TRAFFIC MANAGEMENT IN VEHICULAR AD-HOC NETWORK (VANET)

Project Report submitted in partial fulfilment of the requirement for the degree of

Bachelor of Technology

in

Information Technology

under the Supervision of

Mr. Ravindra Bhatt

Submitted By

Dhruv Arora(111445)

Jaypee University Of Infomation Technology,

Waknaghat, Solan – 173234,

Himachal Pradesh

Certificate

This is to certify that project report entitled "TRAFFIC MANAGEMENT IN VEHICULAR AD-HOC NETWORK (VANET)", submitted by DHRUV ARORA (111445) in partial fulfilment for the award of degree of Bachelor of Technology in Information Technology (IT) to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Date:

Mr. Ravindra Bhatt

Assistant Professor

Dept. of Computer Science & Engineering

Jaypee University of Information Technology

Acknowledgement

I would like to articulate my profound gratitude and indebtedness to those people who helped me in the project. First of all, I would like to express my obligation to my project guide Mr. Ravindra Bhatt for his motivation, help and supportiveness. I am sincerely thankful to him for his guidance and helping effort in improving my knowledge on the subject. He has always been helpful to me in all aspects and I thank him from the bottom of my heart.

An assemblage of this nature could never have been attempted without reference to and inspiration from the works of others whose details are mentioned in reference section. I acknowledge my indebtedness to all of them.

I am also grateful to Mr. Amit Singh (CSE Project lab) for his practical help and guidance.

(DHRUV ARORA)

Abstract

Vehicular Ad-hoc Network(VANET) are expected to support a large spectrum of mobile distributed applications that range from traffic alert dissemination and dynamic route planning to context-aware advertisement and file sharing. VANET helps in defining the safety measures in the vehicles, streaming communication between different vehicles, infotainment and telematics. Vehicular Ad-hoc Networks are expected to implement a variety of wireless technologies like Dedicated Short Range Communications (DSRC) which is a type of Wi-Fi. Vehicular Ad-hoc Networks can be viewed as component of the Intelligent Transportation Systems (ITS).

Traffic congestion has a number of negative effects and is a major problem in today's society. Several techniques have been deployed to deal with this problem. In this project, we have proposed an innovative approach to solve one of the many problems caused by traffic congestion. The system is developed and tested using the SUMO simulator. "Simulation of Urban Mobility", or "SUMO" for short, is an open source, microscopic, multi-model traffic simulation tool. It allows us to simulate the given traffic demand, that consists of many vehicles moves through a given roadmap. The simulation allows addressing a large number of traffic management topics. It is purely microscopic in nature: each vehicle is modelled explicitly, has its own route, and moves individually through the network. Using simulations the applicability of the algorithm in the domain of vehicle traffic congestion in a VANET, is demonstrated.

With the introduction of intelligent vehicles and to rely less on infrastructure, a distributed protocol is designed for the spontaneous creation of virtual traffic lights in a collaborative and synchronized fashion between vehicles, but without a centralized control infra-structure. For cooperation between all the vehicles that are approaching the intersection , one of them must be chosen as a *Leader*. This temporary elected leader will work as an infrastructure and has the responsibility of controlling the traffic. We have used Bully Algorithm for leader election.

Contents

Certificate i

Acknowledgement ii

Abstract iii

List of Figures vi

1. Introduction

1.1 What is VANET? 1

1.2 Properties of Vehicle for VANETs 2

1.3 What is Smart Vehicle? 2

1.4 VANET Applications 4

1.5 Implementation of VANETs 7

1.6 Security Architecture of VANETs 9

1.6.1 Tamper Proof Device 9

1.6.2 Digital Signature 10

1.6.3 Vehicular PKI 10

1.7 Types of Communication in VANETs 11

1.7.1 Vehicle to Vehicle Communication 11

1.7.2 Vehicle to Infrastructure Communication 12

1.8 VANET-enabled In-Vehicle Traffic Signs 14

1.8.1 Intersection Management with Virtual Traffic Lights 15

1.9 Challenging Issues in VANET 17

1.10 Motivation 17

1.11 Proposed Work 18

1.12 Organisation of the Work 18

2. Literature Survey

2.1 Traffic Signal Control Systems 19

2.1.1 Classification of Traffic Signal Control Systems 20

2.2 VANET Settings 22
2.3 Issues in Data Forwarding 23
2.4 Simulation of Urban Mobility (SUMO) 25

3. Proposed Work

3.1 Priority Based Signalling 29
3.2 Infrastructure less System 31
    3.2.1 Bully Algorithm (Leader Election) 31

4. Simulation and Results

4.1 Creation of Road network 33
4.2Traffic Movement in a Network 34
4.3 Simulation of Bully Algorithm 34
4.4 Results 36
6. Conclusion 39
7. References 40

LIST OF FIGURES

1.1 Vehicular Ad- hoc Network 1
1.2 Components of Smart Vehicle 3
1.3 Congestion Detection 7
1.4 Security Architecture of VANET 9
1.5 Tamper Proof Device 9
1.6 Application of Digital signature 10
1.7 Lane Obstruction Example 15
1.8 Vehicles approaching an intersection 16
1.9 Flowchart representing the process for creating a virtual traffic light in an intersection 17
2.1 Static Systems 20
2.2 Dynamic Systems 21
2.3 Communication via VANET 22
2.4 Mobile Vehicle Tracking System 24
2.5 Flow Chart of creation of VANET 28
3.1 Work Flow Diagram 29
3.2 Bully Algorithm 31

INTRODUCTION

1.1 **What is VANET?**

A Vehicular Ad-Hoc Network or **VANET** is a technology that has moving vehicles as nodes in a network for creating a mobile network. We can say that VANET turns each and every vehicle into a wireless node, allowing cars to connect to each other which are 100-300 metres apart and, in turn, create a wide range of network. As cars fall out due to signal range and drop out of the present network, other cars can join in to connect vehicles to one another so a mobile Internet can be created. It is assumed that the first systems in which it will be integrated are police and fire vehicles, to communicate with one another to provide safety.
It is a term which is used to describe the spontaneous ad hoc network that is formed over vehicles moving on the roads. Vehicular networks are very fast emerging for deploying and developing new and traditional applications. It is characterized by rapidly changing topology, high mobility, and ephemeral, one-time interactions. Both MANETs and VANETs are characterized from the movement and self-organization of the nodes (*i.e.*, in the case of VANETs it is Vehicles).

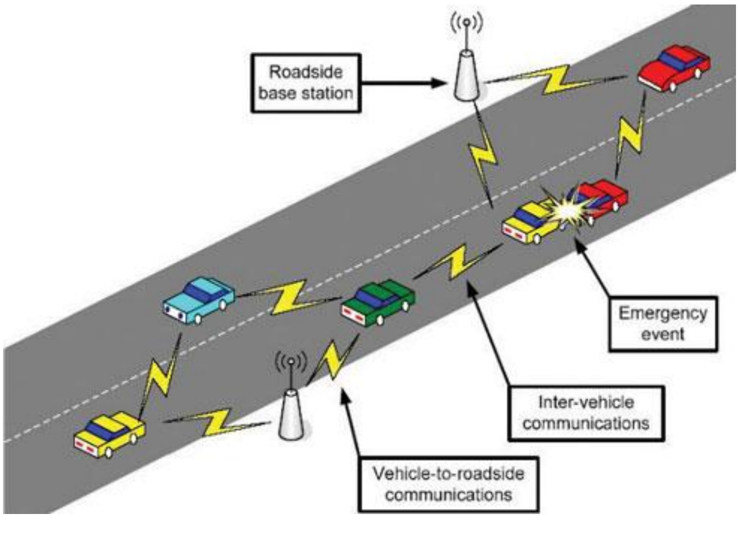


Fig 1.1:- Vehicular Ad- hoc Network (Source: http://www. vehicularcommunication.blogspot.com [1])

1.2 Properties of Vehicle for VANET:

The moving vehicles are the main component in VANET. These vehicles are known as nodes in VANET. The following properties of vehicles are used for better operation in VANET. They are following.

a) **Sensing:** The different types of sensor are used for sense the Different vehicular and environmental conditions (state of the vehicle, state of road, weather condition, pollution and others).

b) **Processing:** the data or information coming from the different sensors are processed by the vehicles.

c) **Storage:** to store the different type of data and processing results for future uses require a large storage space.

d) **Routing:** The vehicles (nodes) should have the potential to communicate with each other in the VANET (IP or Cellular for example).

1.3 What is Smart Vehicle?

Vehicles which are equipped with multi interface cards, sensors and on-board units are called Smart vehicles. The number of vehicles equipped with on-board wireless devices (*e.g.*, UMTS, IEEE 802.11*p*, Bluetooth, etc.) and sensors (*e.g.*, radar, etc.), are increasing for efficient transport and management applications are focused on optimizing flows of vehicles by reducing the time taken to travel and avoiding any traffic congestions. As an instance, the radar present on on-board could be used to sense traffic congestions and automatically slow the vehicle. In another accident warning system, sensors can be used to determine that a crash can be occurred, so air bags are deployed; this kind of information is then relayed via V2I or V2V within the vehicular network.

It provides different levels of functionality by using number of systems and sensors. The major systems and sensors exploited for intra-vehicle communications we cite: crash sensors, the data recorder, the braking system, the engine control unit, the electronic stability control, the infotainment system, the integrated starter generator, the electronic steering, the tire pressure monitoring system , the power distribution and connectivity, the lighting system, seat belt sensors , etc. For the brake systems, there are also the antilock brake system and the parking brake system. The parking brake is also referred to as an emergency brake; it controls the rear brakes using a series of steel cables. It allows the vehicle to be stopped in the event of a total brake failure. Vehicle-mounted cameras are mainly used to display images on the vehicle console in a smart vehicle.

Commonly, a smart vehicle is equipped with the following technologies and devices:

(i) A wireless transceiver for data transmissions among vehicles (V2V) and from vehicles to RSUs (V2I);

(ii) A Central Processing Unit (CPU) which implements the applications and communication protocols;

(iii) A Global Positioning Service (GPS) receiver for navigation and positioning services;

(iv) An input/output interface for the interaction of human with the system;

(v) Different sensors lying outside and inside the vehicle are used to measure various types of parameters (*i.e.*, acceleration, speed, distance between the neighbouring vehicles, etc.

The following figure depicts the different components of a smart vehicle: Controller Area Network, Media Oriented System Support and Local Interconnect Network and their location inside a vehicle.

CAN(Controller Area Network)

- Lock
- Instruments
- Central Body Control
- Climate
- Light

MOST(Media Oriented System Support)

- Speakers
- Digital Radio
- Navigation
- GPS
- GSM

LIN(Local Inteconnect Network)

- Heating
- Mirror
- Roof

Fig1.2:- Components of a Smart Vehicle and their location inside the vehicle

The basic idea behind smart vehicles is to address safety issues of vehicles, and then with a proper combination of functionalities like communications, control and computing technologies, it will become possible to assist the driver decisions, and will also help to prevent driver’s wrong behaviours’ .The control functionality is directly added into smart vehicles for connecting it with the vehicle’s electronic equipment.

1.4 VANET APPLICATIONS

Vehicular applications are classified in the following categories [2]:-

1) Safety oriented
2) Commercial oriented
3) Convenience oriented
4) Productive Applications

- **Safety Applications**

  Safety applications include monitoring of the surrounding road, approaching vehicles, surface of the road, road curves etc [2]. The Road safety applications can be classified as:

  i. **Real-time traffic**: The real time traffic data can be stored at the RSU and can be available to the vehicles whenever and wherever needed. This can play an important role in solving the problems such as traffic jams, avoid congestions and in emergency alerts such as accidents etc.

  ii. **Co-operative Message Transfer**: Slow/Stopped Vehicle will exchange messages and co-operate to help other vehicles. Though reliability and latency would be of major concern, it may automate things like emergency braking to avoid potential accidents. Similarly, emergency electronic brake-light may be another application.

  iii. **Post Crash Notification**: A vehicle involved in an accident would broadcast warning messages about its position to trailing vehicles so that it can take decision with time in hand as well as to the highway patrol for tow away support as depicted in Figure 2.

  iv. **Road Hazard Control Notification**: Cars notifying other cars about road having landslide or information regarding road feature notification due to road curve, sudden downhill etc.

  v. **Cooperative Collision Warning**: Alerts two drivers potentially under crash route so that they can mend their ways.

  vi. **Traffic Vigilance**: The cameras can be installed at the RSU that can work as input and act as the latest tool in low or zero tolerance campaign against driving offenses.

- **Commercial Applications**

Commercial applications will provide the driver with the entertainment and services as web access, streaming audio and video. The Commercial applications can be classified as:

i. **Remote Vehicle Personalization/ Diagnostics:** It helps in downloading of personalized vehicle settings or uploading of vehicle diagnostics from/to infrastructure.

ii. **Internet Access**: Vehicles can access internet through RSU if RSU is working as a router.

iii. **Digital map downloading**: Map of regions can be downloaded by the drivers as per the requirement before travelling to a new area for travel guidance. Also, Content Map Database Download acts as a portal for getting valuable information from mobile hot spots or home stations.

iv. **Real Time Video Relay:** On-demand movie experience will not be confined to the constraints of the home and the driver can ask for real time video relay of his favorite movies.

v. **Value-added advertisement:** This is especially for the service providers, who want to attract customers to their stores. Announcements like petrol pumps, highways restaurants to announce their services to the drivers within communication range. This application can be available even in the absence of the Internet.

- **Convenience Applications**

Convenience application mainly deals in traffic management with a goal to enhance traffic efficiency by boosting the degree of convenience for drivers. The Convenience applications can be classified as:

i. **Route Diversions**: Route and trip planning can be made in case of road congestions.

ii. **Electronic Toll Collection**: Payment of the toll can be done electronically through a Toll Collection Point. A Toll collection Point shall be able to read the OBU of the vehicle. OBUs work via GPS and the on-board odometer or techograph as a back-up to determine how far the Lorries have travelled by reference to a digital map and

GSM to authorize the payment of the toll via a wireless link. TOLL application is beneficial not only to drivers but also to toll operators.

iii. **Parking Availability**: Notifications regarding the availability of parking in the metropolitan cities helps to find the availability of slots in parking lots in a certain geographical area.

iv. **Active Prediction**: It anticipates the upcoming topography of the road, which is expected to optimize fuel usage by adjusting the cruising speed before starting a descent or an ascent. Secondly, driver is also assisted

- **Productive Applications**

We are intentionally calling it productive as this application is additional with the above mentioned applications.

The Productive applications can be classified as:

i. **Environmental Benefits**: AERIS research program [11] is to generate and acquire environmentally-relevant real-time transportation data, and use these data to create actionable information that support and facilitate “green” transportation choices by transportation system users and operators. Employing a multi-modal approach, the AERIS program will work in partnership with the vehicle-to-vehicle (V2V) communications research effort to better define how connected vehicle data and applications might contribute to mitigating some of the negative environmental impacts of surface transportation.

ii. **Time Utilization**: If a traveller downloads his email, he can transform jam traffic into a productive task and read on-board system and read it himself if traffic stuck. One can browse the Internet when someone is waiting in car for a relative or friend.

iii. **Fuel Saving**: When the TOLL system application for vehicle collects toll at the toll booths without stopping the vehicles, the fuel around 3% is saved, which is consumed when a vehicles as an average waits normally for 2-5 minutes.

1.5 IMPLEMENTATION OF VANETs

Let us assume that a driver has been distracted by something on the panorama and moves the steering wheel, because of that the direction of the vehicle changes accidentally. Once the error is recognized, the driver will react very quickly by changing direction or by using quick and strong brakes. If the very first following driver does not experience accidents, then the vehicle does not forward this type of information and false alarm probability is very much reduced. With the use of V2I architecture, the access points must gather information (e.g. Alarms for quick changes in speed), coming from different vehicles and the data is merged that reduces the signalling from the vehicles.

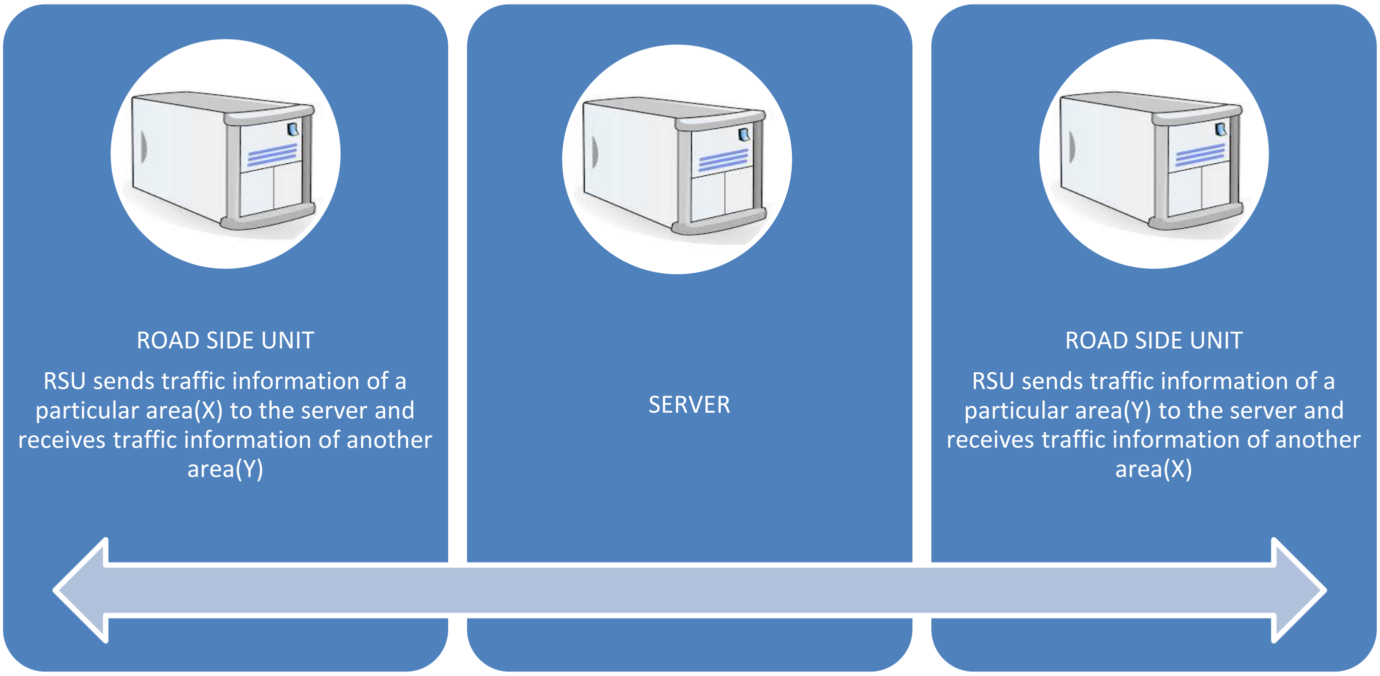


Fig 1.3: Congestion Detection (Continuous exchange of information between RSU and Server)

Traffic monitoring and management are very essential to avoid traffic Congestion and maximize road capacity. Crossing intersections in the streets of city can be dangerous and tricky at times. Traffic light scheduling can be used by the drivers to cross intersections. Allowing a very smooth flow of traffic can mainly reduce travel time and increase vehicle throughput. A token- based intersection traffic management scheme is a scheme in which each vehicle waits to take a token before entering in an intersection. On the other hand, with knowledge of traffic conditions, drivers should optimize their driving routes, thereby the problem of traffic congestion can be minimised.

Congestion Road Notification detects and notifies about road congestions that can be used for route and trip planning. This kind of use is partially implemented in the recent GPS-based applications where a new route is evaluated when heavy congestion has been detected on the present route or in a portion of it. Till now several commercial tools are available for the smart- phones and special purpose devices.

Traffic Message Channel messages have a considerable amount of information:

- Location: the road, area or specific location affected;
- Identification: what is the cause of the traffic problem and its seriousness;
- Extent: how far the problem stretches in each direction;
- Direction: the directions of the traffic which are affected;
- Diversion advice: use alternative routes to avoid the congestion.
- Duration: how long the traffic flow problem is affected;

The service provider (SP) encodes the message and sends it to FM radio broadcasters, which transmit messages as an RDS (Radio Data System) signal within the normal FM radio transmissions. There is usually only about 30 seconds gap between the first report of an incident to the traffic information centre (TIC) and the RDS-TMC receiver getting the message. This application may be implemented according to a V2I configuration or a V2V one. In fact, it is possible to encapsulate information about the direction, the position and the average speed, that are then communicated back to the vehicle following on the street. As it appears very clearly, this solution suffers for a much large amount of data to be processed by the vehicles themselves. The worth in this environment is the use of V2I since access points can process information coming to it and communicate to the incoming vehicles the new route after processing request information about their destination. Finally, Cooperative Collision Warning system works with a cut-out revealing a stopped car or slow-moving car before its arrival at the curve or downhill. All these type of applications require radio transceivers for the exchange of messages sensor and GPS on board car and road side units. Even in this case the dualism between V2I and V2V is renovated.

In general, these applications are motivated by the desire of the passengers to Communicate either with ground-based destinations or other vehicles (e.g., Internet hosts or the Public Service Telephone Network).As an instance, the driver could receive local information about the hotels and restaurants etc. The main aim of infotainment applications is to offer comfort and convenience to drivers and/or passengers. For example, Fleetnet provides a platform for gaming and peer-to-peer file transfer on the road. A real-time parking navigation system is proposed to inform drivers of any available parking space. Digital billboards for the vehicular networks are proposed for advertisement. Internet access can be provided to the vehicle through V2I communications; therefore, the business activities can be performed as usual in a vanet environment, realizing the notion of mobile office. On-the-road media streaming between the vehicles can be available making long travel more pleasant. An envisioned goal is to have human-vehicle-interfaces, such as colour reconfigurable head-up and head-down displays, and very large touch screen active matrix Liquid Crystal Displays, for the high-quality video-streaming services.

1.6 SECURITY ARCHITECHTURE OF VANETS

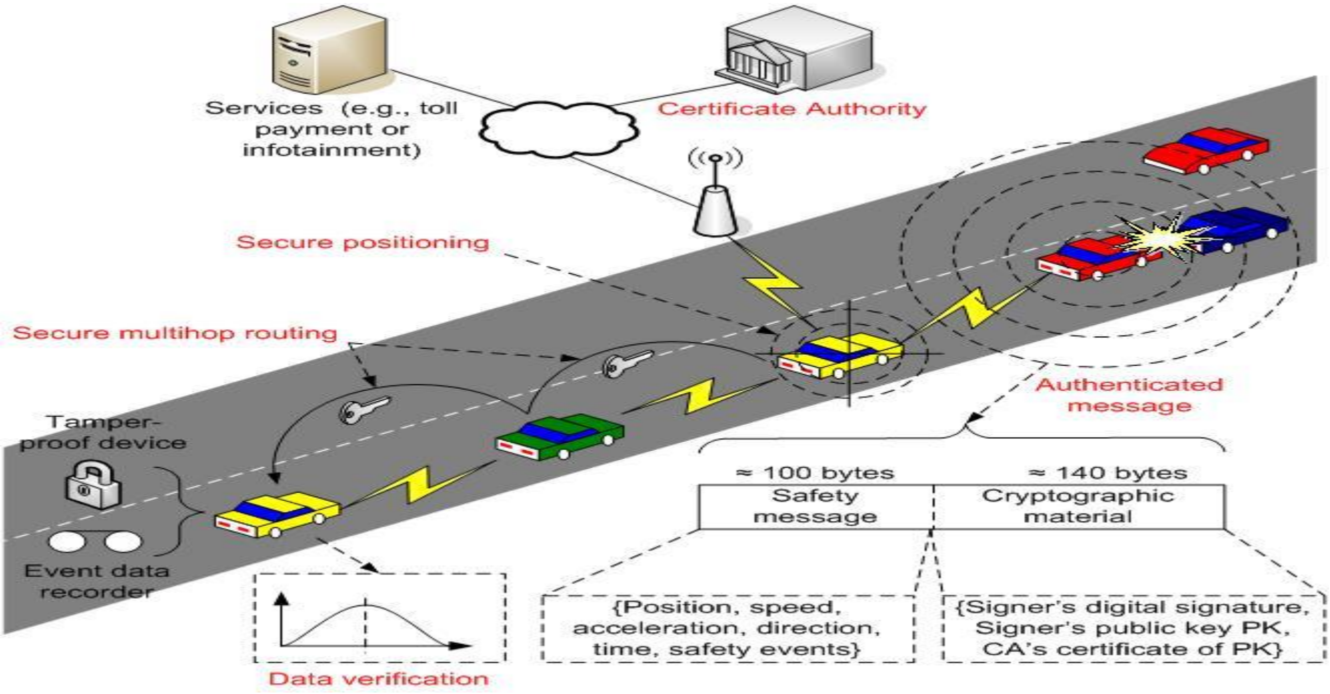


Fig 1.4: Security Architecture Of Vanet (Source: http://www. http://comp.ist.utl.pt/[3])

1.6.1 Tamper Proof Device

- Every smart vehicle carries a tamper-proof device

1. Contains the secrets of the vehicle itself
2. Has its own battery
3. Has its own clock
4. Is in charge of all the security operations
5. Is accessible only by authorized users

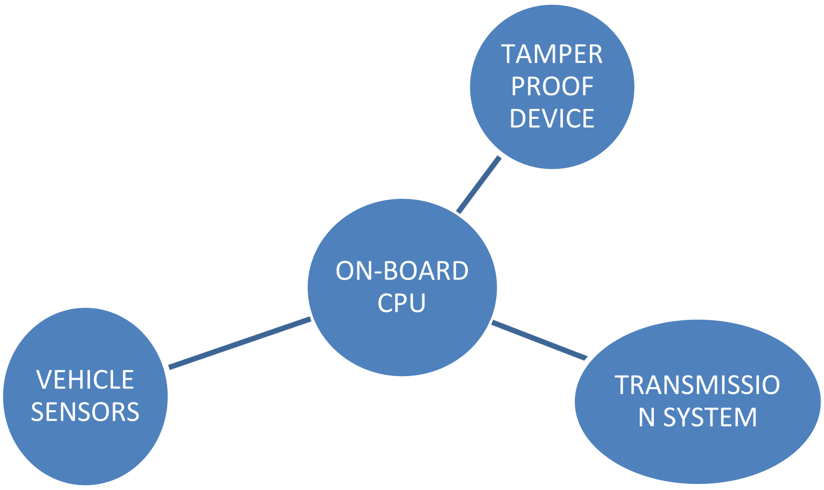


Fig 1.5:- Tamper Proof Device

1.6.2 DIGITAL SIGNATURE

- Symmetric cryptography is not good for messages are of large scale, non-repudiation requirement and standalone.
- Each message that has to be send should be signed with a Digital signature (DS).
- Liability messages must be stored in the EDR.

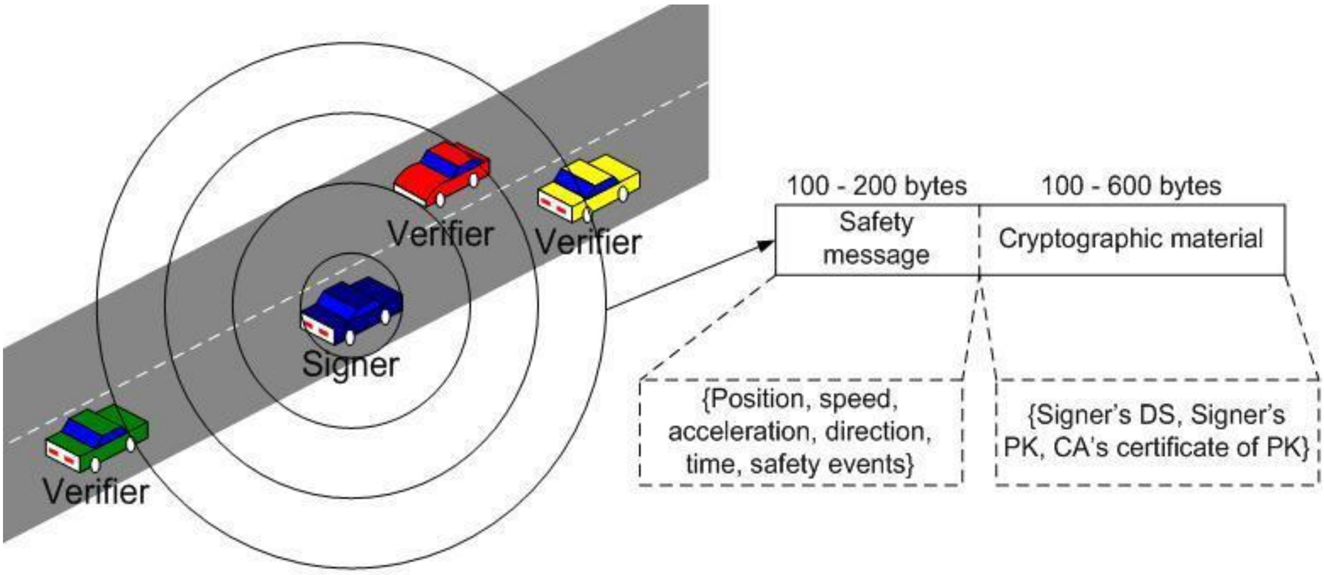


Fig 1.6: Application of Digital signature (Source: http://www. http://comp.ist.utl.pt/[3])

1.6.3 VPKI (Vehicular PKI)

- Each smart vehicle carries in its Tamper-Proof Device (TPD):

    1. A certified and unique identity: Electronic License Plate (ELP)
    2. The set of certified anonymous public / private key pairs

- Before a smart vehicle sends a safety message, it signs the message with its private key and includes the essential CA's certificate.

    1. Mutual authentication can be done without using a server

- Authorities (regional or national) are cross-certified.

1.7 Types of Communications in Vanet

1.7.1 Vehicle to vehicle communication

Vehicle-to-vehicle (V2V) communications comprises a wireless network where automobiles send messages to each other with information about what they're doing. This data would include speed, location, and direction of travel, braking, and loss of stability. Vehicle-to-vehicle technology uses dedicated short-range communications (DSRC), a standard set forth by bodies like FCC and ISO. Sometimes it's described as being a Wi-Fi network because one of the possible frequencies is 5.9GHz, which is used by Wi-Fi, but it's more accurate to say "Wi-Fi-like." The range is up to 300 meters or 1000 feet or about 10 seconds at highway speeds (not 3 seconds as some reports say).

V2V would be a mesh network, meaning every node (car, smart traffic signal, etc.) could send, capture and retransmit signals. Five to 10 hops on the network would gather traffic conditions a mile ahead. That's enough time for even the most distracted driver to take his foot off the gas.

On the first cars, V2V warnings might come to the driver as an alert, perhaps a red light that flashes in the instrument panel, or an amber then red alert for escalating problems. It might indicate the direction of the threat. All that is fluid for now since V2V is still a concept with several thousand working prototypes or retrofitted test cars. Most of the prototypes have advanced to stage where the cars brake and sometimes steer around hazards. Why? It's more exciting for a legislator or journalist to see a car that stops or swerves, not one with a flashing lamp. It uses two types of broadcasting i.e.

1. **Naive broadcasting**

   In naive broadcast, the vehicle detecting an emergency event starts sending warnings periodically. Upon receiving it, the other vehicles start sending their own periodical warnings if the message they received come from their front. In intelligent broadcast with implicit acknowledgement, both the initiator as well as the repeaters cancels their periodical transmission when they hear the same warning coming from a node at their back. All receivers wait for a random time before starting to send their own warnings to see whether another node starts before them. If they do, they come to the conclusion that the warning has already propagated successfully, and do not start sending messages.

2. **Intelligent broadcasting**

   In this more intelligent version, the first vehicle initiates its sequence as in naive broadcast, but it stops sending messages as soon as it overhears another vehicle at its back sending the same message, which is a sign showing that the warning has successfully propagated further down the road. The repeaters also start their sequence as in naive broadcast, but they, too, stop if they overhear others at their back repeating the warning. In addition to their periodic broadcast interval, the repeaters must also wait for a random duration t(wait) , where 0<t(wait)<t(max) and t(max) is the maximum waiting time, before sending their messages. If, while they are waiting, the stopping condition is satisfied, they cancel their sequence immediately since there is no need to repeat the warning any more.

In an emergency situation like the possible collision of vehicles, it is of extreme importance that the application in charge of delivering the warning does the right thing. Thus, the effectiveness metric is defined as the percentage of the vehicles having received the collision warning in a timely manner at the end of the simulation. There are four performance criteria to measure the effectiveness of these algorithms:-

i. Warning Effectiveness

In an emergency situation like the possible collision of vehicles, it is of extreme importance that the application in charge of delivering the warning does the right thing. Thus, the effectiveness metric is defined as the percentage of the vehicles having received the collision warning in a timely manner at the end of the simulation.

ii. Warning Efficiency

This metric is about evaluating the algorithms' ability to do things right. It is measured by observing two phenomena. The first one is the number of messages generated per vehicle until all reachable vehicles have been warned. If all the vehicles in the network can actually be reached before the simulation ends, this number gives us a per-vehicle average of the number of required warnings for a particular algorithm to cover completely the safety area. Otherwise, it gives us the average number of warnings to be sent by each vehicle to reach all the reachable vehicles.

iii. Warning Propagation

The time required to reach all vehicles, or the last reachable vehicle in case not all are reachable, is the time for the algorithm to complete. So the propagation metric gives us an idea about how quickly the warning messages are disseminated throughout the vehicular network and, thus, how quickly the algorithm converges.

iv. Warning Overhead

Based on the volume of the traffic generated by the broadcast algorithm, other network applications may be affected by the safety warning application in various levels.

1.7.2 Vehicle to infrastructure communication

Vehicle to Infrastructure provides solution to longer-range vehicular networks. It makes use of preexisting network infrastructure such as wireless access points (Road-Side Units, RSUs). Communications between vehicles and RSUs are supported by Vehicle-to-Infrastructure (V2I) protocol and Vehicle-to-Roadside (V2R) protocol. The Roadside infrastructure involves additional installation costs. The V2I infrastructure needs to leverage on its large area coverage and needs more feature enhancements for Vehicle Applications.

The characteristic of highly dynamic topology makes the design of efficient routing protocols for VANET is challenging. The routing protocol of VANET can be classified into two categories such as Topology based routing protocols & Position based routing protocols.

1. POSITION-BASED ROUTING PROTOCOL

   Position-based routing provides multi-hop communication in a wireless ad hoc network. It assumes that every node knows its geographic position, e.g. by GPS, and maintains a location table with ID and geographic positions of other nodes as soft state. PBR comprises three core components: beaconing, a location service, and forwarding

2. TOPOLOGY BASED ROUTING PROTOCOLS

   Topology based routing protocols use link’s information within the network to send the data packets from source to destination. Topology based routing approach can be further categorized into proactive (table-driven) and reactive (on-demand) routing.

Proactive (table-driven):

Proactive routing carries the distinct feature: the routing information such as the next forwarding hop is maintained in the background regardless of communication requests. Control packets are constantly broadcast and flooded among nodes to maintain the paths or the link states between any pair of nodes even though some of paths are never used. A table is then constructed within a node such that each entry in the table indicates the next hop node toward a certain destination. The advantage of the proactive routing protocols is that there is no route discovery since route to the destination is maintained in the background and is always available upon lookup.

Reactive (On Demand):

Reactive routing opens a route only when it is necessary for a node to communicate with another node. It maintains only the routes that are currently in use, thereby reducing the burden on the network. Reactive routings typically have a route discovery phase where query packets are flooded into the network in search of a path. The phase completes when a route is found.

1.8 VANET-enabled In-Vehicle Traffic Signs

Understanding traffic sign information correctly is crucial. It helps the driver to anticipate future situations, make decisions and respond in an appropriate way. There are many different kinds of road signs and they are mostly placed above or beside highways and streets. However, these traditional static traffic signs have known limitations. The period of time the drivers have to analyze the information is limited, and even if the road signaling is predominantly standardized, most of the signs use text to convoy meaning restricting the understanding to readers of the language. They are likely to be overlooked during complex driving tasks, and sometimes, the adverse weather conditions or vehicles blocking the line-of-sight between the driver and the sign, make its recognition very difficult. In contrast to traditional traffic signs, traffic signs displayed within the vehicle will solve a big amount of these limitations, as well as they will provide additional help to the driver on his driving tasks. In fact, in-vehicle traffic signs are one of the main ITS technologies.

Regulatory signs play an important role in road safety and traffic efficiency. Traffic light systems are used to regulate and control conflicts between opposing vehicular, or pedestrian traffic movements. While these systems can improve junction capacity and road safety, they also still have some limitations [3]. Most of the signal control systems rely on timing plans generated offline by traffic engineers using optimization models. Traffic, however, seems to be an adaptation problem rather than an optimization problem. Improperly operated traffic lights cause excessive delays that sacrifice productivity, waste fuel, pollute the air, and increase the levels of stress.

Traditional traffic signs have static information and cannot be easily updated to cope with the frequent changes in the environment. Once the information is placed, it remains there until the sign is removed or replaced. Modern traffic signs should allow faster adaptation to the current road or traffic situations like icy roads or accident warnings. Temporary road works, road accidents, or natural disasters are responsible for changes in the road network topology. Nevertheless, traffic should be able to recover rapid and efficiently from these changes. Drivers should be informed about the best actions to follow in order to achieve a efficient traffic flow. In a simple situation of road works, static traffic signs are placed in a near position to inform the driver about the new action to follow. But, in road accidents or disaster situations, placing static traffic signs in order to provide rapid and efficient traffic adaptation is impracticable.

Mobile Ad-hoc Networks (MANETs) are often associated to crisis management applications, such as in a disaster recovery, where the entire communication infrastructure is destroyed and establishing communications quickly is crucial. As VANETs are considered one of the MANETs real-life applications, we consider these networks as a key technology to provide, not only rapid creation and adaptation of wireless networks (that already inherits from MANET features), but also as a mean to rapidly and spontaneously create new traffic rules that provides fast adaption in traffic change situations. For instance (see Figure 1.7), imagine a lane obstruction caused by an accident in a road which has two lanes, one in each direction. As one of the lanes is blocked, vehicles approaching the zone of the obstruction must act cooperatively to create a just-in-time and just-in-place virtual traffic light to regulate and control conflicts between vehicles in different lanes, improving the traffic efficiency and safety in the only lane available. With the introduction of intelligent vehicles, we believe that road transportation and traffic management also needs intelligent traffic signs. Hence, we propose a novel approach to spontaneously create traffic signs through the collaboration of the local vehicles, in a synchronized manner, but without a centralized control infrastructure.

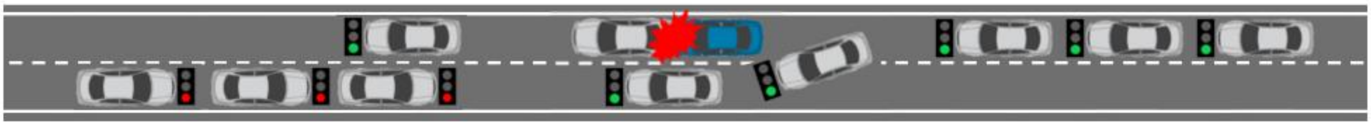
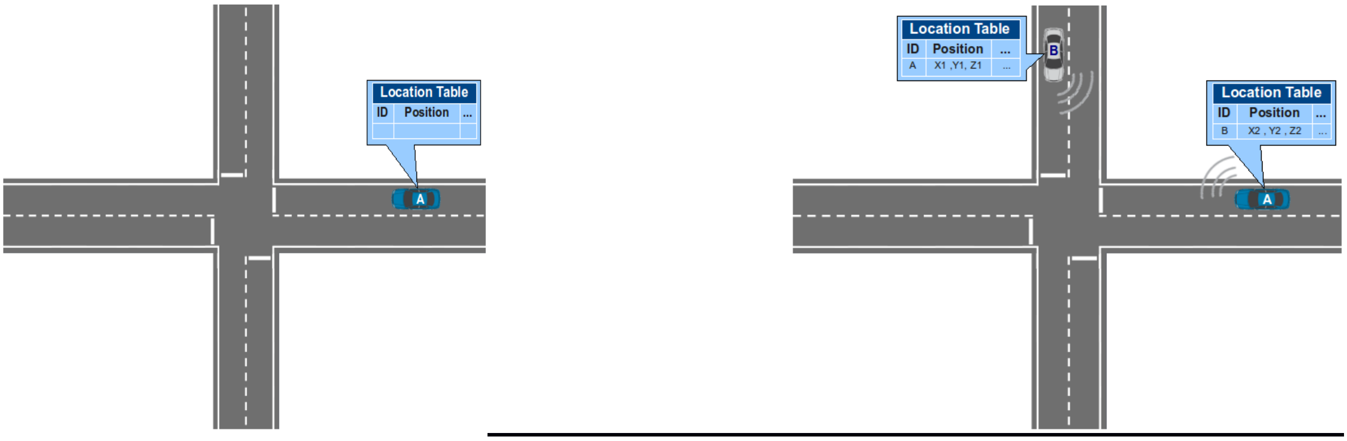


Fig1.7:- Lane obstruction example

1.8.1 Intersection Management with Virtual Traffic Lights

Virtual traffic lights can be used not only at intersections, but in a variety of environments where the crossing conflicts needs to be solved. Regarding the lane obstruction example illustrated in Fig. 1.1, virtual traffic lights can be used to coordinate the traffic flow in the only two available lanes. However, we can see this example as an intersection scenario, where the exact location of the accident is the center of the junction, and the two available lanes represent two entry roads. In fact, every crossing conflict between vehicles can be seen as a conventional intersection of roads. When an accident occurs, the involved vehicles must inform their vicinity about the exact location of the accident. Once the neighbors are informed, they must temporary update its map topology assuming that a new intersection is now present in the exact location of the accident. Then, the virtual traffic light is created as if an intersection really exists in that location.

There are several goals that can be taken into consideration when designing an intersection control system based on traffic lights [4]. As we envision a system where the traffic light infrastructure does not physical exists, a distributed and cooperative protocol must solve the crossing conflicts, minimize the average delay of vehicles approaching an intersection, reduce the queue length of all approaches, and even reduce overall fuel consumption and pollutant missions. Nevertheless, traffic does not follow a well defined pattern, and thus, traffic lights should be adapted to different traffic demands. In the example illustrated in Figure 1.8(a), when the vehicle A approaches the intersection and notices (looking to its LT) that there are no other vehicles moving to cross the same intersection, there is no need for traffic lights. However (Figure 1.8(b)), when vehicle A notices the presence of vehicle B, which is driving to the same intersection, the two vehicles must solve this conflict with a traffic light.



(a) Without vehicles in opposite road

(b) With vehicles in opposite roads

Fig 1.8: Vehicles approaching an intersection

The decision process that might create a Virtual Traffic Light (VTL) at an intersection is illustrated in Fig. 1.9. When a vehicle approaches an intersection and notices that are neighbors attempting to cross the same intersection, it must communicate cooperatively with them in order to create a VTL just-in-time and just-in-place. However, as such VTL does not physically exist; a distributed algorithm is required to ensure that all vehicles can be aware of the traffic light. Thus, as result of the cooperation between all the vehicles that are approaching the intersection, one of them must be chosen to create the VTL and to propagate it - called Leader. This temporary elected Leader will work as an infra-structure and has the responsibility of controlling the VTL. Once the Leader is elected and starts broadcasting (periodically) the VTL information to its vicinity, the other vehicles act as passive nodes in the protocol (just listen the channel for traffic lights).

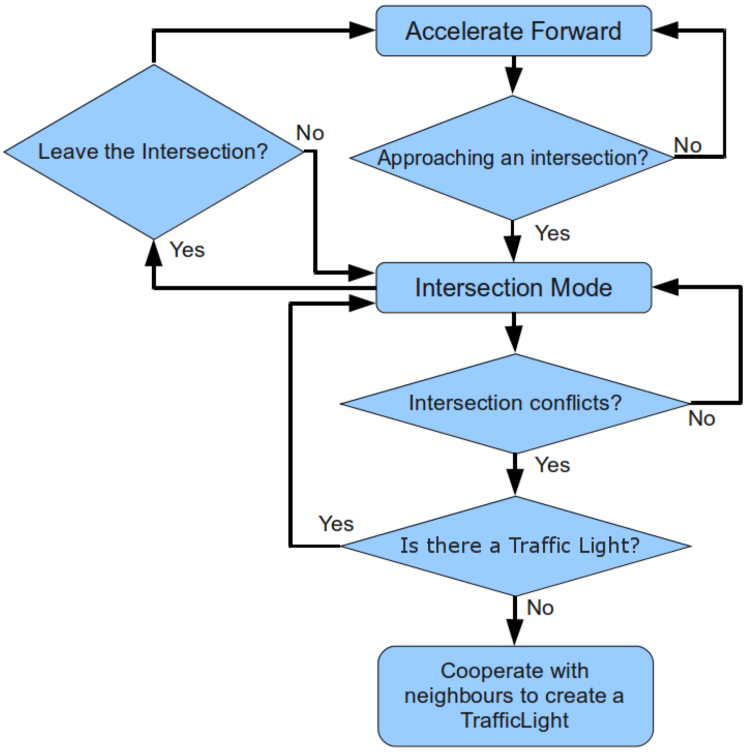
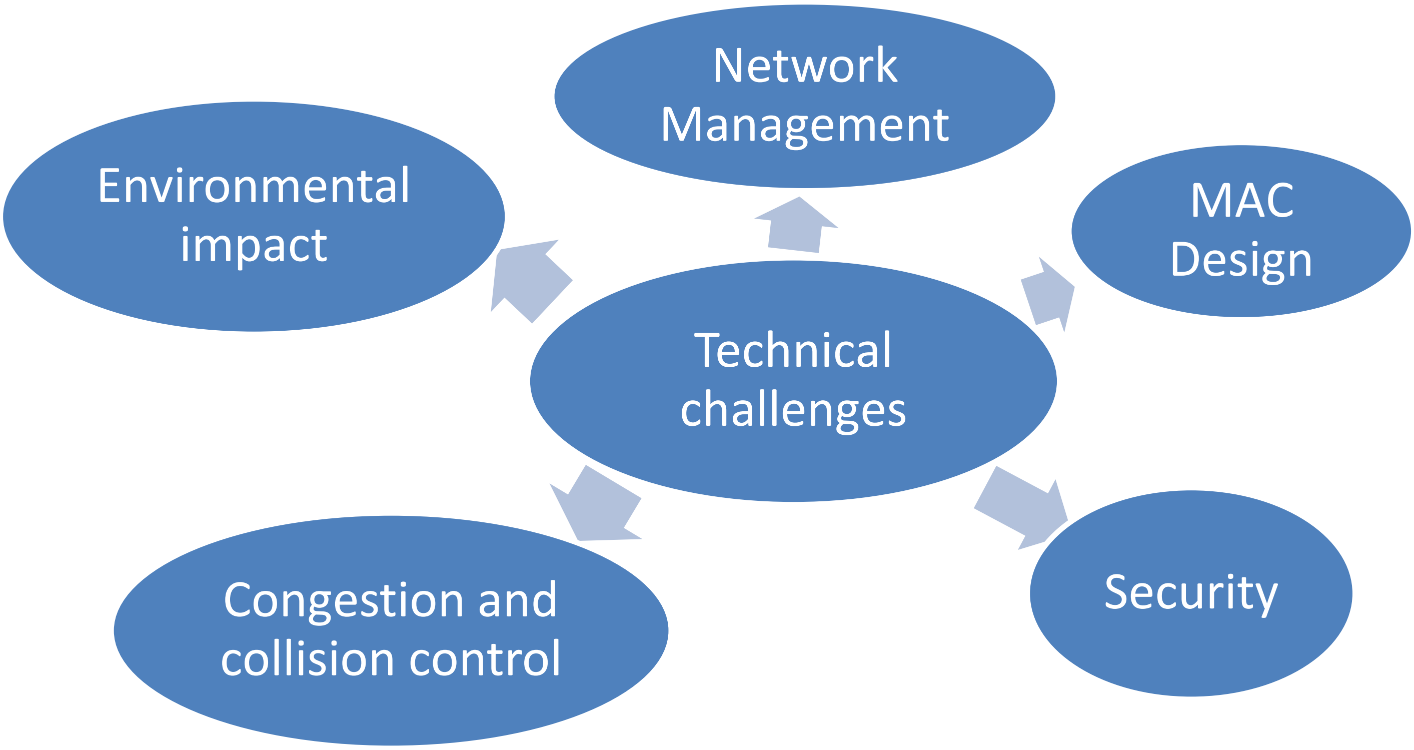


Fig 1.9: Flowchart representing the process for creating a virtual traffic light in an intersection

1.9 CHALLENGING ISSUES IN VANET



1.10 MOTIVATION

The original impetus for the interest in VANET was provided by the need to solve the issue of congestion and its related adverse effects. Traffic management is a serious problem. The need to inform fellow drivers of actual or imminent road conditions, delays, congestion, hazardous driving conditions and other similar concerns is vital. Also due to the heavy traffic, vehicles providing public services like ambulance, fire brigade, etc. are unable to reach the destination on time. Therefore there must be a way so that these vehicles are given priority over normal vehicles. Adaptive traffic light system based on wireless communication between vehicles and fixed controller nodes can significantly improve traffic fluency in intersections. Moreover, the cost of installation of traffic lights at one intersection is in the order of several thousands of dollars. Thus, in this work we want to go a step further than the described state of the art and design communication protocols that enable fully distributed and self-organized traffic signs in a VANET environment. In particular, we propose that the local intersection management can be achieved through the collaboration of the local vehicles creating virtual traffic lights in a synchronized manner, but without a centralized control infrastructure. As an example (see Figure 1.1), imagine a lane obstruction caused by an accident in a road which has two lanes, one in each direction. As one of the lanes is blocked, vehicles approaching the zone of obstruction must act cooperatively to create a just-in-time and just-in-place virtual traffic light to regulate and control conflicts between vehicles in different lanes, improving the traffic efficiency and safety in the only lane available.

1.11 PROBLEM STATEMENT

- To address the issue of traffic congestion at an intersection using traffic light management and the characteristics of VANET.
- To create a distributed, infra-structure less system of virtual traffic lights by electing a coordinating leader using the Bully algorithm.

1.12 ORGANISATION OF THE WORK

Following this chapter, the next chapter presents the basic principles and concepts of Traffic Light Control Systems. In particular, we describe the current specifications, architecture and technologies already proposed by Car-to-Car Communication Consortium (C2C-CC), which are the base reference for a correct understanding of this work. Also, we discuss the working of the Simulation Of Urban Mobility (SUMO) simulator. The main contributions are present in Chapter 3 and Chapter 4. Chapter 3 focuses on the proposed architecture framework and their associated algorithms. The results and observations of the proposed algorithms are discussed in Chapter 4. Different cases have been taken to evaluate the results.

Finally, this work concludes with a brief summary and outlook for future research paths.

LITERATURE SURVEY

2.1 TRAFFIC SIGNAL CONTROL SYSTEMS

Since their inception in 1868 outside the British Houses of Parliament in London, traffic signalling systems have been one of the major components on traffic management strategies. In general, their purpose is to regulate the vehicular flow at the intersection by managing the movements of different vehicles approaching the intersection with reference to time. Precisely, the platoon of vehicles is allowed to cross the intersection in turn, or in phases, thus making it easier and safer for drivers as well as pedestrians. The basic concepts involved in any traffic signal systems are the following:

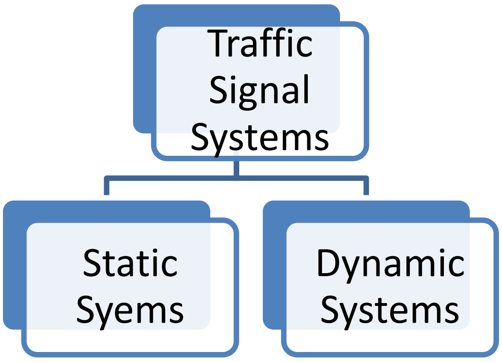
- Cycle: This is a complete sequence of all signal indications at an intersection.
- Phase: This is used to describe any part of a cycle allocated to any specific traffic movement at the intersection. A typical signal cycle is shown below.
- Green Split: This is a division of the cycle allocated to each of the various directions that will allow the passage of the vehicles.
- Red time: This is an interval during which signal indications at an intersection notify a stoppage of vehicles in one direction.

Most of the existing traffic signal systems work with a set of pre-rimed cycles that are generated off-line using the historical data of traffic demand [5]. These systems try to match the best possible timing plan, from the existing database, for the current hour of the day. However, in recent years due to exponential increase of vehicular traffic, the traditional traffic signal control system fails to fulfil the requirements in urban areas for traffic management. Improper traffic signal control systems at the intersections are one of the major contributors for the increase in recurring traffic congestion. Millions of people deal with such traffic on a daily basis. Along with the effects on drivers, environment and health, such traffic congestion has shown its effects on economy as well. Hence it is imperative to improve the efficiency of existing transportation facilities.

To ameliorate the traffic conditions in urban scenarios, it is important to develop dynamic traffic signal control that will adapt itself to varying vehicular traffic demand and optimize its timing accordingly. Such optimizations will result into smoother traffic flows at the intersections which will reduce the travel time for the users. This, in turn, reduces the number of vehicles that stop at the intersections hence cuts the vehicular emissions.

To achieve this, many works have been proposed in the literature. In the following section, a classification and description of the existing signal control methods are discussed.

2.1.1 CLASSFICATION OF TRAFFIC SIGNAL CONTROL SYSTEMS



As discussed earlier, improper traffic signal controllers at the intersections are one of the factors that contribute to the degradation of traffic flow. This drawback demonstrates the need for improvement in the functioning of traffic signal systems. Within this context, the technology of traffic signal system has evolved significantly over the years from the pre-timed systems to complex adaptive systems. The framework of the available solutions in traffic signal systems can broadly be classified into static systems and dynamic systems.

STATIC SYSTEMS

Static systems represent the most basic form of traffic signal control and operate on a predetermined regularly operated sequence of signal indications. The signal rotates through a defined cycle in a constant fashion, as determined by the controller’s settings [4]. For example, in one complete phase of the cycle, each street may be assigned some seconds of green time, several seconds of yellow interval and the remaining time to the red interval. This timing of static signals is typically determined from visual observations and traffic counts. Once the timing programs are set, they remain fixed until they are changed manually. Static controllers are best suited for intersections where traffic volumes are predictable, stable and fairly constant. In general, static controllers are cheaper to purchase, install and maintain. Their repetitive nature facilitates coordination with adjacent signals, and they are useful in areas where progression is desired. However, there is no guarantee that the system will have a plan for the existing traffic conditions. Thus, such systems fail to respond to the varying traffic demand and accommodate highly congested flow or non-recurring traffic conditions caused by accidents, weather conditions or special events.

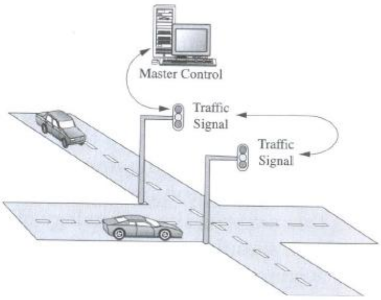


Fig2.1: Static Systems [5]

DYNAMIC SYSTEMS

The inflexibility of fixed systems prompts for the implementation of dynamic signal systems. These systems can accommodate not only recurring traffic congestion but also adequately adjust signal timing for non-recurring traffic congestion caused by random fluctuations in traffic patterns. Such systems are termed as dynamic systems.

These systems optimize the timing plans according to the vehicular demand and help ease the congestion and its negative externalities without the cost and environmental impact of road expansion. In the literature, there are several ways, such as inductive loops or sensor, through which the existing circumstances of the vehicular traffic can be procured. On the basis if the working methodology, the dynamic systems can be further classified into three classes: actuated, responsive and adaptive. Of these, adaptive systems are the most advanced and complex traffic signal systems.

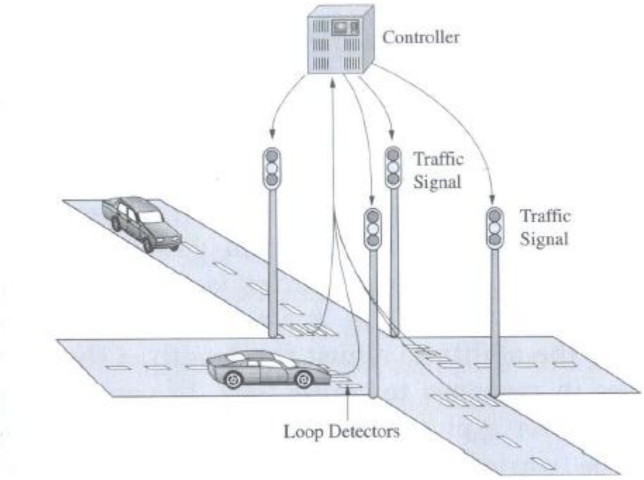


Fig2.2: Dynamic Systems [5]

2.2 VANET settings

1. V2V warning propagation

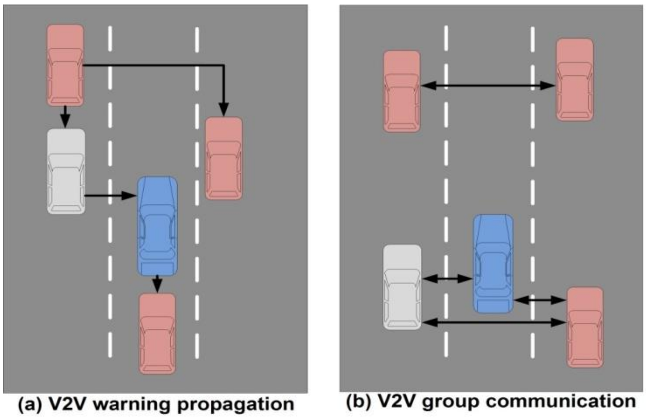
2. V2V group communication

- **Share urgent information**: In case of emergency, such as reporting an accident, the vehicles use V2V communication to propagate the information in their vicinity, up to a certain hop limit (e.g., 10 hops). Therefore, the urgent messages are propagated with the lowest delay in a limited geographical area.
- **Relay other information towards data centres**: Vehicles use V2V communication to relay non-urgent information towards data centres, so that the data centres can store them in the central database.

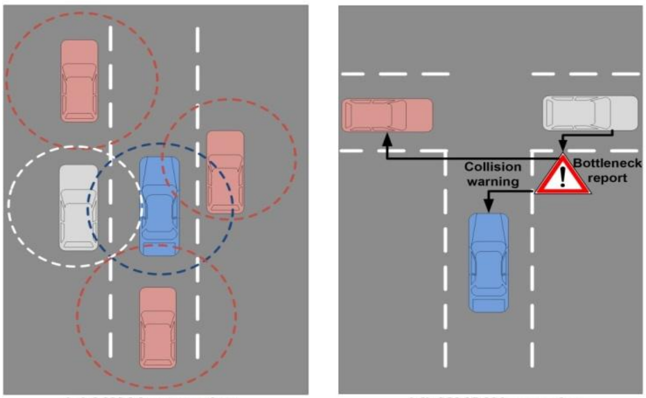
3. V2V beaconing

In Vehicle-to-Vehicle (V2V) communications, an important method of communication is the periodic transmission of short status messages or beacons. They contain such information as speed, position and vehicle state, from which a cooperative awareness can be constructed.

4. I2V/V2I warning



(a) V2V warning propagation (b) V2V group communication



(c) V2V beaconing (d) I2V/V2I warning

Fig 2.3:- Communication via VANET (Source: http://www. http://comp.ist.utl.pt/[3])

2.3 ISSUES IN DATA FORWARDING

The application of inter-vehicular communication over a MANET raises several interesting issues in regard to data forwarding. One such issue is in which an unpredictable traffic has both a limited area of the relevance and a limited time of relevance. For example, carriers should not be forward information about road closures after the road has re-opened. This requires carriers to not only forward received messages, but also to analyse the semantics of the messages to first determine if they must be forwarded. As another example, carriers may be travelling to destinations hundreds of miles away from the traffic slowdown.

VETRAC [6] proposes to introduce relevance filtering at the carriers before forwarding event messages. VETRAC system works on the principle identify of a vehicle, which is assigned as Mobile-IP address to a vehicle. The system tracks the vehicle information through Wi-Fi access points, which are established at various locations in lane or parks or in large campus. The carrier is a navigation server that connects with multiple clients (vehicle) and is also responsible for the client's request data. The client control panel system (provider / consumer) running on client side helps the user for identifying current location, destination client location, and landmark, the distance to be travelled. Traffic intensity of each
Lane at an instant is notified though carrier access points.

A. Functionality of VETRAC

The architectural and functionality overview of Vehicle Tracking System (VETRAC), which has complex modules classified based on functionality. The core functionality of VETRAC focuses on vehicle tracking and controlling the lane (route) selection process tracking system. System establishes connection with the client, updates the client location in the database, gathers the traffic information and controls the streaming traffic video during conference process. The system has various modules
a. Connection establish server
b. Location Management Server and
c. Traffic server

2) Vehicle Tracking System:

System gets connected with Wi-Fi access point and tracks the client within the access point range. The system detects the clients in its range and automatically passes the client, current location to the navigation server.

3) Client Control Panel:

The user should know about current location, destination location, landmark of the said campus, traffic information in a campus, traffic video information about the junction in campus. The registered users can only have access to this navigation system.

The system has various modules.

i. Establish connection
ii. New client registration
iii. Client current location
iv. Destination client's location
v. Traffic information
vi. Traffic video information

4) Display Map System:

The system which runs in the client device and it indicates the client's current location, destination client location, landmark identification and traffic information by the map.

.

VETRAC is currently implemented for locations that to be identified within range of 500 meters to 800 meters. Any user can easily access to the system as well VETRAC system can be implemented in any vehicles.

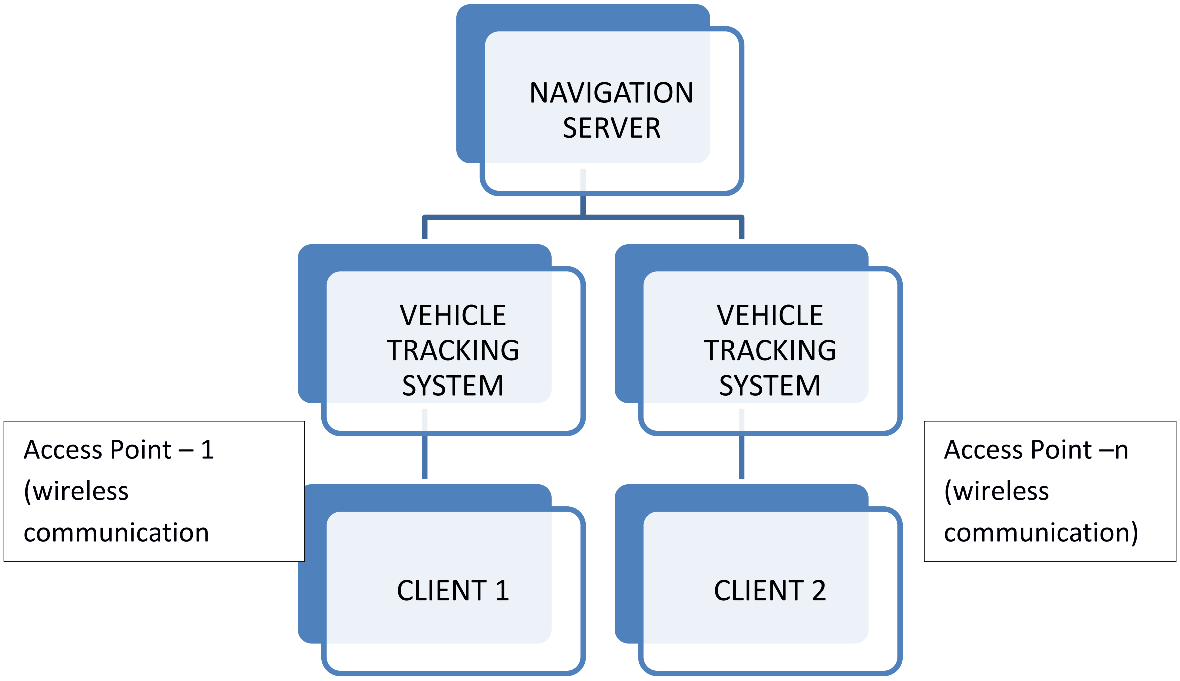


FIG 2.4: Mobile Vehicle Tracking System [6]

2.4 Simulation of Urban Mobility (SUMO)

INTRODUCTION

Simulator: - Device or system that simulates specific conditions or the characteristics of a real process or machine for the purposes of research or operator training.

"**S**imulation of **U**rban **Mo**bility", (SUMO) [7] is an open source, microscopic, multi-model traffic simulation. It allows us to simulate how a given traffic demands or network which consists of single vehicle moves through a given road map. This simulator allows addressing a large set number of traffic management topics. It is purely microscopic in nature: each vehicle is modelled explicitly, has its own route, and moves individually through the given network.

To build a network in SUMO a street network consists of nodes i.e junctions and edges (streets connecting the given junctions) are necessary [8].

A **SUMO** network file describes the traffic-related part of a map. It mainly contains the network of roads/ways, intersections/junctions, and traffic lights in a map

Network Format

 At a coarse scale, a SUMO network is a directed graph. "Nodes" represent intersections/junctions, and "edges" represent roads/streets.

A SUMO network contains further traffic related information:

- every street (edge) as a collection of lanes,
- the position, shape and speed limit of every lane,
- the right of way regulation,
- the connections between lanes at junctions (nodes), and
- The position and logic of the traffic lights.

Files used in SUMO to make a Road Map:-

- Node file(.nod.xml)
- Edge file(.edg.xml
- Route file(.rou.xml)
- Network file (.net.xml)
- Configuration file(.sumo.cfg.xml)

Nodes

All nodes have specific location (x- and y-coordinate, that describe distance to the origin in meters) and also an id for future references. Thus our simple node file looks as follows

```
<nodes>
<node id="0" x="-500.0" y="0.0" type="traffic light" />
<node id="1" x="500.0" y="0.0" type="priority" />
<node id="2" x="0.0" y="-500.0" type="priority" />
<node id="3" x="0.0" y="500.0" type="priority" />
</nodes>
```

File can be edited with a text editor and saved as name.nod.xml where .nod.xml is a default suffix for Sumo node files.

Edges

Now, we have to connect the nodes with the edges. We have a target node id, a source node id, and an edge id for future reference. Edges are directed, thus every vehicle that is travelling this edge will start at the node given in *from* and end at the node given in *to*.

```
 <edges>
<edge id="1m1" from="0" to="1" priority="2" speed="11.11" />
<edge id="m10" from="1" to="0" priority="3" speed="13.89" />
<edge id="01" from="2" to="3" priority="1" speed="11.11" />
<edge id="2m2" from="3" to="2" priority="2" speed="11.11" />
```

```
</edges>
```

Save this data into a file called name.edg.xml. Now that we have nodes and edges we can call the first SUMO tool to create a network. Make sure that the net convert command is somewhere in the path and call:

```
net-convert –node-files=name.nod.xml –edge-files=name.edg.xml –output-file=name.net.xml
```

This will generate our network file that is called name.net.xml.

Routes

Now that we have a network file, we still need a car. In **SUMO** the vehicles have different types defining their basic properties such as the length, acceleration and the deceleration, and given maximum speed. Furthermore it needs a so called sigma parameter, which introduces some random behaviour. Setting it to 0 implies a deterministic car.

Now we define a route for the car which simply consists of the two edges that are defined. The reason why we need two edges is that in **SUMO** the car disappears as soon as it will reach the last edge of its own route (the position of a car is defined by the position of its front).Last but not the least we define the single car mainly referring to the entries giving it a departure time as the type name.rou.xml file.

```
<routes>
<vType accel="1.0" decel="5.0" id="C1" length="2.0" maxspeed="100.0" sigma='"0.0"'' />
<route id="route01" edges="1to2 out"/>
<vehicle depart="1" id="veh0" route="route0" type="Car" />
</routes>
```

Configuration

Now we attach everything together into the configuration file (.cfg.xml)

```
<configuration>
<input>
<net-file value="name.net.xml"/>
<route-files value="name.rou.xml"/>
</input>
```

```
<time>
<begin value="0"/>
<end value="10000"/>
</time>
</configuration>
```

Saving this to name.sumo.cfg we can start the simulation by either:

```
sumo -c name.sumocfg
```

or with GUI by:

```
sumo-gui -c name.sumocfg
```

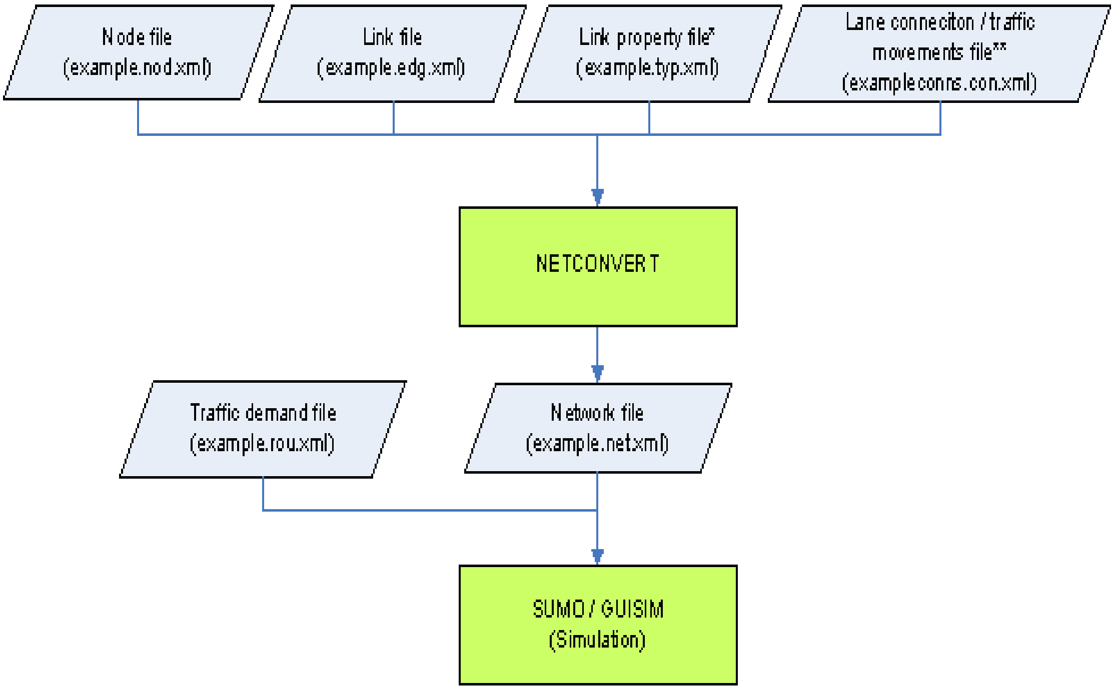
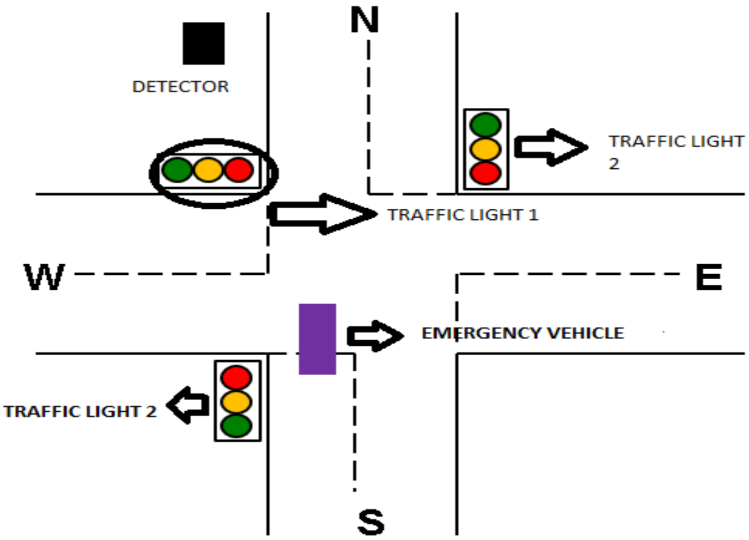


Fig 2.5:- Flow chart of creation of a VANET

PROPSED WORK



The above diagram depicts a typical intersection. We have placed a detector on the road (show in black) to detect the passage of emergency vehicles. For ease of implementation, emergency vehicles only travel in the North-South direction.

3.1 Priority Based Signalling

- Whenever an emergency vehicle like Fire Brigade, Ambulance or Police on pursuit is travelling, a detector placed on the road will transmit the RF signals to RF receiver fitted with the traffic poles, they will automatically turn green and rest of the signal stay red. After passing of that vehicle, all the functionality of the traffic signal will be normal as per specified. We will simulate this scenario at a junction where the emergency vehicle will be coming from one side only.

FIG3.1 :WORK FLOW DIAGRAM

PSEUDOCODE FOR DETECTOR

```
for( ; ;)
{
    receive signal_from_emergveh()   // the detector receives the signal that a emergency vehicle is arriving
   {
     send_sig_to_trffic_light1();    // the detector sends the signal for the smooth passage of emerg.vehicle
      send_sig_to_trffic_light2(); // the detector sends the signal to stop all other vehicles on the other junction
    }
}
```

PSEUDOCODE FOR TRAFFIC LIGHT SYSTEM

(This is the junction through which the emergency vehicle is passing; we assume that emergency vehicle passes only through this junction)

```
for( ; ;)
{
Boolean flag=TRUE, detector=FALSE;
if detector==FALSE
{
if(flag==TRUE)
{
 Change_trafficlight1 ← Green
 Change_trafficlight2 ← Red
 Wait (for quantum time)
flag= FALSE
 } a;
if (flag==FALSE)
{
Change_trafficlight1 → Red
Change_trafficlight2 → Green
Wait (for quantum time or detector=TRUE)
flag= TRUE
}
}
If detector==TRUE          // on passage of emergency vehicle, detector will be TRUE
{
goto(a)
}
```

3.2 Infrastructure less System

As vehicles under a green light only take small time to cross intersections, choosing one of these vehicles to act as a Leader led us to additional issues. For instance, hand-over techniques are required when the current leader leaves the intersection region and a new leader needs to be selected. However, once a vehicle is stopped under a red light, it can act as a leader and control the VTL until it receives a green light and leave the intersection. Only whenever the Leader receives the green light, a new Leader election is required and must elect another stopped vehicle to be the new Leader. Thus, assuming that road leaders are the closest vehicles to the intersection for each entry road, the Leader must be one vehicle elected between these road leaders. Several factors can be taken into account when electing a vehicle as Leader, which might be the distance to the intersection, the time a vehicle, may take to cross the intersection, or the level of congestion that is behind each road leader. Because connectivity is unpredictable in vehicular networks, the leader election process should not rely on a long dialogue between road leaders. Furthermore, this process should never led vehicles to ambiguities, i.e. it must guarantee that only one vehicle can be the Leader and if all vehicles follow the protocol correctly, intersection safety should result. We have implemented Bully Algorithm for leader election.

3.2.1 Bully Algorithm

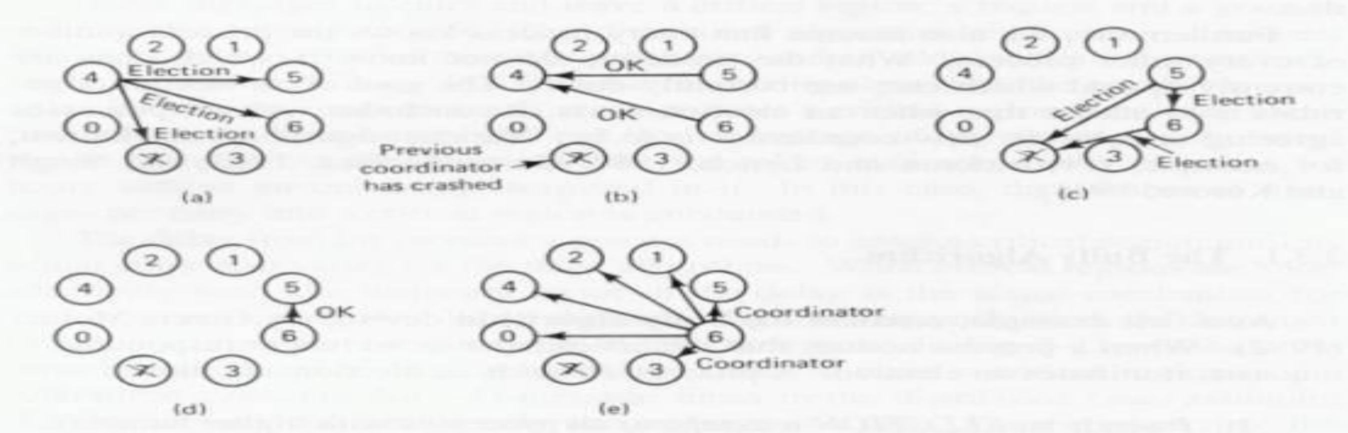


FIG 3.2 Bully Algorithm

Here fig 3.2 shows that current leader 7 has crashed and node 4 detected the crash, and it started an election by sending election messages to nodes having process id higher than itself ie nodes 5, 6 and 7. Nodes 5 and 6 reply to this election message of node 4 by sending an OK message. After that, nodes 5 and 6 themselves start a new election by sending election messages as described above. Finally node 6 does not get any Ok message when it sends election message to nodes above it as there exists no node with process id higher than 6. Thus node 6 elects itself as the leader and sends coordinator messages to all the nodes in the network. Now if node 7 recovers and enters the network again, it will send Query message to nodes having process id higher than itself. Now again, since 7 is the highest process id, node 7 gets no Answer message back. Thus is elects itself as the leader, thereby bullying current leader node 6. This is how original Bully works.

Background: any process P_i sends a message to the current coordinator; if no response in T time units, P_i tries to elect itself as leader. Details follow:

Algorithm for process P_i that detected the lack of coordinator

1. *Process P_i sends an "Election" message to every process with higher priority.*
2. *If no other process responds, process P_i starts the coordinator code running and sends a message to all processes with lower priorities saying "Elected P_i"*
3. *Else, P_i waits for T' time units to hear from the new coordinator, and if there is no response → start from step (1) again.*

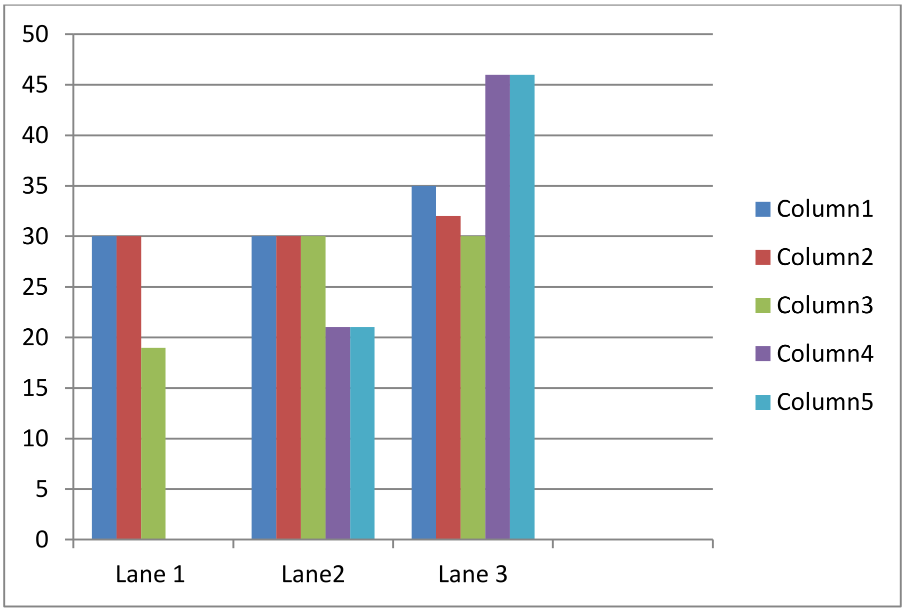
Algorithm for other processes (also called P_i)

```
    If $P_i$ is not the coordinator then $P_i$ may receive either of these messages from $P_j$

    if Pi sends "Elected $P_j$"; [this message is only received if $i < j$]

        Pi updates its records to say that $P_j$ is the coordinator.
    Else if $P_j$ sends "election" message $(i > j)$
        Pi sends a response to $P_j$ saying it is alive
        Pi starts an election.
```

In a traffic management scenario the waiting time of every vehicle standing on the intersection will be calculated. From that data the average waiting time of every lane will be evaluated. The lane having the highest average waiting time will be elected as the leader, using the Bully algorithm. This will allow passage for the leader lane and henceforth a new leader will be elected. Consider a sparse distribution of 20 vehicles. We have modelled the traffic movement of these 20 vehicles in SUMO and calculated the waiting time (in milliseconds) for them at the intersection. Following graph depicts this information.



Hence average waiting time for the vehicles at each lane is: Lane 1 (13.16), Lane 2 (18.85) , Lane 3(27).

Hence Lane 3 will be the leader.

SIMULATION AND RESULTS

4.1 Creation of the Road Network

➢ To create a road map

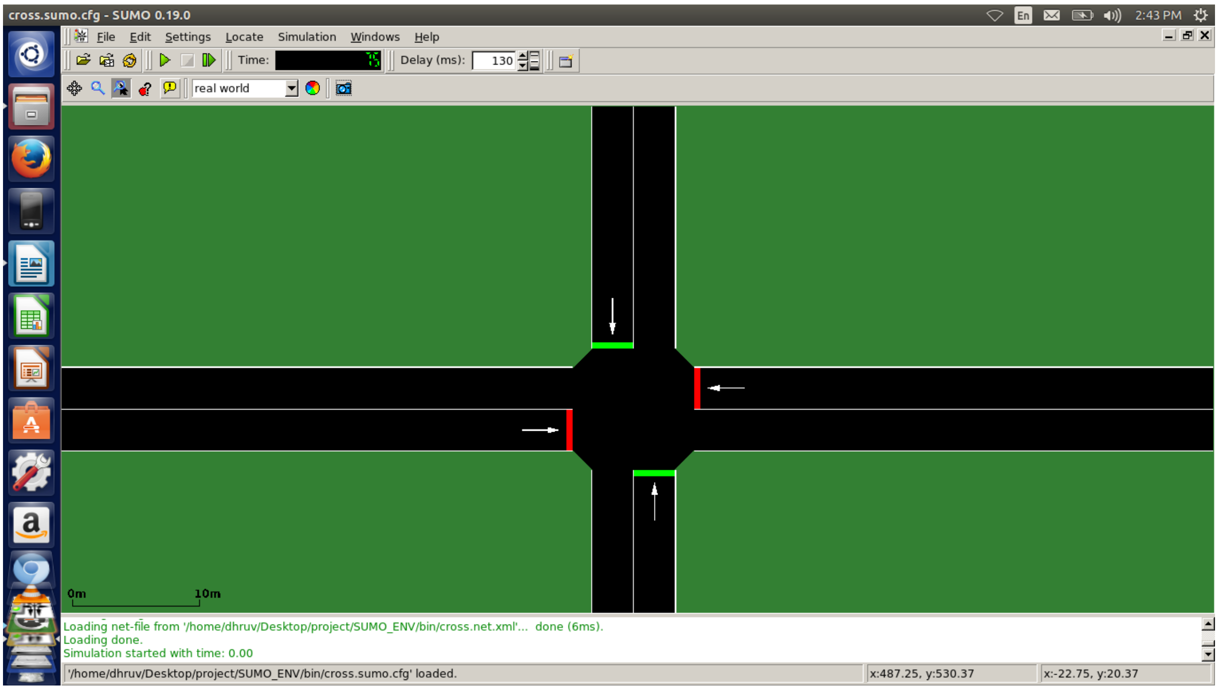
- Node (.nod.xml) and Edge file (.edg.xml) is to be created for road map.
- Transformed into a Network file(net.xml) with the help of NETCONVERT

Command of NETCONVERT **:**

```
netconvert --node-files=MyNodes.nod.xml –edge files=MyEdges.edg.xml \ --output-file=MySUMONet.net.xml
```

- Now, Create **Route** file (rou.xml) to define the path followed by the vehicles to reach from one place to the destination.
- At last network file and route files are combined to make the **configuration** file.

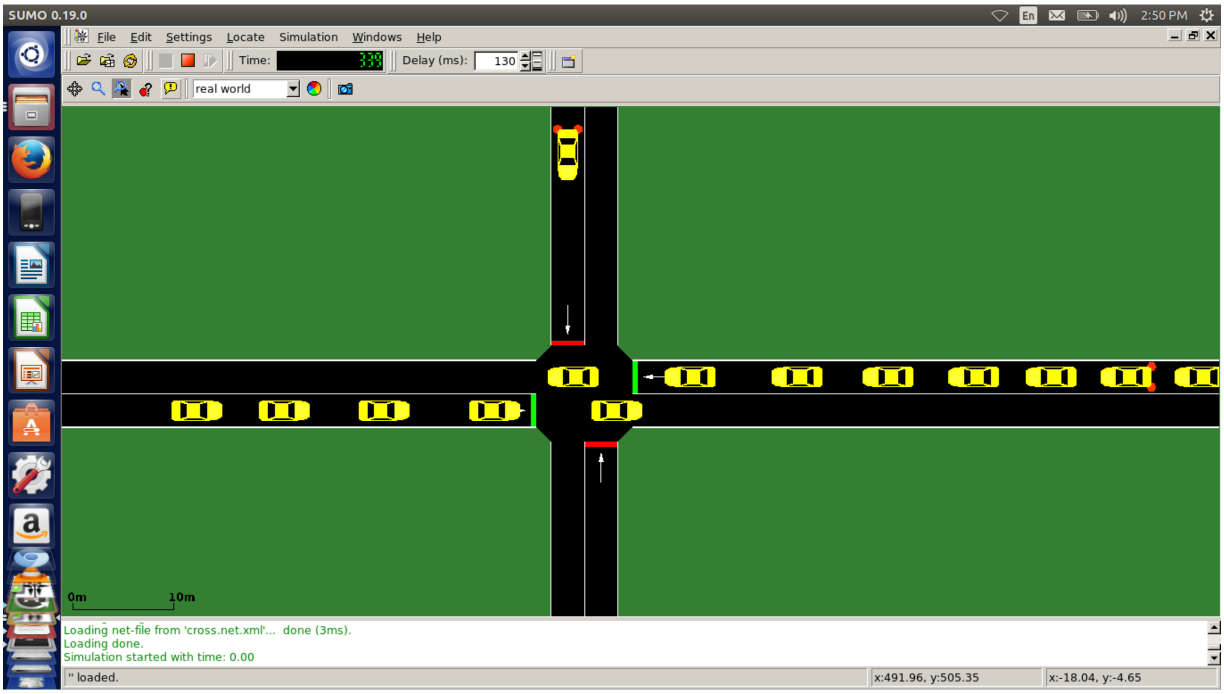
This is how the road map will look like:



```
Loading net-file from '/home/dhruv/Desktop/project/SUMO_ENV/bin/cross.net.xml'...  done (6ms).
Loading done.
Simulation started with time: 0.00
'/home/dhruv/Desktop/project/SUMO_ENV/bin/cross.sumo.cfg' loaded.    x:487.25, y:530.37    x:-22.75, y:20.37
```

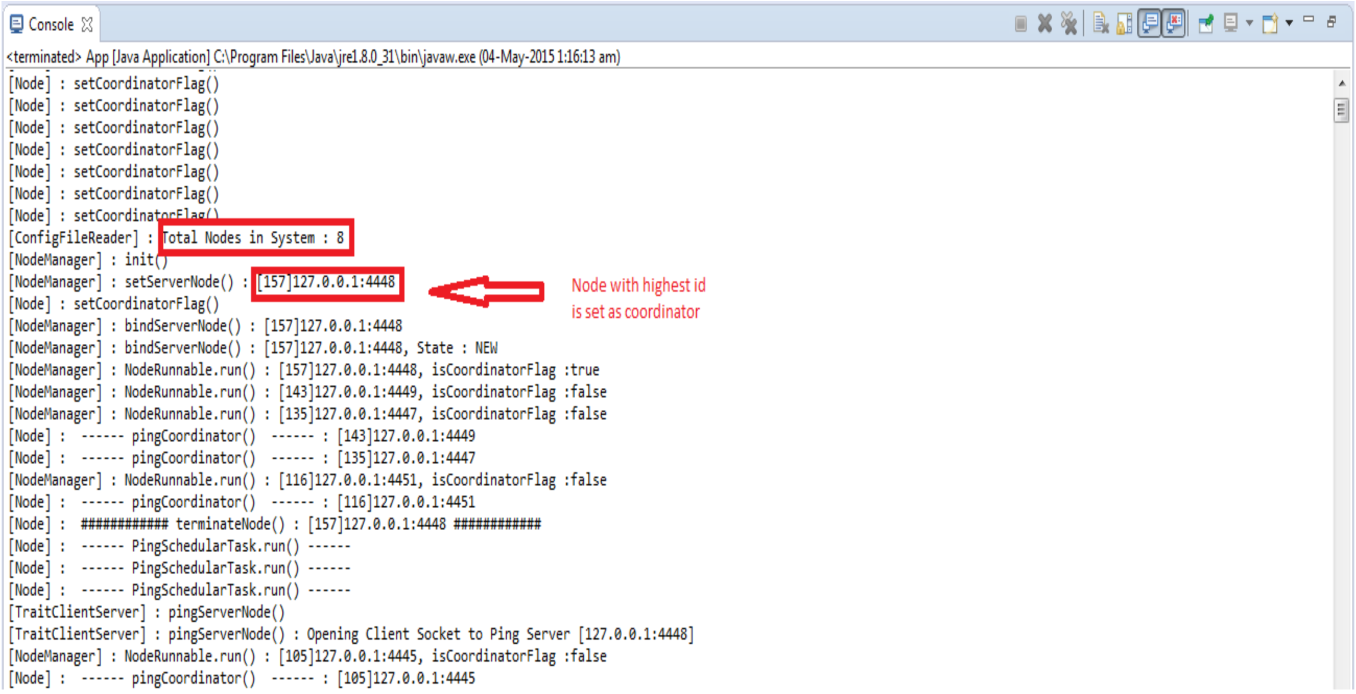
4.2. Traffic movement in a network

- There are two types of vehicles: normal vehicle and an emergency vehicle.
- The vehicles travel in NORTH-SOUTH direction, WEST-EAST direction and EAST-WEST direction.
- Vehicles halt on RED light and move on GREEN light.



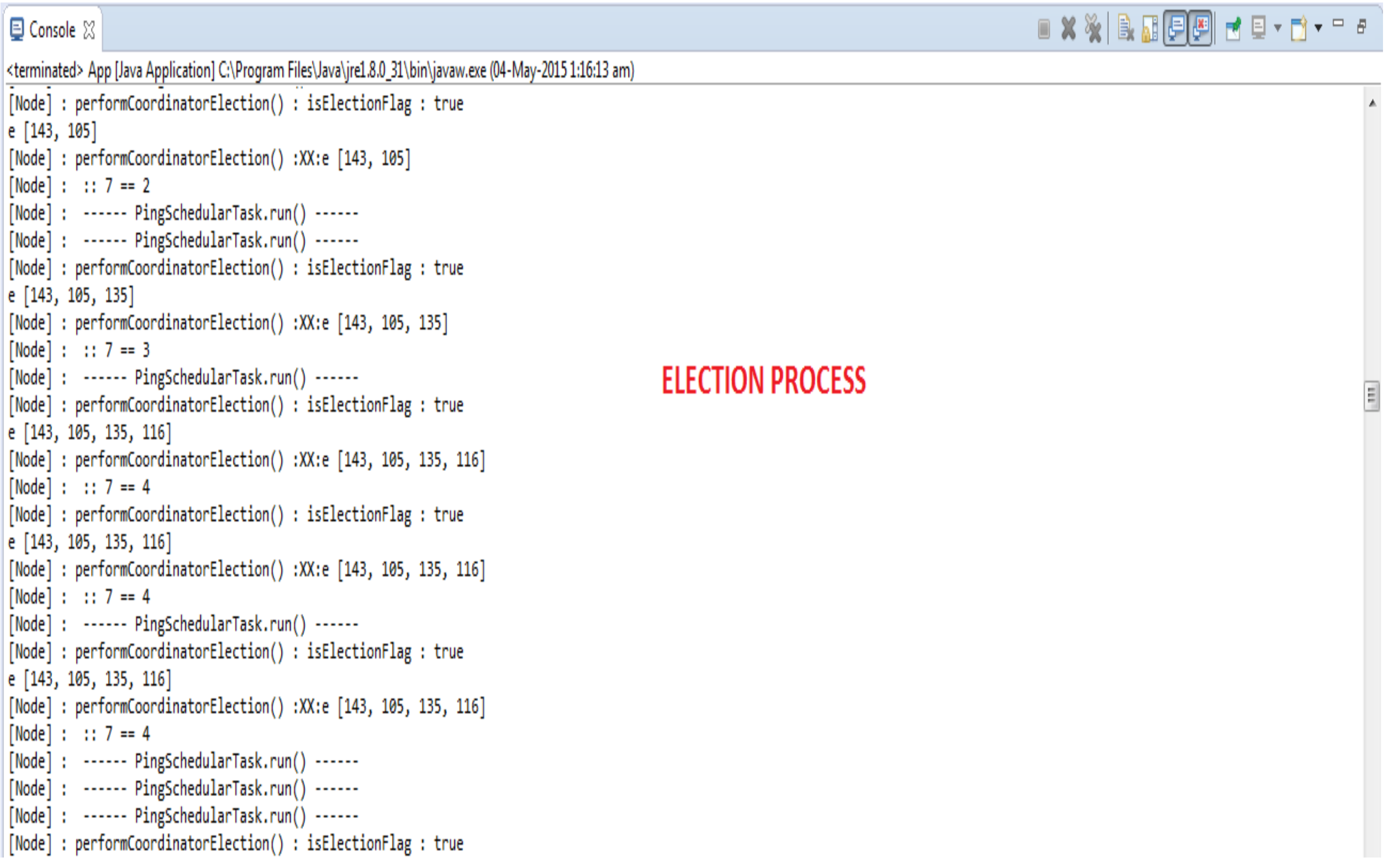
```
Loading net-file from 'cross.net.xml'...  done (3ms).
Loading done.
Simulation started with time: 0.00
" loaded.
```

4.3Bully Algorithm



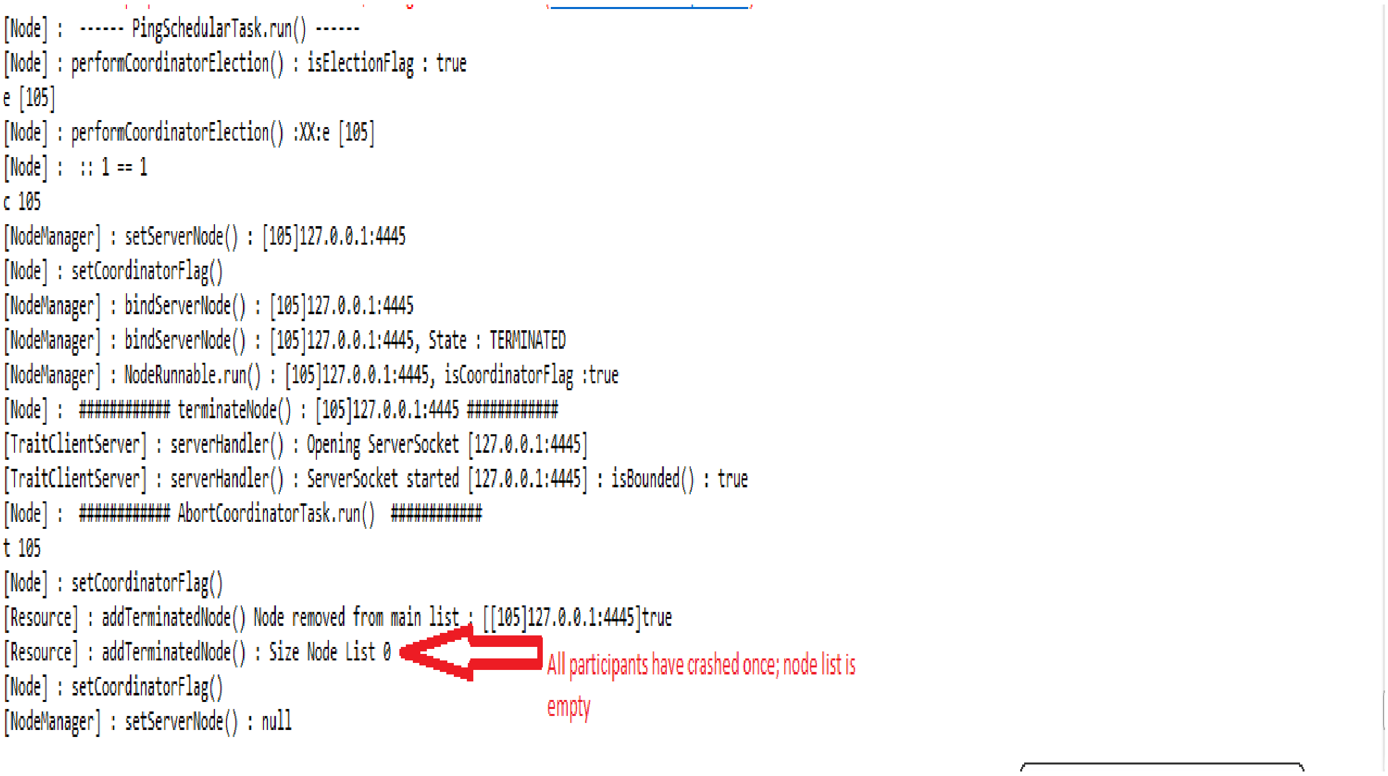
```
<terminated> App [Java Application] C:\Program Files\Java\jre1.8.0_31\bin\javaw.exe (04-May-2015 1:16:13 am)
[Node] : setCoordinatorFlag()
[Node] : setCoordinatorFlag()
[Node] : setCoordinatorFlag()
[Node] : setCoordinatorFlag()
[Node] : setCoordinatorFlag()
[Node] : setCoordinatorFlag()
[Node] : setCoordinatorFlag()
[ConfigFileReader] : Total Nodes in System : 8
[NodeManager] : init()
[NodeManager] : setServerNode() : [157]127.0.0.1:4448
[Node] : setCoordinatorFlag()
[NodeManager] : bindServerNode() : [157]127.0.0.1:4448
[NodeManager] : bindServerNode() : [157]127.0.0.1:4448, State : NEW
[NodeManager] : NodeRunnable.run() : [157]127.0.0.1:4448, isCoordinatorFlag :true
[NodeManager] : NodeRunnable.run() : [143]127.0.0.1:4449, isCoordinatorFlag :false
[NodeManager] : NodeRunnable.run() : [135]127.0.0.1:4447, isCoordinatorFlag :false
[Node] :  ------ pingCoordinator()  ------ : [143]127.0.0.1:4449
[Node] :  ------ pingCoordinator()  ------ : [135]127.0.0.1:4447
[NodeManager] : NodeRunnable.run() : [116]127.0.0.1:4451, isCoordinatorFlag :false
[Node] :  ------ pingCoordinator()  ------ : [116]127.0.0.1:4451
[Node] :  ########### terminateNode() : [157]127.0.0.1:4448 ###########
[Node] :  ------ PingSchedularTask.run() ------
[Node] :  ------ PingSchedularTask.run() ------
[Node] :  ------ PingSchedularTask.run() ------
[TraitClientServer] : pingServerNode()
[TraitClientServer] : pingServerNode() : Opening Client Socket to Ping Server [127.0.0.1:4448]
[NodeManager] : NodeRunnable.run() : [105]127.0.0.1:4445, isCoordinatorFlag :false
[Node] :  ------ pingCoordinator()  ------ : [105]127.0.0.1:4445
```

Node with highest id is set as coordinator



```
<terminated> App [Java Application] C:\Program Files\Java\jre1.8.0_31\bin\javaw.exe (04-May-2015 1:16:13 am)
[Node] : performCoordinatorElection() : isElectionFlag : true
e [143, 105]
[Node] : performCoordinatorElection() :XX:e [143, 105]
[Node] :  :: 7 == 2
[Node] :  ------ PingSchedularTask.run() ------
[Node] :  ------ PingSchedularTask.run() ------
[Node] : performCoordinatorElection() : isElectionFlag : true
e [143, 105, 135]
[Node] : performCoordinatorElection() :XX:e [143, 105, 135]
[Node] :  :: 7 == 3
[Node] :  ------ PingSchedularTask.run() ------
[Node] : performCoordinatorElection() : isElectionFlag : true
e [143, 105, 135, 116]
[Node] : performCoordinatorElection() :XX:e [143, 105, 135, 116]
[Node] :  :: 7 == 4
[Node] : performCoordinatorElection() : isElectionFlag : true
e [143, 105, 135, 116]
[Node] : performCoordinatorElection() :XX:e [143, 105, 135, 116]
[Node] :  :: 7 == 4
[Node] :  ------ PingSchedularTask.run() ------
[Node] : performCoordinatorElection() : isElectionFlag : true
e [143, 105, 135, 116]
[Node] : performCoordinatorElection() :XX:e [143, 105, 135, 116]
[Node] :  :: 7 == 4
[Node] :  ------ PingSchedularTask.run() ------
[Node] :  ------ PingSchedularTask.run() ------
[Node] :  ------ PingSchedularTask.run() ------
[Node] : performCoordinatorElection() : isElectionFlag : true
```

ELECTION PROCESS

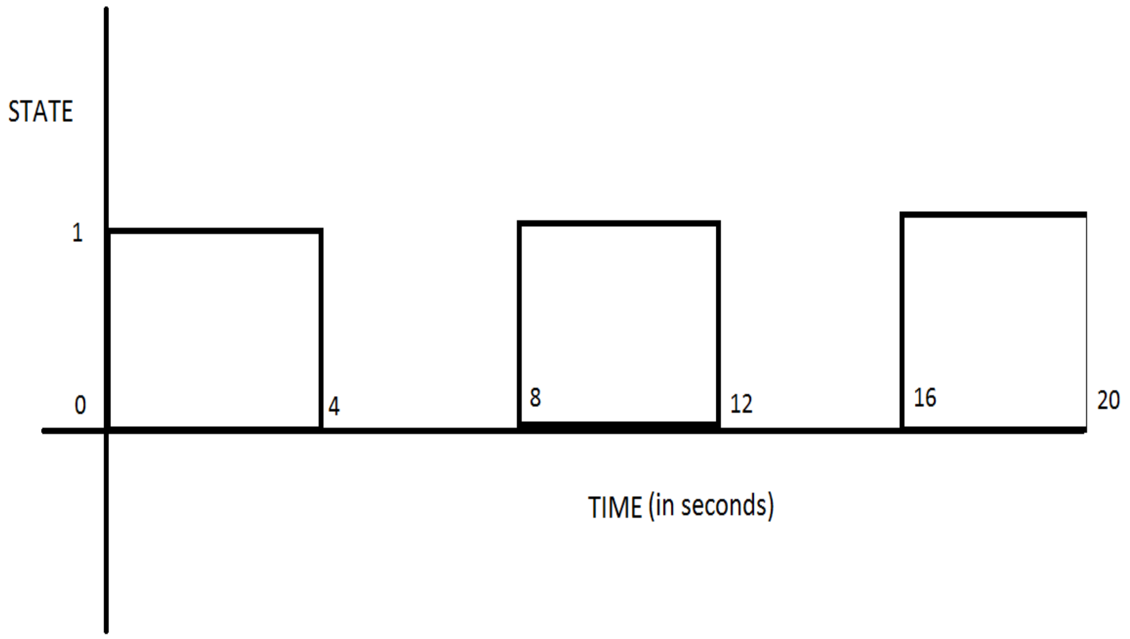


```
[Node] :  ------ PingSchedularTask.run() ------
[Node] : performCoordinatorElection() : isElectionFlag : true
e [105]
[Node] : performCoordinatorElection() :XX:e [105]
[Node] :  :: 1 == 1
c 105
[NodeManager] : setServerNode() : [105]127.0.0.1:4445
[Node] : setCoordinatorFlag()
[NodeManager] : bindServerNode() : [105]127.0.0.1:4445
[NodeManager] : bindServerNode() : [105]127.0.0.1:4445, State : TERMINATED
[NodeManager] : NodeRunnable.run() : [105]127.0.0.1:4445, isCoordinatorFlag :true
[Node] :  ########### terminateNode() : [105]127.0.0.1:4445 ###########
[TraitClientServer] : serverHandler() : Opening ServerSocket [127.0.0.1:4445]
[TraitClientServer] : serverHandler() : ServerSocket started [127.0.0.1:4445] : isBounded() : true
[Node] :  ########### AbortCoordinatorTask.run()  ###########
t 105
[Node] : setCoordinatorFlag()
[Resource] : addTerminatedNode() Node removed from main list : [[105]127.0.0.1:4445]true
[Resource] : addTerminatedNode() : Size Node List 0
[Node] : setCoordinatorFlag()
[NodeManager] : setServerNode() : null
```

All participants have crashed once; node list is empty

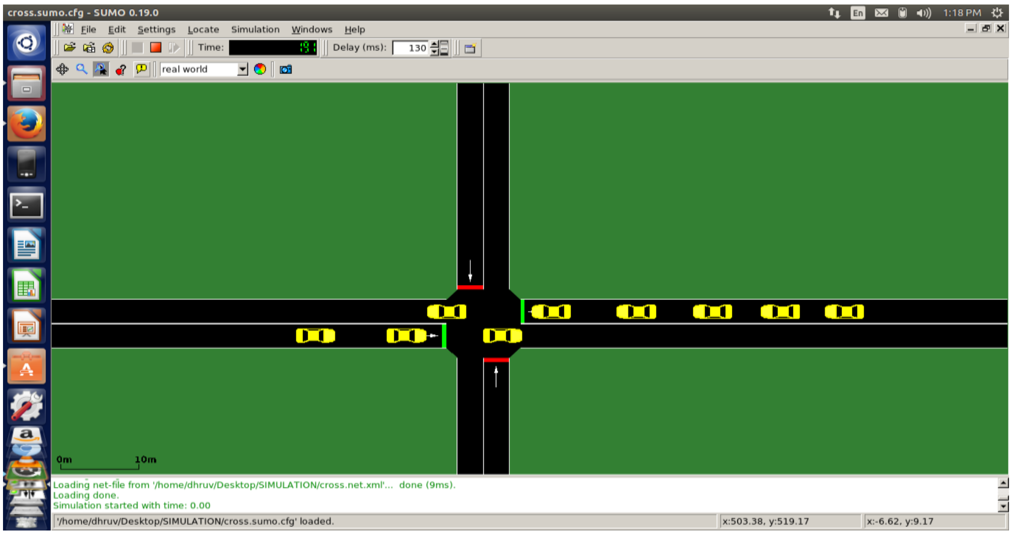
4.4 RESULTS

a) Static System: When no detector is placed on the road, it corresponds to a system of static traffic lights. The following graph (timing diagram) shows the cycle of traffic lights at the emergency intersection. State 0 corresponds to RED and State 1 corresponds to GREEN.

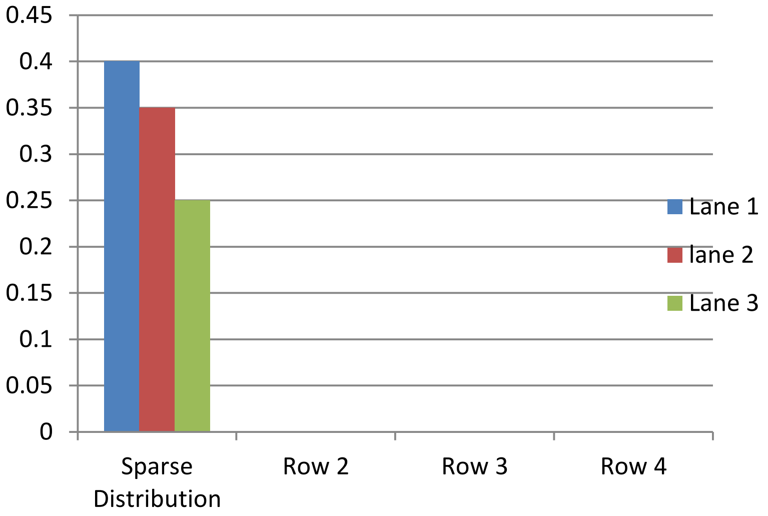


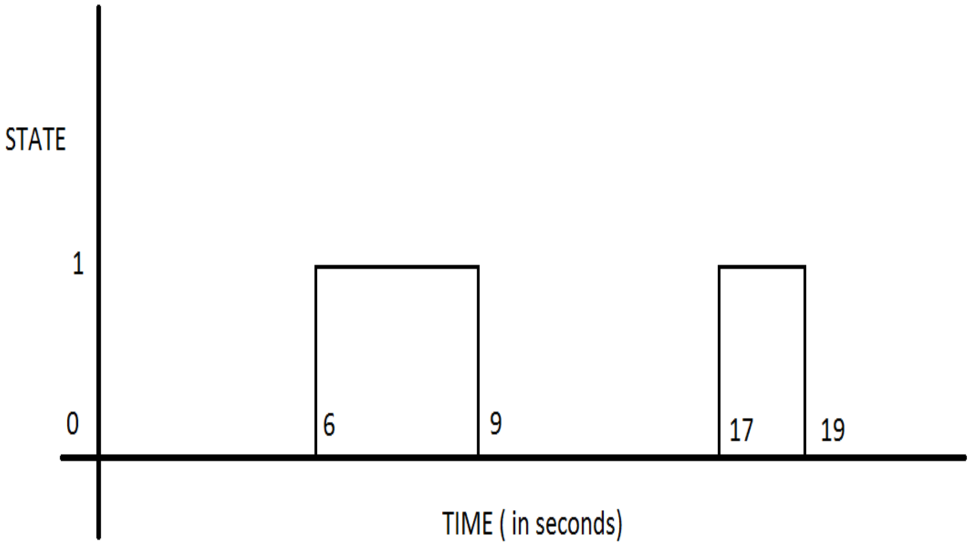
b) Dynamic System: When there is a detector placed on the road, the traffic light only changes when it detects the passage of an emergency vehicle. We have assumed that only emergency vehicle passes in the North-South direction. Here we have considered two cases: Sparse traffic distribution and Dense traffic distribution.

(i) Sparse traffic distribution: The following graphs show the distribution of traffic for 25 vehicles at the intersection and the cycle of traffic light at the emergency intersection. State 0 corresponds to RED and State 1 corresponds to GREEN.

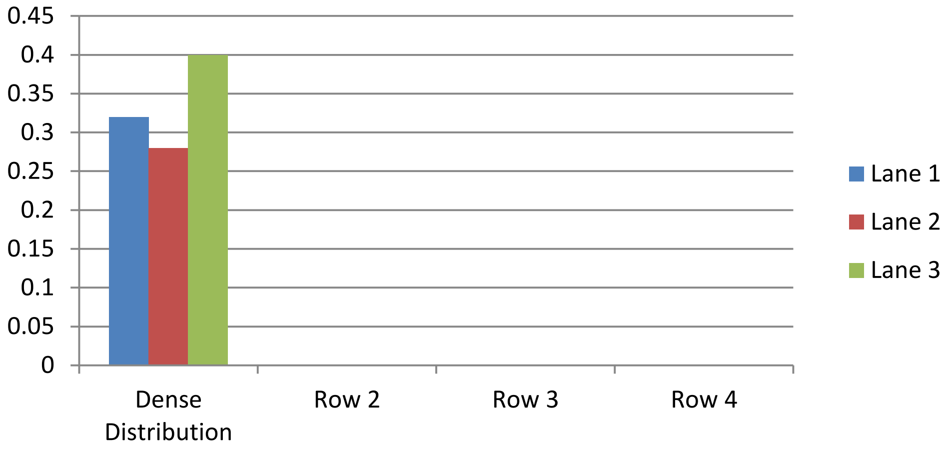


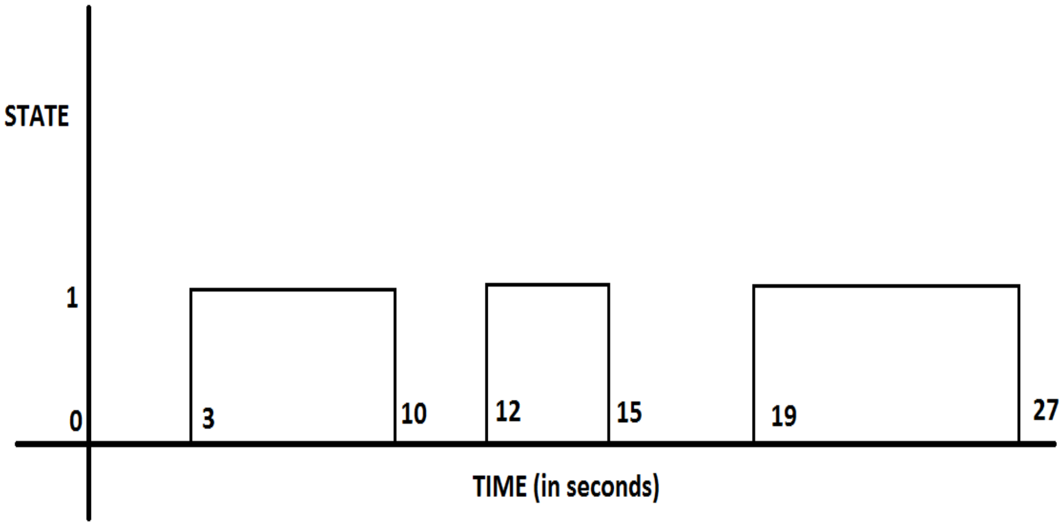
```
Loading net-file from '/home/dhruv/Desktop/SIMULATION/cross.net.xml'...  done (9ms).
Loading done.
Simulation started with time: 0.00
'/home/dhruv/Desktop/SIMULATION/cross.sumo.cfg' loaded.
x:503.38, y:519.17
x:-6.62, y:9.17
```

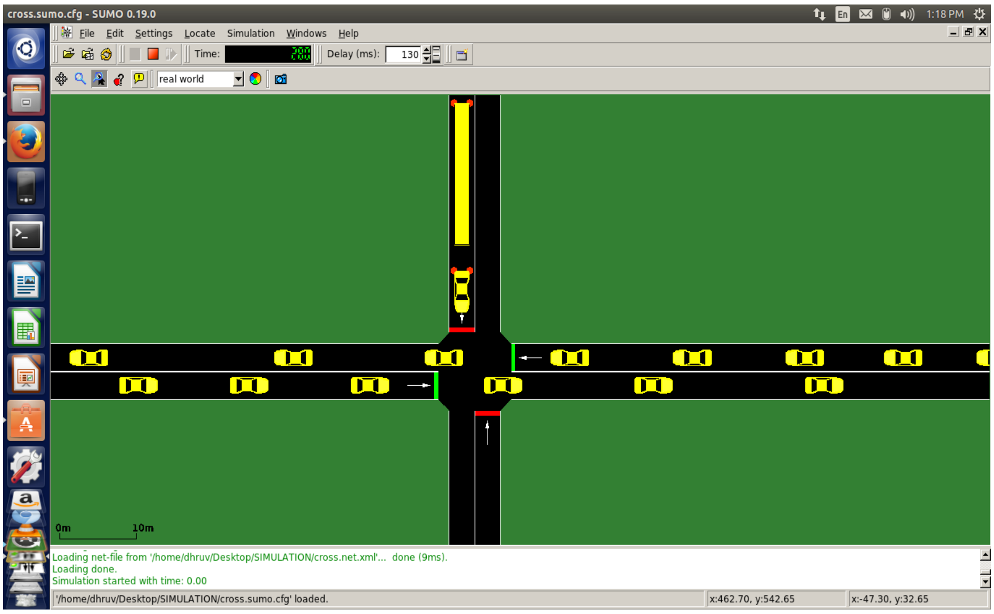




(ii) Dense traffic distribution: The following graphs show the distribution of traffic for 250 vehicles and the cycle of traffic lights at the emergency intersection. More number of emergency vehicles pass through the intersection, changing the cycle of traffic lights accordingly.







```
Loading net-file from '/home/dhruv/Desktop/SIMULATION/cross.net.xml'...  done (9ms).
Loading done.
Simulation started with time: 0.00
'/home/dhruv/Desktop/SIMULATION/cross.sumo.cfg' loaded.    x:462.70, y:542.65    x:-47.30, y:32.65
```

CONCLUSION

In our work, we first gave a description of architecture, standards & protocols of vehicular ad hoc networks. We then proposed a scheme to show how a VANET can be used to aid in traffic signal control. We have implemented an adaptive traffic signal control algorithm that reduces the delays experienced by the emergency vehicles as they pass through the intersection. We have tested this algorithm in the simulator SUMO. This algorithm clearly benefits the emergency vehicles without major concern.

SUMO allows us to generate various outputs for each simulation run. These range from simulated induction loops to single vehicle positions written in each time steps for all vehicles and up to complex values as information about each vehicle's trip or aggregated measures along a street or lane.

With the introduction of intelligent vehicles, we argue that road transportation and traffic management are also demanding for intelligent traffic signs. Based on the emergent vehicular communication technologies, we have described a system that enables virtual traffic signs to increase road safety by fast adaptation to current road and traffic situations. Like real road signs, they are related to a certain geographical region and when a vehicle enters that region, the existing virtual signs are displayed for the driver inside the vehicle. This traffic signs might be stored in an in-vehicle database and might be generated by a vehicle or a RSU as well as they might be created in a collaborative manner by a group of vehicles in an ad-hoc fashion. Virtual traffic lights can be used not only at intersections, but also in a variety of environments where the crossing conflicts must be solved. However, any crossing conflict can be seen like an intersection.

Although the works are numerous, there are still issues which may be untouched. In the future, we want to reduce the dependency, for the fast movement of emergency vehicles, on traffic lights. We will develop a distributed system in which vehicles will communicate with each other and will be diligent enough to give way to emergency vehicles.

REFERENCES

[1]. www. vehicularcommunication.blogspot.com

[2]. Kumar, Vishal, Shailendra Mishra, and Narottam Chand. "Applications of VANETs: Present & Future." *Communications and Network* 5.01 (2013): 12.

[3]. www. http://comp.ist.utl.pt

[4]. N. Chaudhary, V. Kovvali, S. Alam, Research, T. I. Office, Texas, T. T. Institute, and D. of Transportation, Guidelines for Selecting Signal Timing Software. Texas Transportation Institute, Texas A & M University System; Available through the National Technical Information Service, 2002.

[5]. André-Luc Beylot and Houda Labiod, *Vehicular Networks: Models and Algorithms, 2013*

[6]. "Location Identification and Vehicle Tracking using VANET (VETRAC)" given by Arunkumar Thangavelu, K. Bhuvaneswari, K. Kumar, K. Senthil Kumar and S.N. Sivanandam, IEEE - ICSCN 2007, MIT Campus, Anna University, Chennai, India. Feb. 22-24, 2007. pp. 112-116.

[7]. "A framework to simulate VANET scenarios with SUMO" given by Florent Kaisser, Christophe Gransart, Mohamed Kassab and Marion Berbineau *of Université Lille Nord de France IFSTTAR-LEOST F-59650 Villeneuve d'Ascq, France*

[8]. Simulation Of Urban Mobility : www.sumo-sim.org