



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6

Issue: IX

Month of publication: September 2018

DOI:

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Realistic Simulation of Vehicular Network in Urban and Semi-Urban Area of India

Shivani Nishad¹, Dr.Tulika Pandey²

^{1,2}Department of Computer Science and Information Technology, SHUATS, Allahabad, Uttar Pradesh, INDIA

Abstract: These days wireless communication has impacted our daily lives. The developments achieved in this field have made our lives amazingly simpler, easier, convenient and comfortable. One of these developments has occurred in Car to Car Communication (C2CC) in Vehicular Ad-hoc Network (VANET).

For a closer look on C2CC studies the necessity of simulation is obvious. Network simulators can simulate the Ad-hoc network but they cannot simulate the huge traffic of cities. To simulate huge traffic of the city, traffic simulator is required. So to simulate VANET both network and traffic simulators are required. Lots of network and traffic simulators are developed.

In this paper, Traffic simulator (SUMO) Network Simulator (OMNet++) and Veins framework is used to simulate VANET in Urban and Semi Urban Area of India. Realistic simulation results are observed like how much BSM (Basic Safety Message) sent and received and CO₂ Emissions produced in Urban Area (Delhi) & Semi-Urban Area (Allahabad).

Keywords: VANET; SUMO; OMNet++; Veins; TraCI; BSM; CO₂ Emissions;

I. INTRODUCTION

VANETs were first mentioned and introduced in 2001 under "car-to-car ad-hoc mobile communication and networking" applications, where networks can be formed and information can be relayed among cars. VANETs are a key part of the intelligent transportation systems (ITS) framework. A form of mobile ad hoc network, to provide communications among nearby vehicles and between vehicles and nearby fixed infrastructure, usually described as roadside unit. While, in the early 2000s, VANETs were seen as a mere one-to-one application of MANET principles, they have since then developed into a field of research in their own right. The term VANET became mostly synonymous with the more generic term **inter-vehicle communication (IVC)**, although the focus remains on the aspect of spontaneous networking, much less on the use of infrastructure like Road Side Units (RSUs) or cellular networks. VANETs to provide road safety, navigation, and other roadside services.

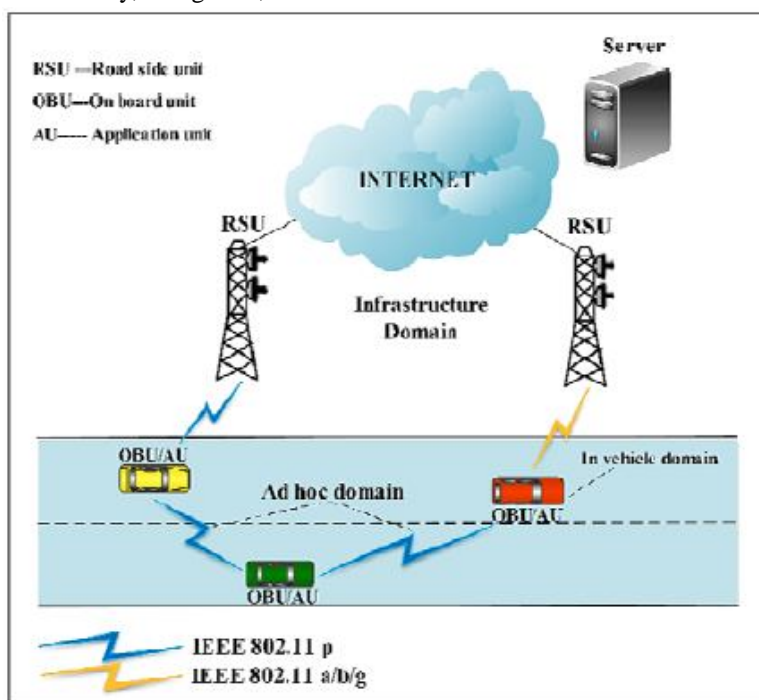


Fig 1: VANET Architecture

A. VANET Communication are Three Types

- 1) *Inter-vehicle communication:* The inter-vehicle communication configuration uses multi-hop multicast/broadcast to transmit traffic related information over multiple hops to a group of receivers.
- 2) *Vehicle-To-Roadside Communication:* The vehicle-to-roadside communication configuration represents a single hop broadcast where the roadside unit sends a broadcast message to all equipped vehicles in the vicinity.
- 3) *Inter-roadside Communication:* The Inter-roadside communication in vehicular ad-hoc network represents communication between RSU-to-RSU, RSU-to-Internet (WiMax/3G Base Station) and RSU-to-Traffic Probe.

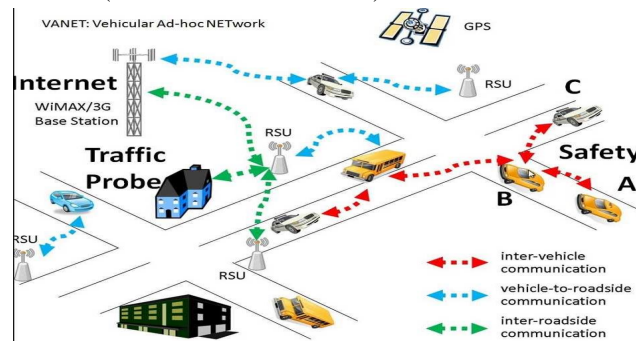


Fig 2: Communication in VANET [1]

B. Characteristics of VANET

VANET is an application of MANET but it has its own distinct characteristics which can be summarized as:

- 1) *High Mobility:* The nodes in VANETs usually are moving at high speed. This makes harder to predict a node's position and making protection of node privacy.
- 2) *Network Topology:* Due to high node mobility and random speed of vehicles, the position of node changes frequently.
- 3) *Unbounded Network Size:* VANET can be implemented for one city, several cities or for countries.
- 4) *Frequent Exchange of Information:* The ad-hoc nature of VANET motivates the nodes to gather information from the other vehicles and road side units.
- 5) *Wireless Communication:* VANET is designed for the wireless environment. Nodes are connected and exchange their information via wireless.

To simulate VANET mobility model is used. Mobility model proposed for MANET cannot be because of the intrinsic characteristics of VANET. Traces form the mobility model is fed to the network simulator. But mobility model does not give realistic results. Road traffic simulator can be used to simulate traffic. Simulation framework which couples network & road traffic simulation is already developed. Some of them are VEINS, NCTU'ns.

This Paper addresses the need of bidirectional coupling of traffic simulator with network simulator. In this paper, veins framework is used to simulate VANET for Urban and Semi-Urban Area of India.

The rest of the paper is organized as follows: section II of the paper describe mobility models & simulators, section III of the paper gives all the simulations set-up, proposed work are presented in section IV of the paper, and finally section V of the paper concludes the Conclusion and future scope.

II. MOBILITY MODELS & SIMULATORS

A. Mobility Models

VANET is a subclass of MANET, yet the mobility model designed for MANET cannot be used in VANET because of the inherent characteristics of VANET. Successful mobility models like random waypoint model or Manhattan Model are not suitable for VANET.

The mobility model is described by the framework, which includes topological maps like lanes, roads, streets, obstacles in mobility and communication model, car velocities, the attraction and repulsion points, based on traffic densities relating to how the simulation time could vary, vehicular distribution on roads and intelligent driving pattern. VANET mobility model [2] can be Macroscopic and Microscopic. The node mobility includes streets, lights, roads, buildings etc. and is classified as Macroscopic, whereas the movement of vehicles and their behaviors are classified as Microscopic.

Vehicular mobility generators are needed to increase the level of realism in VANET simulations. They generate realistic vehicular mobility traces to be used as an input for a network simulator. The output of the trace details the location of each vehicle at every time instant for the entire simulation time and their mobility profiles. Some traffic simulators are VANETMobiSim [3], CityMob [4], STRAW [5], and SUMO [6].

A Network Simulator is software that predicts the behaviour of a computer networks. since communication networks have become too complex for traditional analytical methods to provide an accurate understanding of system behaviour, network simulators are used. Most simulators use discrete event simulation- the modeling of system in which state variables change at discrete points in time. Some networks simulators are OPNET [7], QualNet [8], OMNet++ [9], NS2 [10], and NS3 [11].

There are very few integrated frameworks available. Currently, the mobility and network models in integrated frameworks are implemented in two separated simulation tools. Therefore, there is a clear need for an integrated mobility and network simulators in order to evaluate effectively performance of IVC systems. Some Hybrid Simulators are available like TraNS (Traffic and Network Simulation Environment) [12], NCTU'ns [13], MobiREAL [14] and VEINS [15].

B. Simulators Used

1) *Traffic Simulator (SUMO)*: "Simulation of Urban MOBility" (SUMO) is an open source, highly portable, microscopic road traffic simulation package designed to handle large road networks. SUMO is a free and open traffic simulation suite which is available since 2001. SUMO is a wealth of supporting tools which handle tasks such as route finding, visualization, network import and emission calculation. SUMO is implemented in C++ and uses only portable libraries. It Supports import formats like OpenStreetMap, VISUM, VISSIM, NavTeq. SUMO is widely used by the V2X community for both, providing realistic vehicle traces, and for evaluating applications in an on-line loop with a network simulator.

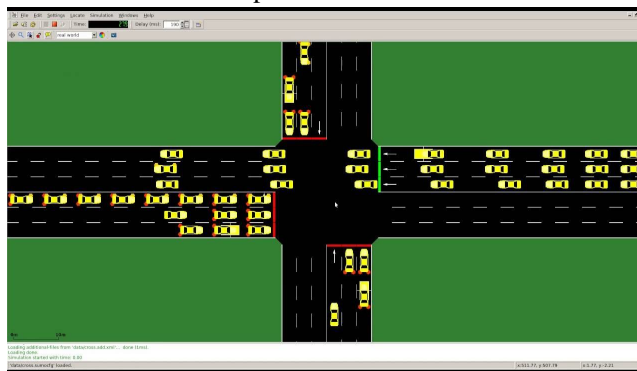


Fig 3: SUMO Graphical Environment

2) *Network Simulator (OMNET++)*: The OMNeT++ Integrated Development Environment is based on the Eclipse platform. It is developed by Andras and his team since 2001. The most common use of OMNeT++ is for simulation of computer networks. It provides component architecture for models. OMNet++ adds functionality for creating and configuring models (NED and INI files), performing batch executions, and analyzing simulation results, while Eclipse provides C++ editing, SVN/GIT integration, and other optional features (UML modeling, bugtracker integration, database access, etc.) via various open-source and commercial plug-ins. OMNet++ is compatible with traffic simulator SUMO.

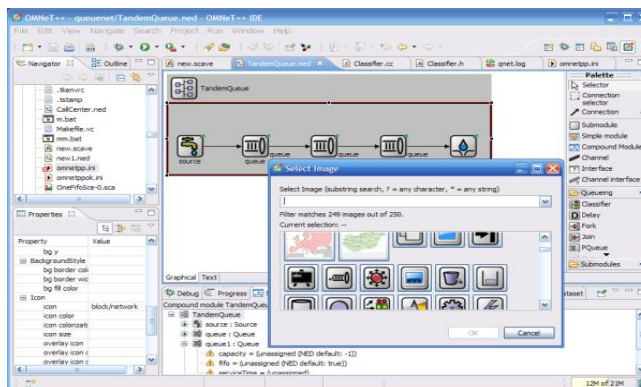


Fig 4: The OMNet++ IDE, editing a NED file

C. Hybrid Simulator (VEINS)

Veins is the Open Source vehicular network simulation framework, ships as a suite of simulation models for vehicular networking. These models are executed by an event based Network simulator (OMNeT++) while interacting with a road traffic simulator (SUMO). With the help of Veins each simulation is performed by executing two simulators in parallel: OMNeT++ (for network simulation) and SUMO (for road traffic simulation). Both simulators are connected via a TCP socket. The protocol for this communication has been standardized as the Traffic Control Interface (TraCI) [16].

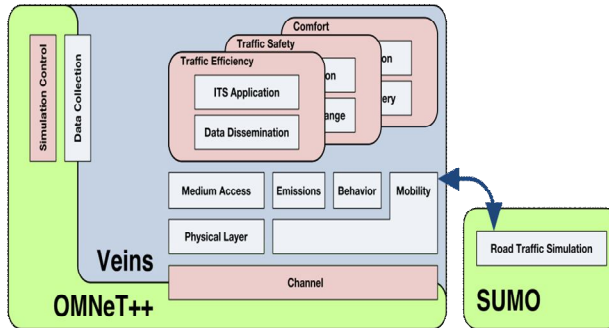


Fig 5: Veins diagram

D. TraCI (Traffic Control Interface)

Veins instantiates one network node per vehicle driving in SUMO. This task is handled by TraCIScenario Manager Launchd module: it connects to a TraCI server (SUMO or sumolaunchd (/documentation/sumo-launchd/)) and subscribes to events like vehicle creation and movement. For each vehicle created in SUMO, it instantiates one OMNeT++ Compound Module in the OMNeT++ simulation. This module is assumed to contain a mobility submodule of type TraCIMobility. At regular intervals it will use this module to advance the simulation in SUMO and update the node's mobility information (such as position, speed, and direction) based on the behavior of the vehicle. For quick tests, the TraCIMobility module also includes functionality to stop a vehicle at a predefined point in time (configured via its accident Start, and accident Duration parameters).

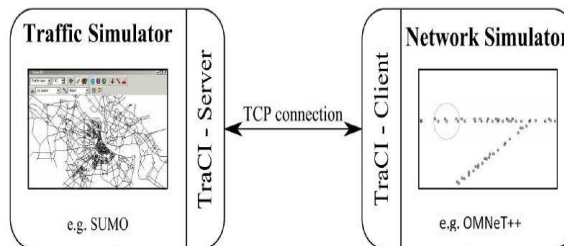


Fig 6: Network Simulator OMNeT++ coupled with Traffic simulator SUMO Using TraCI

III. SIMULATION SETUP

We configured vehicles to drive at a maximum speed of 14 m/s and modeled dense inner-city traffic with inattentive drivers. For all communications, the complete network stack, including ARP, is simulated and wireless modules are configured to closely resemble IEEE 802.11b network cards transmitting at 11 Mbit/s with RTS/CTS disabled.

TABLE 1

Simulation Software	Versions
LINUX PLATFORM	16.04.1
OMNET++	5.1.1
SUMO	0.30.0
INET	3.6.0
VEINS	4.6

TABLE 2
Road Traffic MicroSimulation Setup

PARAMETERS	VALUE
Maximum vehicle speed	14 m/s
Maximum vehicle acceleration	2.6 m/s ²
Assumed vehicle length	5m
Driver imperfection ("dawdling")	0.5

TABLE 3
INET Framework Module Parameters

PARAMETERS	VALUE
TCP.mss	1024 Byte
TCP.advertisedWindow	14336 Byte
TCP.tcpAlgorithmClass	TCPReo
ARP.retryTimeout	1s
ARP.retryCount	3
ARP.cacheTimeout	100s
Mirate	Auto
Mac.broadcastBackoff	11 Mbit/s
Mac.maxQueueSize	31 slots
Ma.rtsCtsac.address	14 Packs
Mac.b	False
decider.bitrate	11 Mbit/s
decider.snirThreshold	4 db
snrEval.bitrate	11 Mbit/s
snrEval.headerLength	192 bit
snrEval.ThresholdLevel	3 dB
snrEval.ThermalNoise	-110 dB
snrEval.sensitivity	-85 Db
snrEval.pathLossAlpha	1.9
snrEval.carrierFrequency	2.4 GHz
snrEval.transmitterPower	2 mW
channelcontrol.carrierFrequency	2.4
channelcontrol.pMax	2m W
channelcontrol.sat	-80 dBm
channelcontrol.alpha	1.9

To provide ad-hoc routing among the nodes, we use our implementation of the Dynamic MANET On Demand (DYMO) routing protocols as an application layer module of INET Framework module set. As per the specification, it uses a node's UDP module to communicate with other instances of DYMO, to discover and maintain routes, and thus, establish a VANET.

IV. PROPOSED WORK

A. A Realistic Example – Street Map of ALLAHABAD and DELHI

Building on the first proof-of-concept examples in which vehicles traveled on an artificial grid of roads, we now used the coupled simulations environment to model IVC among traffic in the city of Allahabad and Delhi.

In this section step by step method of the simulation is presented.

1) **STEP1. Exporting the map from Open Street Map:** The road map on which the VANET is to be simulated is imported from openstreetmap.org by manually selecting the desired area and using the 'export as .osm' option to download a map.osm file, which is an xml file with an open street map defined schema [17].

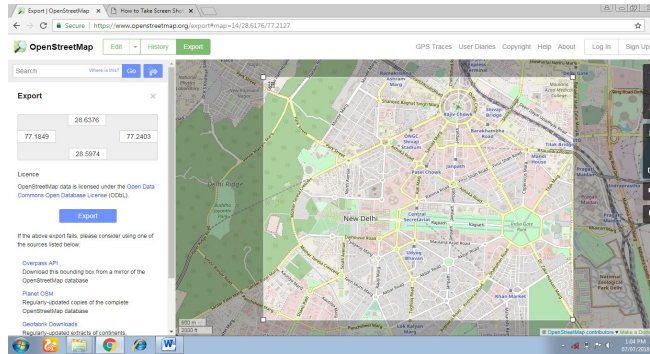


Fig. 7: DELHI MAP (Urban Area)

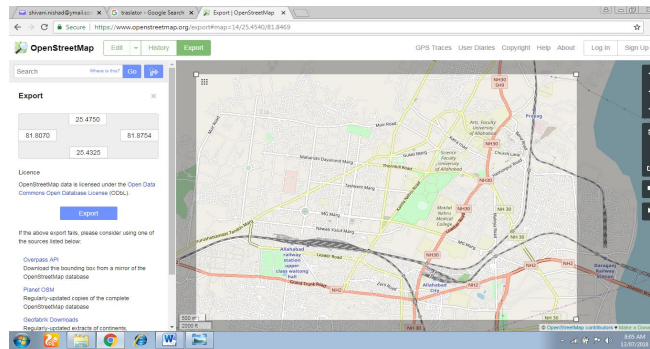


Fig. 8: ALLAHABAD MAP (Semi-Urban Area)

2) **STEP2. Generating a Sumo network using NETCONVERT:** SUMO provides a command line application called Netconvert which is used to import and generate road networks that can be used by SUMO and other tools present in the SUMO package. Also, it projects the geographic co-ordinates (Latitude and Longitude) to x-y coordinates and also applies the required offset to translate the axes to the first quadrant. The following command line arguments are passed to achieve the desired road network. Generating a sumo network by Netconvert from these simple steps.

- a) Open terminal and run these commands
- b) Change current location to working directory where the file (map.osm) have been saved in folder Delhi.
- c) Run these command
- d) Copy the polygon structure from this website sumo.dlr.de/wiki/Networks/Import/OpenStreetMap
- e) Copy that polygon code in gedit (editor) and delete this line shown in picture below.

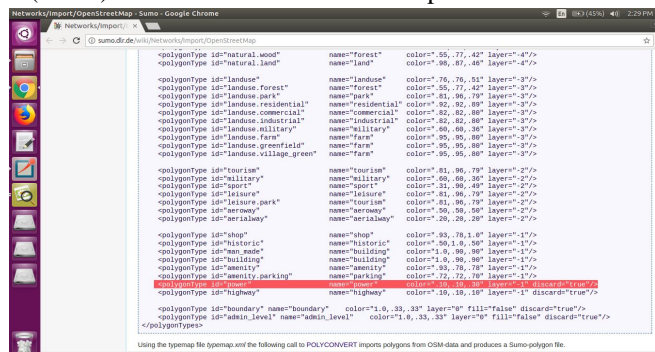


Fig. 9: Polygon Code

f) Save this file named typemap.xml (typemap.xml file saved in your working folder delhi)

3) *STEP3. Generating Geometrical shapes using polyconvert*

Polyconvert imports geometrical shapes (polygons or points of interest) from different sources, converts them to a representation that may be visualized using sumo-gui. Run these command in command-line.

```
polyconvert - -net-file map.net.xml - -osm-files map.osm - -type-file typemap.xml -o map.poly.xml
```

4) *STEP4. Generating vehicle routes using DUAROUTER:* The resulting map.net.xml file from the previous step is now used to generate a Trips file using the RandomTrips python script provided as part of the SUMO package. This script generated a set of random trip definitions- demand for routes between two points in the road network. These are then used in the creation of vehicle routes.

Run these command in command line

```
python/home/shivani/sumo-0.30.0/tools/trip/randomTrips.py -n map.net.xml -e 100 -l
```

The map.trips.xml along with the map.net.xml is now used to generate the vehicle routes using the Duarouter.exe application which again is a part of the SUMO package.

Run these command in command line

```
python/home/shivani/sumo-0.30.0/tools/trip/randomTrips.py -n map.net.xml -r map.rou.xml -e 100 -l
```

5) *STEP5. SUMO Configuration File:* Each scenario in sumo has a .sumo.cfg file associated with it which points to the corresponding .net.xml and .rou.xml files to be used, and also the start and end time of the simulation. The xml schema used is as described below:

a) *SUMO-GUI Configuration:* To display the right vehicle colors as set by the Public Safety Application in OMNET through TraCI, the 'View Settings' in the SUMO-GUI application are set to display the 'given/assigned vehicle color'.

6) *STEP6. Import project in OMNet++:* Import your own project (delhi) in OMNet++. These are the steps to import project in OMNet++.

- a) Click on file →New→OMNet++ Project→Give name of your own project
- b) Select Empty Project
- c) Click on OMNet++ Simulation
- d) Then check debug and release
- e) Finish

7) *STEP7. OMNet++ Simulation Setup:* In Network Description file (NED) describe all the network modules used in the scenario like World, Connection Manager, Obstacle, Annotations, RSU.

The simulation in OMNet++ can be controlled by providing the required modifiable parameters per module that is being used in the simulation using the .ini file (Initialization File) corresponding to that simulation. The playground size is set to the maximum size of the map in order to prevent vehicles from going out of bounds. The mobility of vehicle in x and y direction is handled by the mobility model and hence set to zero. But the position on the z-axis is set to a constant value representing the height of the antenna from the ground.

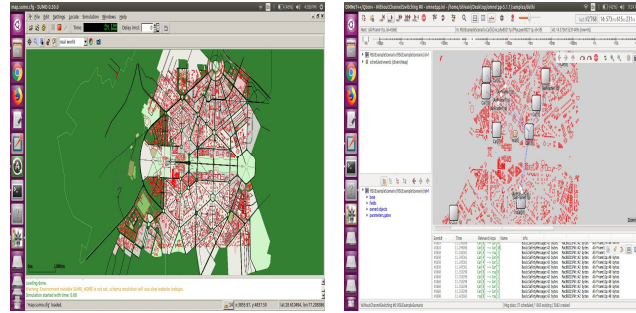
INI is prepared according to scenario created for Delhi and Allahabad traffic.

After completing NED and INI launch file is made in xml format. Then the project is built and TraCI is launched using the following command.

```
cd omnetpp-5.1.1/samples/veins/python sumo-launchd.py -vv -c
```

Bidirectionally coupled network (traffic simulator + network simulator) is established now and running parallel via TCP Socket.

Graphical view in Sumo and OMNet++ after launching TraCI:



a) Graphical view in SUMO b) graphical view in OMNet++

IV. RESULTS

Most of the VANET researches are simulated for managed traffic of very urban cities of USA, Europe and JAPAN. Very few researchers simulate VANET communication in a realistic manner for Indian Cities where the traffic is not managed. VEINS, which combines SUMO and OMNet++ is used for getting the realistic result for Delhi and Allahabad. The C2CC program used is very simple which is just showing simple Car to Car communication. For simplicity, UDP is used as transport protocol, RTS/CTS is disabled. The results are evaluated on the following parameters.

- 1) *Basic Safety Message (BSM)*: Basic Safety Message (BSM) is broadcasted periodically to announce vehicle's status information (i.e., position, direction, velocity, neighbor number, etc.) to its neighbors for collision detecting, for instance.
- 2) *CO₂ Emission*: In order to estimate CO₂ emissions, we integrated the EMIT model of vehicle emissions with our OMNet++ mobility model. EMIT calculates emissions depending on vehicle speed and acceleration, taking into account vehicle characteristics such as total mass, engine, and installed catalytic converter.

A. Urban Area (DELHI)

Figure 10 shows the Number of BSM sent and received by the car. On the x-axis car numbers are shown and on the y axis number of BSMs are shown. Red line shows the generated BSM by the vehicle and the green line shown the received BSM. For a simple VANET scenario, RTS/CTS mechanism in IEEE 802.11p is disabled. Due to this, there may be chances of collision of broadcasted packets. When there is one car in the network, it is periodically broadcasting BSMs. Because there is no other car to receive the BSM, number of BSMs generated by car 1 is 25. When more number of cars joins BSMs are received and periodic broadcasts are less. As a result, car 20 generates only 2 BSMs. Because BSMs are getting collided, some of the vehicles are receiving BSM and some are not receiving. Number of BSM received by a vehicle is also less than the generated BSM.

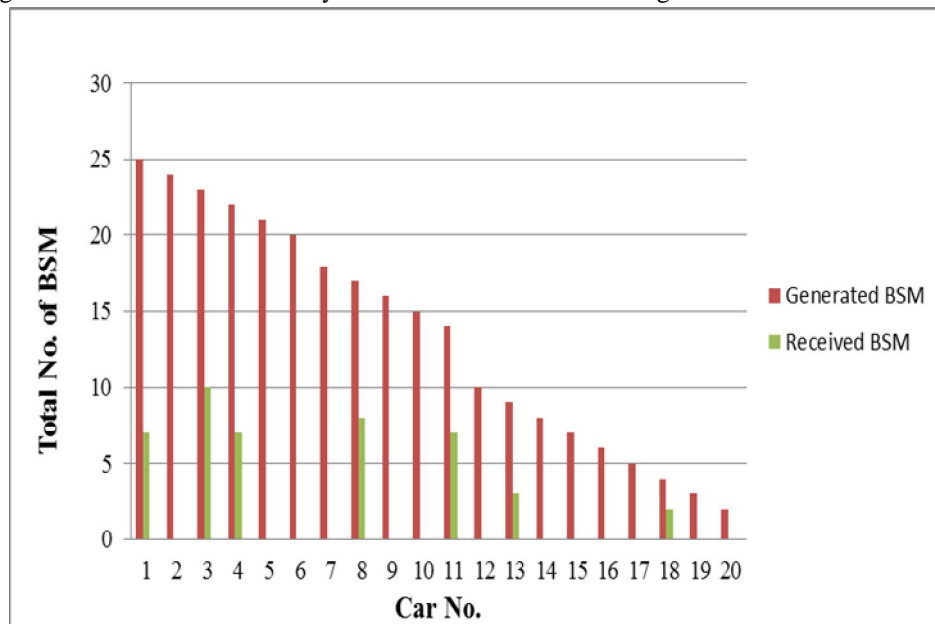


Fig 10: BSM for Delhi

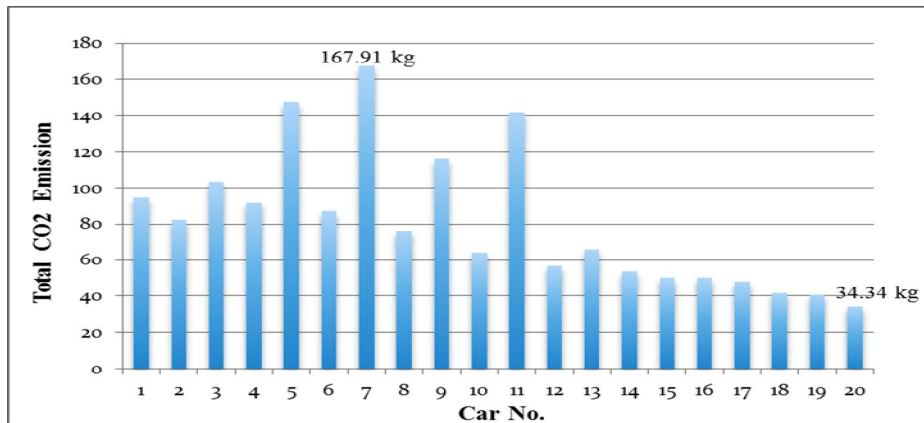


Fig 11: CO₂ Emission for Delhi

Fig 11 shows the CO₂ emission in Delhi scenario. CO₂ emission in kilogram for all the vehicles are calculated Car no.7 and 5, 11 recorded a much larger amount of CO₂ emitted per meter traveled, and reflecting the large number of car caught in the jam and emitting CO₂, but not traveling.

B. Semi-Urban Area (ALLAHABAD)

Figure 12 shows the generated and received BSM in Allahabad. In Allahabad, as the total numbers of vehicles are less in the network, BSM generated are more in comparison to Delhi. Car 1 is generating 32 BSM. As it is alone in the network, it is kept on broadcasting BSM and not getting any reply. Car 2 is also generating 30 BSMs whereas car 2 in Delhi is generating 24 BSM. This is because in Allahabad, car 2 is joining late in the network.

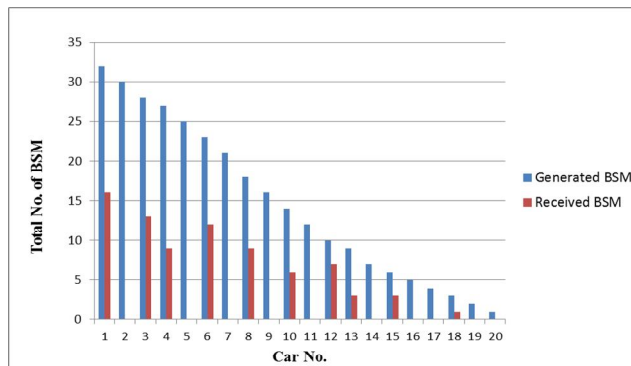


Fig 12: BSM for Allahabad

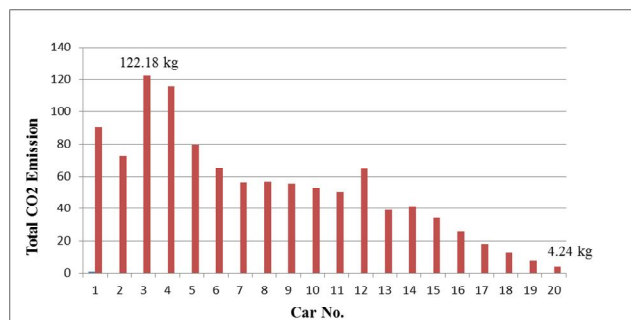


Fig 13: CO₂ Emission for Allahabad

Fig 4.4 shows the CO₂ emission in Allahabad scenario. Car number 3, 4, and 1 recorded a much larger amount of CO₂ emitted per meter traveled, and reflecting the large number of car caught in the jam and emitting CO₂, but not traveling. In Allahabad, large numbers of cars are not caught in traffic jam, so CO₂ emission is less than Delhi.

Simulation results shown are expected.

V. CONCLUSIONS & FUTURE SCOPE

We simulate the realistic traffic of the Delhi and Allahabad as an ad-hoc network. And expected results are observed. In Delhi, we found that the density of the number of vehicles is very high and there is a lot more traffic as compared to Allahabad. Due to traffic, a large number of vehicles are caught in the jam and producing high CO₂ emission. The vehicle nos. 7,5,11 produced 167.91 kg, 147.71 kg, and 142.24 kg CO₂ emission respectively. In Allahabad, the numbers of vehicles are less in the network and there is no traffic signal. Whenever so many vehicles are caught in the jam and producing fuel then CO₂ emission may high like vehicle no. 3,4,1 (122.18 kg, 115.72 kg, 90.77 kg) respectively. In Delhi, total BSM generated and received are less as compared to Allahabad. Some packets collide with each other as shown in the results because RTS/CTS is disabled.

A. Future Scope

In this paper, a simple VANET scenario is simulated and presented. So, lots of things can be done in future. Some of them are listed below.

- 1) TCP can be used as a transport protocol.
- 2) RTS/ CTS mechanism can be enabled. When RTS/CTS will be enabled, chances of collision are decreased. There by more number of packets can be received.
- 3) No obstacle and incident were added in the presented VANET scenario. So after adding obstacles and incidents, the results will be more realistic.
- 4) Results can be found to compare TCP and UDP packets.
- 5) I have used DYMO routing protocol. Some other routing protocols can also be simulated and a comparison can be presented.

REFERENCES

- [1] Sourav Kumar Bhoi & Pabitra Mohan Khilar (2013), "Vehicular communication: a survey".
- [2] F. Karnadi, Z. Mo & K. Lan "Rapid Generation of Realistic Mobility Models for VANET," WCNC'07
- [3] H  rri, Jerome, Fethi Filali & Christian Bonnet (2009), "Mobility Models for Vehicular Ad-Hoc Networks: A Survey and Taxonomy. IEEE Communications Surveys & Tutorials, 11:4, 19-41. ISSN:1553-877X.
- [4] Martinez FJ, Cano JC, Calafate CT & Manzoni P. Citymob (2008), "A mobility model pattern generator for VANETs". In IEEE Vehicular Networks and Applications Workshop (VehiMobi, held with ICC), Beijing, China, May 2008.
- [5] Bustamante & Fabian E (2013), "STRAW - STreet Random Waypoint - vehicular mobility model for network simulation". Available from Internet: <URL: <http://www.aqualab.cs.northwestern.edu/projects/144-straw-street-random-waypoint-vehicular-mobility-model-for-network-simulations-e-g-car-networks>>.
- [6] Michael Behrisch, Laura Bieker, Jakob Erdmann & Daniel Krajzewicz (2011), "SUMO - Simulation of Urban Mobility An Overview" SIMUL 2011", The Third International Conference on Advances in System Simulation.
- [7] Alain, et al. (1986) available at: <https://en.wikipedia.org/wiki/OPNET>
- [8] Dr. Rajive Bagrodia (2001), "Scalable Network Technologies (SNT)" available at: <https://web.scalable-networks.com/history>
- [9] Andras.Varga, "The OMNET++ Discrete Event Simulation System," Proc. European Simulation Multiconf. (ESM '01), June 2001
- [10] Fall. K & Varadhan. K. (2000), "The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC", February 2000. Available at: <http://www.isi.edu/nsnam/ns/ns-documentation.html>.
- [11] Tom Handerson, et al. (2008), available at: [http://en.wikipedia.org/wiki/Ns_\(simulator\)#ns-3](http://en.wikipedia.org/wiki/Ns_(simulator)#ns-3).
- [12] M. Piorkowski M. Raya A. Lezama Lugo P. Papadimitratos M. Grossglauser & J.-P. Hubaux "TraNS: Realistic Joint Traffic and Network Simulator for VANETs"
- [13] S.Y. Wang, C.L. Chou, Y.H. Chiu, Y.S. Tseng, M.S. Hsu, Y.W. Cheng, W.L. Liu & T.W. Ho, "NCTUns 4.0: An Integrated Simulation Platform for Vehicular Traffic, Communication, and Network Researches," WiVec'07.
- [14] F. Karnadi, Z. Mo & K. Lan "Rapid Generation of Realistic Mobility Models for VANET," WCNC'07.
- [15] C. Sommer, R. German, & F. Dressler (2011), "Bidirectionally coupled network and road traffic simulation for improved IVC analysis"
- [16] Axel Wegener, Michał Piorkowski, Maxim Raya, Horst Hellbruck, Stefan Fischer & Jean-Pierre Hubaux "TraCI: An Interface for Coupling Road Traffic and Network Simulators"
- [17] Harshad Phule (2012), "Public Safety Application for Approaching Emergency Vehicle Alert and Accident Reporting in VANETs Using WAVE"



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)