggers a communication simulation again. This algorithm enables the network simulator be synchronized in time to the vehicular simulator which runs in real-time. Utilizing the simulator, test cases which are demonstrated in real life, are reproduced and will be explained in detail with results in Section 3.3.2.

3.3.2 Reproduction of Test Scenarios in the Simulator

Test cases defined in Table 3.1 are reproduced using the simulator framework which was explained in section 3.3.1. In order to do this, vehicles are driven using the two vehicle crash simulator by the user to mimic the maneuvers of the host vehicle and the target vehicle depending on the test case. The output of the intersection access safety system application is recorded, and the Figures 3.12, 3.13, and 3.14 are generated via the output of the safety application.

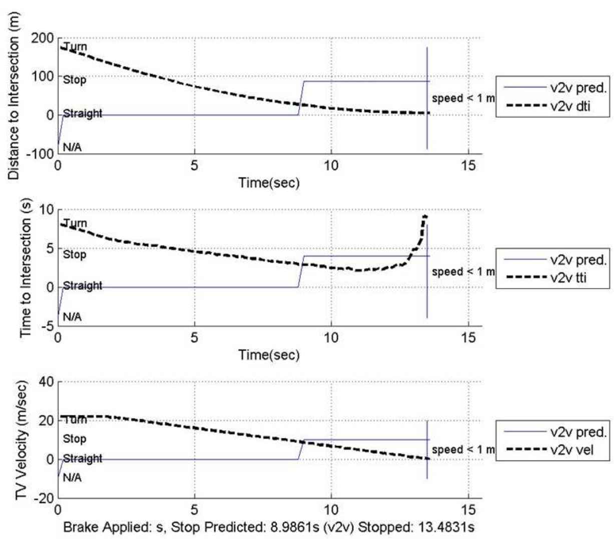

Figure 3.12: Results from the Reproduction of Test Case 1.1

Test case 1.1 consists of two vehicles in an intersection access scenario. Target vehicle approaches the intersection with 45mph speed and stops right before the intersection while the host vehicle is located perpendicular to the target vehicle and stationary right before the intersection. The nonlinear trace in the figures identifies the estimator state which can be stop, straight, turn, and N/A, showing the predicted estimator state transitions. The figures also provide the distance-to-intersection(DTI), time-to-intersection(TTI), and velocity of the target vehicle. From the figure 3.12, one can observe that the difference between the time point that the brake is applied and the time point that the car stopped is the time in advance the host vehicle driver is aware of the stop intention of the target vehicle. For test case 1.1, this is approximately 4.5 seconds.

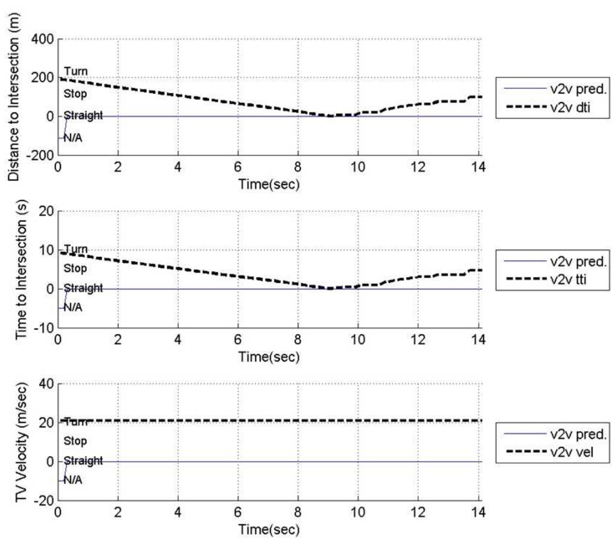

Figure 3.13: Results from the Reproduction of Test Case 1.2

Test case 1.2 consists of two vehicles in an intersection access scenario. Target vehicle approaches the intersection with 45mph and goes through the intersection while the host vehicle is located perpendicularly to the target vehicle and stationary right before the intersection. The driver state estimator results are shown in Figure 3.13 with a similar format to the presentation of the results of the previous case. From the figure, one can conclude that the host vehicle driver will be notified that accessing to the intersection is not safe immediately when target vehicle enters the driver state estimator model range.

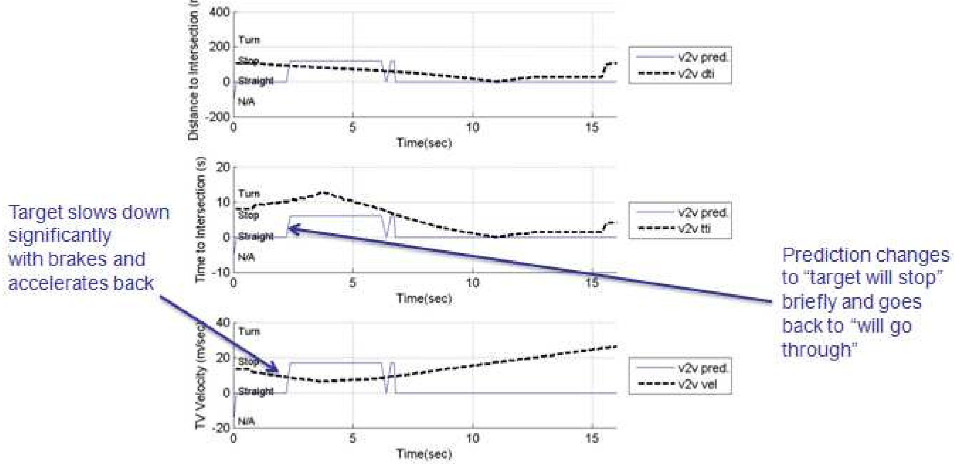

Figure 3.14: Results from the Reproduction of Test Case "Yellow Light Behavior"

Test case 2.6 consists of two vehicles in an intersection access scenario. Target vehicle approaches the intersection with 45mph and slows down before the intersection intending to stop. When target vehicle driver decides that a safe stop is not possible, he changes the intent and accelerates to go through the intersection imitating a yellow light behavior. In the meanwhile, host vehicle is located parallel but in opposite directions to the target vehicle and stationary right before the intersection. The driver state estimator results are shown in Figure 3.14 with a similar format to the presentation of the results of the previous case. From the figure, one can conclude that the driver state estimator catches the stop intent with the brake and updates the predicted state back to the straight state slightly after the accelerator is pressed.

Therefore, the host vehicle driver is immediately warned that intersection access is not safe.

Chapter 3 defined the active intersection access safety system and presented the work done within this research project in detail. The next chapter describes similar simulation studies in a topology including several vehicles and intersections.

CHAPTER 4

REAL-TIME INTEGRATED WIRELESS AND VEHICULAR SIMULATOR WITH COLLISION WARNING SYSTEM

The Ohio State University Vehicle-to-Vehicle Communication group has been investigating vehicular networking area in the last few decades. During the studies, being able to simulate vehicular traffic networks and vehicular wireless communication networks has been identified to be of great importance due to the high costs associated with the field tests performed with several vehicles. Chapter 4 introduces a new real-time integrated wireless and vehicular simulator developed, targeting multiple vehicle scenarios. An overview of integrated simulators, details about the design and implementation of the simulator, and verification of the simulator framework are presented throughout the chapter.

4.1 Overview of Integrated Wireless and Vehicular Simulators

Demand for the integration of wireless network simulators and the vehicular network simulators has been increasing in the past decade. With the approval of DSRC band in the USA, research focusing on the VANET environment became a hot topic rapidly. The topic is interdisciplinary; therefore it requires a broad knowledge and understanding of the components of this environment. Simulation of VANET scenarios are highly desirable but it comes at the expense of complex simulators. Basically, the integrated VANET simulators have three fundamental components: vehicular simulator, wireless network simulator and the possible applications running on the platform. Vehicular simulator and wireless network simulators are the most important components of an integrated simulator. As the modeling of vehicles and the communication processes gets more realistic, the computation complexity of the integrated simulators increase rapidly.

The vehicular simulator for the integrated VANET simulation environment preferably would be able to simulate complex highway and intersection scenarios along with collisions while allowing the user to set the parameters for vehicular traffic environment. It should also be noted that the detailed vehicle kinematics such as lateral slips and skidding are not the main concern of a integrated VANET environment and can be omitted. Other desirable properties of a vehicular simulator would be as follows; allowing applications to reach the vehicular information which is analogous to CANBUS access for a real life scenario and being able to import real maps and traffic information to the simulator.

Wireless networks simulators are also one of the crucial components of a integrated VANET simulation environment. Depending on the focus of the research group, one might choose to use either a simple model or a complex model for wireless network simulations. A complete wireless network simulator inherently requires computations of high complexity. Therefore, research groups with focus on the vehicular network simulation part usually use simple models to avoid computation complexities and vice versa. While using ray tracing for the propagation model for the wireless communication channel may be computationally complex and interesting for some users, only two-ray propagation model might be enough for some other users. Considering that the wireless simulators are already complex because of NxN transmission and reception processes, it might be a good idea to compromise between the computation complexity and the simulation results.

Prolonged runtime durations of wireless simulators have attracted the attention of the Ohio State University High Performance Computation Laboratory and a solution has been developed by generating offline and online wireless simulations. The basic idea behind this approach is to generate a statistical database for the packet error rate and the transmission delay for various traffic scenarios beforehand and use the statistical results from this database while running the vehicular simulations to incorporate for the wireless network effects. While this approach decreases the runtime complexity of the integrated wireless and vehicular network simulators, it results in omitting the instantaneous events occurring in the environment. Chapter 5 presents a simulator using this approach, and Chapter 4 presents a simulator using a real-time approach without the incorporation of statistical wireless network simulations.

4.2 Overview of the Vehicular Simulator, VATSIM

VATSIM is a traffic simulator which integrates the individual elements of a traffic situation such as automobiles, traffic signals, road structure and advanced ITS sensors. Then it attempts to model their interactions in a realistic way. The traffic interactions are continuously displayed on a two dimensional, top-down view of the road data as illustrated in Figure 4.1. The positions of individual automobiles are displayed, as well as the state of the traffic signals. Models of ITS sensors such as loop sensors and vision sensors can be added to the traffic simulator as well. The state of the simulated ITS sensors used in VATSIM can be interrogated via internet protocols. Thus the simulated sensors may be used to provide information which is functionally interchangeable with real world data provided via wired or wireless connections to actual sensors. For more information about the simulator please refer to publications [24] and [25].

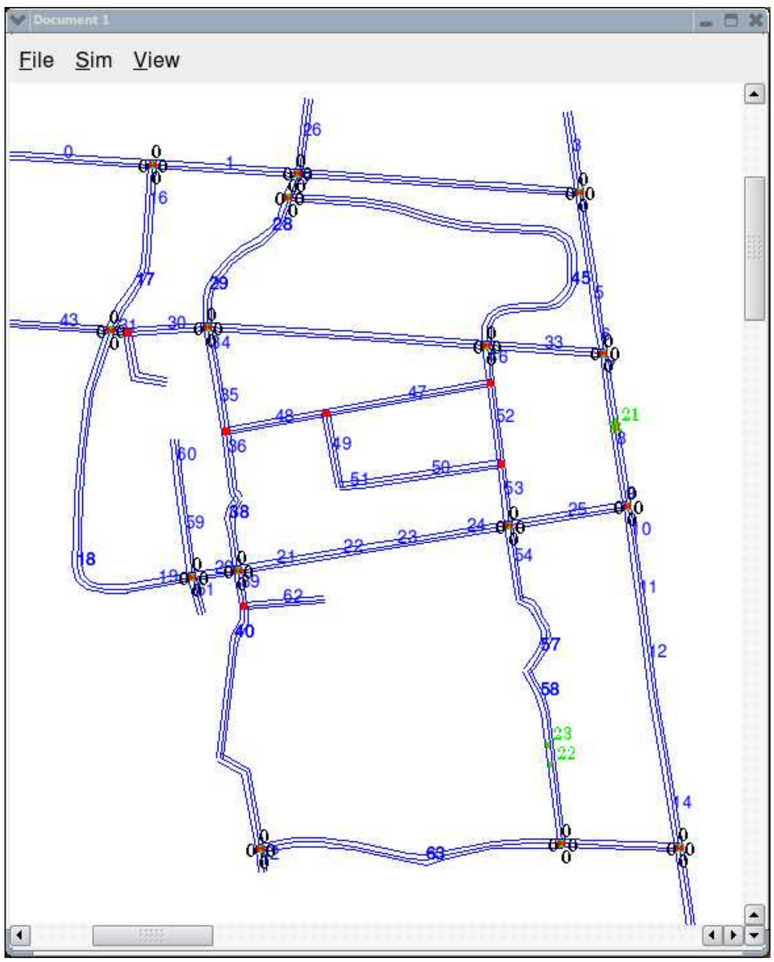

Figure 4.1: Graphical user Interface of the Vehicular Simulator VATSIM.

4.3 Overview of the Wireless Network Simulator, Modified NS-2.33

Network Simulator, NS-2 was originally developed for wired networking research with a considerable support for the simulation of transport control, routing and multicast protocols. The new versions of NS-2, starting with the version 2.31, expanded the simulation platform to wireless networks. IEEE 802.11 protocol has been implemented with the support of universities and the industry. It has been used in many research publications thereby going through extensive tests. However, the wireless network simulator has been found to have shortcomings which is recognized by the user groups. NS-2.33 has included a completely new architecture for the physical and MAC layer simulations of the IEEE 802.11 networks with the contributions of the University of Karlsruhe. [21]

The new IEEE 802.11 simulation framework in NS-2 models the MAC and PHY layer of the IEEE standards in a generic way making it possible to support standards like IEEE 802.11a,b,g and the draft p. The Tcl files allowing the user to configure and run NS-2 can be set up depending on the IEEE standard targeted in the simulations. Moreover, the RF propagation model in the new package supports Nakagami model, which has proven itself as a suitable model for vehicular communication channels. Simulators described in this chapter and the other chapters are based on this new available package with some modifications done to incorporate bit error rate, frame error rate and wireless channel performance to match field tests' results indicated in Chapter 3.

4.3.1 Simulator MAC Layer Performance

In this section, simulations regarding the evaluation of the MAC layer performance are described. Before going into the details of the simulation, it is helpful for the reader to go through the overview of the IEEE 802.11 transmission and reception. Overview is intended to have a medium detail level which is corresponding to the implementation detail of the simulator.

- **Modulation and coding rate**

The data rates of the IEEE 802.11 radios are controlled by the modulation and coding rate used in transmission process. Modulation is the process by which periodic waveforms are varied in order to convey information. Looking at the Table 4.1, one can see that QPSK is able to carry 2 bits per sub-carrier. Noting that every symbol has 48 subcarriers, maximum possible bits per OFDM symbol are 96. However, not all of this capacity is used for data transmission. The data is coded for error detection and forward error correction resulting in less number of actual data bits transferable with every symbol. Coding rate defines the efficiency of coding. A coding rate of 0.5 would result in half of the capacity being used for actual data transfer. For a 10Mhz channel, each periodic waveform plus a guard interval to prevent inter-symbol interference takes 8us. Therefore, with a 0.5 coding rate and QPSK modulation, the resulting data rate is 6 Mbps. A lower coding rate means the reception will be more resilient to low SNR values while a higher coding rate would require a higher SNR value to match same the packet error rate. Similarly, BPSK signals are resilient to lower SNR values compared to QPSK, 16-QAM, and 64-QAM. Depending on the channel quality and the resulting SNR value, a radio may be able to get the PLCP header and fail during the body reception process if the SNR value is not good enough for the selected modulation scheme for the frame body.

Modulation	Coded bits per sub-carrier	Coded bits per OFDM symbol	Coding rate	Data bits per OFDM symbol	Data rate for 10Mhz channel
BPSK	1	48	0.5	24	3 Mbps
BPSK	1	48	0.75	36	4.5 Mbps
QPSK	2	96	0.5	48	6 Mbps
QPSK	2	96	0.75	72	9 Mbps
16-QAM	4	192	0.5	96	12 Mbps
16-QAM	4	192	0.75	144	18 Mbps
64-QAM	6	288	0.5	192	24 Mbps
64-QAM	6	288	0.75	216	27 Mbps

Table 4.1: IEEE 802.11p Modulation and coding rates

- **PHY Layer frame structure**

The transmitter side selects a modulation and coding rate combination based on the current SNR value using the guidelines of the IEEE 802.11 standard [4]. However, the receiver side needs to know the settings of the transmitter to distinguish the incoming signal from noise. Therefore, the header of the packet is supposed to notify the receiver side using a generic and fixed configuration about the coding rate, modulation and the packet duration of the incoming packet. Figure 4.2 shows the basic PHY layer frame of IEEE 802.11 radios. Transmission starts with a training sequence called preamble to notify the receiver and let it synchronize. This pre-RX phase is continued with a PLCP (Physical Layer Convergence Procedure) header which contains the coding rate, modulation, frame length for the upcoming data. The whole pre-RX phase, which consists of preamble and the PLCP, is modulated with BPSK scheme where preamble is not coded and PLCP is 0.5 coded. When the pre-RX phase is complete, the receiver side knows the modulation scheme and the coding rate it needs, in order to decode the incoming data stream containing the payload.

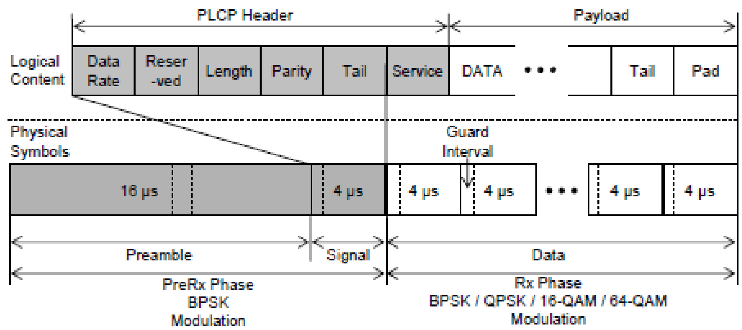

Figure 4.2: IEEE 802.11a PPDU format

- **Frame Reception Process**

In general IEEE 802.11 radios share a channel for transmission and reception. Moreover, generally wireless radios cannot send and receive simultaneously from the same channel. That is why, a radio can be considered as a state machine which is either looking for incoming frames(listening), transmitting, or receiving a frame.

When a radio is in the listening state, it continuously looks for the known preamble pattern by decoding the incoming data using BPSK. If preamble is received, radio will try to decode the signal portion of the PLCP header and extract the configuration for the upcoming data stream. If successful, radio will dedicate itself for the frame length duration to the incoming signal and send the decoded data to MAC layer for error detection and processing. Preamble and body frame capture processes which can divert the radio to another signal will be discussed shortly.

In the transmission state, the radio will miss any incoming frame even if the transmission is complete. This is due to the fact that the PLCP header contains the vital information about the incoming frame and there is no way to decode it once the PLCP header is missed. IEEE 802.11 radios do not interrupt their transmission to receive a detected signal.

On the other hand, while in reception state, radio will not be able to receive any other incoming frame as it is already receiving one. This is due to the fact that the radio is not looking for a preamble signal at that point. If the new incoming signal is strong enough, it will result in collision and prevent a successful reception of the current frame. This situation usually occurs in hidden node cases and CSMA and CSMA/CA type of mechanisms attempt to prevent this phenomena by coordinating the radios through their distributed coordination function(DCF) or in IEEE 802.11p case Enhanced Distributed Channel Access(EDCA) algorithms. Preamble capture is a method where the receiver continuously monitors the incoming data with a BPSK decoder separately from the reception process and detects a preamble from a stronger new transmitter while it is still receiving the PLCP header of the former transmitter. When the SINR of the new transmitter is above a certain threshold compared to the current transmitter, the radio understands that it will not be able to receive the current frame correctly and switches transmitter to the new incoming frame.

It is also feasible to have a frame body capture process during the reception of the body portion of the current frame. Even if this method is not in the standard for IEEE 802.11 [2], it is implemented by the chip producers. The algorithm continuously monitors the received signal strength indicator and looks for jumps that increase RSSI by approximately 10dB. If a sudden power increase like this is observed it discards the current frame reception and tries to decode the preamble and PLCP of the new frame. This implementation is usually useful for urban residential areas. For example, while receiving service from the neighbors' wireless router, your laptop's wireless radio would lock immediately to your wireless router when it sees the strong signal coming from a shorter distance.

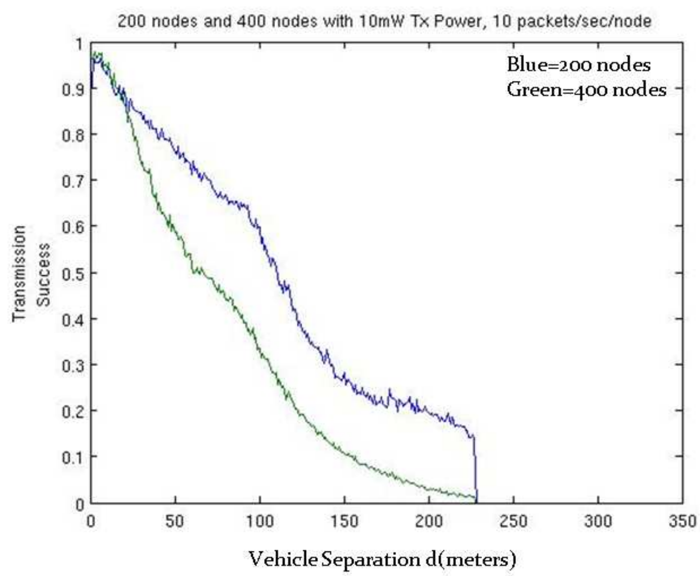

Figure 4.3: MAC Layer Performance of the NS-2.33 IEEE 802.11p Model

- **NS-2 IEEE 802.11p MAC Layer Performance**

Assuming the physical and medium access control layer modeling of the simulator are similar enough to the actual wireless networking chips, one can do simulations to evaluate the MAC layer performance of the IEEE 802.11p networks. As expected, dense networks result in more collisions and more back-off done by the vehicles.

As, one does not want to fund network saturation tests which need hundreds of actual vehicles, doing these tests with the simulator is rational. Figure 4.3 shows the successful packet transmission ratio vs. vehicle separation for networks of density 200 and 400. Simulations are done with each vehicle transmitting 10 packets per second at 10mW power and 6 Mbps. The range at which every vehicle can have successful communication with 90% confidence is observed as low as 20m and 30m for 400 and 200 vehicles respectively in a 1000m x 1000m topology. The results show easily periodic broadcast can saturate the network in highly populated urban environments. Considering the high buildings in an urban environment, one can suggest that the results may be even more critical. The results stress the importance of having smarter packet generation techniques such as distance based or event based packet generation.

4.4 Collision Warning System and Driver Model

Collision warning system in this context is just a safety application running on the simulator architecture. In other words, after the V2V communication process is completed, the vehicles have their database full with both their and neighbors' information which allows them to run any desired safety application. The safety applications may try to increase the awareness of the driver against possible traffic events. In this case, it is a collision warning system which warns the driver of possible accidents that might occur by displaying warning messages.

Collision warning system runs the following algorithm. At each time step, all the vehicles, assuming the speed and the yaw-rate of the vehicles are constant, generates a trajectory for the next t seconds. After the trajectories are calculated, every vehicle finds the neighbor vehicles closer than the collision warning system maximum

VITS Driver Model	Level 1	Level 2	Level 3
Conservative	Break	Break	Break
Normal	Release Gas	Break	Break
Aggressive	Ignore	Release Gas	Break

Table 4.2: Driver responses for warning messages

neighbor range. Now, each vehicle calculates if the bodies of itself and any neighbor overlap in the next t seconds. If a overlap is present, depending on how soon the crash is predicted to happen, a warning message flag with the corresponding level is raised. The ratio of time-to-crash and prediction duration t are utilized for the warning message level decision. If this ratio is smaller than 0.33, the highest level warning message is raised. Similarly, if the ratio is higher than 0.66 warning level is determined to be one. Level two warning message is raised if this ratio is between 0.33 and 0.66.

Contrary to the case in Chapter 3, now there is a driver model which simulates the behavior of the driver along with the drivers decision when warning messages are raised. Assuming a real-life scenario, one can easily deduct that drivers would have different reactions to different levels of warning messages depending on their vehicle's characteristics and personalities. This driver model is a simplified approach in pursuit of reflecting the real-life scenario. Table 4.2 illustrates the outline of the driver model used in the simulator. For example, while an aggressive driver ignores a low level collision warning message, a normal driver would release the gas pedal and a conservative driver would apply brakes. A true model would need statistical studies

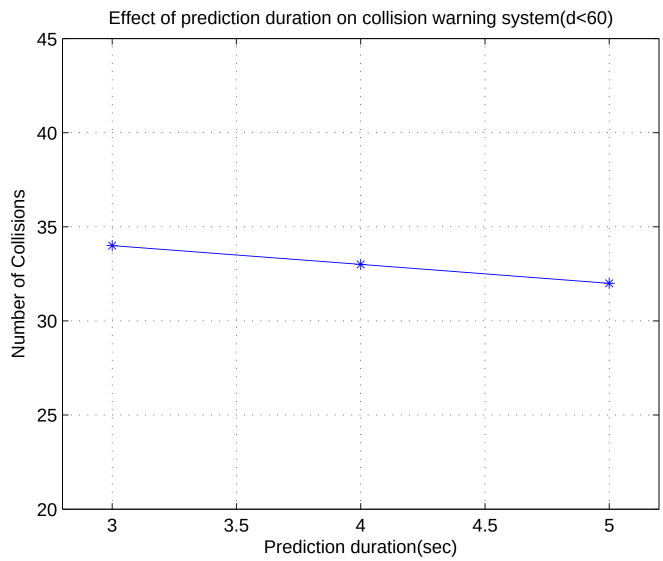

Figure 4.4: Effect of prediction duration on collision warning system(d<60)

to assign the percentages of the driver types in a community. In this study types of drivers are assumed to be uniformly distributed.

Collision warning system parameters such as prediction duration t(ttp) and maximum neighbor distance(d) have to be determined after a careful analysis of their effects on the safety application.

Analysis of the effect of prediction duration t on the safety application is shown in Figure 4.4. One can clearly see that the number of collisions are not significantly decreased as the prediction duration is increased. This is due to the fact that the trajectories are calculated assuming the speed and the yaw rate of the vehicles remain constant during the next ttp seconds. The estimated trajectories are not reflecting the true trajectories that the vehicles will traverse for the later portions of the prediction which are generated when ttp is increased. Therefore, including computational

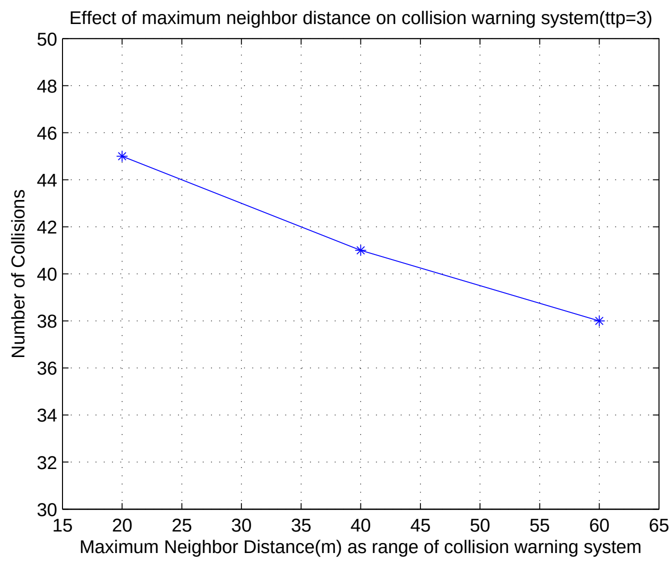

Figure 4.5: Effect of maximum neighbor distance range on collision warning system(ttp=3)

complexity concerns, the one resulting in less intensive computations is selected and the ttp is defined as three seconds. For a better trajectory generation algorithm, ttp may be increased effectively and result in a better safety application.

Maximum neighbor distance is defined as the maximum distance between the vehicles for which the collision warning system is executed. Analysis of the effect of maximum neighbor range(d) on the collision warning system in done by sweeping the values of d and observing the number of accidents and the cost of the accidents. Figure 4.5 and 4.6 shows the results of the analysis. Looking at the resulting graphs, one can deduce that the number of accidents and the cost of the accidents are reduced as d increases. The cost of the accidents that still occur is considered to be the directly proportional to the speed of the vehicles at the time of the accident. Therefore,

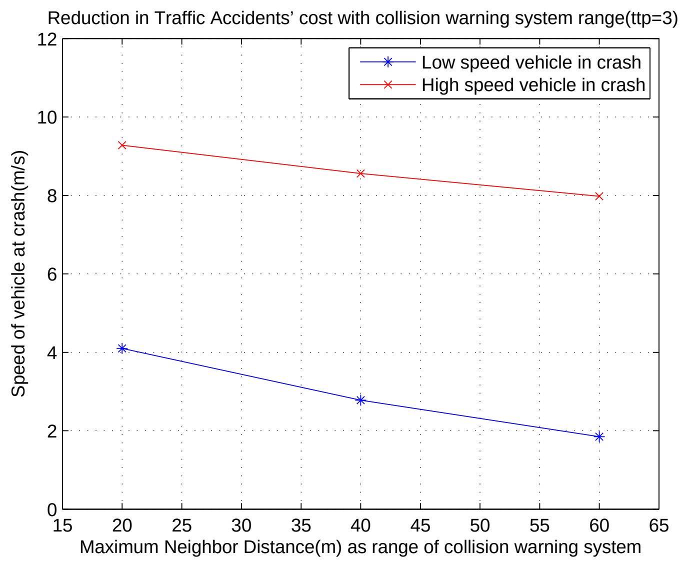

Figure 4.6: Reduction in Traffic Accidents' cost with collision warning system(ttp=3)

eventhough computationally the costs are expensive, maximum neighbor range is selected to be 60meters.

4.5 Integrated Real-time Simulator Architecture

The Vehicle and Traffic Simulator (VATSIM) is implemented as a modular simulation system with an object oriented flavor. The definition of the simulator consists of specifications of the program's databases, which include input databases, dynamic databases, and model databases and the execution flow of the simulator software. The contents and format of these databases are given below.

Input databases specify features of the world such as road geometry, buildings, signs, obstacles, and annotations, as well as the conditions under which vehicles will be added or removed from the network. In this implementation, we have assumed that these features are static and that their number is fixed for an entire simulation.

However, it would not be difficult to extend our software to overcome this assumption. The internal dynamic vehicle database contains the modeling and state information for each vehicle in the simulation. Its size can change as vehicles are added and removed from the simulation over the course of simulated time. Each vehicle possesses its own copy of its state database and information about the vehicle model to be used in its simulation.

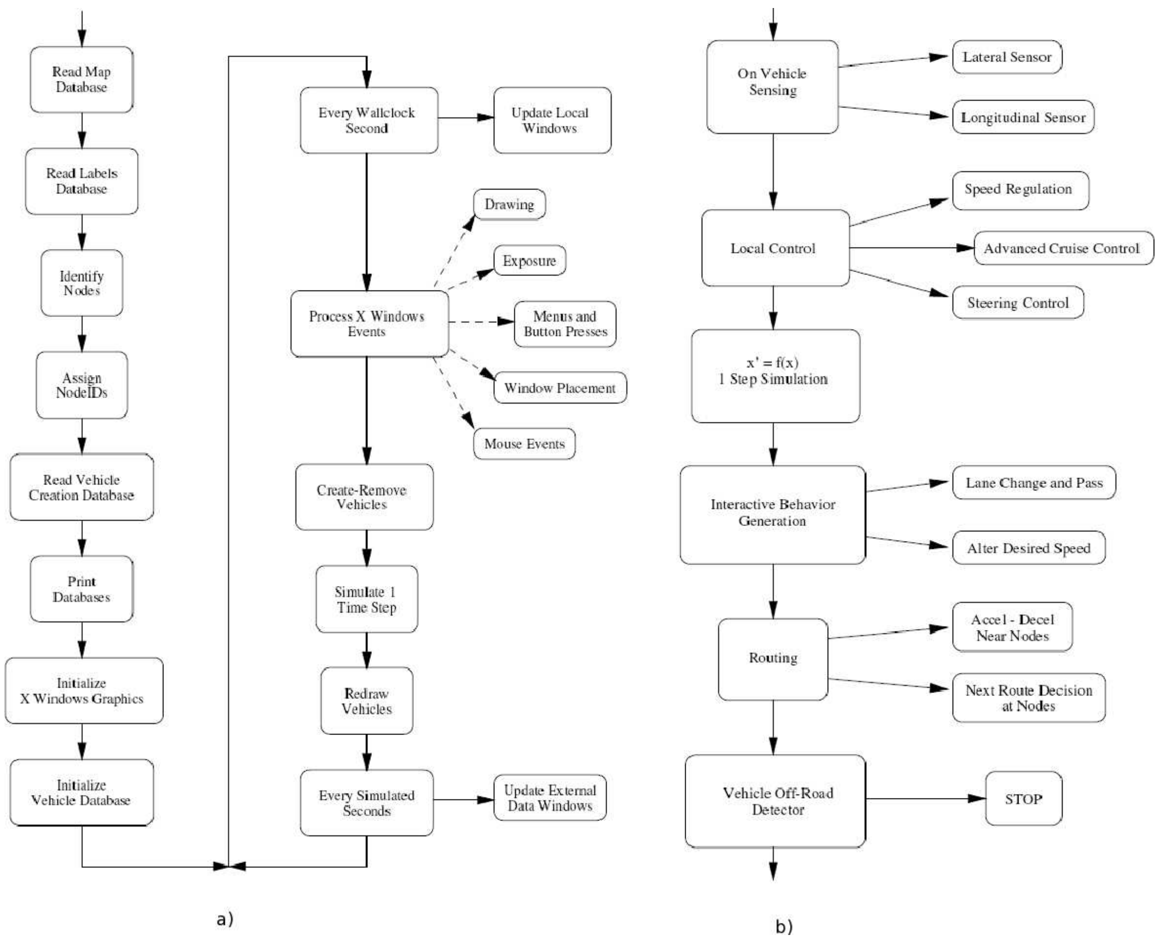

Figure 4.7: a)Overall flow of simulator program execution, b)Program execution in main simulation loop

Finally, the agent model database contains the functions which are used to implement models for the discrete and continuous-time sensors, actuators, vehicle dynamics of motion, controllers, and behavior generation modules used in the simulation of each class of vehicles. This simulator performs a micro-simulation, meaning that each car is simulated as an individual entity with, potentially, its own unique dynamics and controllers. This database also contains the description of the local variables associated with each class of vehicles. Although this database is designed for the simulation of vehicles, it could also be used to specify dynamic models for non-vehicle agents, such as traffic management systems, that might be introduced into the simulation.

The simulation proceeds as follows. After the initialization of the simulator and the X-Windows graphical interface, the initial vehicles present in the simulation are created and their states initialized as described in the vehicle source-sink input database. After initialization is complete, the simulation will begin and simulated time will move forward. In this simulator the time step is fixed, normally at 10ms. At each time step, vehicles are added or removed as necessary and each vehicle in the dynamic database has its sensor, local controller, dynamics of motion, behavior generation, and routing functions called in sequence. Finally, the graphical map based and text-based displays are updated and the values of selected state variables for each simulated vehicle are recorded in an output file. The overall execution flow of the simulator is illustrated in the block diagram of Figure 4.7.a. The details of simulator execution at each time step are shown in the block diagram of Figure 4.7.b. This simulator has been implemented in the C programming language and provides an X-Windows based graphical interface.

4.5.1 Real-time operation

Figure 4.8 illustrates the real-time operation of the integrated wireless and vehicular simulator. The synchronization of NS-2 and VATSIM is achieved by using semaphores in the shared memory. As one can see, the vehicular simulator VATSIM determines the coordinates, speed, heading and yaw-rate of each vehicle every 100ms and records the values to the database defined in the shared memory. At this point, NS-2 is triggered to continue the simulation for 100ms using the configuration defined by the TCL scripts. When the wireless network simulation is finished, it sleeps until the next time vehicular simulator triggers it. Based on the results of the wireless network simulation of that 100ms duration, receivers fill their database in the shared memory with the information extracted from the received WSM. Now, the collision warning system starts running its algorithm as defined in section 4.4 and the driver takes action depending on the collision warning messages and the driver's type.

4.5.2 Shared Memory Architecture Utilization

Vehicular simulator, VATSIM, has unique identifiers for every vehicle throughout the simulation. When a vehicle leaves the topology, that unique identifier is never used again. Therefore, to keep track of the vehicles' ids in the shared memory database a dictionary is implemented. Using the dictionary, both NS-2 and VATSIM can access the correct indexes in the shared memory to read or write data. A database entry containing neighbor vehicle data is removed if it is older than 500ms. Shared memory using the POSIX library is useful and fast for integrating processes and passing variables between processes thereby making the use of this library very desirable.

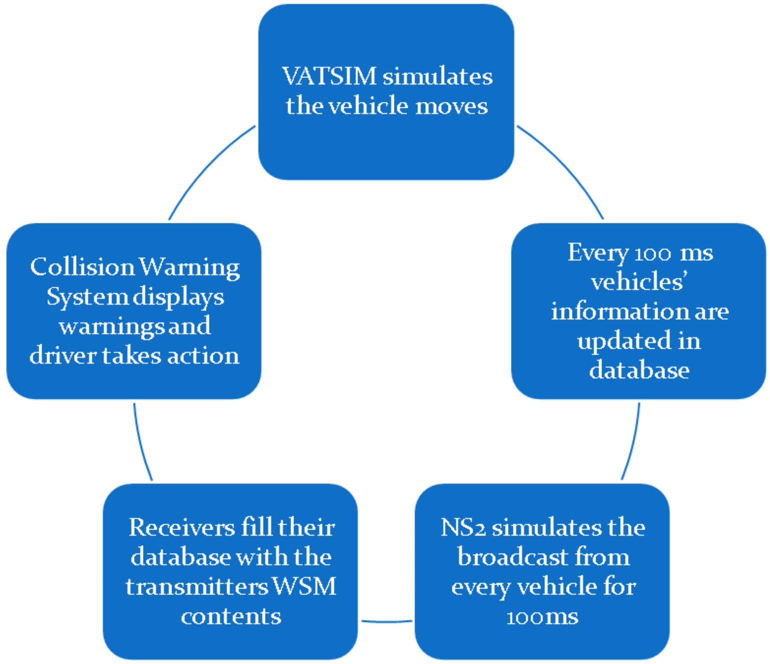

Figure 4.8: Real-time Integrated Vehicular and Wireless Network Simulator Architecture

4.5.3 Computation Complexities and Parallelization

Collision warning system and the wireless network simulator have high computation complexities. As the number of vehicles increase the load of the NS-2 and the collision warning system increase very fast. This is due to the fact that these algorithms calculate the interactions of each vehicle with all the other vehicles. In case of the wireless network simulator, a distributed computation approach needs very careful modeling because the wireless channel is inherently not suitable for parallelization. On the other hand, collision warning system is inherently parallel in real life scenario. Every vehicle has its Wireless Safety Unit, which runs the collision warning application for itself only. The collision warning system in the simulator is ran for all the vehicles, and OpenMP is used to parallelize this process on a Intel Quad Core Xeon CPU.

4.6 Real-time Integrated Simulator Results

In this section, Integrated Wireless and Vehicular simulator is explained with all of its components, and the necessary verification and the simulator results are given in this section. Figure 4.9 and Figure 4.10 are the graphs related with verification of the simulator collision warning system while Figure 4.11 illustrates the result of the safety system.

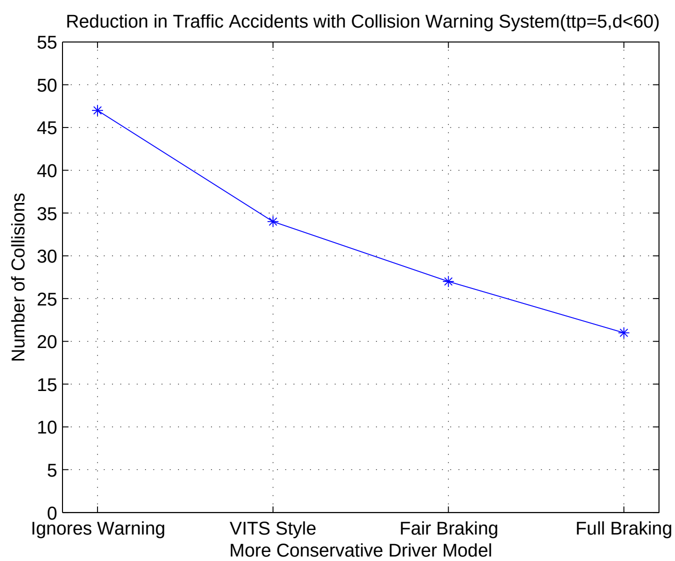

Figure 4.9: Reduction in Traffic Accidents with Collision Warning System (ttp=5, d<60)

In order to verify the operation of the collision warning system, the conservativeness of the driver model responses to the warning messages is swept. Figure 4.9

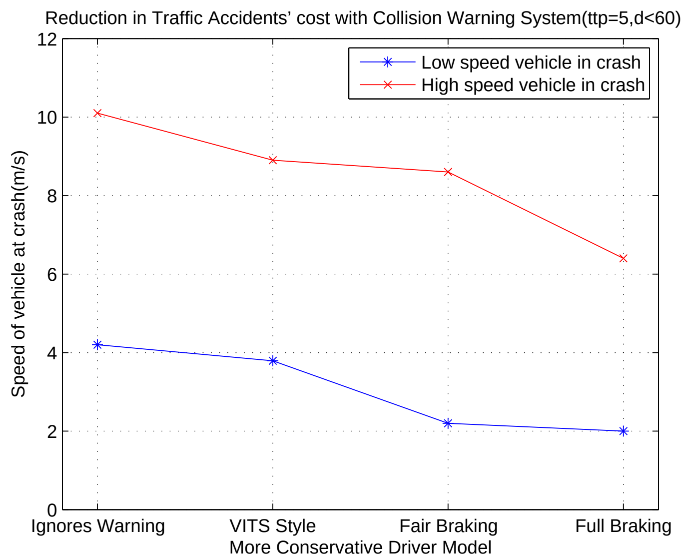

Figure 4.10: Reduction in traffic accidents' cost with Collision Warning System (ttp=5,d<60)

shows that as the drivers are more conservative, the number of collisions are decreasing thereby verifying the operation of the collision warning system. Moreover, Figure 4.10 shows that the cost of the accidents that still occur while collision warning system is operating also reduces as the conservativeness of the driver model is increased. In this case, the cost of the collision is assumed to be directly proportional to the velocity of the cars at the moment of impact. Consequently, one can say that the collision warning system operation is verified according to the explained experiment setup.

Figure 4.11 illustrates the contributions of the collision warning system using the simulator framework. As the vehicle flux into the topology is increased, the number of collisions increases as expected. From the figure, one can deduce that with the collision warning system in operation the number of collisions is decreased approximately by 30%. Moreover, even if not graphically illustrated, one knows the

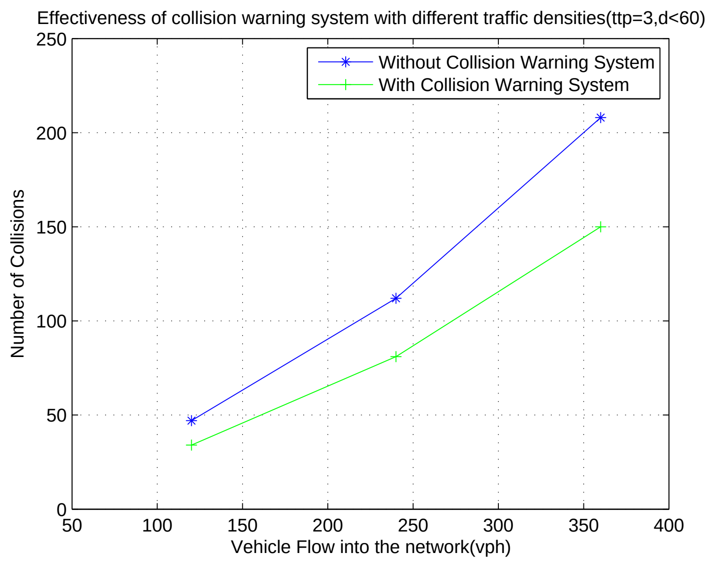

Figure 4.11: Effectiveness of collision warning system with vs. traffic flow density(ttp=3,d<60)

costs of the accidents that still occur while collision warning system is operating are reduced.

Considering these results, the operation of the collision warning system is verified, and the contribution of the vehicular active safety system is illustrated. The study in this chapter is a good example of an integrated simulator with multiple intersections and many vehicles. In the next chapter, a single intersection, multiple vehicle integrated simulator will be presented.

CHAPTER 5

INTEGRATED WIRELESS INTERSECTION SIMULATOR(IWIS)

5.1 Introduction

In this chapter, the complete architecture of the integrated wireless and vehicular simulator is presented; it is called *Integrated Wireless Intersection Simulator* (IWIS) and it has been developed since 2003 [5, 7]. The chapter is based on a published collaborative study [19] by me and Boangoat Jarupan. For completeness, the parts which are not my individual work are also included. All the experiments done by the simulator are my individual work.

IWIS is built upon our previous research [5, 16, 20, 7] and it consists of two main components, *Vehicle Intersection Traffic Simulator* (VITS) and *Wireless Simulator* (WS) (Figure 5.1). In this chapter, we adopt the architecture of VITS from our previous work which is built using a Visual C++ environment and provides a friendly graphical user interface (GUI). The VITS traffic simulator includes a driver model which takes action upon receiving a warning message. VITS is based on a realistic traffic scenario which includes multiple lanes and different type and size of vehicles. IWIS relies on detailed propagation models and accurately accounts for noise and

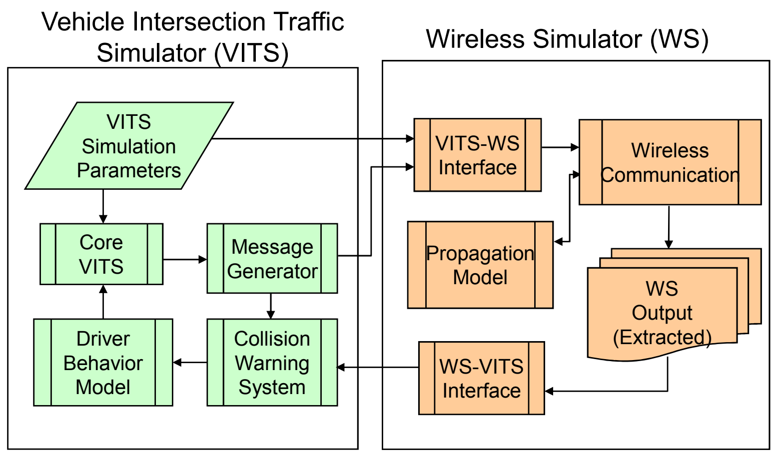

Figure 5.1: Architecture of Integrated Wireless Intersection Simulator [19]

interference. The model includes building blockage and vehicle shadowing considering different size vehicles. Performance of IWIS is investigated on an urban road network with an intersection.

Unlike our previous work where WS was built using CSIM [6], WS in this work is built using the network simulator NS-2 [18]. NS-2 brings in the flexibility and modularity to our system architecture where individual components can be changed transparently without affecting the rest of the system. In comparison with our previous work and existing simulators, it is found that IWIS not only evaluates the collision warning system but it also can be used in evaluating and comparing different MAC protocols such as 802.11p [4], UMB [14], and DOLPHIN [27]. These MAC protocols were simulated in various traffic conditions using IWIS. The evaluation results are presented in Section 5.5.

Both VITS and WS are stand-alone simulators, and they can be run independently. To seamlessly integrate the simulation traffic flow and communication among vehicles, we introduce the *automated interface model* in this chapter (Figure 5.2). The *VITS-WS interface* reads the traffic flow input files from VITS and translates them into the WS environment. The *WS-VITS interface* collects statistical results from the WS and automatically places them into the VITS environment. IWIS runs VITS and WS simulations interlinked by the interface models, and generates the results based on user defined inputs.

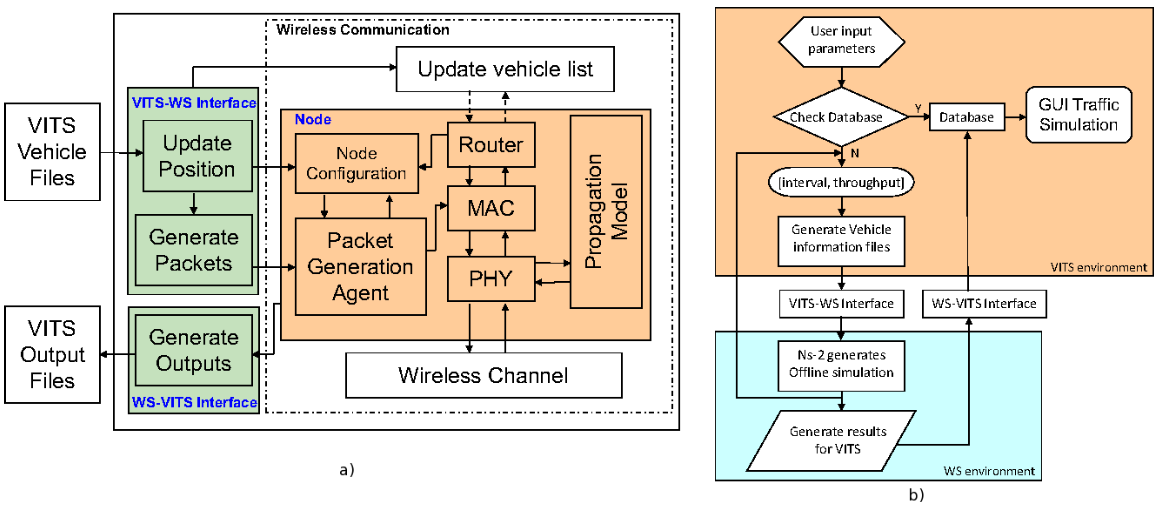

Figure 5.2: a) Architecture of Wireless Simulator b) Flow of IWIS

5.2 IWIS Overview

IWIS is a collision warning system simulator which consists of VITS and WS. VITS simulates vehicular behavior and produces traffic flow information, which is fed to WS. Based on traffic flow information from VITS, WS simulates the wireless communication among vehicles. WS produces several packet transmission statistics which are fed back into VITS, and then they are used in generating traffic alerts on important events (such as possible vehicle collisions).

VITS and WS operate in different time scales. VITS simulates ITS events which are fewer in number and in milliseconds timescale level compared to WS which generates exhaustive communication events and operates at the microsecond level. The *WS-VITS interface* is used to avoid the timing conflicts between the two simulators by performing a statistical approximation of wireless transmission behavior.

5.2.1 IWIS intersection environment and input variables

IWIS is developed in a Visual C++ environment. It provides a GUI based interface which allows users to configure the following list of parameters.

MAC protocols: IWIS supports three different MAC protocols: 802.11p, UMB, and DOLPHIN.

Vehicle type and dimension: IWIS supports four vehicle types, passenger vehicles (cars), busses, trucks, and motorcycles. The vehicle type not only determines the physical dimensions but also properties such as maximum acceleration/deceleration rates, turning radius etc. IWIS uses these properties in accurately simulating the communication among vehicles. For example, the vehicle height contributes to a shadowing effect in the propagation model (Section 5.4.2).

Intersection vehicle throughput: This parameter controls the *average* number of vehicles at an intersection. It helps in studying the performance of IWIS in different traffic conditions. However, in the actual simulation, the number of vehicles and the time at which they enter and leave the intersection area are modeled as probabilistic distributions.

Transmission interval: The road is divided into segments based on the user specified range. A transmission interval is the length of segments within a given range. Vehicles generate data packets in each transmission interval. Therefore, a smaller interval causes a higher packet generation rate.

Building presence: Buildings can block signals, and therefore greatly hinder communication among vehicles. Section 5.4.2 describes the effect of signal path loss in detail.

Repeater presence: A repeater placed on a high structure above the road can be optionally present at the intersection. The repeater re-broadcasts all received packets. Hence, it greatly improves the packet transmission success rate.

Traffic signal presence: Traffic lights control the flow of traffic at intersections.

A screen shot of the GUI used by IWIS is shown in Figure 5.3. The left side of the window displays the animation of vehicle movement, whereas the right hand side demonstrates vehicle collision warning messages generated by the simulator.

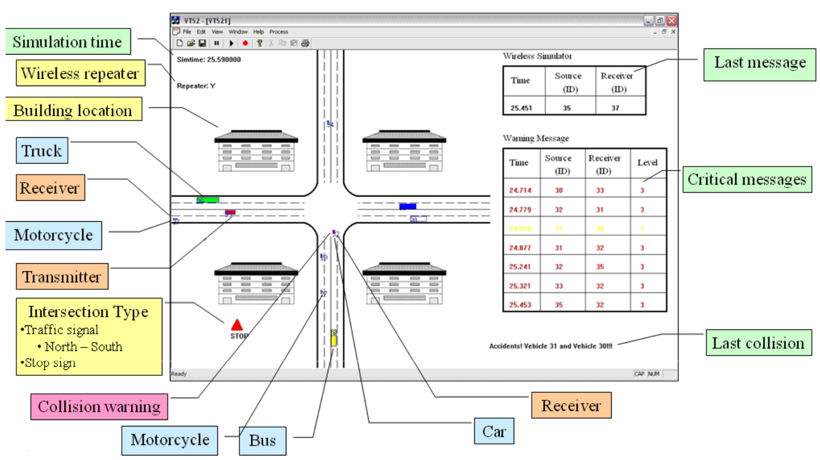

Figure 5.3: IWIS Graphical User Interface

5.3 VITS

VITS simulates the movements of different vehicles according to the user-defined parameters described in Section 5.2. It produces traffic flow information such as the position and speed of vehicles at different times. VITS consists of the collision warning system, the message generator, and the driver behavior model (Figure 5.1) as described below.

5.3.1 Collision Warning System and Message Generator

The Message Generator generates warning messages of three different levels: *elevated*; *high*; and *severe*. The data packets generated by a given vehicle contain the vehicle's current position, velocity, and acceleration. These messages are generated based on the collision probability for a given vehicle that is computed from received data packets (from neighboring vehicles) and expected destination of the vehicle. The simulator detects collisions when two vehicles have overlapping bodies. Once the drivers receive a warning message, they take different actions to avoid the collision based on the driver behavior model.

5.3.2 Driver Behavior Model

This model determines driver's response to different warning messages. IWIS defines three types of driver characteristics: aggressive, normal, and conservative. Based on the driver type and the level of the warning message, the driver responds by accelerating, decelerating, or breaking. For instance, aggressive drivers may ignore low level messages while conservative drivers may decide to decelerate the vehicle.

The process of developing a new version of VITS which includes more realistic traffic environments and additional user input parameters is an ongoing work.

5.4 Wireless Simulator

The *Wireless Simulator* (WS) performs the detailed simulation of wireless packet transmission among vehicles and repeaters. WS incorporates fundamental protocol properties, such as carrier sensing, random back-off, transmission interval, and packet retransmission which are developed using the well-known wireless Network Simulator (NS-2). This simulator also models the functioning of repeaters which are optionally placed at the intersection. Due to the modularity of the NS-2 architecture, the packet transmission behavior of different MAC protocols are evaluated. The physical layer and channel propagation in NS-2 are enhanced to include the effect of signal path loss and frame error rate.

Figure 5.2 shows the architecture of the WS. WS has three main functionalities: *VITS-WS interface*, *WS-VITS interface*, and *wireless communication*. The interfaces are responsible for transferring traffic flow information from VITS to WS (Section 5.4.1) and statistics from WS to VITS (Section 5.4.3). The wireless communication module (Section 5.4.2) simulates the actual packet exchange among vehicles and the repeater.

5.4.1 VITS-WS interface

The interface reads input files with the traffic flow information from VITS, and performs two actions: *vehicle position update* and *packet generation*. The VITS-WS interface reads the vehicle information and positions from VITS input files and periodically updates the vehicle configurations in the WS environment.

The VITS-WS interface has a packet generation agent that monitors each vehicle position and its transmission border. Whenever the vehicle crosses the border, it broadcasts a packet with its current position, velocity, and acceleration. The data packet information helps other neighboring vehicles in tracking the position of the vehicle that sent the packet. Each vehicle employs a *distance-based* protocol for generating packets where it determines the *transmission border* by using its current position (given by input files from VITS) and transmission interval (a user-defined parameter).

By employing such a distance-based protocol that relies on transmission borders, the load on the wireless channel is reduced when compared to a time-based strategy that sends packets at constant time intervals. For example, slow or stationary vehicles do not generate any redundant messages. Therefore in our protocol, the vehicle speed determines the frequency of packets and the number of packets depends on the distance traveled.

5.4.2 Wireless communication

The *communication* module of the Wireless Simulator is responsible for transmitting packets among vehicles and repeaters. The user can select the desired MAC protocol and other input parameters using the IWIS GUI window. All vehicles share a common wireless channel, and employ the propagation model.

MAC layer

Once a packet is enqueued into the queue, the vehicle determines the method of transmission based on the MAC protocol that the user has selected. Currently, IWIS supports three different MAC protocols: DOLPHIN [27], UMB [14], and 802.11p [4].

- *DOLPHIN*: DOLPHIN [27] is a p-persistent CSMA MAC protocol. The time is divided into slots, and in any given slot, a single packet is allowed to transmit with probability p. Before sending the packet, the vehicle senses the channel and waits for it to be idle. Once the channel becomes idle, it transmits the packet with probability p, a parameter that the user can specify. In case of packet collisions, the sender waits for a random amount of time and restarts the contention process. The maximum number of retransmissions is set to 5 times, and the slot size is set to 20 ms.

- *UMB*: UMB proposed in [14] is an IEEE 802.11-based directional broadcasting protocol. In UMB, each successful packet transmission involves the RTB/BB/CTB/DATA/ACK handshake, where RTB is a request-to-broadcast packet, BB is a black-burst packet, and CTB is a clear-to-broadcast packet. Unlike in 802.11, the sender selects the farthest vehicle along the road in the direction of packet dissemination through the RTB/BB/CTB handshake. UMB relies on vehicles or a repeater at intersections to broadcast the packets in all directions. In other words, a selected vehicle or repeater issues new directional broadcasts to all road directions to cover an intersection area.

- *IEEE 802.11p*: IEEE 802.11p [4] is an extension to the well-known IEEE 802.11 standard [2] to support Intelligent Transportation Systems (ITS). The packets are transmitted using the flooding mechanism available in 802.11 that does not use the network topology or the neighborhood information. Vehicles employ a random back-off strategy to reduce the number of packet collisions. We use the latest version of NS-2 (NS-2.33) that has extensions to MAC and physical layers to support wireless access in vehicular environments.

Routing layer

Since IWIS is based on broadcasts, we implemented a simple routing layer that is applicable to all MAC protocols. Unlike most existing routing protocols for vehicular networks, the routing layer neither performs neighbor discovery nor neighbor management through explicit packet exchange. In our protocol, vehicles directly forward the received packets to the application layer, and they do not rebroadcast the packets. The repeater nodes installed at intersections however rebroadcast the packets to different road directions.

Physical layer

The physical layer in IWIS is designed to model a wireless modem operating at the DSRC band. It computes the received power and bit error probability, which was proposed in [5]. Here, we briefly describe the ways in which signal path loss is computed.

- *Line-of-sight*: In this case, there is no obstruction between the transmitting vehicle and the receiving vehicle. Therefore, the direct signal and the reflected signal (from ground) are the dominant contributors to the received power. The received signal path loss is computed using a Two-Ray model [23].
- *Building blockage*: Buildings present between the sender and the receiver may block the communication signal, and therefore they act as major obstacles in vehicular networks. Here, the signals have no line-of-sight (between sender

and receiver) and are received as a result of reflection and diffraction. In such scenarios, we employ a simple virtual source model [8] to compute the received signal path loss.

- *Shadowing effect*: In addition to buildings, large vehicles such as trucks and busses can potentially block the signals destined to nearby smaller vehicles. Such an effect is called the shadowing effect where the signal has no line-of-sight. We use the knife-edge diffraction model proposed by Rappaport to compute the path loss under shadowing effects [23].

5.4.3 WS-VITS interface

WS collects the following statistics during simulation.

Packet collision rate is the total number of packet collisions experienced by all vehicles (or repeater if it is present) during the simulation period.

Packet latency is the packet delay from the source to the destination including contention and transmission delay.

WS statistical results are fed into the IWIS database. Before starting the VITS simulation, IWIS checks the database availability. If the database is not available from previous simulations, IWIS automatically runs WS and generates the database for that simulation configuration. If the WS database is ready, IWIS can directly access it during VITS collision warning analysis.

5.5 Evaluation

We study an urban environment with a four-way intersection. The intersection area is 215 x 215 m. Each road has two lanes running in opposite directions. The speed of vehicles is set to 45 mph (20 m/s). Additional IWIS user-defined parameters are set as per the discussion in Section 5.2.

WS simulation is run for 100 seconds. The data packet size is 500 bytes. Further, the maximum transmission power, data rate, and radio frequency is set 10 dBm, 3 Mbps, and 5.9 GHz (DSRC) respectively.

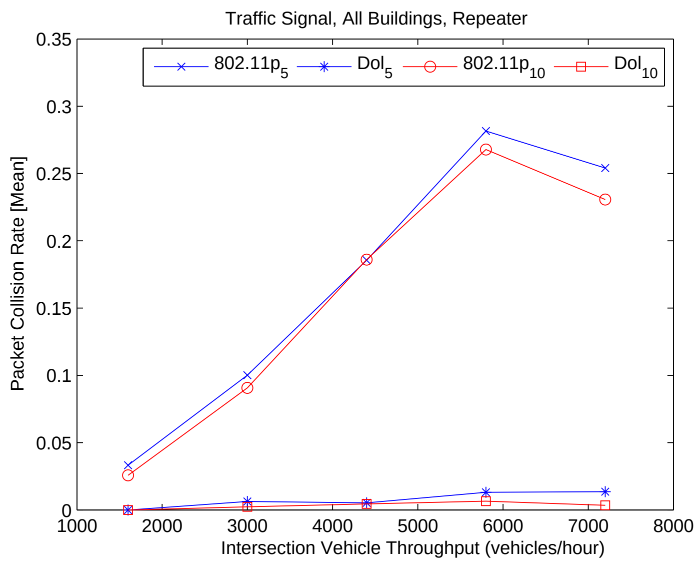

Figure 5.4: WS Packet Collision Rate, Transmission Interval=5,10

First, we evaluate performance of the WS. In Figure 5.4, we compare average packet collision rate and vehicle throughput of 802.11p and DOLPHIN protocols. For this experiment, we set up the simulation with buildings at all four corners, a traffic signal, and a repeater. We also considered other simulation scenarios. Due to space limitations, we omit the results from those experiments – however, we note that the trends are similar to the ones in Figure 5.4. We simulated with the transmission interval equal to 5 m (as 802.11p_5, Dol_5) and 10 m (as 802.11p_{10}, Dol_{10}). Overall,

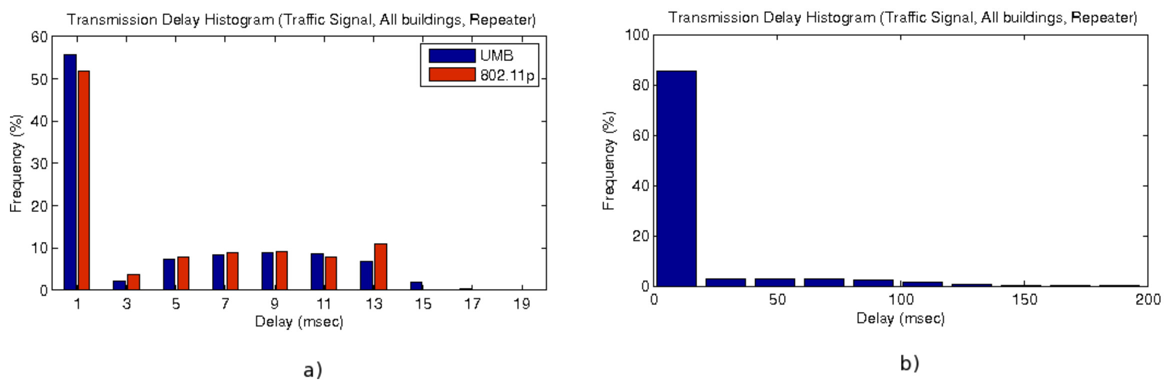

Figure 5.5: a)UMB and CSMA Packet Delay Distribution, b)DOLPHIN Packet Delay Distribution

the results demonstrate that the packet collision rate is increased with the vehicle throughput. Furthermore, the transmission interval of 5 has a higher packet collision rate than the transmission interval of 10. This is because a smaller transmission interval causes higher number of packets to be generated. Hence, the packet collision rate is increased with smaller transmission interval and higher vehicle throughput.

Between 802.11p and DOLPHIN, 802.11p has a higher collision rate because DOLPHIN uses p-persistent random access ($p = 0.1$) whereas 802.11p uses CSMA (broadcast mechanism). Since p (the probability of sending the packet at each time slot) is low, the vehicles could potentially avoid some of the collisions. Note that when vehicle throughput is from 5800 to 7200 vehicles/hour, 802.11p packet collision rate drops. This may be a result of channel overload where the sender drops the packets when it finds the channel busy (even after random back-off and retransmission). Although the results of UMB are not shown in the Figure 5.4, we found that the packet collision rate of UMB coincided with the X-axis and it is significantly lower than 802.11p and DOLPHIN. The low collision rate is mainly due to the handshake mechanism and also due to the presence of repeaters at the intersection.

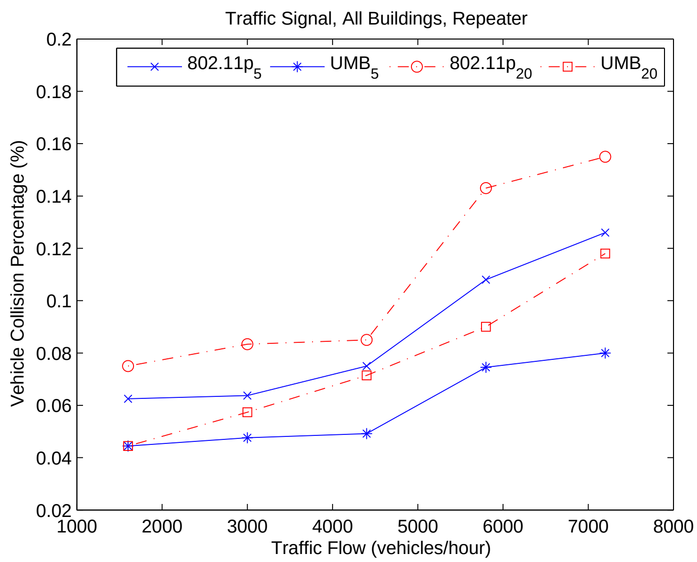

Figure 5.6: UMB and CSMA Vehicle Collision

Figure 5.5 and Figure 5.5 show the delay distribution of DOLPHIN, UMB, and 802.11p. Out of all three protocols, DOLPHIN has a very high delay (up to 200ms), which is caused by the slot based p-persistent contention mechanism. When compared to 802.11p which has a delay up to 12ms, UMB incurs marginally higher delays (up to 19ms). This may be due to the extra packet overhead introduced by the handshaking mechanism. It is important to note that the increase in delay is much smaller than the decrease in collision rate achieved by UMB.

In another experiment, we evaluated the performance of IWIS by measuring the vehicle collision percentage of MAC (802.11p and UMB) protocols and vehicle throughput (Figure 5.6). Overall, we find that vehicle collision rate increases with increased traffic flow. When packet transmission interval is reduced (packet generation rate is increased), the vehicle collision rate is reduced. Furthermore, we find that 802.11p has a higher vehicle collision rate than UMB. Although we do not show DOLPHIN results due to space limitation, we observed similar trends in DOLPHIN. Additional DOLPHIN simulation results can be found in [5].

5.6 Conclusion

An Integrated Wireless Intersection Simulator (IWIS) in an urban environment that contains two stand-alone simulators, VITS and WS, is presented. VITS simulates traffic behavior while WS simulates communication events. The two simulators are connected through interfaces that transfer vehicle and statistical information from one simulator to the other. A distance-based protocol to generate the data packets is proposed in the system. The simulator in this chapter is evaluated under different traffic scenarios, and use the simulator to compare three different MAC protocols (DOLPHIN, 802.11p, UMB). WS results show that the packet collision percentage is increased with the packet generation rate. On the other hand, IWIS results show that vehicle collision rate is reduced when vehicles generate more messages. Furthermore, our simulations also show that UMB outperforms 802.11p and DOLPHIN, in terms of the packet collision rate, the transmission delay and the vehicle collision rate.

5.7 Ongoing and Future Work

Designing integrated wireless and vehicular simulators is a inter-disciplinary study. It requires background from many fields such as control, network, software development. Targeting a simulator which focuses on every aspect of this framework in great detail requires very good software programming and distributed computation for wireless network simulations. In the sections discussed in this chapter offline & online wireless network simulations have been utilized to overcome the computation complexities rising from the dense vehicular networks. In simple terms, by doing online simulations the simulator generates statistical data for critical wireless network performance metrics such as delay and packet transmission ratio for different scenarios emerging from the combinations of building existence, selected MAC protocol, repeater existence etc... Online simulator uses the generated statistical data depending on the selected configuration to make a faster quasi network simulation that could keep up with the speed of the vehicular network simulator in dense networks. Computation of these statistical data for all possible configurations is a time consuming simulation period. Noting all these offline simulations being done for different scenarios are independent processes, Ohio SuperComputer Center's IBM Opteron Cluster is utilized to execute all the processes in parallel. The windows terminal running the IWIS, transparently connects to the cluster using automation scripts and the simulations are executed with a speedup greater than 30. Source code for this ongoing work is included with the disc accompanying this thesis.

It has also been argued that having offline and online wireless network simulations result in averaging network performance metrics over time and results in discarding necessary details that occur as peaks in performance metrics. For example; a sudden vehicular traffic pattern change may result in a peak in demand for wireless network traffic depending on the packet generating application. This case would be averaged out by online and offline wireless network simulation approach. To account for this drawback, the third generation IWIS is developed and it is in testing phase. This

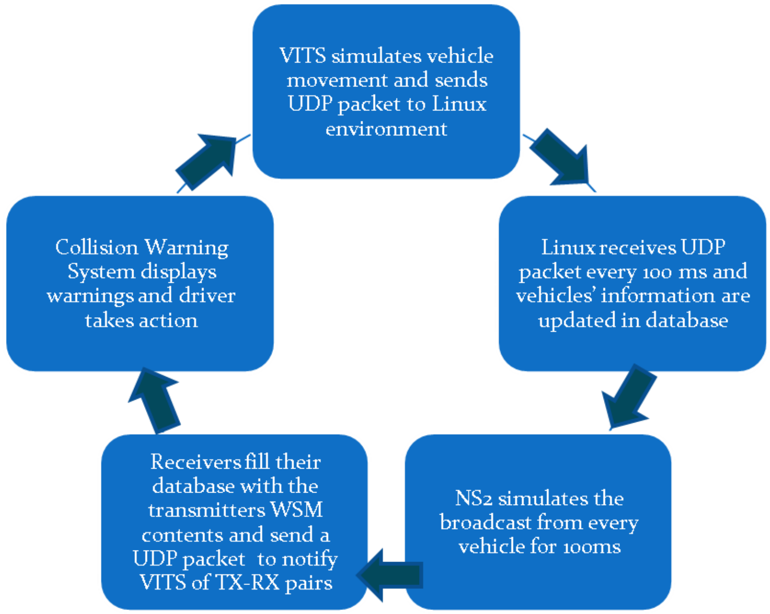

Figure 5.7: Simulation cycle for the real-time IWIS

simulator incorporates a detailed real-time wireless network simulation parallel to the vehicular network simulator. The NS-2 wireless network simulator is modified in the house and using UDP packets to communicate with the windows terminal running the vehicular simulator a simulation cycle as depicted in Figure 5.7 is developed. As the simulation progresses every 100ms the vehicular simulator updates the positions and speeds of the vehicles in the wireless network simulator. When the nodes in wireless network simulator is updated a wireless network simulation for 100ms duration is triggered and the successfully communicating pairs are conveyed to the vehicular network simulator. At this point the vehicular simulator runs the collision warning system and depending on the driver model incorporated the drivers take action depending on the warning message level just like they did in the version 2.0 of the IWIS.

CHAPTER 6

CONCLUSION AND FUTURE WORK

This study includes design and performance analysis of three different integrated wireless and vehicular simulators. While conceptually sharing some design considerations they all concentrate on different aspects of the vehicular communication environment.

The first one of these simulators is a two vehicle and single intersection simulator presented in Chapter 3. It is developed as a part of a larger project including HONDA and The Ohio State University. Main purpose of the simulator is to replicate the behavior of the V2V safety system hardware testbed developed in this research project. A vehicular simulator and a wireless network simulator are integrated to each other by modifying vehicular simulators developed in OSU and NS-2 wireless network simulator. The IEEE 802.11p radios of the hardware testbed are tested for their RF performances and wireless simulator is tweaked to reflect the performance of these radios. Finally, the intersection access safety system application developed for the hardware testbed is successfully implemented on the software testbed, and its operation is demonstrated.

The second integrated wireless and vehicular simulator is presented in Chapter 4. It is a multiple vehicle and multiple intersection simulator developed for analyzing micro-traffic networks as a project for the V2V Consortium of The Ohio State University. Main purpose of this simulator is to analyze the performance of the IEEE 802.11p standard performance for dense networks and implement a collision warning system utilizing the V2V communication. A vehicular network simulator developed in The Ohio State University is modified and integrated with the a modified NS-2 wireless network simulator. The design of the collision warning system, driver model and the performance of the IEEE 802.11p MAC layer is carefully presented with results verifying the design goals.

The third and final integrated wireless and vehicular simulator is presented in Chapter 5. It is a single intersection and multiple vehicle simulator developed as a project for the V2V consortium of OSU. The main target of this simulator is investigating the effects of buildings and a dedicated repeater on V2V communication performance and comparing three different MAC protocols. Also, a collision warning system application is implemented along with a driver model to analyze the traffic network.

Throughout this thesis, the design of three different integrated wireless and vehicular simulators are presented in detail. This work focuses on the high performance integration of various pieces of simulator blocks. It resembles the system engineering in private sector. The designer not necessarily designs all the small pieces resulting in the big picture, but he surely knows how each functional block is designed and works and how-to modify, manipulate, use and connect functional blocks gracefully. One of the main advantages of such a study including interaction with team members and getting familiar with many functional blocks from different disciplines is the steep learning curve and the contributions to the system designer. Having learned all the aspects of integrated wireless and vehicular simulators, I will continue my studies by having deeper understanding of the current functional blocks of our simulator and replacing them with appropriate methods that would result in a better simulator. At this point, depending on the drawbacks of computational efficiency that I observed in these systems, I can conclude that distributed computation of the NS-2 wireless network simulation looks like a promising area full of research potential. I would like to continue my PhD work on the topic and contribute to the NS-2 users community by providing a faster simulator.

APPENDIX A

WIRELESS SAFETY MESSAGES IN WAVE

IEEE 802.11p is a draft amendment to the IEEE 802.11 standard to add wireless access in vehicular environments (WAVE). It defines enhancements to 802.11 required to support Intelligent Transportation Systems (ITS) applications. This includes data exchange between high-speed vehicles and between the vehicles and the roadside infrastructure in the licensed ITS band of 5.9 GHz (5.85-5.925 GHz). IEEE 1609 is a higher layer standard on which IEEE 802.11p is based.

802.11p will be used as the groundwork for Dedicated Short Range Communications (DSRC), a U.S. Department of Transportation project based on the ISO Communications, Air-interface, Long and Medium range (CALM) architecture standard looking at vehicle-based communication networks, particularly for applications such as toll collection, vehicle safety services, and commerce transactions via cars. The ultimate vision is a nationwide network that enables communications between vehicles and roadside access points or other vehicles. This work builds on its predecessor ASTM E2213-03.

- The wireless safety message packet in WAVE/DSRC is generated depending on the details given by Figure A.1 and Figure A.2.

All multi-byte integer fields are in network byte order (i.e., Big-Endian)

byte	8 (MSB)	7	6	5	4	3	2	1 (LSB)
1	DSRC Message Id *(Set to the fixed value of 0x02)*							
2	milliseconds in a minute 0-60000							
3								
4	Temporary Id - Represents dynamic MAC or IP address of the radio							
5	-do-							
6	-do-							
7	-do-							
8	-do-							
9	-do-							
10	Latitude of the Host Vehicle Center Position (MSB) 32-bit signed							
11	-do-							
12	-do-							
13	LSBit = 1.25-e7							
14	Longitude of the Host Vehicle Center Position (MSB) 32-bit signed							
15	-do-							
16	-do-							
17	LSBit = 1.25-e7							
18	Elevation (MSB) 24-bit unsigned							
19	Range 0 - 16777215 m; Offset =1km (value 0 = 1km below the WGS84 reference ellipsoid							
20	LSBit = 0.1m							
21	Vehicle Speed (MSB); Range (-327.65...327.65) m/s; Forward positive							
22	(LSBit = 0.01 m/s)							

Figure A.1: WSM packet structure.

byte	8 (MSB)	7	6	5	4	3	2	1 (LSB)
23	Heading (MSB); Range (0...65535); North is 0 deg, and Clockwise is Positive Angle							
24	(LSBit = 0.00549 degrees)							
25	Longitudinal Acceleration (MSB); Range (-327.65...327.65) m/s^2; Forward positive							
26	(LSBit = 0.01 m/s^2)							
27	Lateral Acceleration (MSB); Range (-327.65...327.65) m/s^2; Right positive							
28	(LSBit = 0.01 m/s^2)							
29	Vertical Acceleration; Range (-12.7...12.7) m/s^2; LSB = 0.08 m/s^2							
30	Yaw rate (MSB); Range (0...360.00) deg/sec; Clockwise positive							
31	(LSBit = 0.01 deg/sec)							
32	Brake Active Status LF = 0x01 RF = 0x02 LR = 0x04 RR = 0x08				Traction Control 00 = Not equipped 01 = OFF 10 = ON 11 = ENGAGED		ABS 00 = Not equipped 01 = OFF 10 = ON 11 = ENGAGED	
33	Steering Wheel Angle (MSB); Range (-655.36 ... 655.36) deg; Clockwise positive							
34	(LSBit = 0.02 deg)							
35	Throttle Position; Range (0 – 100) %; LSBit = 0.5 %							
36	Vehicle Exterior Lights; 8-bit enumerated values is defined: AllLightsOff = 0x00 LowBeamHeadlightsOn = 0x01 HighBeamHeadlightsOn = 0x02 LeftTurnSignalOn = 0x04 RightTurnSignalOn = 0x08 HazardSignalOn = 0x0C AutomaticLightControlOn = 0x10 DaytimeRunningLightsOn = 0x20 FogLightOn = 0x40 ParkingLightsOn = 0x80							
37	Vehicle Width (MSB); 10-bit; Range (0-1023) cm; LSBit = 1cm							
38	Vehicle Length (MSB)							
39	14-bit; Range (0-16383) cm; LSBit = 1cm							
	Obj tag for Path History							

Figure A.2: WSM packet structure continued.

APPENDIX B

IWIS 2.0 QUICK INSTALLATION GUIDE

B.1 Quick Installation Guide

1. Install Visual Studio (VC++) 2005

2. Add cygwin binary files to window environment: Control Panel - System. Select Advanced and Environment Variables buttons and edit the following system variable. Add to the end of already existing variables with a semi colon in front as shown below. Path=;c:/cygwin/bin

3. Install Cygwin (http://www.cygwin.com/) in C: directory. The default packets plus diffutils, gcc, gcc-g++, gawk, tar, gzip, make, patch, perl, w32api, bc, X11, perl, awk packets should be installed.

4. Install the NS-2 (Download link given in www.car.osu.edu/v2v) in C:/cygwin/ home/user name directory. NS-2 release version used in this project is ns-allinone-2.33. By default, a directory named ns-2.33 is created under C:/cygwin/home/user/ns-allinone-2.33.

5. Add the ns variables to the cygwin path. Edit or create c:/cygwin/home /usernamebashrc and add the following variables: Be sure not to convert cygwin file .bashrc to DOS format.

PATH=PATH:HOME/bin: HOME/ns-allinone-2.33/ ns-veh:HOME/ns-allinone-2.33/bin: HOME/ns-allinone2.33/tcl8.4.18/ unix:HOME/ns-allinone-2.33/tk8.4.18/unix:. export LDLIBRARYPATH =HOME/ns- allinone-2.33/ otcl-1.13:/home/uther/ ns-allinone-2.33/lib export TCLLIBRARY=HOME/ ns-allinone-2.33/tcl8.4.18/library

6. Now your environment is ready for installing IWIS Version 2.0. Extract The IWIS-2.0_setup.zip and run the install_iws.bat. After first copy is done it will ask for your cygwin user name which is usually the same as the windows user name. If everything from the previous steps of this guide are done properly, it will install the c++ portion of the project to c:/IWIS2.0 and copy the necessary files (ns-veh folder) to ns simulator. Moreover it will make all the necessary modifications on these files and build the wireless simulation part. When this long copy and build process is done, you can go and open IWIS2.0.sln from c:/IWIS-2.0. When you run this solution everything should work correctly.

B.2 Additional Information

C:/cygwin/home/user name/nsallinone2.33/nsveh/tcl/veh. This directory includes all the scripts and the result files. Cygwin is a Linuxlike environment for Windows. Cygwin DLL release version used in this project is 1.5.242. Check the requirement and installation tips at http://nsnam.isi.edu/nsnam/index.php/Running_Ns_and_Nam_ Under_Windows_9x/2000/XP_Using_Cygwin.

If you have a problem about Install GSL (http://www.gnu.org/software/gsl/) in any directory that is not in ns directory i.e., /home/username directory. Once installation is complete, run "make" in the gsl main directory to generate gslhistogram.exe. Copy the gslhistogram.exe into the project directory. Note that, the gslhistogram.exe that is available by default in cygwin does not run when invoked from a batch script. It is thus recommended to compile the gsl source and use the compiled version. Gsl-histogram.exe is invoked in order to calculate the latency probability.

APPENDIX C

DOCUMENTS CONTAINED IN THE ATTACHED DVD

The DVD attached with this thesis contains the thesis soft copy as well as the source code for all the simulators mentioned throughout the thesis.

Folder names with chapter number present the chapter that folder is affiliated with. In the folder for Chapter 3, there are three different folders containing projects. The readme.txt file explains how to install and execute the programs in this folder to get the simulator up and running. In the folder for Chapter 4, the projects for the vehicular simulator and the wireless network simulator are given in separate folders and the readme.txt file explains howto install and execute the simulator. Folder related to the Chapter 5, has all the versions of the IWIS simulator and each version is contained in a separate folder with the necessary readme.txt file explaining howto install and execute the program. For online information please refer to "http://car.osu.edu/v2v"

BIBLIOGRAPHY

[1] Electronic Toll Collection: Market Analysis and Technology Update. *Alexander Communications Group*, 1999.

[2] IEEE 802.11 wireless LAN medium access control (MAC) and physical layer (PHY) specifications, 1999.

[3] ASTM PS (draft) specifications for dedicated short range communication (DSRC) - data link layer; medium access and logical link control, 2000.

[4] IEEE 802.11p Draft Amendment for Wireless Access in Vehicular Environments (WAVE), 2006.

[5] A. Avila, G. Korkmaz, Y. Liu, H. Teh, E. Ekici, F. Özgüner, Ü. Özgüner, K. Redmill, O. Takeshita, K. Tokuda, M. Hamaguchi, S. Nakabayashi, and H. Tsutsui. A complete simulator architecture for inter-vehicle communication based intersection warning systems. *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pages 461–466, 2005.

[6] CSIM. "http://www.mesquite.com/".

[7] A. Dogan, G. Korkmaz, Y. Liu, F. Özgüner, Ü. Özgüner, K. Redmill, O. Takeshita, and K. Tokuda. Evaluation of intersection collision warning system using an inter-vehicle communication simulator. *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, pages 1103–1108, 2004.

[8] H.M. El-Sallabi. Fast path loss prediction by using virtual source technique for urban microcells. *Proceedings of the IEEE Vehicular Technology Conference*, pages 2183–2187, 2000.

[9] GloMoSim. "http://pcl.cs.ucla.edu/projects/glomosim/".

[10] J. Härri, F. Filali, C. Bonnet, and M. Fiore. VanetMobiSim: Generating realistic mobility patterns for VANETs. *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks*, pages 96–97, 2006.

[11] T. Hikita, T. Kasai, and A. Yoshioka. Integrated simulator platform for evaluation of vehicular communication applications. *Proceedings of IEEE ICVES'08*, Sept. 2008.

[12] Intelligent Transportation Society of America. "http://www.itsa.org".

[13] B. Kerner, S. Klenov, and A. Brakemeier. Testbed for wireless vehicle communication: A simulation approach based on three-phase traffic theory. *Proceedings of the IEEE Intelligent Vehicles Symposium (IV'08)*, pages 180–185, 2008.

[14] G. Korkmaz, E. Ekici, and F. Özgüner. Black-burst-based multihop broadcast protocols for vehicular networks. *IEEE Transactions on Vehicular Technology*, 56(5):3159–3167, 2007.

[15] Y Liu and Ü. Özgüner. Human Driver Model and Driver Decision Making for Intersection Driving. *Proceedings of the ITSC Conference*, pages 642–647, 2007.

[16] Y. Liu, Ü. Özgüner, and E. Ekici. Performance evaluation of intersection warning system using a vehicle traffic and wireless simulator. *Proceedings of the IEEE Intelligent Vehicles Symposium (IV'05)*, pages 171–176, 2005.

[17] Jun Luo and Jean-Pierre Hubaux. A Survey of Inter-Vehicle Communication. *IC Technical Report*, pages pp.1–12, 2004.

[18] Network Simulator NS-2. "http://www.isi.edu/nsnam/ns".

[19] B. Jarupan;Y. Balcioglu;E. Ekici;F. Ozguner;U. Ozguner. An Integrated Wireless Intersection Simulator for collision warning systems in vehicular networks. *Proceedings of IEEE International Conference on Vehicular Electronics and Safety,ICVES*, pages 340–345, 2008.

[20] F. Özgüner, Ü. Özgüner, O. Takeshita, K. Redmill, Y. Liu, G. Korkmaz, and A. Dogan. A simulation study of an intersection collision warning system. *Proceedings of the International Workshop on ITS Telecommunications*, Singapore, June 2004.

[21] D. Jiang M. Torrent Moreno L. Delgrossi Q. Chen, F. Schmidt Eisenlohr and H. Hartenstein. Overhaul of IEEE 802.11 modeling and simulation in NS-2. *Proceedings of the 10th ACM international symposium on modeling, analysis, and simulation of wireless and mobile systems*, pages 159–168, 2007.

[22] QualNet. "http://www.qualnet.com/".

[23] T.S. Rappaport. *Wireless Communications: Principles and Practice*. IEEE Press Piscataway, NJ, USA, 1996.

[24] K Redmill and Ü. Özgüner. VATSIM: A Vehicle and Traffic Simulator. *Proceedings of the ITSC Conference*, pages 656–661, 1999.

[25] K Redmill and Ü. Özgüner. VATSIM: A Simulator for Vehicles and Traffic. *Proceedings of the ITSC Conference*, pages 686–691, 2001.

[26] Automotive Company Research and Development Department. "Vehicle to Vehicle Communication Project 2008 Annual Presentation", 2008.

[27] K. Tokuda, M. Akiyama, and H. Fujii. DOLPHIN for inter-vehicle communications system. *Proceedings of the IEEE Intelligent Vehicles Symposium (IV 2000)*, pages 504–509, 2000.

[28] Kiyohito Tokuda. Application of Wireless Technology to ITS. *OKI Technical Review*, pages vol. 68, no. 3, pp.6–7,, September 2001.

[29] C. Young and B. Chang. A highway traffic simulator with dedicated short range communications based cooperative collision prediction and warning mechanism. *Proceedings of the IEEE Intelligent Vehicles Symposium (IV'08)*, pages 114–119, 2008.