2018

Study of Obstacle effect on the GPSR protocol and a Novel Intelligent Greedy Routing protocol for VANETs

Ravikumar Chilmula
Michigan Technological University, rchilmul@mtu.edu

STUDY OF OBSTACLE EFFECT ON THE GPSR PROTOCOL AND A NOVEL INTELLIGENT GREEDY ROUTING PROTOCOL FOR VANETS

By

Ravikumar Chilmula

A THESIS

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Electrical Engineering

MICHIGAN TECHNOLOGICAL UNIVERSITY

2018

This thesis has been approved in partial fulfillment of the requirements for the Degree of MASTER OF SCIENCE in Electrical Engineering.

Department of Electrical and Computer Engineering

Thesis Advisor:	*Dr. Aurenice M. Oliveira*
Committee Member:	*Dr. Glen E. Archer*
Committee Member:	*Dr. Zhaohui Wang*
Department Chair:	*Dr. Daniel R. Fuhrmann*

Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my deepest gratitude to my advisor, Dr. Aurenice M. Oliveira for being the cornerstone of this thesis and for being constant source of encouragement, guidance, and support. She consistently allowed this thesis to be my own work but steered me in the right direction whenever she thought I needed it. I would thank Dr. Glen Archer and Dr. Zhaohui Wang for being on the thesis committee and for giving the necessary feedback.

I would like to thank my parents Narsimhachari and Sunita, and my sister Rakshanda for their love, support, and for being my constant source of inspiration. I am indebted to my dearest friend Priya who is always with me in all my difficulties, happiness and for her unfailing support throughout my years of study.

I thank my best friends Thanmai and Aklesh for giving the necessary confidence to pursue Master's in the US. I thank my dear friend Sarala for reviewing my work and for invaluable constructive criticism while developing this thesis. Lastly, I would like to thank my friends at Michigan Tech: Kiran, Karan, Meghana, and Karthik for all the happy talks, prolonged debates,and the best dinner times. You guys made my Masters life, away from family, an easier and fun journey.

Finally, Thank You Michigan Tech for molding me into a better person.

List of Abbreviations

DSRC	Dedicated Short-Range Communications
E2ED	End-to-End Delay
GPSR	Greedy Perimeter Stateless Routing
IGR	Intelligent Greedy Routing
ITS	Intelligent Transportation Systems
MAC	Medium Access Control
MANET	Mobile Ad-Hoc Network
NS-3	Network Simulator-3
OSM	OpenStreetMap
PDR	Packet Delivery Ratio
RSU	Road Side Unit
SUMO	Simulation of Urban Mobility
VANET	Vehicular Ad-Hoc Network
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Anything
US DOT	United States Department of Transportation
WAVE	Wireless Access in Vehicle Environment

Abstract

In recent years, connected vehicle technologies have been developed by automotive companies, academia, and researchers as part of Intelligent Transportation Systems (ITS). This group of stakeholders continue to work on these technologies to make them as reliable and cost-effective as possible. This attention is because of the increasing connected vehicles safety-related, entertainment, and traffic management applications, which have the potential to decrease the number of road accidents, save fuel and time for millions of daily commuters worldwide.

Vehicular Ad-Hoc Network (VANET), which is a subgroup of Mobile Ad-Hoc Network (MANET), is being developed and implemented in vehicles as the critical structure for connected vehicles applications. VANET provides a promising concept to reduce the number of fatalities caused by road accidents, to improve traffic efficiency, and to provide infotainment. To support the increasing number of safety-related applications, VANETs are required to perform reliably. Since VANETs promise numerous safety applications requiring time-bound delivery of data packets, it is also necessary to replicate real-world scenarios in simulations as accurately as possible.

Taking into account the effect of realistic obstacles while simulating a variety of case scenarios increases the reliability of the tested routing protocol to appropriately perform in real-world situations. It also exposes routing protocols to possible vulnerabilities caused by obstacles. Nevertheless, it is not uncommon for researchers to omit real-world physical layer communication hurdles in simulation-based tests, including not considering the effect of obstacles on their routing protocol performance evaluation simulations. Consequently, the performance of these protocols is usually overestimated and do not support in real-world environment. Failure to account for obstacle effects overstate the network performance. In this thesis, a framework for measuring obstacle effects on routing protocols is defined. We also propose, a new routing protocol based on the traditional Greedy Perimeter Stateless Routing (GPSR) protocol called Intelligent Greedy Routing (IGR) protocol.

The proposed IGR protocol considers a parameter called *Receptivity* to chose the next hop in a route. We implemented the new protocol using the Simulation of Urban Mobility (SUMO) and the Network Simulator (NS-3). An analysis of Packet Delivery Ratio (PDR), End-to-End Delay (E2ED) and Mean Hop count with the assumption that nodes (vehicles) are moving in various topologies is presented in this thesis. The study presented here gives a general idea of the effects of obstacles on the Greedy Perimeter Stateless Routing (GPSR) protocol considering multiple realistic scenarios such as Urban, Residential and Highway. In addition, we compare the performance of GPSR and the new IGR protocols with the presence of obstacles considering various topologies. The new proposed IGR protocol performs better compared to the traditional GPSR for all the investigated metrics.

Chapter 1

Introduction

Vehicular communication has emerged as a critical technology for safety applications as it has the potential to prevent collisions and save thousands of lives yearly. Over five million crashes occur on U.S. roads and highways every year, of which over 30,000 include at least one fatality [1]. The potential to prevent up to 80% of these accidents has fueled researchers, industry, and governments to invest in vehicle communication and autonomous vehicle technologies [2].

The connected vehicles technology implemented through a wireless communication network is called Vehicular Ad-Hoc Network (VANET). VANET is a subclass of Mobile Ad-Hoc Network (MANET). It provides a promising concept to make users aware of the real-time information like crash scenarios, weather information, road conditions,

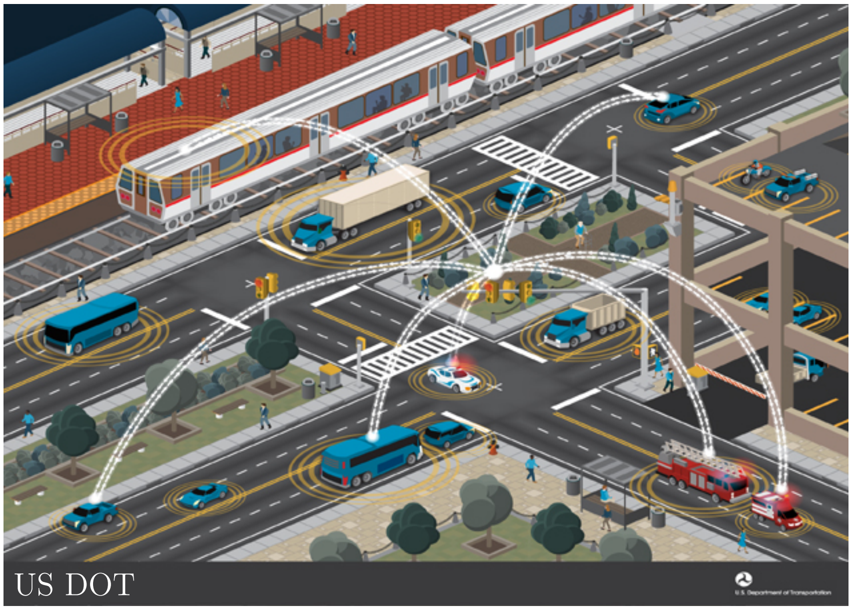


Figure 1.1: Implementation of Connected Vehicle Technology

and so on. This can significantly reduce the number of fatalities caused by road accidents, improve traffic efficiency, and provide infotainment. It is also predicted that inter-vehicle communication will become a major component in autonomous (driverless) vehicles. According to reports from the U.S. Department of Transportation, the potential connected vehicles transportation applications for public safety and traffic management include warnings such as: blind spot; forward collision; sudden braking ahead; do not pass; approaching emergency vehicle; transit or emergency vehicle signal priority; electronic parking and toll payments; and traffic and travel condition data to improve traveler information and maintenance services.

Fig. 1.1 illustrates the implementation of typical VANET applications. VANET is a technology that uses moving vehicles as nodes to create a mobile network. It turns every participating vehicle into a wireless node, allowing vehicles approximately 300-1000 meters distant from each other to connect and create a network with a wide range. As vehicles fall out of the signal range and drop out of the network, other vehicles join in, connecting vehicles to one another so that it creates a mobile network. With connected vehicle technology, drivers will get warning messages in their vehicles, for instance, when a potential crash is imminent.

Existing wireless protocols for Mobile Ad-hoc Networks (MANETs) cannot be directly applied to Vehicular Ad-hoc Networks (VANETs) because of their unique characteristics. VANETs main characteristics include high dynamic topology, sufficient energy and storage space, topology changes, and critical latency requirements. The highly dynamic nature of VANET topology leads to frequent disconnections and excessive overhead generation due to route repairs. Moreover, the network topology in VANETs changes with the time of day, as well as with location. For example, when a vehicle moves from highway environment into an urban environment the density of nodes surrounding it changes drastically, and so does the quality and quantity of network obstacles. Therefore, developing and choosing a robust routing protocol for VANETs is challenging. Vehicle velocity, direction, and location predominately affect the efficiency of the protocols. Researchers have been working on many issues, such as routing, Quality of Service(QoS), applications, protocols, and standards, just to name a few. Routing is the process of selecting the best path to send messages in a network. The design of ad-hoc network routing protocols is challenging because of the constant changes in the network.

The main ways to implement connected vehicle's communications as suggested by ITS [3] are: vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications collectively known as V2X.

- “Vehicle-to-Vehicle (V2V) Communications for Safety is the wireless exchange of data among vehicles (including cars, trucks, transit vehicles, and motorcycles) traveling in the same vicinity”.
- “Vehicle-to-Infrastructure (V2I) Communications for Safety is the wireless exchange of critical safety and operational data between vehicles and roadway infrastructure”.

1.1 Literature Review

The idea of communication between the vehicles using radios is a concept known well before the digital communication times. A patent ”Radio Warning Systems for use in Vehicles” was issued in 1925, which mentions the concept of peer-to-peer communication using similar systems installed on both vehicles [4]. Later on, various forms of communication between vehicles were used in the military for safety purposes. However, there was not much attention towards this technology for public vehicles until 1984. Starting in 1984, the hardware of“Radios Data Systems” started to be installed on new vehicles to facilitate Frequency Modulation (FM) radio broadcasts [4]. It was the first example of Infrastructure-to-vehicle type of communication, however, it was unidirectional communication. The first bidirectional communication system was introduced using Radio Frequency IDentification (RFID) for tolling systems in 1980s [4]. In the early 1990s, Philips proposed the idea to have Dedicated Short Range Communication (DSRC) systems to operate at 5.8 GHz for use in the called “Intelligent vehicle-highway systems” (IVHS) [4].

In 1999, the US Federal Communications Commission (FCC) allotted 75 GHz in the spectrum of 5.850-5.925GHz for Intelligent Transport Systems (ITS). Until the early 2000s, there was not a standardized network plan suitable for vehicle communication. The book titled ”Ad hoc mobile wireless networks: protocols and systems” authored by Chai K. Toh introduced the term VANET in 2001. It was referred as an application of MANET allowing cars to form an ad hoc network to help overcome blind spots, avoid accidents, and so on. It was later considered as a sub-class of MANET and got significant attention due to its ability to solve road safety and traffic issues.

Due to the dynamic nature of VANET, the process of passing the information in a time-efficient way was a point of concern. Safety-related applications are very time sensitive and latency easily becomes an issue as some applications require very low latency. Forwarding the information is a process of relaying the data packets to the destination by intermediary vehicles on behalf of the source. A Routing protocol defines this process. Routing as first defined in ”Internet Protocol–DARPA Internet Program Protocol” [5] is ”a process selection of a path for transmission.” The nodes/routers/Internet modules use the addresses carried in the Internet header to transmit Internet datagrams to their destinations using the routing protocol [5]. Routing is challenging in VANETs due to the varying density of the vehicles. Many researchers have proposed different types of routing protocols based on topology, position, cluster, traffic based and so on.

One of the first proposed protocol for wireless is Ad-Hoc On-demand Vector routing (AODV) protocol [6]. It establishes a route based on calculating the distance sequence number based on the request by the source node. Although it provides up-to-date path information, suitable for VANET implementation, and less memory requirement, the drawback is the time taken to setup a route, consuming high bandwidth and inconsistency in providing a standard route [7]. Another drawback is the issue of scalability when facing higher density of nodes, as expected in urban VANETs.

The issue of scalability was successfully tackled by the use of geography-based routing, which in general does not require any topological information other than the positions of destination nodes and the positions of potential next-hop nodes. The readily available location information from vehicle's Geographic Positioning System

(GPS) made position-based or geographic routing protocols very popular. Position-based protocols have proved to be a promising approach towards developing routing protocols for VANETs. The Greedy Perimeter Stateless Routing (GPSR) protocol with the use of geography-based routing, first proposed for wireless networks by Karp and Kung[8] in 2000, was one of the protocols that successfully tackled the issue of scalability. A typical topology based routing protocols requires them to maintain routing tables and needs to be frequently updated. A high bandwidth is consumed to maintain these routing tables. GPSR and in general, any geography-based routing protocol do not require any topological information other than the positions of destination nodes and the positions of potential next-hop nodes. Packet forwarding decisions in GPSR are done based on greedy forwarding algorithm where nodes closest to the destination are chosen to receive the packet. The perimeter forwarding algorithm, which is the recovery mechanism of GPSR, makes use of the right-hand rule when a node encounters the problem of local maximum [8].

Another approach to solving problems in urban environments is Geographic Source Routing (GSR) which is proposed by Lochert et al. [9]. In this protocol, the routing decisions are made based on a map of the environment, avoiding the routes where there is a possibility of disconnection due to obstacles.

Although a universal routing protocol which can work in all the scenarios and applications is not possible at this time, designing a VANET routing protocol that can work in most of the scenarios is still achievable. Designing such a protocol is a multi-faceted problem, and involves consideration of the specific application and type of implementation scenario.

1.2 Motivation and Research Questions

Several studies have been conducted in the field of VANETs, however, due to large vehicle system complexity more research and testing needs to be performed for VANETs to be implemented flawlessly and save lives. Simulations have always been an integral part in testing research ideas, especially in VANETs where real testbeds are still expensive or not easily accessible and where real traffic experiments with a large number of nodes (vehicles) are usually not feasible.

Since VANETs promise numerous safety applications requiring time-bound delivery of data packets, it is necessary to replicate real-world scenarios in simulations as accurately as possible. Nevertheless, it is not uncommon for researchers to omit real-world physical layer communication hurdles in simulation-based tests. For instance, many researchers do not consider the effect of obstacles on the routing protocol in performance evaluation simulations they have presented. Consequently, the performance of these protocols often overestimated and deviated from the realistic behavior [10].

According to the research conducted by Harri, Tchouankem et al. [11], the communication range may significantly depend on the antenna location and direction of the destination w.r.t source. Failure to account for the effects of obstacles can, therefore, inaccurately overstate network performance. In this thesis, a framework for measuring obstacle effects on routing protocols is defined. We also propose, a new routing protocol based on the traditional Greedy Perimeter Stateless Routing (GPSR) protocol called Intelligent Greedy Routing (IGR) protocol.

Two main research questions contributed to this thesis:

- **Research Question 1:** How significantly real-world obstacles affect the performance of VANETs position-based routing protocols?
- **Research Question 2:** Is there a realizable way to design a more efficient VANET routing protocol considering real-world obstacles? If yes, how can such a routing protocol be designed?

1.3 Thesis Structure

This thesis is organized as follows:

Chapter 1: We provided an overview of connected vehicle technology, a summary of the history of connected vehicles, and a literature review of VANETs. Moreover, we provided the motivation behind the study described in this thesis and its overall structure.

Chapter 2: We introduce VANET, explain its main characteristics, challenges, and applications. We also explain the main U.S. VANET standards and its stack of protocols including descriptions of the MAC layer, Physical layer, IEEE802.11p, DSRC, WAVE, Mobility models, security and privacy, and NLOS issues.

Chapter 3: We briefly explain the simulation tools necessary to implement VANET routing protocols such as Network simulator 3 (NS-3), Simulation of urban Mobility (SUMO) and Open street maps. We also describe different VANET applications scenarios setup such as Urban, Highway, and Residential.

Chapter 4: We introduce the Greedy Perimeter Stateless Routing (GPSR) protocol with the simulation setup and a framework of how an obstacle shadowing model was added to the GPSR implementation. The chapter also contains information about setting up the application scenarios and how the GPSR can be analyzed with obstacle shadowing, as well as the combination of obstacle shadowing and free-space models. The simulation results are presented showing the performance of the GPSR protocol for three different scenarios with obstacles and without obstacles. An assertion of how significantly obstacles affect the GPSR protocol is explained in this chapter.

Chapter 5: We present the proposed new routing protocol: Intelligent Greedy Routing (IGR). This routing protocol can overcome the effects of obstacles as observed in *Chapter 4*. The developed algorithm for the new proposed protocol is also explained. In addition, we present the simulated results for the performance comparison of the traditional GPSR and the Intelligent Greedy routing (IGR) in the presence of obstacles. The performance metrics considered for comparison are packet delivery ratio (PDR), mean hop count and end-to-end delay.

Chapter 6: We present our conclusions and suggest areas of improvement for future work.

Chapter 2

Overview of VANET

VANET is the supporting network for ITS services. An ad-hoc network is a collection of wireless nodes that can communicate among themselves without the support of any additional infrastructure, such as routers or base stations. In VANET, each vehicle acts as sender, router, and receiver. VANETs main characteristics include high dynamic topology, sufficient energy and storage space, topology changes, and critical latency requirements. VANETs are challenging networks because of the nodes high mobility and frequent link disruption.

As described in Chapter 1, there are three basic types of communication in VANETs, V2V, V2I, and I2V. In V2V communication, the nodes (vehicles) communicate with each other directly (single-hop) if in the wireless range or indirectly in a multi-hop mode. Each vehicle has an on-board unit (OBU) with Dedicated Short-Range Communication (DSRC) capabilities. In I2V communication, the infrastructure broadcasts messages to the nodes (moving vehicles), providing road conditions and traffic information. The wireless network infrastructure uses road side units (RSUs) as wireless access points.

Although VANET is a sub-class of MANET, specific standards were developed for VANET mainly because of the type of applications it handles, especially time-sensitive safety-related applications. The standards and regulations for communication in VANETs are established and maintained by agencies and organizations such as Intelligent Transportation Systems (ITS) Program of U.S. Department of Transportation, American Society for Testing and Materials (ASTM), Institute of Electrical and Electronics Engineers(IEEE), and Society of Automotive Engineers (SAE). The following sections provide a brief overview of the components of a connected vehicle, specific set of challenges, standards, security and privacy issues, routing protocols and physical layer issues associated with VANETs.

2.1 Basic Components of a Connected Vehicle

The Intelligent Transportation Systems (ITS) Program of U.S. Department of Transportation has proposed some basic components required by a connected vehicle [3] as

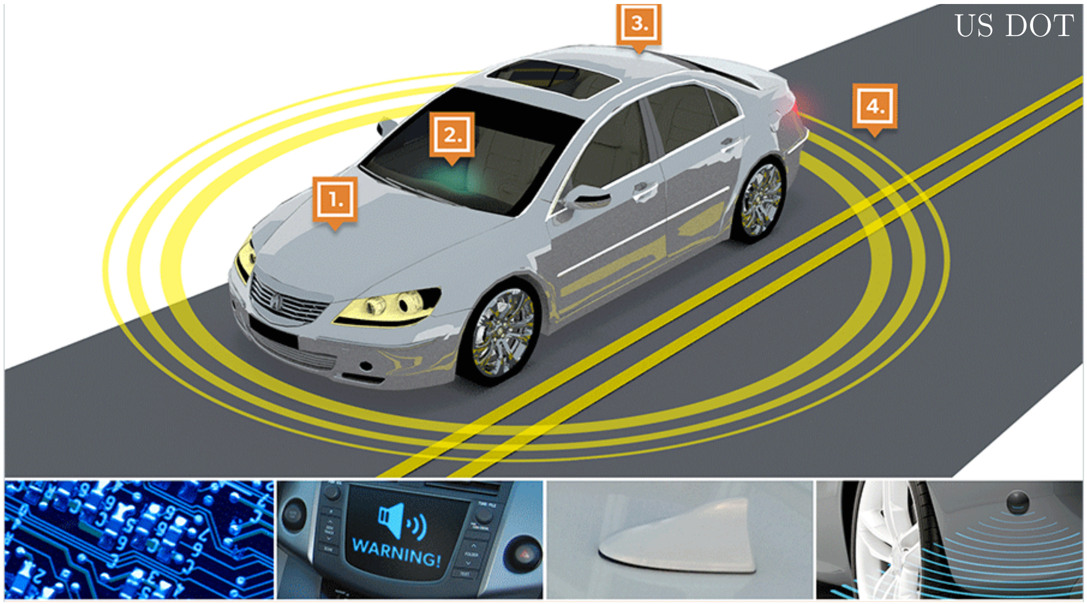


Figure 2.1: Components of a Connected Vehicle.

shown in Fig. 2.1. They are defined as follows:

- **Processing Unit:** It commonly consists of a processor and a memory. It collects and processes the data received from the On Board Equipments (OBEs) and transmits through an antenna to nearby vehicles. It is the brain behind the decisions made to process VANET information.

- **Display:** When a processing unit receives information, depending upon the information, a display works as an indicator to warn or inform the driver about the critical information obtained. Warnings such as lane changer alert, sudden brake alert or a crash event are informed using the display.

- **Antenna:** It contains a DSRC module that receives and transmits signal from the antenna's of the vehicles in range. Acts as entry and exit of information for the vehicle.

- **Sensors:** Obtains surrounding information like obstacles and detects emergency situation such as imminent collision. Sensors give 360 degree awareness of the potential hazards and emergency situations. In case of self-driving cars, more information can be available with wide-range of embedded sensors on the vehicle.

2.2 Challenges in VANETs

Despite VANETs many advantages, several challenges still need to be addressed by the research community and industry. Due to the dynamic nature of VANET, some of the challenges and constraints are:

- **Connectivity:** A critical issue in VANETs is the maintenance of continuous connectivity between vehicles. Due to the dynamic network topology, i.e., the number of vehicles in the network always varying as well as their position and speed, frequent disconnections are observed.

- **Latency:** Latency is the time taken for systems to respond when a signal is received. Many of the applications rely on time-bound delivery of packets. Hence, the On-board unit (OBU) needs to have a low response time and should be able to decide the next step in a time-critical approach. Latency becomes crucial for applications based on emergency broadcasting because VANET has a very tight requirement for latency as the delay of information can be quite harmful in some cases. The IEEE defines the latency for safety application in VANETs to be no more then 50 ms and for non-safety applications as much as 100 ms [12].

- **Limited bandwidth:** The DSRC standard established channel bandwidth as 10MHz. Some channels can be paired to form a 20MHz channel. This relatively low bandwidth can be a limitation to the applications requiring varied/high bandwidth, such as infotainment applications.

- **Congestion control:** Vehicle density varies widely with the type of scenario involved. For instance, high vehicle density in Urban scenarios make the network congested or leads to packet loss.

- **Cooperative communication:** VANET is a technology involving cooperation from the vehicle owners to be a part of the network. Cooperation is a essential factor which can affect multi-hop communication as intermediate vehicle in a route not cooperating to pass the information to further vehicles can disrupt the network.

- **Security and privacy:** Proper authentication to access other vehicles data should be provided. Private information such as vehicle location, drivers personal information should not be shared in the network. This information also needs protection from cybersecurity threats.

- **Reliability:** Safety applications such as alerting about crash scenarios to the nearby vehicles/drivers can save lives. This type of application relies on the assumption that the emergency information is delivered to the appropriate person/vehicle.

- **Implementation of cross-layer protocols:** MAC layer in VANETs is defined by a single MAC protocol. This limits handling of different prioritized messages.

- **Computing and Energy requirements:** Some of the routing protocols associated with VANETs need high processing power, requires efficient processing unit, and may demand more energy. Failure of the processing unit or energy system can be an issue in a multi-hop communication.

- **Frequent exchange of information:** A typical routing protocol works on frequent beaconing of information from the nearby vehicles and Road-side Units (RSUs). This requirement can consume large amounts of bandwidth. An efficient routing protocol design can mitigate this challenge.

- **Attenuations:** The Wireless Access in Vehicular Environment (WAVE) standard ensures to have the transmission power making data reachable to a distance up to 1000m for line of sight (LOS) and up to 300m for non-line of sight

(NLOS) . However, obstacles such as buildings, vehicles and environmental obstacles attenuate the signals. Factors like reflection, diffraction, and dispersion are inevitable to the signal propagation. These factors lead to fading effects, Doppler effects, losses and so on. These issues can be solved using environment-aware routing protocols or placement of RSUs where these effects are strong.

2.3 VANET applications

Applications in VANETS are categorized broadly as Safety applications and Non-Safety applications.

2.3.1 Safety Applications

The main agenda for safety applications in VANETs is to prevent vehicle collisions. The sensors available in vehicles can monitor the environment, collect real-time traffic information and pass the information among vehicles using VANET [13]. Some of these applications are depicted in Fig. 2.2. Displaying simple warnings based on the activity of nearby vehicles is one of the VANET salient applications. For instance, warnings such as Electronic Brake Warning (EBW), Lane Change Warning (LCW), Intersection Violation Warning (IVW), and Curve Speed Warning(CPW) help the

Figure 2.2: Safety Applications in VANET (Source: U.S. DOT)

driver's consciousness in driving. An EBW warns the driver when the preceding vehicle attempts a sudden brake; an LCW warns the driver if he/she intends to change the lanes over a blind spot zone, and an IVW warns the driver and nearby authority when he/she violates the traffic signal. A CSW warns the driver about an approaching vehicle in a blind zone during a curve. These warnings come under cooperative driving applications. Another application is a warning to the driver just before a possibility of an accident. According to the Rajkumar, Nithya et al. [14], a warning just about half a second before a crash can avoid 60 percent of accidents [14].

Another warning *Post Crash Notification* is warning message broadcasted by the vehicle involved in an accident about its position and situation of the vehicles behind it so that the other vehicles can take appropriate actions. With a Road Hazard Control Notification, a vehicle can notify other vehicles about a road having landslide or information regarding road feature such as a road curve, sudden downhill and so on.

The eight main application identified by several U.S. transportation departments in 2006 are listed below [15]. The list of envisioned safety applications goes on, and there are bound-less safety applications.

- Pre-crash sensing
- Emergency brake lights
- Collision warning
- Traffic signal violation
- Curve speed warning
- Left turn assist
- Stop sign assist
- Lane change warning

2.3.2 Non-Safety Applications

The agenda of non-safety applications is to bring convenience, traffic efficiency, infotainment and comfort to drivers and vehicles. The readily available real-time traffic information can also be considered as a non-safety application as it helps the driver to navigate better. The advantage of the non-safety applications is that they do not require standardization and cooperation among vehicles, as these applications are individual vehicle's basis [13]. These applications can also be linked with smartphones to increase their scope. Convenience and efficiency applications provide weather or traffic information and the location of restaurants or hotels nearby. Entertainment applications provide internet services, media downloading, media sharing, and online gaming. Although safety applications have high priority, non-safety applications are also considered by researchers and the industry. They work continuously to develop, improve and implement new non-safety applications in the near future. Non-safety applications can serve as a large source of revenue for businesses and stakeholders.

2.4 Standards and regulations (DSRC/WAVE)

In the U.S., the standards and regulations for communication in VANETs are established and maintained by agencies and organizations such as Intelligent Transportation Systems (ITS) Program of U.S. Department of Transportation, American Society for Testing and Materials (ASTM), Institute of Electrical and Electronics Engineers (IEEE), and Society of Automotive Engineers (SAE). These agencies and organizations develop theses standards in consultation with other government agencies, academia and automotive manufacturers.

The architecture, services, interfaces and a set of standard protocols for VANETs often defined as Wireless Access in Vehicular Environment (WAVE) is significantly different from the Wi-Fi standard and cellular environment. WAVE is the current U.S. chosen technology suitable for VANETs because it has the short latency necessary for road safety control and messaging. These standards come under the Dedicated Short Range Communication (DSRC) standards. The communication standards specified under DSRC are IEEE 802.11p and IEEE 1609. In the U.S., a 75 MHz spectrum is allocated by ITS for DSRC in the 5.9 GHz frequency band [12]. The spectrum is configured with one control channel (CCH) reserved for carrying high priority messages and six service channels (SCHs) for safety-related and non-safety applications. The band and allocated channels are portrayed in the Fig. 2.3. The range of transmission

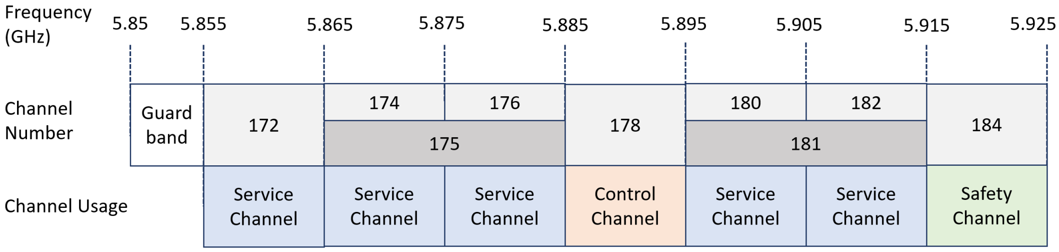


Figure 2.3: Allocated DSRC frequency band in U.S.

power allowed by DSRC is 0 dBm to 28.8 dBm depending on the type of vehicle [16]. The DSRC set of standards also describes the message set SAE J2735 consisting of 15 message types depending upon application purpose[4]. Some of these are Basic Safety Messages (BSMs), Emergency Vehicle Alert Message, Roadside Alert, and Common Safety Request. The standards IEEE 802.11p and IEEE 1609 contribute to different layers of DSRC as compared to a simple OSI model is shown in Fig. 2.4.

Standard IEEE 802.11p

The IEEE 802.11p is the standard applied to provide messages transmission between vehicle nodes in VANET, thus it has a significant influence on the efficiency of VANET. The primary challenge of the physical layer is the VANET propagation concerning outdoor environment and high-speed mobility. IEEE802.11p consists of PHY and MAC layers. The PHY layer of IEEE 802.11p manages the interface between MAC layer and the media that allows sending and receiving frames. The Physical

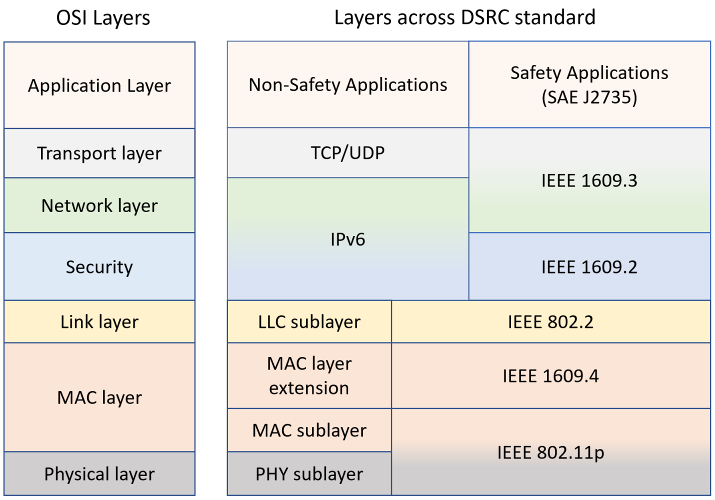


Figure 2.4: The layered architecture for DSRC as compared to the standard OSI model.

layer (PHY) of VANET is responsible for hardware specification, bits conversion, signal coding and data formatting [12]. Orthogonal Frequency Division Multiplexing (OFDM) technique with 64 sub-carriers is employed for wireless communication as a part of PHY layer. PHY layer consists of two sublayers: Physical Layer Convergence Protocol (PLCP), which is responsible for communicating with MAC layer, and Physical Medium Access (PMD), which manages data encoding and modulation. The MAC layer is responsible for sensing the medium for idleness and then transmit. IEEE 802.11P MAC is a contention-based MAC protocol where the node listens to the medium whether it is free or occupied. To avoid message collision in MAC, the IEEE802.11p employees Enhanced Distributed Channel Access (EDCA) where high-priority messages get access to the medium first [17].

Standards: IEEE 1609.2, IEEE 1609.3, and IEEE 1609.4

The middle layers of transport, network and upper MAC layer and other security features are defined under the IEEE 1609 standards. IEEE 1609.2 outlines standard security mechanisms and authentication involved in safety applications. It uses a combination of symmetric and asymmetric cryptography [15]. IEEE 1609.3 is the standard for network and transport layer services. IEEE 1609.4 is the upper MAC layer extension responsible for the multi-channel operation, providing ease in switching between channels and tuning of safety messages to their respective channels[4].

2.5 Routing protocols in VANETs

To ensure reliable, continuous and secure communication among the vehicles is a challenge for VANETs due to moving vehicles. The topology of the network varies often making it difficult to establish fixed routes. The vital consideration for routing protocols to work in VANETs efficiently is the elimination of establishing and saving of routes. This state is due to the high mobility nature of nodes in VANETs. A saved route will not be working most of the time, as the nodes involved in the route may change their position. Therefore, routing protocols in VANETs need to be designed in such way that they dynamically establish routes and maintain them through a process of constant communication between nodes via beacons [18]. According to the authors of [19], VANETs routing protocols are classified based on the strategy used to route the packets from source to destination. A brief discussion about each category is presented below.

- **Topology-based:** Establishes link information between nodes with periodic updates. The routing protocols under topology-based can further be classified into proactive, reactive and hybrid. Proactive routing protocols maintain lists of destinations and their routes in route tables and periodically update the nodes in the network. Reactive routing protocols find the route based on the demand of the source node by flooding the network with Route Request packets. A hybrid approach is a combination of both of these types. Due to periodic updates used in topology-based routing protocols, they experience additional overhead leading to large delays. Ad-Hoc Demand Vector(AODV) routing, Distance-sequenced distance vector (DSDV) Dynamic Source Routing (DSR), Zone Routing Protocol (ZRP), Optimized link state routing (OLSR) are examples of topology-based protocols.

- **Cluster-based:** Nodes based on vicinity try to create a cluster and a cluster head is chosen [19]. This cluster head is in turn responsible for managing intra/inter-cluster communications. The advantage of cluster-based protocols is their scalability, the disadvantage is high overhead and long delay. Examples of cluster-based protocols are: Cluster-based routing (CBR), Clustering for Open IVC Networks (COIN) and Cluster-based directional routing protocol (CBDRP).

- **Geocast-based:** This type of protocol forwards packet to the vehicles in a particular geographical region called a Zone of Relevance (ZOR) [20]. Hence, packets outside the ZOR will not receive packets. Some geocast routing protocols are Robust Vehicular Routing (ROVER), Cached Geocast Routing (CBR), and Inter-vehicle Geocast Routing (IGR). The pros include reduced overhead, and reduced network congestion, and the con is that undesirable neighbors preventing forwarding of packets.

- **Broadcast-based:** This is a simple routing scheme where packets are broadcast to the network and are mostly used for broadcasting of emergency information such as crash incidents. These protocols are efficient for small networks. Some of the broadcast protocols proposed are BROADCOMM, Distributed Vehicle Broadcast (DV-CAST), and Density-Aware reliable broadcasting (DECA). The disadvantage of broadcast protocols is the high chance of collision and heavy bandwidth utilization.

- **Multi-cast based:** In this type of protocols, the packets are forwarded to the group of nodes. The selection of the type of nodes may be based on geographical area, address based, type of vehicle (cars, buses or trucks). Some examples include Multi-cast On-demand Vector (MAODV), Adaptive Demand-Driven Multicast routing (ADMR), and Destination Driven On-demand Multicast Routing (D-ODMRP). These protocols are proven to be practical in MANETs but need appropriate changes to efficiently work in VANETs.

- **Position-based:** Position-based routing protocols use the geographic position of the nodes to route the packets. The position usually is available via the Global Positioning System (GPS) data from the vehicle. In this type of protocol, the source node must know its location and the location of the destination node. The efficiency of a position-based routing protocol is dependent on driving environments. Numerous position-based routing protocols are proposed for VANETs such as Greedy Perimeter Stateless Routing (GPSR), Geographic Source Routing (GSR), Greedy Perimeter Coordinator Routing (GPCR), Movement Prediction-based Routing and A-STAR (Anchor-based Street Traffic-Aware Routing). The GPSR routing protocol forwards the packet to a node which is nearest to the destination. The GSR finds the route using Dijkstra shortest path algorithm based on the map data available. The A-STAR protocol is similar to that of GSR but uses an additional weighing feature to select the nodes for a route. The GPCR protocol is designed to overcome obstacles by assigning nodes at the junction as coordinators. When the performances of different routing protocols are compared, the position-based Greedy

Perimeter Stateless Routing (GPSR) protocol is considered more suitable for VANETS considering real-time traffic, throughput, and higher mobility models [21]. Many routing protocols proposed recently use the basic idea of GPSR and modify it accordingly with the necessity of the topology/scenario/application.

2.6 Non-line of sight (NLOS) issues for Routing protocols in VANETs

Wireless signals emitted by vehicles traveling through a downtown or urban scenario will most likely encounter Non-Line of Sight (NLOS) situation, which can affect signal propagation [22]. According to the Federal Communications Commission (FCC) proposals for DSRC devices, the desired communications range is from 15 to 1000 meters with typical operation expected in the 400 m range [10]. While vehicles and infrastructure expect to communicate reliably over these ranges, radio-blocking obstacles and other interference that prevent message delivery challenge the effectiveness of all the safety applications [10]. Moreover, communicating at 5.9 GHz is challenging for NLOS transmissions, as waves experience refraction, reflection, fading, among others by almost any obstacle [11]. In city scenarios, baseline position-based routing protocols have difficulties to handle two-dimensional scenarios with obstacles (buildings) and voids [23].

Many proposed protocols are built based on the idea that the signal may reach the next node in the route, but the physical challenge of NLOS can significantly effect the propagation of the signal. Designing and simulating routing protocols without taking into consideration obstacle effects can lead to inaccuracy and overestimation of network performance [10]

Urban scenarios with almost all the area between streets covered with buildings significantly limits the applicability of purely greedy position-based routing and corresponding recovery strategies [23]. Due to these obstacles, nodes that would have seen each other in a free space model might not be aware of each other anymore [23]. Planarization methods proposed for greedy based protocols are also affected by obstacles. The authors in [10] have investigated how to appropriately include realistic obstacles into simulations leading to a promising simulation model considering loss due to obstacles – the Obstacle Shadowing Model – which is based on the empirical formula provided by authors in [24].

Chapter 3

Simulation tools

The low penetration rate (number of vehicles with OBUs), high complexity, and high cost involved in the development, evaluation, and implementation of connected vehicle solutions requires large-scale performance evaluations that are not frequently explored in field test experiments. Real testbeds are not widely available or easily accessible. Furthermore, field tests require a large number of vehicles, which is expensive and difficult to control. Thus, systematic assessment is difficult. As a result, most of the VANET research on protocol design has relied on simulations. Simulation is the most efficient tool for performance evaluation of VANET routing protocols (network layer) and is expected to remain the most used evaluation tool for the next several years.

Simulation of VANET protocols and applications requires two main components: a traffic simulator and a network simulator.

- **Traffic Simulation:** A traffic simulation consists of creating a map/scenario with roads and buildings as needed. It helps in creating the traffic environment and constructing traffic utilities: roads, vehicles, lanes, real-time parameters and build real-world like scenarios. The vehicles are deployed with information such as position, route, velocity and so on. Some traffic simulators available that are open-source are: SUMO [25], MOVE [26, 27], CityMob [28, 29], VanetMobiSim [30], STRAW [31],and Freesim [32]. Commercial simulators include VISSIM [33], NETSTREAM [34], CORSIM [35], and PARAMICS [36].The most commonly used mobility modeling software is SUMO (Simulation of Urban Mobility). Real-time maps can also be imported into this software from OpenStreetMap [37]; hence it allows simulations as close as possible to a real-world environment.

- **Network Simulation:** A network simulation consists of creating a network, establishing communication among the nodes and analyzing its performance. The most commonly used network simulators are Network Simulator 3 (NS-3) [38], GloMoSim [39], SNS [40], GTNetS [41], JiST/SWANS [42], Oualnet [43], OMNET++ [44], NetSim [45], J-Sim [46] and OPNET [47, 48], of which NetSim and Qualnet are commercial software.

To perform simulations for this thesis, the tools used were OpenStreetMap, SUMO, and NS-3, which are discussed in detail in sections 3.1, 3.2 and 3.3.

3.1 OpenStreetMap (OSM)

OpenStreetMap (OSM) is the project created by OpenStreetMap Foundation which provides free geographic data for the entire world. It is an open-source platform built by volunteers across the world providing map data with intricate details which are essential for simulating platforms such as SUMO to produce real-world like experiments. In our case, to perform simulations for VANETS, we chose different scenarios urban, highway, and residential scenarios depending on the locality, building structure, and placement, among others. Using the OpenStreetMap Website [37], the region of interest over a map is selected and exported as a file with extension *.osm*. The map represented in the Fig. 3.1 is of Houghton County exported using OSM.

3.2 Simulation of Urban Mobility (SUMO)

As stated in the SUMO official pages [25, 49] "Simulation of Urban Mobility (SUMO) is an open source, highly portable, microscopic and continuous road traffic simulation package designed to handle large road networks" [25]. It was developed by Institute of Transportation Systems at the German Aerospace Center. The term *microscopic*

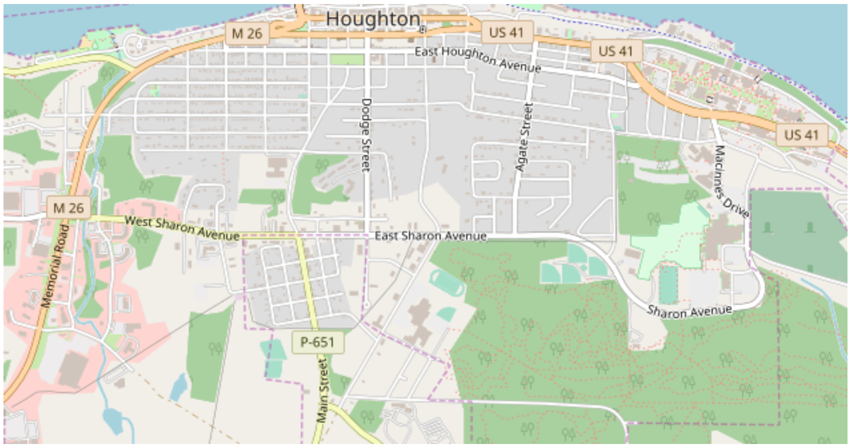


Figure 3.1: OpenStreetMap for the Houghton County, MI.

indicates how detailed the simulations can be defined such as the vehicle's speed, route, departure time, type of the vehicle and an explicit identifier for each vehicle in the network. The salient features of SUMO include collision free vehicle movement, different vehicle types, single-vehicle routing, multi-lane streets with lane changing, junction-based, right-of-way rules, a hierarchy of junction types, an OpenGL graphical user interface (GUI), and dynamic routing [50]. SUMO performs simulations in a time-discrete manner, i.e., the information of vehicles such as position, velocity, and direction are stored in time steps (the default is 1 second). SUMO can be operated in two ways: One is using the command line application, and another is using the SUMO-GUI. In this thesis, to create different scenarios such as Urban, Highway or Residential with vehicles deployed at different speeds and positions, SUMO command line application is used. The map of Houghton County, MI imported to SUMO and displayed in SUMO-GUI is shown in Fig. 3.2. The flowchart depicted in Fig. 3.3 shows different files created by SUMO application and the flow to generate such

Figure 3.2: SUMO-GUI for the Houghton County, MI

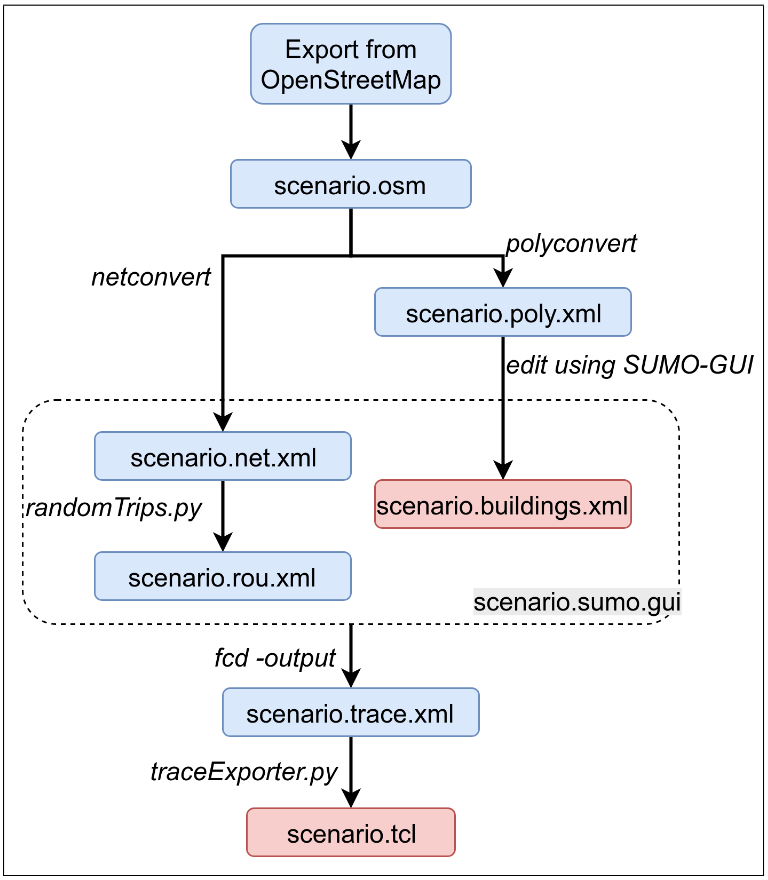


Figure 3.3: Flow-chart to generate tcl file using SUMO

scenario. The generated files and the commands used for SUMO to create scenario compatible with NS-3 are described next.

scenario.net.xml : *NETCONVERT* is command line application. This command can be used to detect road networks and export them to XML extension files. The command shown below has an input of osm file *scenario.osm*, which then extracts the road networks from osm file and exports network file as *scenario.net.xml*. A network file consists of a list of nodes and edges which can represent a road network.

```
$ netconvert −−osm−files scenario.osm −o scenario.net.xml
```

typemap.xml : A typemap.xml is downloaded from the SUMO official website [25] and can be used to detect features such as rivers and buildings from the osm file.

scenario.poly.xml : *POLYCONVERT* command line function extracts geometrical shapes (polygons or points of interest) from and converts them to a representation that can be visualized using SUMO-GUI. The *POLYCONVERT* command used with *typemap.xml* on osm file to generate the polygon file *scenario.poly.xml* is as given below.

```
$ polyconvert − −net − file scenario.net.xml − −osm −
files map.osm −−type−file typemap.xml −o scenario.poly.xml
```

scenario.rou.xml : To deploy vehicles, inbuilt SUMO program *randomTRips.py* is used. It generates a set of random trips for a given network. This is done with the help of *DUAROUTER* function while executing which assigns routes with Dynamic user assignment (DUA). It is responsible for defining routes, compute trips and repair connectivity problems. The command function shown below generates the route file *scenario.rou.xml* with 100 seconds duration and 1000 vehicles deployed with random routes.

```
$   <   SUMO − HOME   >   /tools/randomTrips.py   −
r  scenario.rou.xml  − n  scenario.net.xml  − b  1  − e  100  −
p  0.1
```

scenario.sumo.cfg : SUMO configuration file is used to stick together all the essential files to be used at once. It can combine, route file, network, and poly file. This file is necessary to generate trace files.

scenario.trace.xml : When simulating the communication using a network simulator, it needs vehicle traces ignoring the background information. This can be generated using the configuration file with fcd output. fcd is the short form for Float Car Data.

```
$ sumo − c scenario.sumo.cfg − −fcd −
output scenario.trace.xml
```

scenario.tcl : For trace file to generate a car data file which can be compatible with NS-3, *traceExporter.py* is used with *ns2mobility-output*. The command is as shown below.

```
$ < SUMO − HOME > /tools/traceExporter.py − −fcd −
input scenario.trace.xml − −ns2mobility − output scenario.tcl
```

scenario.buildings.xml : The polygon file *scenario.poly.xml* has many other features which are not needed for our simulation. These are manually removed using a text editor, and only the building data is retained and copied to file *scenario.buildings.xml.*

3.3 Network Simulator-3 (NS-3)

“Network Simulator-3 (NS-3) is a discrete-event network simulator for Internet systems, targeted primarily for research and educational use. NS-3 is a free software, licensed under the GNU GPLv2 license, and is publicly available for research, development, and use” [38]. The official NS-3 source code with different releases is available

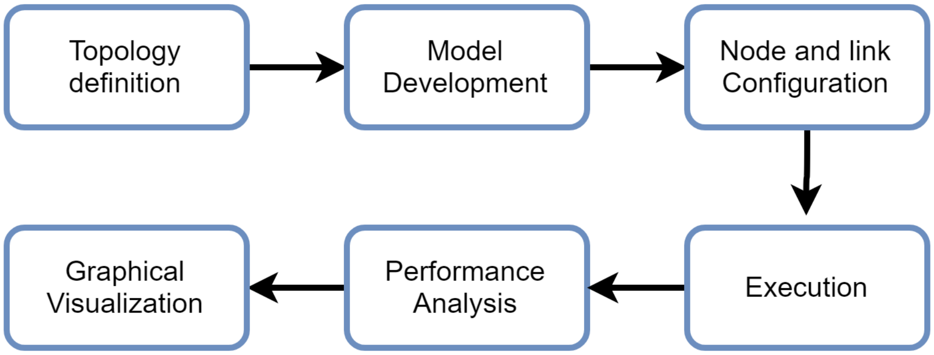


Figure 3.4: Simulation Flow in NS-3

at `code.nsnam.org`. NS-3 is built with a combination of software libraries written in C++ or Python programming languages. User programs usually are written and linked with the predefined software libraries for the functionalities of the network communication model. It is used to create the communication among different nodes and implement routing protocols.

3.3.1 Simulation Flow in NS-3

The NS3 simulator is developed and distributed completely in the C++ programming language. To construct a simulation using NS3, the user needs to write a C++ main program describing various elements necessary for the simulated communication network. The program is then compiled, and linked with the library of network models distributed with NS3. The process of creating a simulation in NS-3 can be divided into steps as described below and a flowchart is shown in Fig. 3.4.

1. **Topology definition:** Topology definition is the process of defining nodes(sometimes imported traces from software like SUMO), devices, channels(physical layer definition) and network protocols (routing protocols/ network initialization) that are necessary for the simulation. These are defined with the help of C++ objects/classes. Classes like Containers and helper from NS-3 repository ease this process.

2. **Model development:** The nodes characteristics are setup using the model development. Model development includes defining the source node, the destination node, assigning addresses to the nodes, and criterion for nodes to send/receive the packets. Modules based on Ipv4, UDP, point-to-point devices, links, and applications that are available in the NS-3 repository can be used for model development.

3. **Node and link configuration:** Attribute systems define the size of packets sent by an application, or the link configuration of point-to-point communication.

4. **Execution:** Simulation of the built models are compiled and run using the Waf tool. Waf is a Python-based framework for configuring, compiling and building models in NS-3. Execution generates events and the data requested by the user.

5. **Performance analysis:** Performance analysis is done after the simulation is finished and data is available as a time-stamped event trace. Typically, this

results in the simulator entering the main event loop, which reads and removes events in timestamp order from the sorted event data structure described earlier. This process repeats continuously until either the event list becomes empty, or a predetermined stop time is reached. Using the optional *pcap* trace format, any of the publicly available tools such as Wireshark can be used for analyzing *pcap* traces. The data generated can also be statistically analyzed with languages such as R and Python to derive conclusions.

6. **Graphical Visualization:** Raw or processed data collected in a simulation can be graphed using tools such as Gnuplot [51], matplotlib [52] or XGRAPH [53].

Chapter 4

Obstacle Effect on Greedy Perimeter Stateless Routing Protocol

As discussed in Chapter 3, non-line of sight (NLOS) issues can severely change the performance of routing protocols implemented for real-world scenarios. In this Chapter, a framework to study the effects of obstacles on routing protocols is defined. For this study, we considered a widely used position-based protocol: the Greedy Perimeter Stateless Routing (GPSR) protocol [8]. The authors Fonseca et al. presented a simulation setup for GPSR in the research paper "Implementation of GPSR in NS-3" [54]. To analyze the obstacle effect, we applied the empirical formula proposed by

Sommer et al., which is based on experimental data [24] and an NS-3 based obstacle shadowing model [10]. The GPSR protocol and its implementation in NS-3, a framework to calculate obstacle effect, the simulation setup in NS-3, and the simulation results are discussed in the following sections.

4.1 Greedy Perimeter Stateless Routing and its implementation in NS-3

The GPSR protocol typical works in 2 modes as described below: *Greedy forwarding* and *Perimeter forwarding* (or *recovery*) modes:

Greedy Forwarding: When each node chooses its next hop as the nearest node to the destination in its transmission range. In Fig. 4.1 we can see the *greedy mode* hops selection for the packet to travel from source to destination.

Perimeter Forwarding: The GPSR switches to *perimeter mode* when it reaches a *local maximum condition*, i.e., when the node itself is nearest to the destination than the other nodes in its transmission range. In *perimeter mode*, the node chooses the next node which is the first node in counter-clockwise to the direction of the destination. It will switch back into *greedy mode* once it is out of *local maximum condition.* In Fig. 4.2, the first hop is in perimeter mode due to local maximum

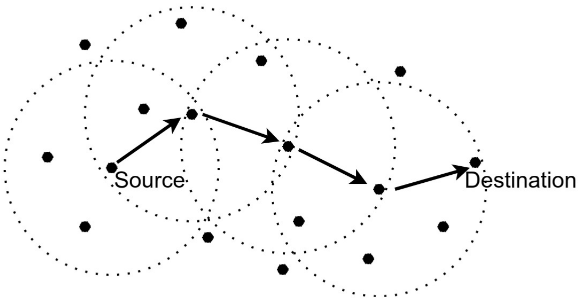


Figure 4.1: GPSR in full greedy mode

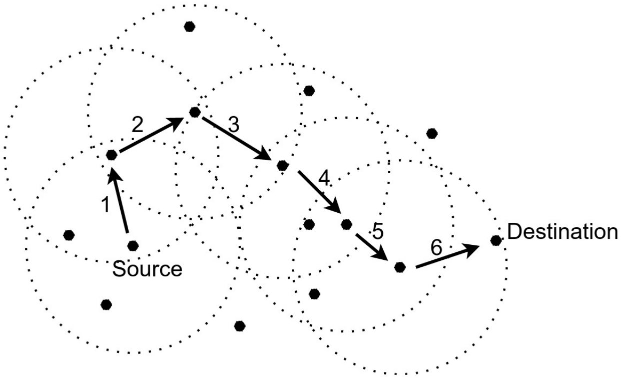


Figure 4.2: GPSR in Perimeter mode during hop 1.

condition at source, and later hops are in greedy mode since the source node has come out of the local maximum condition.

For GPSR to work efficiently, every node needs to have the location of the other nodes in its network. This is achieved through beaconing of *hello* packets to the network by each vehicle. The *hello* packets contain location information, vehicle speed, and other safety-related information. Therefore, each vehicle knows the location of every other vehicle in the network or knows the location of vehicles which knows other vehicle's location information. Hence, a standard location service is maintained throughout the network to make the GPSR work efficiently.

The algorithm for GPSR code is described in the **Algorithm** 1 where:

1. p is the packet received by node R for destination D.

2. set of neighbors for R is N.

3. n is the selected node for next hop from set N.

Algorithm 1 GPSR Implementation

```
if $\exists \quad n \in N : Distance(n, D) \leq Distance(R, D)$ then
    Greedy Forwarding
    n = Min Distance (N, D)
    ForwardPacket (p, n)
    Return
else
    local maximum, use right-hand rule
    n= RightHandRule (N)
    ForwardPacket (p, n)
    Return
end if
```

The implementation of GPSR in NS-3 comprises of various components/classes as represented in the GPSR block diagram shown in the Fig. 4.3. The description of the GPSR components is as follows:

- **GPSR packet** A GPSR packet class is responsible for defining the packets and consists of the criteria to select the next hop for the packet.

  GPSR Header: This class is used to build the GPSR packet. A GPSR packet is 16 bytes long and includes the position of the destination and a timestamp to check the positions freshness.

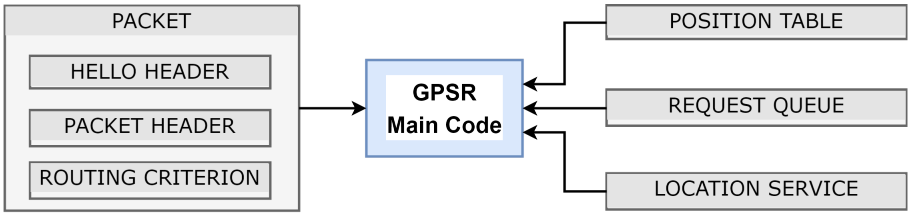


Figure 4.3: Components of GPSR in NS-3

Hello header: This class is used to build the *hello* packet used by the location service. The *hello* packet is of 8 bytes long. This packet contains the position of the node that sent the packet. The time interval for these packets is parameterizable, and the default value is 1 second. To avoid collision of *hello* packets a random jitter of duration 0.5 and 1.5 seconds is added between two hello packets[55]. **Routing Criterion:** This class contains the main logic flow of the GPSR algorithm. Calling and using the functionality of this class enables us to implement the required GPSR routing protocol.

- **Location Service:** In order to manage the location information of all the nodes in the network, Location service module is used. It manages the location information of each vehicle in the network. This module is the replication of Global Positioning System (GPS) available in the vehicles.

- **Request Queue:** Sometimes the route may not be available when an individual node receives a packet. The Request Queue class enables us to store the packet temporarily until the route information is available. The route is periodically checked when a packet is stored in Request Queue.

- **Position Table:** This class is for each node. The position table has the list of nodes, one entry for each neighbor for the corresponding node. Each entry has the neighbor identification number, its coordinates and the time-stamp at which the last *hello* message was received.

4.2 Path-loss Models

Attenuation models reflecting the realistic behavior of the physical layer play a critical role in the simulations. A model widely used in VANETs is Free space propagation model which is based on Friis' equation [56] and is given in Eq. (4.1:

$$L_{freespace}[dB] = 10log(\frac{16\pi^2}{\lambda^2}d^2) \tag{4.1}$$

where λ is the wavelength, d is the distance [57]. A generic formula for the power received by the receiver antenna is given by Eq. (4.2):

$$P_r[dBm] = P_t[dBm] + G_t[dB] + G_r[dB] - \Sigma L_x[dB] \tag{4.2}$$

where P_r is the received power strength at the receiver, P_t is transmitted power strength, G_t is the transmitter antenna gain, G_r is the receiver antenna gain [24]. $\Sigma L_x[dB]$ is the summation of all losses encountered by the signal. In addition to propagation losses, we also need to consider the obstacle losses. The Obstacle shadowing model considering obstacle losses based on empirical data proposed by Sommer el al. [24] produced realistic results. Sommer et al. [24] performed real-world experiments with DSRC/IEEE802.11p modules and estimated the effect of buildings on the signal propagation. A computationally inexpensive empirical model was used to generate the signal attenuation formula and is shown in Eq. (4.3):

$$L_{obs}[dB] = \beta n + \gamma d_m \tag{4.3}$$

Where n is number of walls intersecting the Line of Sight (LOS) when establishing communication between any two nodes, β is the per wall attenuation in dB, d_m is total length of building in LOS, γ is the coefficient factor for attenuation caused by internal structure of the building. Based on the experimental data of [24], the chosen parameter values are $\beta = 9$ dB and $\gamma = 0.4$ dB/m. The obstacle shadowing model was implemented in NS-3 by Carpenter,Sichitiu et al. [10]. The model used in our simulation is a combination of Free space propagation model and the obstacle shadowing model. The total path-loss $\Sigma L_x[dB]$ is given by Eq. (4.4):

$$\Sigma L_x[dB] = L_{freespace}[dB] + L_{obs}[dB] \tag{4.4}$$

In Fig. 4.4, we shown the city area used to simulate the shadowing effect. The blocks in blue numbered 1 to 12 represent buildings of different sizes. Using the model in

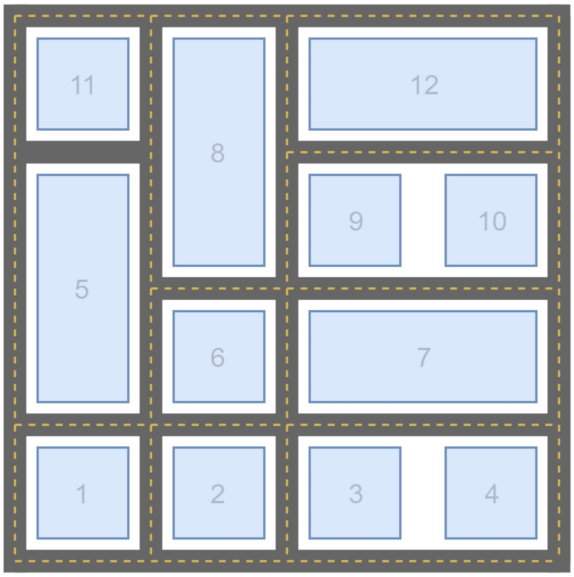


Figure 4.4: Typical urban scenario designed to test path-loss formula

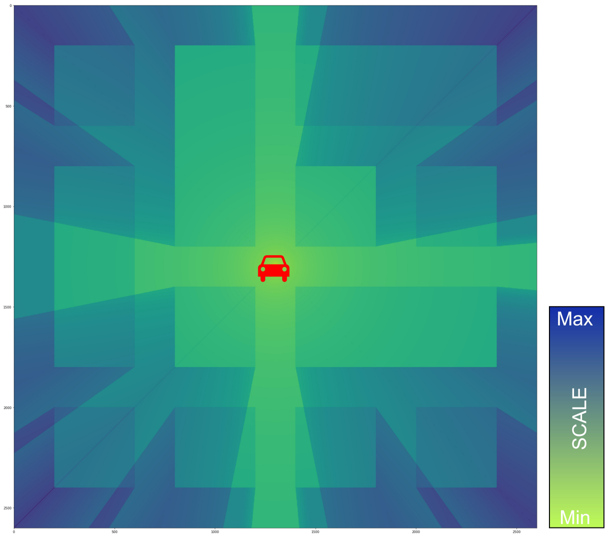


Figure 4.5: Shadowing effect when vehicle positioned as shown

Equation 4.4, the path losses are estimated for the region in Fig. 4.4 when a car is placed at center and has a omni-directional antenna. The measured value of signal strength is represented with color intensity and is shown in Fig. 4.5. The darker the color, the lower the signal strength. We can observe in Fig. 4.5 how the buildings

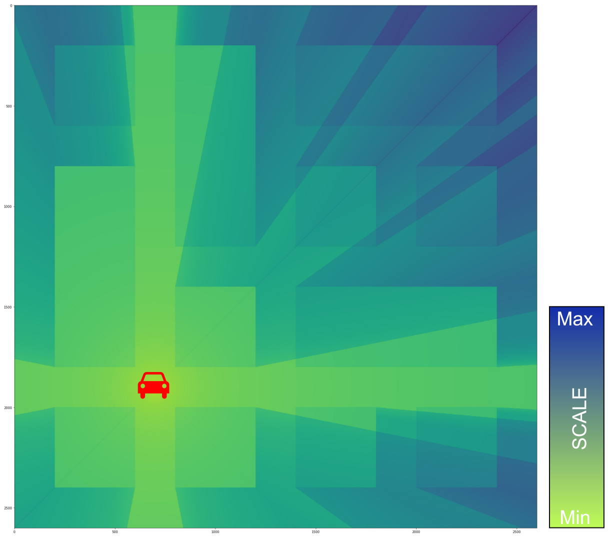


Figure 4.6: Shadowing effect when vehicle positioned at a different location as compared to Fig. 4.5

create shadowing effect to the signal strength. When a car is moved to a different location, the change in shadowing effect is shown in Fig. 4.6.

4.3 Framework to Calculate Obstacle Effect on GPSR

The obstacle effect on GPSR protocol is simulated using some inbuilt repository programs of NS-3 for maintaining features like addressing, physical layer, MAC layer description, and monitoring the parameters. The overall model built and the different classes/models called in the main programs are represented in the Fig. 4.7. The different models used can be called as source models and are described as follows:

- **GPSR:** In the main program, the components mentioned in the section 4.1 are integrated with other source models.

- **Addressing:** For maintaining addresses and communication between the nodes Ipv4 class is used. Source and destination nodes are managed by UDP: Server client application class.

- **PHY and MAC:** $YansWifiPhy$ module is used to configure physical layer, and $NqosWaveMac$ module used to configure the MAC layer. These two modules provide the configurations which resemble the IEEE 802.11p standard.

- **Flowmonitor:** It is NS-3 repository module built by Carneiro et al. [58]. It helps to detect all flows passing through the network and stores in a file Metrics such as packet delivery rate (PDR), mean hops, delays, packet sizes, and packet loss ratio is analyzed using flow-monitor module.

- **Propagation model:** The propagation model is built based on both the obstacle shadowing model and Friis path loss combined as discussed in the section 4.2. The obstacle model uses a *Computational Geometry Algorithms Library* (CGAL) [59], an extended library, to NS-3 to detect building walls/obstacles in a simulation for Obstacle Shadowing Model.

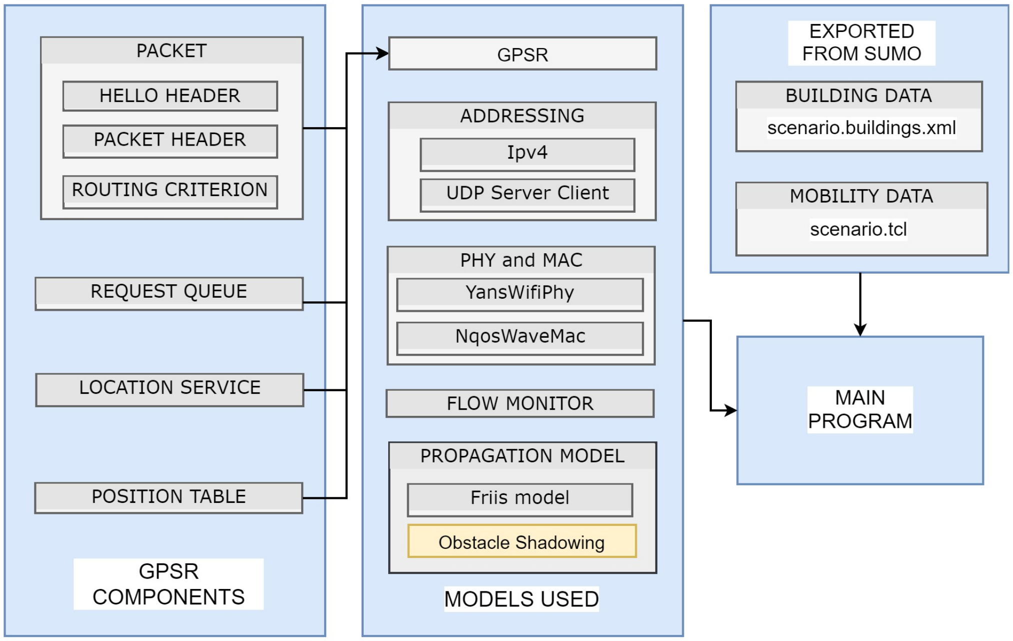


Figure 4.7: Framework to calculate obstacle effect of GPSR in NS-3

4.4 Simulations and Analysis

Two types of simulations are performed for comparison, one is GPSR routing protocol using the Friis path loss model neglecting the obstacle effect, and the GPSR model with the combined path loss Equation 4.4 of obstacle shadowing and Friis model. In this way, the obstacle effect on GPSR is analyzed. The simulation parameters used for the simulations are listed in the Table 4.1.

Parameter	Value
Scenario Area	1000mx500m
Scenario types	Urban, Residential, Highway
Simulation time for each run	60 seconds
Propagation model	Free space, Obstacle Shadowing model
Frequency	5.9 GHz
Channel Access	IEEE 802.11p OCB
Routing protocol	GPSR
Packet rate	2 packets/sec
No. of Vehicles	20, 40, 60, 80, 100, 120, 140, 160, 180, 200

Table 4.1
Simulation Parameters in simulating GPSR protocol

4.4.1 Scenario setup

The scenarios used in the simulation are real-world environment scenarios, such as Urban, Residential and Highway. The scenarios are defined as follows:

- **Urban Scenario:** It is similar to a Manhattan grid with high rise buildings that are close knit. A downtown area would be a good example of such an Urban scenario.

- **Residential Scenario:** It is a housing locality where small buildings (houses) are sparsely distributed across an area.

- **Highway Scenario:** The roads are wider, like freeways, and buildings are rarely present.

Based on these definitions, several scenarios were chosen from real map data available at *www. openstreetmap. org* [37]. Urban and Highway scenarios are from the Chicago,IL,USA area, and Residential is from Houghton, MI, USA. The longitude and latitude information with the number of buildings for each scenario is given in the Table 4.2. The road data, building data, and the mobility traces are exported from SUMO in the appropriate formats usable by NS-3. The building data imported with scenario file is then modified by removing data not related to buildings as the obstacle shadowing model employed here considers only buildings as obstacles. The scenario maps after importing the building data and road data into SUMO are shown in Fig. 4.8, Fig. 4.9, and Fig. 4.10. Each simulation consists of a source node and destination node with 60 seconds of simulation time transferring a total of 120 packets with a rate of two packets/sec. The parameters measured in our simulations are defined as follows:

- **Packet Delivery Ratio (PDR):** The percentage of packets received by the destination for the total number of transmitted packets by the source.

- **Mean Hop Count:** Average number of hops for all the packets received by the destination.

- **Mean End-End Delay:** Average of time-taken for the packets to travel from source to destination.

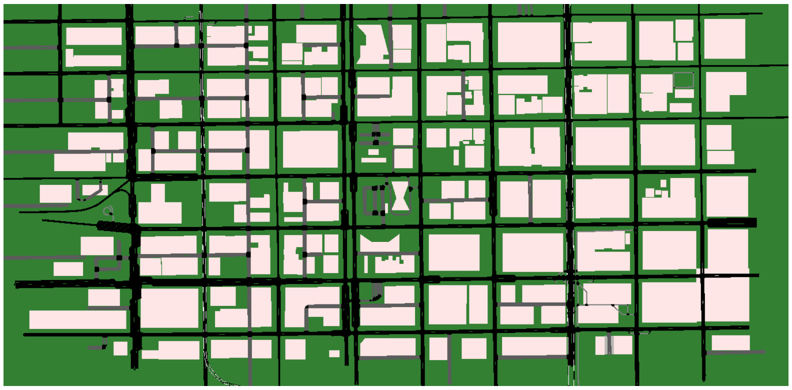


Figure 4.8: Urban Scenario in SUMO-GUI

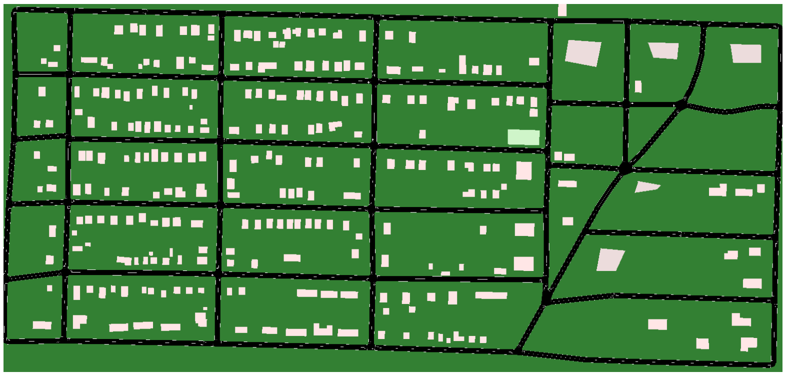


Figure 4.9: Residential Scenario in SUMO-GUI

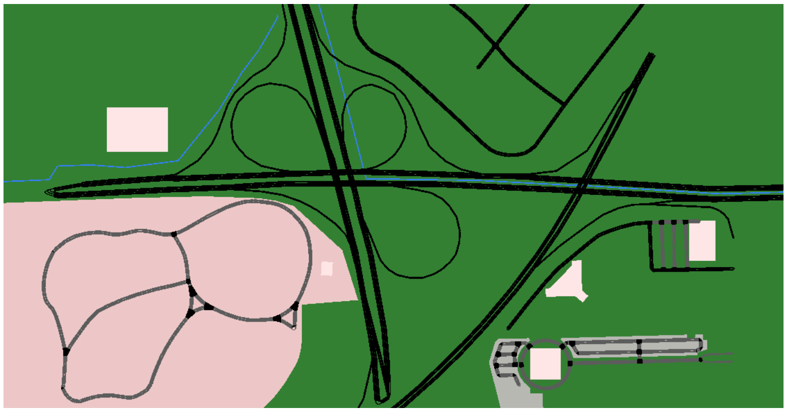


Figure 4.10: Highway Scenario in SUMO-GUI

Table 4.2
Scenario parameters

Scenario	Latitude		Longitude		Buildings
	Min	Max	Min	Max	
Urban	41.8848090	41.9415960	-87.6533460	-87.6202870	270
Residential	47.1160000	47.1198000	-88.5815000	-88.5690000	280
Highway	41.8581000	41.8629000	-87.9622000	-87.9509000	6

4.4.2 Results and Analysis

The road structure and building structures differ in all the three scenarios: Urban, Residential and Highway. Because the simulation includes mobile nodes with varying speeds, the distance between the source node and the destination node always varies. To have unbiased results due to the varying location of nodes, the measured metrics (PDR, E2ED, and Mean Hop count) are the average of simulation results equal to 10 percent of the total number of nodes. In each simulation, the destination is chosen at random. For instance, if a simulation of a scenario has 40 nodes, four different destination nodes are chosen and simulated four different times and the mean of these values is calculated. Similarly, 20 different destination nodes are chosen for the scenario with 200 nodes. This averaging gives the general behavior of the scenario rather than having the results based on a single destination node. A total of three scenarios is studied for each of the two cases - with obstacles and without obstacles. This gives us six different cases, which makes it easier to study obstacle effect in each scenario. Following is the analysis of results for each metric measured in this thesis.

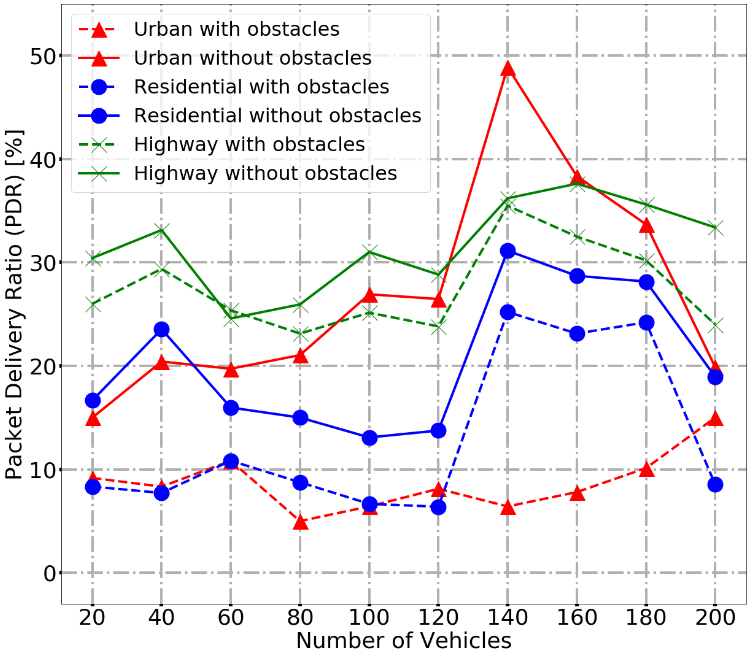


Figure 4.11: Packet Delivery Ratio in percentage using GPSR protocol (With obstacles vs. Without obstacles)

Packet Delivery Ratio (PDR)

The PDR is simulated and averaged considering the six different cases. Based on the results presented in Figs. 4.11, 4.12, 4.13 and 4.14, the highway scenarios have the highest PDR, with or without obstacle. This is because very few buildings are present in highways. On the other hand, the PDR for urban scenario and residential scenarios are lower when considering obstacles. Obstacles can affect the packet delivery rate significantly for urban scenarios (Fig. 4.12) due to the presence of a large number of buildings with more area (more walls, longer length). The attenuation effects are less

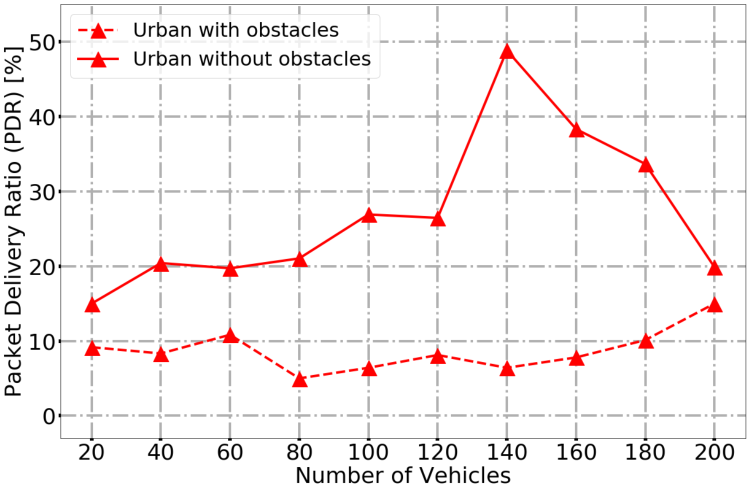


Figure 4.12: PDR for Urban Scenario

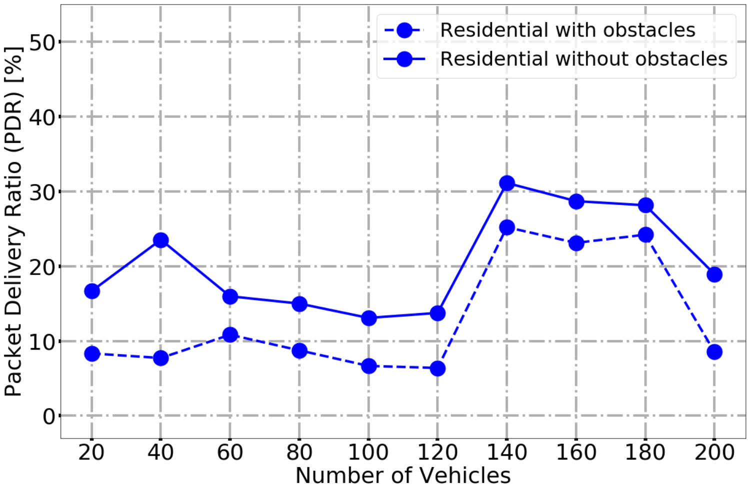


Figure 4.13: PDR for Residential Scenario

in the residential scenario as compared to urban since obstacles in residential areas are mostly houses which are sparsely distributed and not as tall. It is also important to consider that the Manhattan grid structure for urban scenario makes it more easy for vehicles to communicate when compared to the residential scenario. With the Manhattan grid structure, the likelihood that vehicles can communicate with each

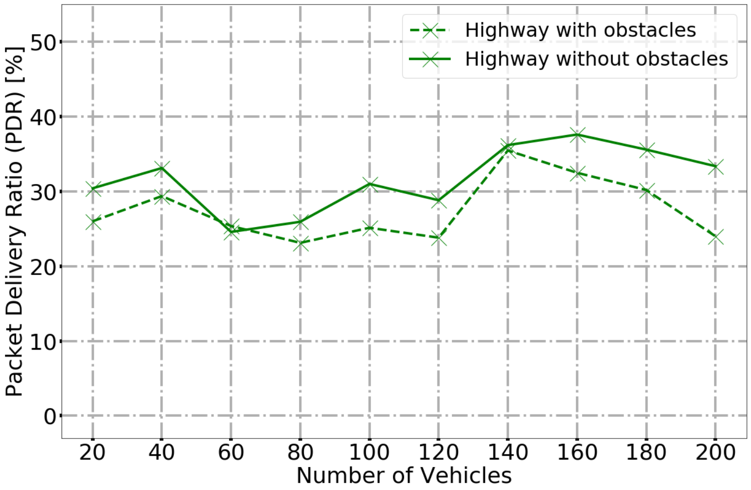


Figure 4.14: PDR for Highway Scenario

other on parallel roads is higher than in the residential grid where the roads are not as well structured and are distant from each other. This behavior is obvious in the Fig. 4.11 as the general trend of PDR of *residential without obstacle* is less than that of *urban without obstacle*. The PDR of any scenario with the different number of vehicles did not cross 50 percent. This is because the nodes are highly mobile, which is huge backset for VANETs. A traditional GPSR cannot handle this high mobility, and the PDR is even more detrimental when we consider the obstacles. This is due to the varied distribution of the signal strength caused by the obstacles. Considering scenarios with vehicles from 20 to 80, the PDR is observed to be fluctuating, but there is increasing trend of PDR in almost all the scenarios for vehicles greater than 80. This increase is due to the easy availability of the next hop in the network due to more number of vehicles, i.e., higher density (penetration) of vehicles.

Mean Hop count

The averaged mean hop count is presented in Fig. 4.15. Mean hop give us the information of how many hops the signal can make before dropping the packet. The number of hops gives us the idea of how far the packets can reach. The general trend as observed in Fig. 4.16 for all the scenarios is that the number of hops is less for the cases with obstacles as compared to that of without obstacles. The packet is normally dropped for mean hops less than two in most cases for scenarios with obstacles as the buildings increase the chance of disrupting the signals. The obstacle-free GPSR works better as the number of hops is higher in most of the cases in all the scenarios. It is worthwhile noticing that in situations where the destination is too distant, the likelihood that the obstacles will disrupt the signal increases with the number of hops, and the packet will simply be dropped before continuing with additional hops. Note that, if the packet is dropped, its hop count towards the calculation of average hop count is not considered. It is also observed during simulations that in some cases less than 100 nodes, the GPSR often goes into perimeter mode (recovery mode) due to lower vehicle density. The GPSR recovery mode is not efficient regarding the number of hops. This is also a contributing factor to the unstable number of hops for vehicles less than 100.

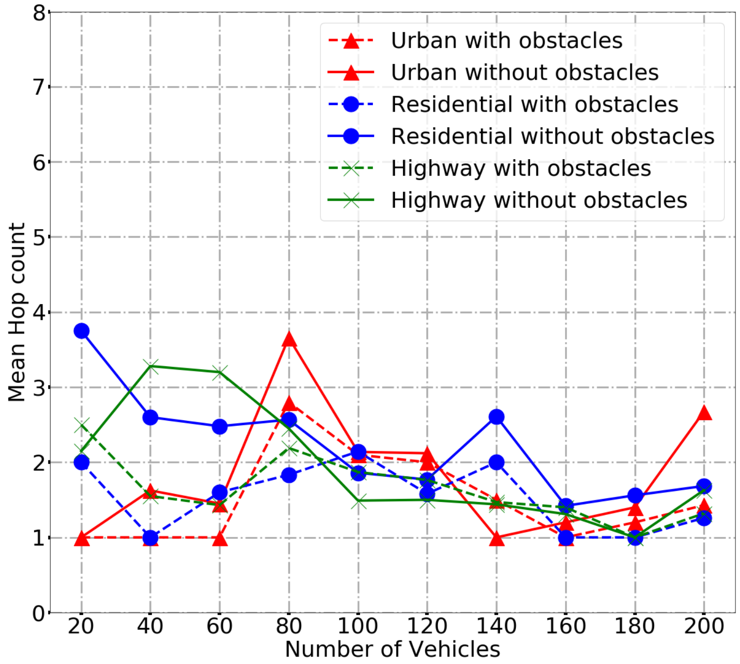


Figure 4.15: Mean Hop count using GPSR protocol (With obstacles vs. Without obstacles)

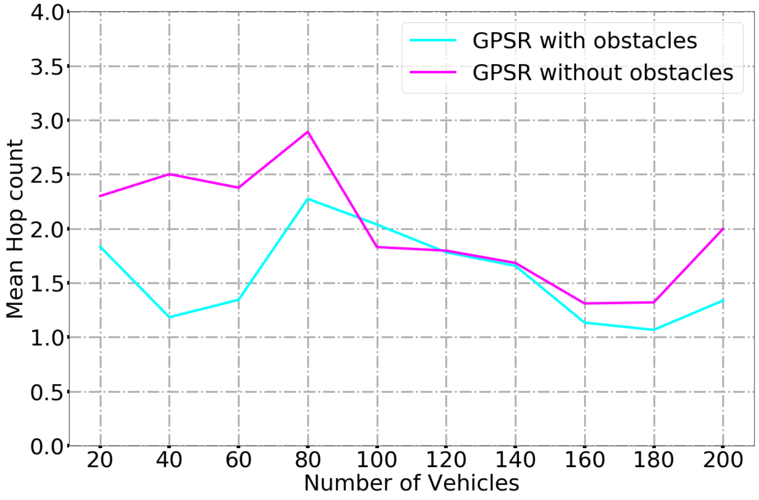


Figure 4.16: Average of Mean Hop count using GPSR protocol (All scenarios with obstacles vs. All scenarios without obstacles)

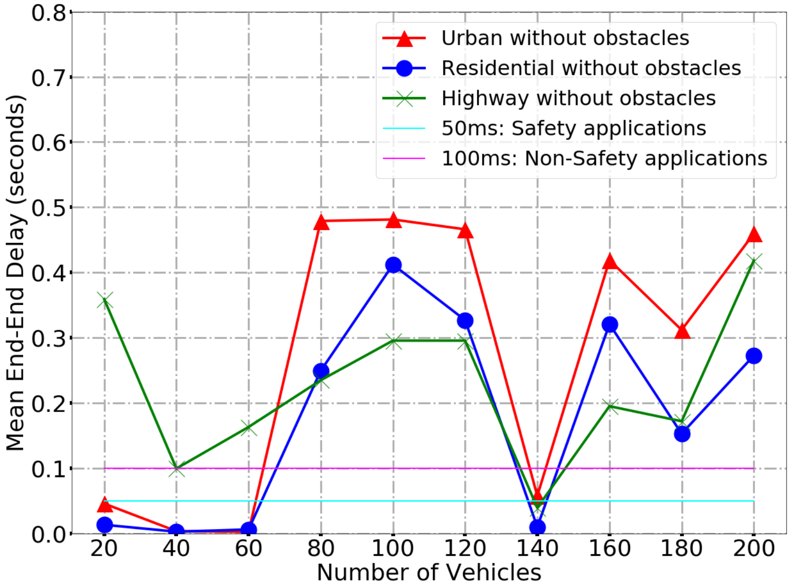


Figure 4.17: Mean End-End delay using GPSR protocol Without Obstacles

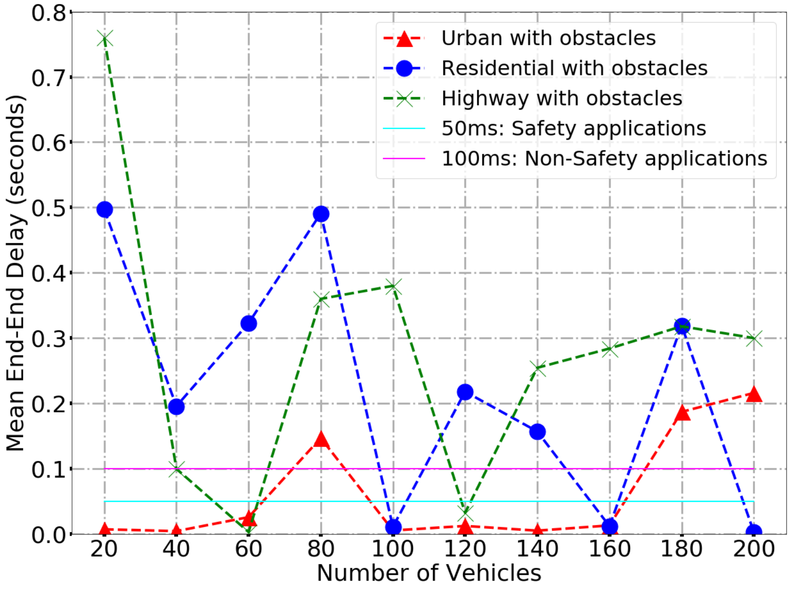


Figure 4.18: Mean End-End delay using GPSR protocol with obstacles

Mean End-End (E2E) Delay

A Mean End-End Delay (E2E), also referred as "Latency" gives the average of time taken for a packet to get from source to destination. When E2E delay is observed, the mean delay is relatively consistent for all the cases without obstacles as shown in Fig. 4.17. This is because there is not much variation in signal strength with an absence of obstacles. It is observed that the delays associated with cases with obstacles (See Fig. 4.18) are highly unstable due to high variation in signal strength as the signal generated by the mobile nodes propagate through the buildings. In general, highway scenarios experienced the least amount of E2E delay due to the absence of buildings and with the GPSR mostly working in the greedy mode. These results demonstrate the compromising effects of obstacles in the network overall performance.

As per authors [12] the latency for Safety applications should be minimum of 50 ms for safety applications and 100 ms for Non-Safety applications. In all the scenarios with and without obstacles implemented with GPSR protocol, there are only a few cases where the E2E delay is less than 100 ms and even few cases less than 50ms. Hence, the traditional GPSR has proved to be unfavorable for VANETs in these scenarios.

Chapter 5

Proposed Routing Protocol: Intelligent Greedy Routing (IGR)

The simulation results presented in chapter 4 clearly demonstrate how adversely obstacles (buildings) can affect the efficiency of the GPSR protocol. Hence, there is a need to develop a new routing protocol able to efficiently work in such environments. In this chapter, we discuss the issues observed with the GPSR protocol, the design approach for the new Intelligent Greedy Routing (IGR) protocol, details of how it works, its implementation in NS-3 and a detailed analysis of results comparing GPSR and the new IGR routing protocol.

5.1 Issues with GPSR in the presence of obstacles

There are many issues found with the study and simulations of GPSR. Some issues are associated with the high mobility nature of the nodes associated with VANETs and others are due to the obstacle shadowing effect. Following we discuss a few of these problems in more detail:

- **Connectivity:** The first and foremost problem with the GPSR is the unstable connectivity between nodes. For example let us consider a node A, trying to choose its next hop. Based on the list of neighbor nodes it has in its position table, it chooses a node B, based on the criterion that it is the nearest node to the destination. The position associated with node B while listed in the position table may be at extreme edge of the transmission range of node A. Chances are that, after selecting the hop, when trying to send a packet to node B, node B may be out of transmission range of node A. A significant number of packets can be dropped due to this issue in GPSR. Another factor is the obstacle effect. The transmission range is unevenly distributed when considering obstacle shadowing. This leads to even more disconnections.

- **Routing Loops:** The high mobile nature of the network makes it difficult for a *perimeter mode* to work, leading to packets going through loops before reaching the destination. This increases the chances that packets drop and increases hop count, which in turn leads to higher delay of the packet.

- **Wrong Direction:** In *greedy mode*, when selecting the next hop, due to high mobile nature of the vehicles, the selected hop (node) may be leading in the opposite direction to that of the destination, and this makes it a wrong choice of selecting the node to be a nearest node to the destination. This also leads to packet drop.

5.2 Design approach for a new routing protocol

A conventional type of Greedy Perimeter Stateless Routing protocol considers only the distance parameter when selecting the next hop in a route. This method works very efficiently in an obstacle-free environment. However, in the real-world, because of varying signal strength the next hop node might be in a region of low signal strength or the node may be moving to an even lower signal strength region. Taking this into consideration, we need to consider the ability of the next hop node to receive the packet in addition to its proximity to the destination. Also, the *perimeter mode* for GPSR has proven to be an issue considering the mobile nature of the nodes in VANETS. The new routing protocol should also solve the looping issue caused by the GPSR. A new routing scheme should also consider the fact that not all the nodes from which it receives *hello* packets will be available when the next hop is chosen. The new routing protocol designed to overcome these issues is the *Intelligent Greedy Routing (IGR)*. It considers the ability of the node to receive packets before selecting it as a hop. The term *Intelligent* is because the criteria used to select the next hop is dependent on the location, path loss, and the region in which the vehicle is moving. It chooses the next hop based on a parameter called *Receptivity.* The details of how the proposed *Intelligent Greedy Routing (IGR)* protocol works is explained in the following section.

5.3 Intelligent Greedy Routing (IGR)

The proposed *Intelligent Greedy Routing* protocol works based on the parameter *Receptivity (R)*, which is the ability of the node to successfully receive the packet from the transmitting node. The parameter R is calculated as:

$$R = -\big(d + \delta W\big) \tag{5.1}$$

Where d is the distance to the destination, W is the average path-loss weight of the neighbor node, and δ is the matching factor parameter between the average path loss and the distance. The *Receptivity* for every neighbor node is calculated, and the next hop will be to the node with maximum *Receptivity.* Algorithm 2 shows the process for the IGP implementation with the following terminology:

1. p is the packet received by node I for destination D.
2. set of neighbors for I is N.
3. $d_n, W_n \quad and \quad R_n$ are the parameters calculated for each neighbor node of I
4. n is the selected node from set of neighbors N which has the maximum *Receptivity.*

Algorithm 2 IGR Implementation

```
if $\exists \quad n \in N : Distance(n, D) \leq Distance(I, D)$ then
    $Intelligent\ Greedy\ Forwarding$
    $d_n = distance(n, D)$
    $W_n = AveragePathloss(I, n)$
    $R_n = -(d_n + \delta W_n)$
    $n = MaxR_n(I, N)$
    ForwardPacket (p, n)
    Return
end if
```

A typical selection of next hop can be explained with an example shown in Fig. 5.1. A packet needs to be transmitted from *source* to *destination*, and the *source* neighbors are A, B, C, D, E, and F as shown within the transmission range in Fig. 5.1. The distance to the destination is represented with d and the calculated average path loss (W) for each neighbor is shown with value W. Table 5.1 shows the calculation of *Receptivity* for each neighbor node, Node B has the maximum *Receptivity* of $R = -300$. Hence, B is chosen as the next hop using the IGR protocol. A typical GPSR would select node C as the next hop as it is the nearest node to the destination. A

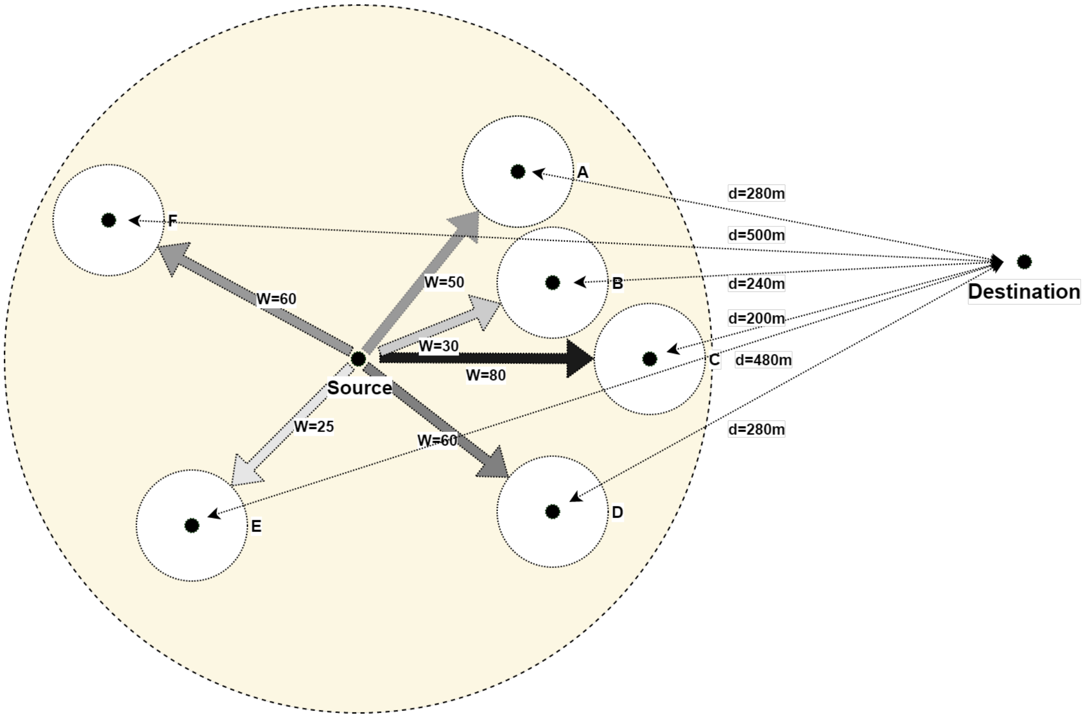


Figure 5.1: Intelligent Greedy Routing

typical GPSR considers only the distance to destination parameter. But, IGR also considers the parameter Average Pathloss (W) in addition to the distance parameter. Average path loss gives the estimated value of the path loss for the packet to reach the region of the neighbor node. This value is too high for node C, i.e. $W = 80$. This is due to node C being located at the extreme edge of the source transmission range. The chances of the packet reaching node C are too low.

Therefore IGR does not consider the criteria of considering the nearest node with the parameter d like a simple GPSR. This distance parameter can be added by selecting the node which can receive the signal with more probability. This process makes us

Table 5.1
Calculation of Receptivity(R) for nodes in Fig. 5.1 with delta=2

Node	Weight(W)	distance (d)	Receptivity(R)
A	50	280	-380
B	30	240	-300
C	80	200	-360
D	60	280	-400
E	25	480	-530
F	60	500	-620

compromise the proximity to destination due to addition of average pathloss parameter. Hence, the IGR protocol with the help of the *Receptivity* parameter selects the node B as next hop. This node may not be the nearest node to the destination, but it is the node with high *Receptivity* in the direction towards the destination.

The definition and calculation of each *Receptivity* parameter value is discussed below:

1. **Distance (d):** is the distance of the neighbor node to the destination.

2. **Average Pathloss W:** The Average pathloss is the estimated value based on the obstacle shadowing model and the buildings map of the scenario. The buildings map can be generated using the map data from OpenStreetMap [37]. Each node in the network needs to have the buildings file with the location information of the buildings to estimate the average pathloss (W) to reach the neighbor node. Based on this data and obstacle shadowing model propagation loss, the average path loss from Source to each neighbor node is calculated using Eq. 5.2.

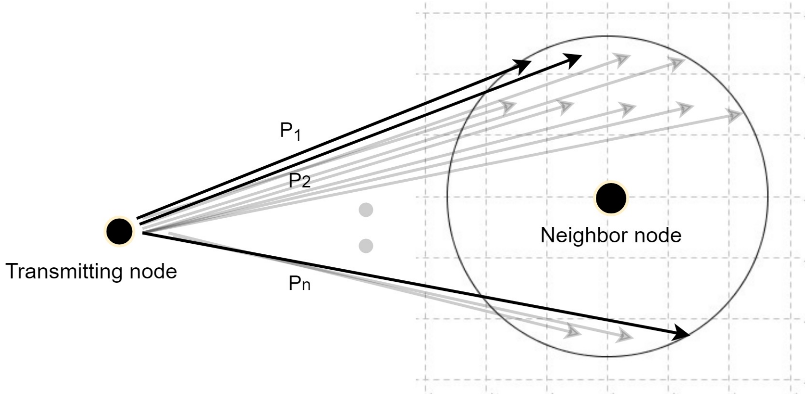


Figure 5.2: Path-loss from transmitting node to every point within the radius r of the neighbor node

$$W = \frac{p_1 + p_2 + p_3 + ... + p_n}{n} \tag{5.2}$$

where $p_1, p_2, ..., p_n$ are the pathlosses to each point within the radius r of the particular neighbor node. This is depicted in the Fig. 5.2. It should be noted that considering the average path-loss in making routing decisions should not be confused with cross-layer optimization. We are not passing any real-time data(such as Signal to Noise Ratio (SNR), link prediction, path-loss) from physical layer to network layer, instead using previously stored building data to estimate the average path-loss.

3. **Matching factor** δ : The appropriate selection of δ plays a vital role in the calculation of R, as it can dominate the distance d parameter leading to a slow progress of the packet to the destination node. Therefore, this parameter should be selected carefully depending on the wall losses, the transmitting power

of antenna, and pathlosses. With varying the location of destination of neighbor nodes and calibrating the δ to properly match between estimated average pathloss and distance to destination in different scenarios: Urban, Residential and Highway, the wall loss $\beta = 9$dB and coefficient of attenuation for internal structure $\gamma = 0.4$dB/m. The values of δ parameter is varied from 0 to 5 with a step size of 0.5 and tested for each scenario and checked for delivery of the packet at a node placed at the extreme of the transmission range with different cases: Urban, Residential and Highway. After numerous simulation in different scenarios/cases, the value of δ is calibrated as 2 such that the receiving node is able to receive the packet without compromising much distance to destination. However, the δ parameter can be chosen more accurately with physical world experimental data of pathloss information between 2 vehicles in different scenarios. Performing physical experiments is beyond the scope of this thesis. The average pathloss will eliminate the chances of the next hop node (vehicle) moving to very low signal region (voids) as the the next neighbor points of the next hop node are also considered in the W parameter. If the building information cannot be obtained or average pathloss cannot be estimated, the *hello* packet when received should be stored with the signal strength information of the *hello* packet in the position table for the corresponding neighbor node. This can also be used as a W parameter replacing the actual W parameter if calculation of average pathloss is beyond the ability of the hardware used in the vehicle.

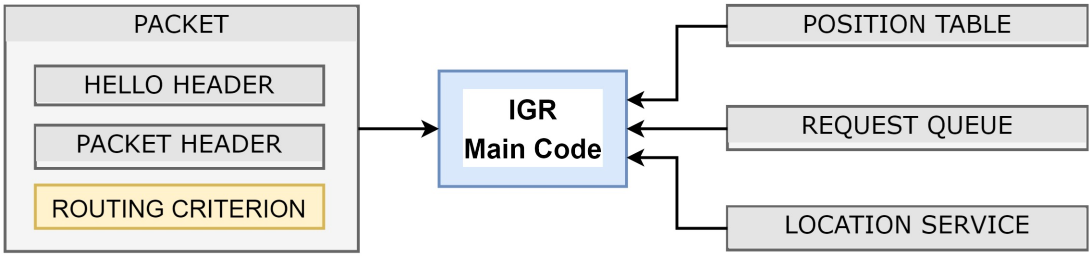


Figure 5.3: Implementation of Intelligent Greedy Routing in NS-3

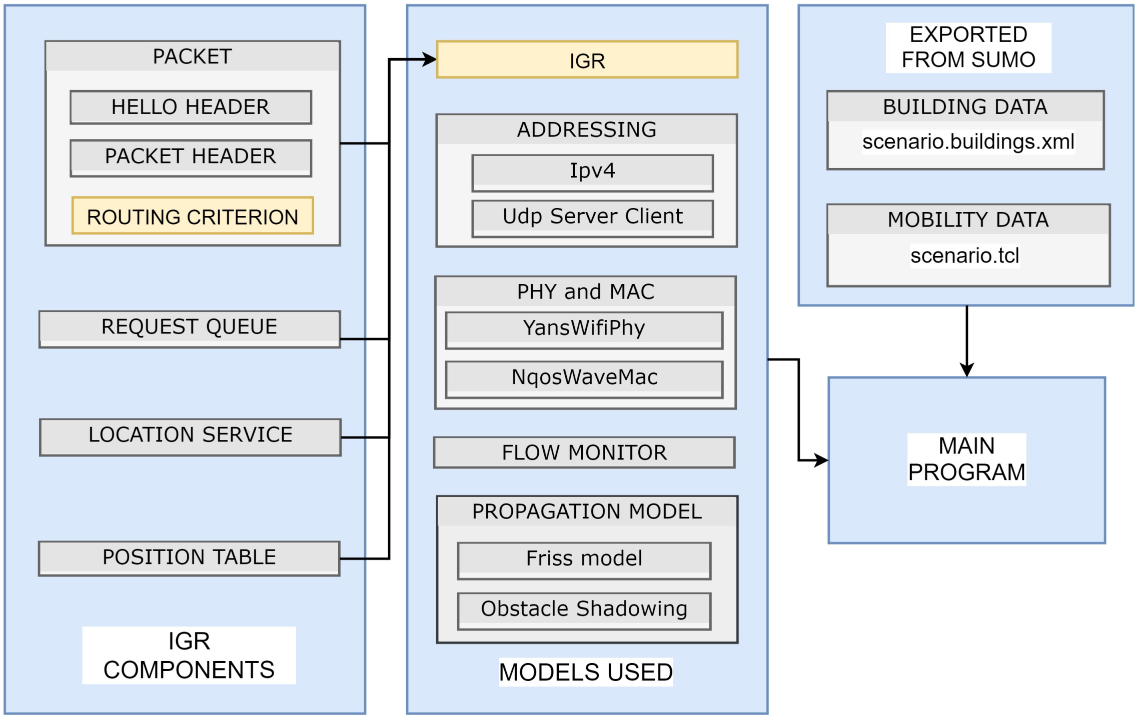


Figure 5.4: Framework for IGR

5.4 Simulation setup for the IGR protocol

The implementation of our proposed IGR protocol is very similar to the implementation of code for NS-3 discussed in Chapter 4. The functionality of the routing criterion in the packet was modified accordingly with the concept of IGR and a new NS-3 main code was built. The routing criterion block is highlighted in Fig. 5.3. The modified framework for simulation of IGR with appropriate model is depicted in Fig. 5.4 with the changes highlighted. The scenario parameters, of the selected scenarios: urban, highway and residential are the same as those used for the simulations in *Chapter 4*. The simulation parameters set for simulations are listed in Table 5.2.

Parameter	**Value**
Scenario Area	1000mx500m
Scenario types	Urban, Residential, Highway
Simulation time for each run	60 seconds
Propagation model	Free space, Obstacle Shadowing model
Frequency	5.9 GHz
Channel Access	IEEE 802.11p OCB
Routing protocol	GPSR, IGR
Packet rate	2 packets/sec
No. of Vehicles	20, 40, 60, 80, 100, 120, 140, 160, 180, 200

Table 5.2
Simulation Parameters for comparison between IGR and GPSR

5.5 Comparison of metrics for GPSR and IGR

The metrics PDR, Mean hop count, and Mean End-End delay were analyzed in *Chapter 4* and used to observe the effects of obstacle on GPSR. Using the same obstacle scenarios, the new IGR protocol is simulated and compared to GPSR. Once again, a total of six different cases are analyzed: urban, highway and residential with GPSR and with IGR. In all the cases, the obstacle effect is considered. Hence

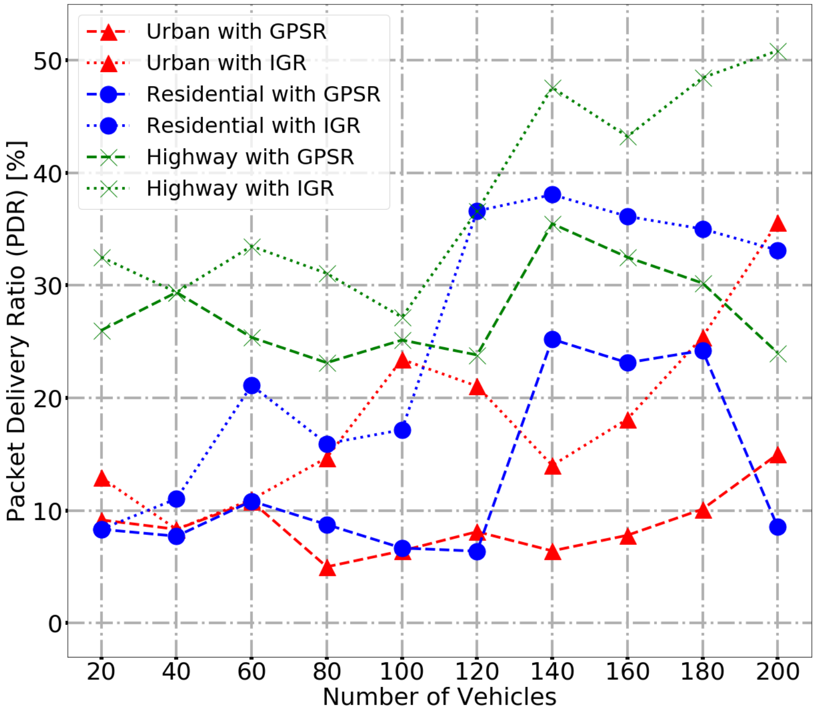


Figure 5.5: PDR comparison for GPSR and IGR for the scenarios

an analysis of how IGR outperforms the GPSR protocol is presented in the following sections. For experimental simulations the value of δ is chosen to be 2 for the scenarios selected.

5.5.1 Packet Delivery Ratio

The Figs. 5.5, 5.6, 5.7 and 5.8 show the Packet Delivery Ratio (PDR) comparison between GPSR and IGR in each scenario. We can observe that there is a significant increase in the PDR in almost all the scenarios when considering IGR. In *Chapter 4*, we have seen that the obstacle effect is not so significant on GPSR in highway scenario

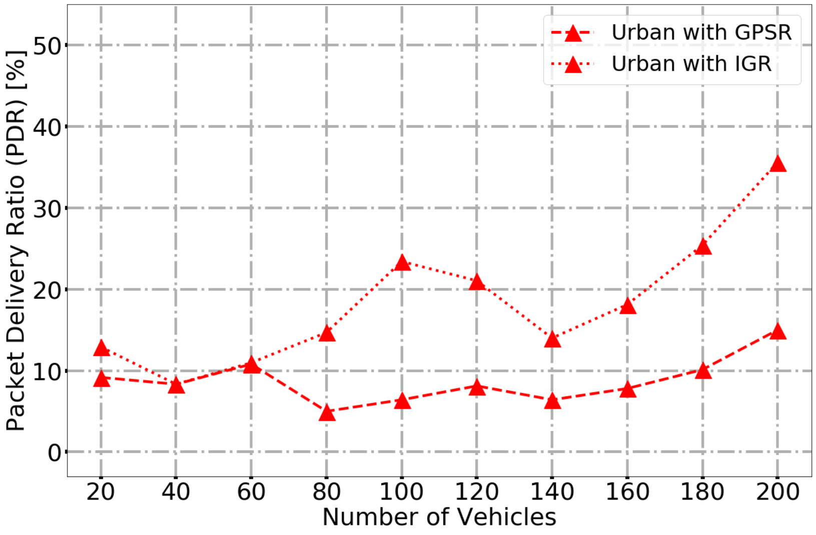


Figure 5.6: PDR (GPSR vs. IGR) for Urban Scenario

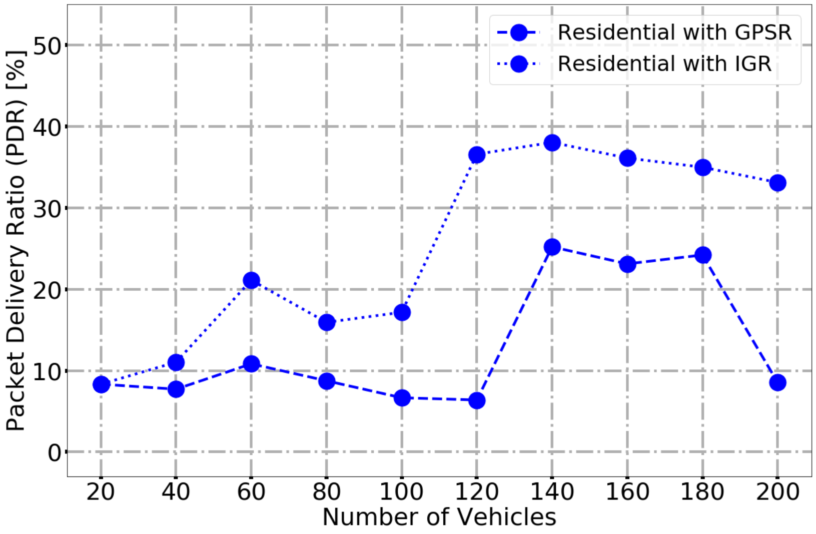


Figure 5.7: PDR (GPSR vs. IGR) for Residential Scenario

due to very few buildings. But, there is even more improvement when IGR protocol is used as evident in the Fig. 5.5. This explains why the packet drop was mostly due to the choosing the edge node in the transmission range as a next hop and hence has

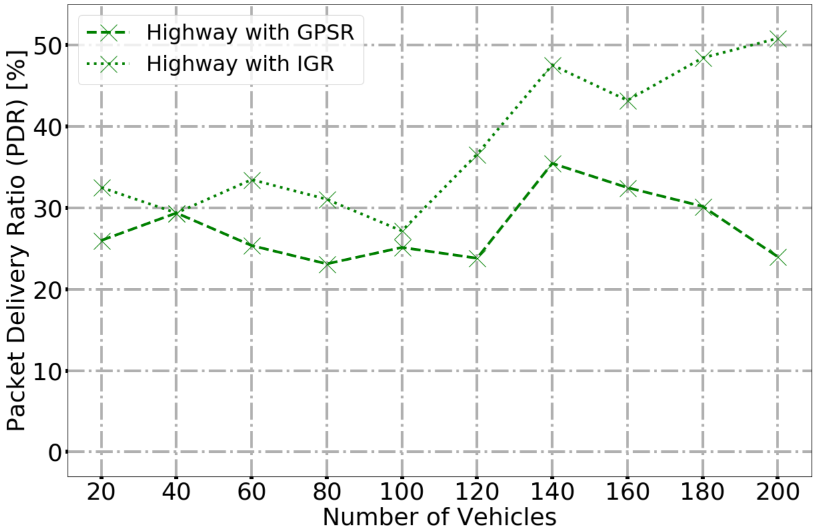


Figure 5.8: PDR (GPSR vs. IGR) for Highway Scenario

a low chances of receiving the packet as it may be out of range before receiving the packet. IGR has improved the PDR significantly for urban (Fig. 5.6) and residential scenarios (Fig. 5.7) when the number of vehicles is more than 80. This is due to easy availability of next hop with the *Receptivity* parameter when vehicle density is high.

5.5.2 Mean Hop Count

Fig. 5.9 shows the averaged mean hop count for both the GPSR and IGR protocols for different scenarios. The averaged mean hop count for IGR appears to be high for all the scenarios. This is because of two reasons:

1. The step for each hop is less distant when compared to that of GPSR because

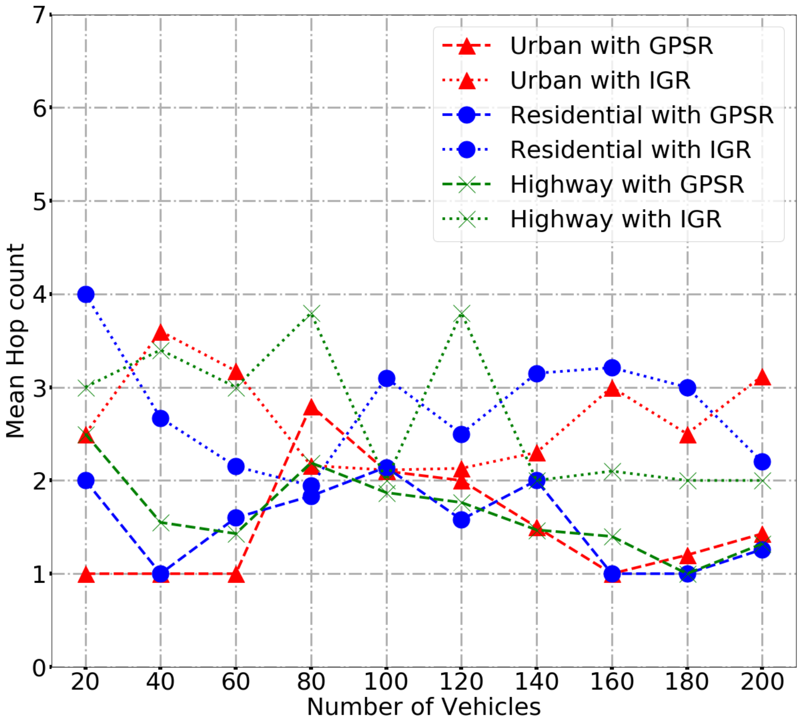


Figure 5.9: Mean Hop Count GPSR vs. IGR for all the scenarios

of the parameter W. This leads to progress of the packet to destination with more number of hops.

2. As the packet is transmitted to the next hop based on the *Receptivity*, there is a less chance of packet drop. So the reach of packets of to distant destinations also increases the hop count.

Fig. 5.10 shows averaged mean for different scenarios for both GPSR and IGR and we can observe how significant is the increase of hops in IGR.

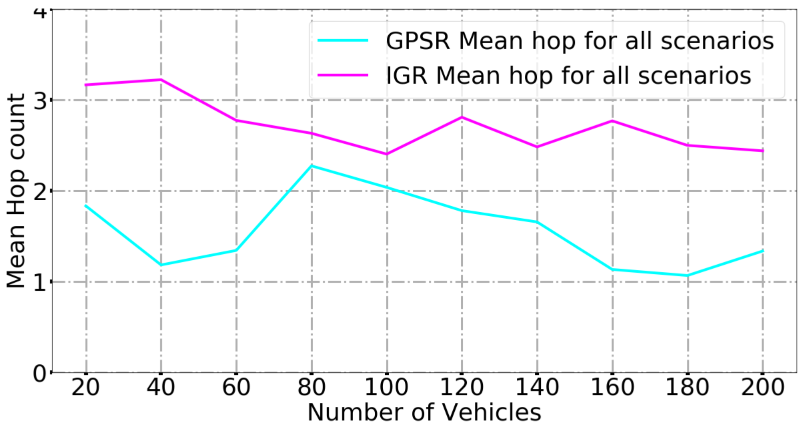


Figure 5.10: Average Mean Hop count for different scenarios GPSR vs. IGR

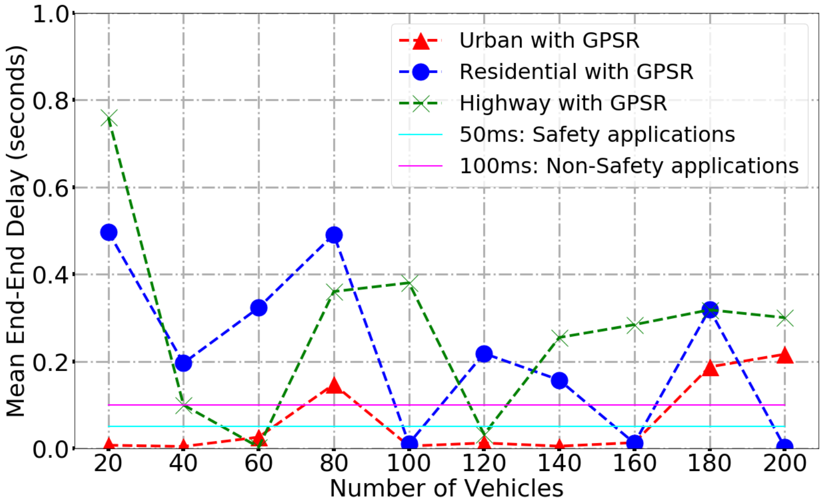


Figure 5.11: Mean End-to-End Delay for GPSR

5.5.3 Mean End-to-End Delay

In Fig. 5.11, we show the E2E delay of the GPSR protocol in presence of obstacles. These results indicate that the criteria for Latency for safety application of 50 ms is not satisfied in most cases for the GPSR protocol. This is due to contributing

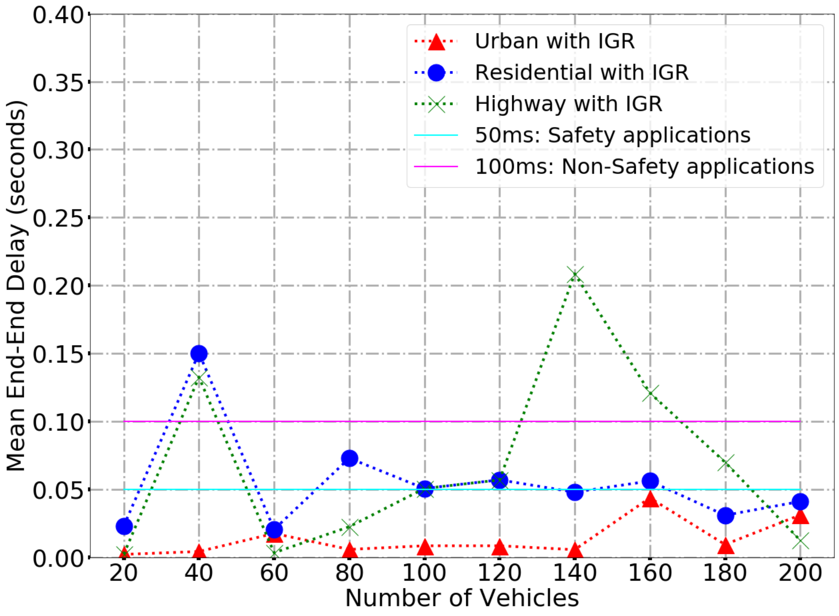


Figure 5.12: Mean End-to-End Delay for IGR

hops by perimeter mode. Due to high mobility the perimeter mode often lead to increased path and delayed the packet in reaching the destination. On the other hand, when the same scenarios are simulated using IGR as shown in Fig. 5.12, the latency is significantly improved. Latency is mostly effected by 2 parameters: multi-hop and weak signal strength. The IGR protocol eliminates the issue of weak signal strength and decreases the latency to values that can be tolerated for safety applications. Therefore, IGR protocol is more suitable for applications in VANET than the traditional GPSR due the satisfied latency parameter. Considering the organized structure of vehicle mobility in Urban scenario, the latency using IGR protocol seems to have very stable latency with any number of of vehicles in-spite of the presence of buildings. Furthermore it is within the latency limit of safety applications.

5.5.4 Limitations with IGR protocol

1. The calculation of estimated average path-loss parameter W requires great computational power, which may be beyond the physical capacity of on-board units.

2. The building data that needs to be available in each on-board unit should be imported from OpenStreetMap [37] and needs to be updated with new building information frequently.

3. Selecting the node that is not the nearest node to the destination can lead to increase in the number of hops. This can also increase the latency, which is undesirable.

4. Careful selection of the matching parameter δ is required during implementation in the physical world. This parameter can dominate the distance to destination criterion or possibly make the IGR ineffective if not selected properly. It is worth noticing that the matching factor $\delta = 0$ makes the IGR perform like simple *greedy* mode.

Chapter 6

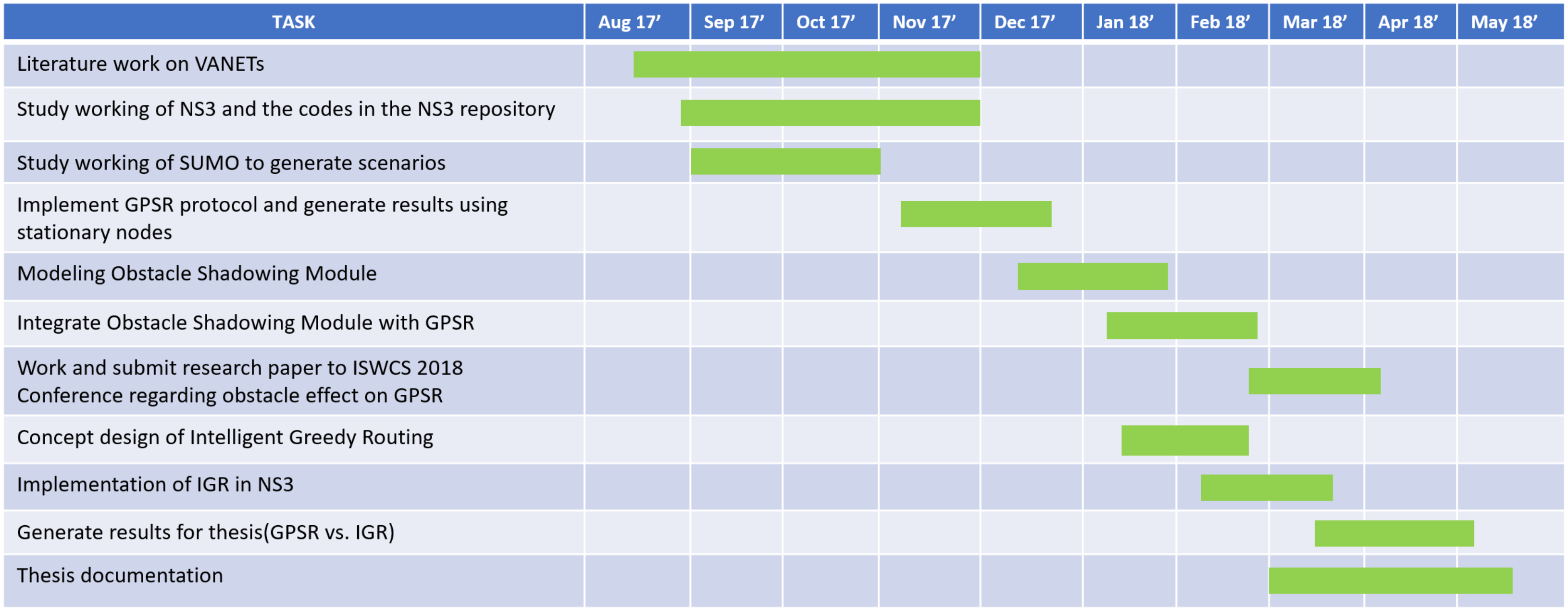
Conclusion and Future Scope

Today, connected vehicle technologies have reached the implementation phase. The increasing focus on the self-driving vehicles has also lead to even more focus on connected vehicle technology. Vehicular Ad-hoc Network (VANET) is the backbone of intelligent transportation systems. Its performance needs to be investigated using accurate research models. The use of realistic models in simulation-based experiments is necessary to test the technology for the reliability of safety-related applications.

The work presented in this thesis regarding the obstacle effects on the GPSR protocol clearly demonstrates the importance of considering losses due to obstacles/buildings. The results in chapter 4 shows, how unreliable the GPSR protocol is when considering realistic topologies (urban, residential, and highway) with obstacles. For the cases we considered, the GPSR protocol proved to significantly decrease the PDR, increase instability in latency and require multiple hops due to perimeter mode. Using the distance parameter as the sole criterion to select the next hop limits GPSR performance. Considering only the distance parameter will not be sufficient for VANETs as there is obstacle effect in real world topologies and also high mobile nodes/vehicles.

The proposed Intelligent Greedy Routing (IGR) proved to solve the issues found with GPSR for obstacle present topologies. In comparison with GPSR, the PDR for the IGR increased significantly; the latency was reduced and now meets implementation specification of VANETs safety and non safety applications. Moreover, packet reach is increased with increased number of hops. Latency for urban topology using the IGR protocol is found to be very promising with the values being less than 50 ms for the maximum number of vehicles in the simulations.

The results presented in this thesis work were generated with extensive simulations. A total of data results from 990 simulations contributed to all the different plots presented in this work, and many more simulations were a part of building and validation of the simulation models. Using a Core i7, 16GB RAM computer system, the time taken to generate the data presented in this work is 25 hours, 17 hours. The timeline followed to develop this work is presented in Fig. 6.1.



TASK	Aug 17’	Sep 17’	Oct 17’	Nov 17’	Dec 17’	Jan 18’	Feb 18’	Mar 18’	Apr 18’	May 18’
Literature work on VANETs	🟩	🟩	🟩	🟩						
Study working of NS3 and the codes in the NS3 repository	🟩	🟩	🟩	🟩						
Study working of SUMO to generate scenarios		🟩	🟩							
Implement GPSR protocol and generate results using stationary nodes				🟩	🟩					
Modeling Obstacle Shadowing Module					🟩	🟩				
Integrate Obstacle Shadowing Module with GPSR						🟩	🟩			
Work and submit research paper to ISWCS 2018 Conference regarding obstacle effect on GPSR							🟩	🟩	🟩	
Concept design of Intelligent Greedy Routing						🟩	🟩			
Implementation of IGR in NS3							🟩	🟩		
Generate results for thesis(GPSR vs. IGR)								🟩	🟩	🟩
Thesis documentation								🟩	🟩	🟩

Figure 6.1: Thesis Timeline

6.1 Future Scope of Work

To estimate the average path loss for the region of neighbor nodes, which is one of the parameters required to calculate *Receptivity* needs building data from OpenStreetMap. This information may be out for scope for the on-board units (OBUs) present in vehicles. It also increases the computational complexity of the system. This parameter needs to be focused and replaced with more realistic path loss estimation models. One possibility could be to have vehicles with advanced sensors such as obstacle detector that could send signal information to the source using *hello* packets. In this case, each vehicle would know the region at which the neighbor nodes are located and what would be their average path loss to send a packet to that region.

The matching factor δ also needs attention and can be configured more accurately with physical world experiments.

With increasing implementation of self-driving vehicles, there is a scope of more data available from different sensors on-board which can detect the surroundings. This information can be shared between vehicles using hello packets and hence more accurate estimation of average path-loss (W) can be possible with this information.

To increase the reliability of IGR and to model more accurately more testing needs be done both in simulation environment and in physical world.

References

[1] https://www.technologyreview.com/s/534981/car-to-car-communication/.

[2] J., K. *Proceedings of the IEEE* **2011**, *99*(7).

[3] Its research 2015-2019 connected vehicles. https://www.its.dot.gov/research_areas/connected_vehicle.htm.

[4] Campolo, C.; Molinaro, A.; Scopigno, R. *Standards, Solutions, and Research* **2015**.

[5] Jon, P. Internet protocol–darpa internet program protocol specification Technical report, RFC-791, DARPA, **1981**.

[6] Perkins, C.; Belding-Royer, E.; Das, S. Ad hoc on-demand distance vector (aodv) routing Technical report, No. RFC 3561, **2003**.

[7] Paul, B.; Ibrahim, M.; Bikas, M.; Naser, A. *arXiv preprint arXiv:1204.1201* **2012**.

[8] Karp, B.; Kung, H.-T. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM, 2000.

[9] Lochert, C.; Hartenstein, H.; Tian, J.; Fussler, H.; Hermann, D.; Mauve, M. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 156–161. IEEE, 2003.

[10] Carpenter, S. E.; Sichitiu, M. L. In *Proceedings of the 2015 Workshop on Ns-3*, WNS3 '15, pages 17–24, New York, NY, USA, 2015. ACM.

[11] Härri, J.; Tchouankem, H.; Klemp, O.; Demchenko, O. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 1645–1650. IEEE, 2013.

[12] Abdelgader, A.; Lenan, W. In *Proceedings of the world congress on engineering and computer science*, Vol. 2, page 71, 2014.

[13] Liang, W.; Li, Z.; Zhang, H.; Wang, S.; Bie, R. *International Journal of Distributed Sensor Networks* **2015**, *11*(8), 745303.

[14] Rajkumar, M. N.; Nithya, M.; HemaLatha, P. *International Research Journal of Engineering and Technology (IRJET)* **2016**, *3*(1), 137–142.

[15] Hartenstein, H.; Laberteaux, K. *VANET: vehicular applications and internetworking technologies*, Vol. 1; John Wiley & Sons, 2009.

[16] Li, Y. J. In *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pages 544–558. Springer, 2010.

[17] Böhm, A.; Jonsson, M. *International Journal of Vehicular Technology* **2011**, *2011*.

[18] Venkatesh, I. *Routing protocols for vehicular adhoc networks (VANETS): a review Journal of Emerging Trends in Computing and Information Sciences* **2014**, *51*.

[19] Jabbarpour, M. R.; Marefat, A.; Jalooli, A.; Noor, R. M.; Khokhar, R. H.; Lloret, J. *Wireless Networks* **2015**, *21*(3), 911–929.

[20] Allal, S.; Boudjit, S. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on*, pages 323–328. IEEE, 2012.

[21] Khairnar, M.; Vaishali, D.; Kotecha, D.; others. *arXiv preprint arXiv:1311.1378* **2013**.

[22] Carpenter, S. E. In *Network Protocols (ICNP), 2014 IEEE 22nd International Conference on*, pages 480–482. IEEE, 2014.

[23] Lochert, C.; Hartenstein, H.; Tian, J.; Fussler, H.; Hermann, D.; Mauve, M. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 156–161. IEEE, 2003.

[24] Sommer, C.; Eckhoff, D.; German, R.; Dressler, F. In *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on*, pages 84–90. IEEE, 2011.

[25] Simulation of urban mobility (sumo). `http://www.sumo.dlr.de/index.html`.

[26] Lan, K.-C. *Telematics Communication Technologies and Vehicular Networks: Wireless Architectures and Applications: Wireless Architectures and Applications* **2009**, page 355.

[27] Lan, K.; Chou, C.-M. *Proceedings of OSDOC* **2011**, *11*, 1–2.

[28] Citymob simulator. `http://www.grc.upv.es/Software/citymob.html`.

[29] Martinez, F. J.; Cano, J.-C.; Calafate, C. T.; Manzoni, P. In *Communications Workshops, 2008. ICC Workshops' 08. IEEE International Conference on*, pages 370–374. IEEE, 2008.

[30] Vanetmobisim project. `http://vanet.eurecom.fr/`.

[31] Straw simulator. `http://www.aqualab.cs.northwestern.edu/projects/111-c3-car-to-car-cooperation-for-vehicular-ad-hoc-networks`.

[32] Miller, J.; Horowitz, E. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pages 18–23. IEEE, 2007.

[33] Vissim simulator. `http://vision-traffic.ptvgroup.com/en-us/products/ptv-vissim/`.

[34] Teramoto, E.; Baba, M.; Mori, H.; Asano, Y.; Morita, H. In *Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on*, pages 484–489. IEEE, 1997.

[35] Corsim simulator. `http://mctrans.ce.ufl.edu/featured/tsis/`.

[36] Cameron, G. D.; Duncan, G. I. *The Journal of Supercomputing* **1996**, *10*(1), 25–53.

[37] Open street map (osm). `https://www.openstreetmap.org/`.

[38] Ns3- official website. `https://www.nsnam.org/`.

[39] Zeng, X.; Bagrodia, R.; Gerla, M. In *ACM SIGSIM Simulation Digest*, Vol. 28, pages 154–161. IEEE Computer Society, 1998.

[40] Staged network simulator. `http://mctrans.ce.ufl.edu/featured/tsis/`.

[41] Georgia tech network simulator (gtnets). `http://griley.ece.gatech.edu/MANIACS/GTNetS/`.

[42] Jist/swans simulator. `http://jist.ece.cornell.edu/`.

[43] Qualnet simulator. `https://web.scalable-networks.com/qualnet-network-simulator-software`.

[44] Omnet++ simulator. `https://www.omnetpp.org/`.

[45] Netsim simulator. `https://www.tetcos.com/`.

[46] J-sim network simulator. http://www.physiome.org/.

[47] Chang, X. In *Proceedings of the 31st Conference on Winter Simulation: Simulation—a Bridge to the Future - Volume 1*, WSC '99, pages 307–314, New York, NY, USA, 1999. ACM.

[48] Opnet simulator. http://www.opnet.com/.

[49] Krajzewicz, D.; Erdmann, J.; Behrisch, M.; Bieker, L. *International Journal On Advances in Systems and Measurements* **2012**, *5*(3&4).

[50] Martinez, F. J.; Toh, C. K.; Cano, J.-C.; Calafate, C. T.; Manzoni, P. *Wireless Communications and Mobile Computing* **2011**, *11*(7), 813–828.

[51] Gnuplot. http://www.gnuplot.info/.

[52] Matplotlib. https://matplotlib.org/.

[53] Xgraph. http://www.xgraph.org/.

[54] Fonseca, A.; Cames, A.; Vazo, T. In *Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques*, pages 353–358. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2012.

[55] Gpsr code review. Fonseca, A. https://codereview.appspot.com/5401042/.

[56] Friis, H. T. *Proceedings of the IRE* **1946**, *34*(5), 254–256.

[57] Ns3- propagation models. https://www.nsnam.org/docs/models/html/propagation.html#id2.

[58] Carneiro, G.; Fortuna, P.; Ricardo, M. In *Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools*, page 1. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[59] Computational geometry algorithms library(cgal). https://www.cgal.org/.

[60] Lee, K. C.; Lee, U.; Gerla, M. *Advances in vehicular ad-hoc networks: Developments and challenges* **2010**, pages 149–170.

[61] Toh, C. K. *Ad hoc mobile wireless networks: protocols and systems;* Pearson Education, 2001.

[62] Johnson, D. B.; Maltz, D. A.; Broch, J.; others. *Ad hoc networking* **2001**, *5*, 139–172.

[63] Lochert, C.; Mauve, M.; Fler, H.; Hartenstein, H. *SIGMOBILE Mob. Comput. Commun. Rev.* **2005**, pages 69–72.

[64] Hong, K.; Xing, D.; Rai, V.; Kenney, J. In *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, pages 63–68. ACM, 2009.

[65] Martinez, F. J.; Fogue, M.; Toh, C.-K.; Cano, J.-C.; Calafate, C. T.; Manzoni, P. *Wireless Personal Communications* **2013**, *69*(2), 639–663.

[66] Obstacle shadowing model: Code review. Carpenter, S. https://codereview.appspot.com/201200043/.

[67] Real network simulator. https://www.cs.cornell.edu/skeshav/real/overview.html.