ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 5, May 2015

Simulation Environment for VANET

Shifali[1] , Yogesh Juneja[2]

Student, M. Tech, (ECE), PDM College of Engineering, Bahadurgarh, India [1]

Assistant Professor, (ECE), PDM College of Engineering, Bahadurgarh,, India [2]

Abstract: Vehicular Ad hoc Networks (VANETs) are classified as an application of Mobile Ad-hoc Networks (MANETs) that has the potential in improving road safety and providing Intelligent Transportation System (ITS). But VANET simulation is different from MANETs (mobile *ad hoc* networks) simulation because in VANETs vehicular environment imposes new issues and requirements, such as constrained road topology, multi-path fading and roadside obstacles, traffic flow models, trip models, varying vehicular speed and mobility, traffic lights, traffic congestion and driver's behavior. There are different type of simulators that we have used. These simulators are traffic simulators network simulators and combination of both VANET simulator. In this paper, we have studied different type of simulators and their software characteristics, graphical user interface, accuracy of simulation, input requirement output visualization.

Keywords: VANET, MANET,V2V, V2I, ITS,RSU.

I. Introduction

The most superior technology for Intelligent transportation system (ITS) is VANET which is subclass of MANET. It provide wireless communication among vehicles and road side unit (RSU).By using wireless links to send information to every user which are in the range of VANET area, it provide a best safety applications to passengers on road side area. VANETs are characterized by: (a) trajectory-based movements with prediction locations and time-varying topology, (b) varying number of vehicles with independent or correlated speeds, (c) fast time-varying channel (e.g., signal transmissions can be blocked by buildings), (d) lane-constrained mobility patterns (e.g., frequent topology partitioning due to high mobility), and (e) reduced power consumption requirements [1]. Due to high-speed mobility, V2V and V2I communication links tend to be short lived. Thus, it is important to propagate traffic-related information toward a certain region of interest instead of sending to a particular vehicle; hence a particular region is to be set to transmit information to from wireless devices. Deploying and testing VANETs involves high cost an intensive labor. Hence, simulation is a useful alternative prior to actual implementation. Simulations of VANET often involve large and heterogeneous scenarios. The models try to closely represent the movement patterns of users [2]. Moreover, it is well known that mobility models can significantly affect simulation results. For results to be useful, it is important that the simulated model is as close to reality as possible [3]. Hence, Simulators have become indispensable tools at least in the initial phases of the VANET application engineering process . Under these conditions, computer simulation has become the main tool in VANET research.

II. BACKGROUND

Vehicular networks have a lot of similarities to mobile ad-hoc network, but network topology in VANET networks is highly dynamic due to fast movement of vehicles and the topology is often constrained by the road structure and different obstacle like traffic lights, buildings, or trees, resulting in poor channel quality and connectivity in wireless connection. Therefore, protocols developed for traditional MANETs is fail to provide Therefore, protocols developed for traditional MANETs is fail to provide reliable, high throughput, and low latency performance in VANETs. Thus, there is a need for effective protocols that take the specific characteristics of vehicular ad-hoc network. VANET simulation is a challenging task, since it involves network simulation and traffic simulation. A number of network simulators are currently available, with ns-2 being the most prominent. However, ns-2 also brings performance issues regarding the nodes behavior as real vehicles. Simulating a VANET involve two different aspect. First aspect is related to communication among vehicles. Second aspect is related to mobility of VANET nodes. Network simulators are used for communication issues and traffic simulators are take into account of node movement [4]. Choffnes and Bustamante (2005) showed that the vehicular mobility (traffic) model is very important, and its integration with the wireless network model could produce more significant results. The authors present an integrated simulator that uses an original vehicular traffic model called Street Random Waypoint (STRAW) implemented on top of JiST/SWANS (2008) [5]. The authors have used the simulator to show that studying routing protocols for a vehicular network without an accurate vehicular traffic model is a wrong approach. The mobility model implemented in some simulators is not a sufficiently accurate representation of actual vehicle mobility. For example, in the model of Saha and Johnson (2004), each vehicle moves completely independent of other vehicles, with a constant speed randomly chosen. Multi-lane roads or traffic control systems are not taken into consideration. Other authors (Mangharam et al., 2005) make similar simplifying assumptions and do not consider multi- lane roads or car following models. The mobility model of Choffnes and Bustamante (2005) is more complex: the motion of a vehicle is influenced by the

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 5, May 2015

preceding vehicle, and traffic control systems are considered. However, multi-lane roads are not taken into consideration. Hence, simulators have been used to analyze the networks which represent thousand of nodes in cities and in highways. Hence simulators allow the evaluation of a large range of vehicular computing applications, which cannot be studied by using other simulators, and can be used to improve both car-to-car communication protocols and traffic control applications.

III.SIMULATION ENVIRONMENT

We have to define two types of simulation environment to define real life scenario of road environment and other critical situation arise in roads.This simulation environment define a particular solution of a given scenario but all time same result is not found. Hence, a precise solution of a given scenario is define by the simulators. Two types of simulators are used. One is traffic simulator and other is network simulator.

A) TRAFFIC SIMULATOR

1 **Simulation in urban mobility(SUMO) [6]]**: SUMO is open source microscopic simulation package first time used in 2001 by GERMAN AEROSPACE CENTRE(DLR) .SUMO is an important tool for researchers in an urban traffic and transportation domain.Hence, it is not only a simple traffic simulator but also a complete suite of application to perform the simulation of traffic on complex environment.

Each vehicle in SUMO is define by its properties like: its arrival time, departure time, vehicle route' through the network, lane to use, position, velocity and vehicle types. Vehicle types define the vehicle physical properties and variable of vehicle movement model.The simulation in SUMO is time discrete and each vehicle's position is define by the no. Of lane in which vehicle move and the distance is from the starting of this lane.Vehicle speed is determined by car following model.

To create a scenario in SUMO we have to create XML files. A node XML and edge XML files act as input and create output as net XML by using NETCONVERT. Now by using node XML and edge XML create flow file that define the flow between edges. When SUMO work on flow file and net file as input we create a route file with an extension rou XML [7].

Route file and net file act as input for creating a SUMO cfg XML file.SUMO has SUMO cfg XML file.SUMO has inbuilt utility to generate trace file.These trace files act as input for network simulator. NS-2 used .tcl trace file while NS-3 used .cc extension trace file by deleting some code in .tcl file.

2 **Mobility model generator for vehicular networks(MOVE)** [8]: MOVE is an open source traffic simulator and its suite is top of SUMO.It is based on JAVA. Hence, move simulator is jar file which open in JAVA. FIG. 1 Define the GUI of MOVE simulator.

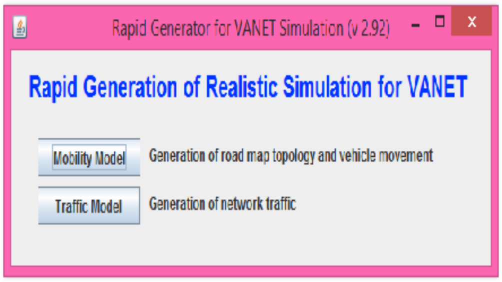

FIG:1 Graphical user interface(GUI) of MOVE

MOVE generate two types of models. One is mobility model and other is traffic model. Mobility model generator provides a user friendly interface for generating mobility model for simulations using SUMO. It also allows the user to create customized topology or import maps. In map editor nodes and edges are define by X-Y co-ordinates to design the road model. Vehicle movement define the vehicle flow between edges and turn position. We can define different map topology or TIGER map in MOVE .And network traffic model generator takes the SUMO trace file as the input and generates the network traffic model as required by either NS-2 or QualNet. MOVE provides a GUI that allows the user to quickly generate realistic simulation scenarios without the hassle of writing simulation scripts as well as learning about the internal details of the simulator.

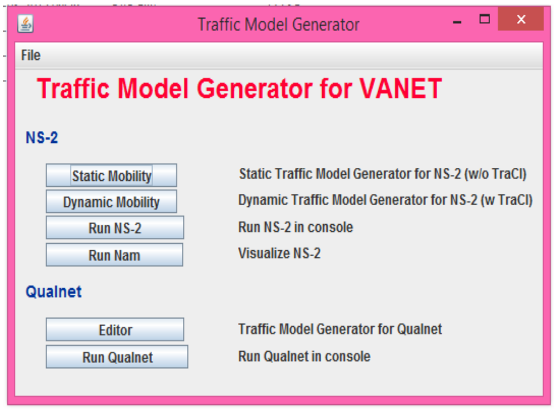

FIG 2: Traffic model generator of MOVE

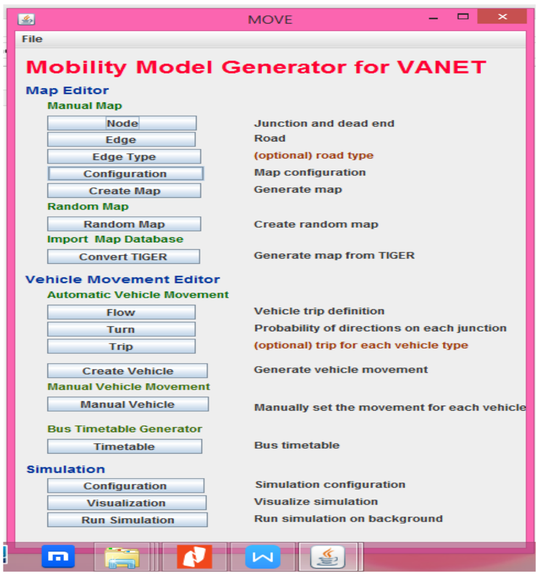

FIG 3: Mobility model generator of MOVE

DOI 10.17148/IJARCCE.2015.45104

3).**VanetMobiSim**: VanetMobiSim is an extension of CanuMobiSim mobility simulator.This is an open source discrete event simulator which work on JAVA and support both macro mobility and micro mobility.It support TIGER(topologically integratedgeographicencoding and referencing database,) database at macroscopic level and support common mobility model at microscopic level. At macroscopic level, TIGER line define the geographical features such as roads , rail roads, river , lake and some legal boundaries.At microscopic level, different mobility models are: intersections. At the microscopic level, it supportsmobility models such as Intelligent Driving Modelwith Intersection Management (IDM/IM), IntelligentDriving Model with Lane Changing (IDM/LC) andan overtaking model (MOBIL), which interacts withIDM/IM to manage lane changes and vehicle accelerations [10]. It generate movement trace pattern that are input of some network simulators such as NS-2,QUALNET,GloMoSim.

4)**FreeSim [12]:** FreeSim is an open source macroscopic microscopic free flow traffic simulator. FreeSim'sgraphical user interface (GUI) runs within a web browserusing the Adobe Flash plug-in and connects to a Java-basedserver application. Free way system define by graph and graph are represented by nodes and edges description for distance measurement.

Each independent entity (vehicle) in a free way system send their current speed and location to central server. Central serve send information regarding shortest path and fastest speed of new path.In FreeSim six inbuilt shortest path algorithms [13] : Dijkstra'sAlgorithm , Bellman-Ford's Algorithm Johnson'sAlgorithm , and the three All-Pairs All-Paths PreComputedalgorithms.

5)**. Street Random Way Point (STRAW) [5]**: STRAW is an open source discrete event simulator basically run on JAVA. Every node is move according to their realistic vehicular traffic model on roads define by real maps In STRAW mobility model define in four section inter segment ,intra segment, route management, and execution management [11].In intra segment mobility model we define a car following model which define the starting and exit point of vehicles in a scenario.

Acceleration speed of vehicles is depend on the distance and speed of the following vehicle.If distance is greater than vehicle speed inc up to highest speed limit and change the lane if adjacent lane has higher speed lane and a perfect room for a given vehicle. Inter segment mobility model define the behavior of vehicles between road segment i.e at intersection.

These are depend o different type of road segments and different type of traffic control. Route management and execution are determine by different random way point models, some are define by origin and destination point, some are define by no origin and destination point.

Comparison of different traffic simulator

Different traffic simulators					
	SUMO	MOVE	VANET MOBISIM	FreeSim	STRAW
Source	Open	Open	Open	Open	Open
language	C++	JAVA	JAVA	JAVA	JAVA
Traffic model	Microscopic	Microscopic	microscopic	Macroscopic ,microscopic	Microscopic
Support network simulator	Vis sim, XML discription	Ns-2, GloMoSim,	Ns-2, GloMoSim, QualNet	-	JiST/SWAN
TIGER Map import	No	No	Yes	Yes	Yes
platform	Window,linux	Window, linux	linux	Window,linux	linux
GUI	Yes	Yes	Yes	Yes	Yes
Map import	Real,User defined, Random	Real User defined,Random	Real, Userdefined ,Ranom	Real	Real, User defined
Set up and usage	Hard	Moderate	Moderate	Easy	Moderate

Different traffic simulators					
	SUMO	MOVE	VANET MOBISIM	FreeSim	STRAW
Source	Open	Open	Open	Open	Open
language	C++	JAVA	JAVA	JAVA	JAVA
Traffic model	Microscopic	Microscopic	microscopic	Macroscopic ,microscopic	Microscopic
Support network simulator	Vis sim, XML discription	Ns-2, GloMoSim,	Ns-2, GloMoSim, QualNet	-	JiST/SWAN
TIGER Map import	No	No	Yes	Yes	Yes
platform	Window,linux	Window, linux	linux	Window,linux	linux
GUI	Yes	Yes	Yes	Yes	Yes
Map import	Real,User defined, Random	Real User defined,Random	Real, Userdefined ,Ranom	Real	Real, User defined
Set up and usage	Hard	Moderate	Moderate	Easy	Moderate

B)NETWORK SIMULATOR

For implementing the whole network area on computer we use network simulator because in real life, it s very difficult to implement whole area network.This network is calculated either by network area entities using some mathematical calculations

.The network simulator provide GUI (graphical user interface) based network designer tool to design and simulate a network with different network protocol.

Network simulator test the scenario that are difficult to design in real environment and test new network protocol and changes in existing protocol for better efficiency [14]. There are different types of network simulators which are used in VANET NS-2,NS-3, OPNET,OMNET++, QUALNET, JiST.

1). Network simulator (NS-2) [15]: NS-2 is an open source event driven network simulator licensed by GNU.It support for simulation of TCP, routing , multicast routing, over all network. NS-2 work on C++ language and has developed under VINT project in 1995; it is ajoint effort by people from University of California atBerkeley, University of Southern California's InformationSciences Institute, Lawrence Berkeley National Laboratoryand Xerox Palo Alto Research Center. The main sponsors arethe Defense Advanced Research Projects Agency and theNational Science Foundation.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 5, May 2015

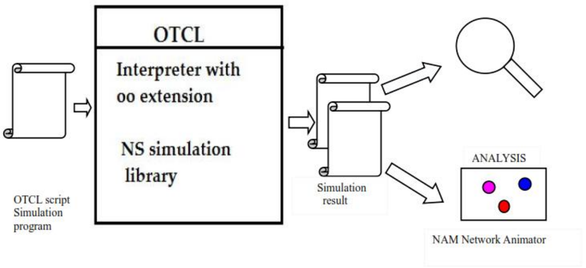

FIG 4: Architecture of NS-2

2).**Network simulator version 3(NS-3) [16]**: NS-3 is an extension of network simulator 2.In NS-2 there is a drawback of scalability . Hence, to improve the scalability of network simulation an advanced version of NS-3 is developed. For improving this ,two approaches are used to extend NS-2; first is grid based and second is list based node organization.By using these approaches on 3000 nodes performance is 30 times faster. This improvement is basically improved by enhancing data structure. Different author may include spatial data structure for efficient radio signal transmission define by wifi channel(medium) some use proximity detection of mobile nodes by using wifiphy which implement in 802.11 physical
layer model for signal transmission, some author define the distributed and parallel computing architecture.

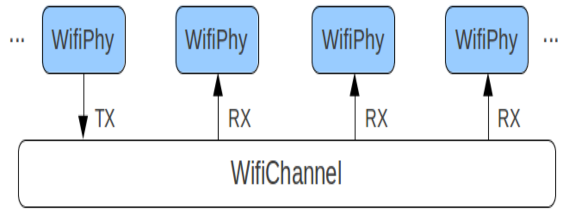

FIG 5: List based data structure used by NS-3

Based on these data structure the simple list based data structure is extended into Quad tree data structure.In Quad tree a given scenario is divided into a small sub scenario.As the distance increase signal strength decrease but with in small scenario the signal strength is not decrease in a specified range, say R, and beyond a specified range R the energy level threshold is zero hence, packet loss and congestion effect are reduced and improvement in scalability.

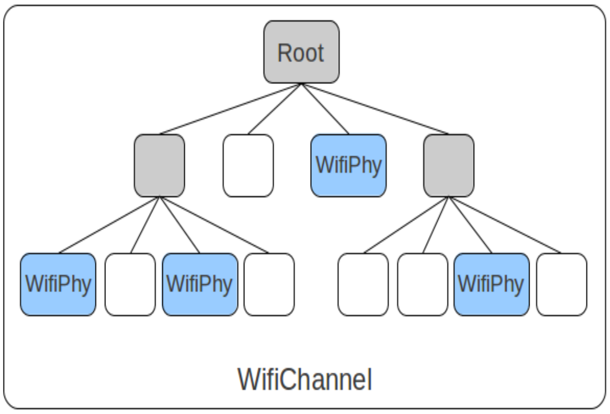

FIG 6: Quad tree data structure used by NS-3

3).**Objective modular network testbed (OMNet++) [17]:**OMNet++ is an open source discrete event based network simulator which work on c++ environment. It contain multiprocessors and distributed or parallel system in its application area.Its topology description language is NED and input of NED is simple module declaration, compound module definition and network definition.Active module are simple module written in C++ using simulation class library and compound module are combination of simple module. Simple module define interface of module and compound module define external module's interfaces(define sub-modules and their interconnections). Network definitions are compound modules thatqualify as self-contained simulation models.

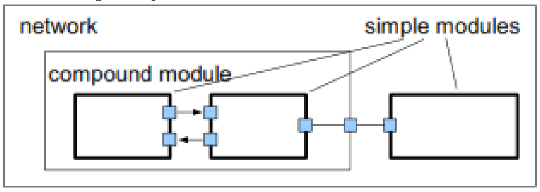

FIG 7: Model structure in OMNet++

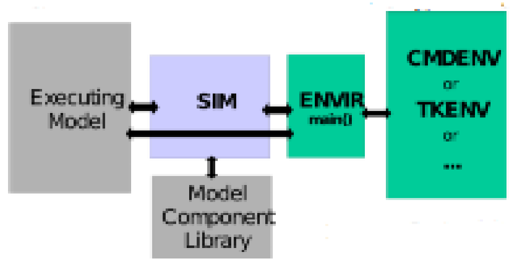

FIG8: Architecture of OMNet++

The executing model execute C++ and NED files. Model component library contain simple module and compound module. Modules are instantiated and theconcrete simulation model is built by the simulation kernel andclass library (*Sim*) at the beginning of the simulation execution.The simulation executes in an environment provided by the userinterface libraries (*Envir*, *Cmdenv* and *Tkenv*) – this environmentdefines where input data come from, where simulation results goto, what happens to debugging output arriving from the simulationmodel, controls the simulation execution, determines how the simulation model is visualized and (possibly) animated, etc.

4). **JAVA in Simulation Time(JiST) [18]**: JiST is an open source discrete event network simulator based on JAVA. JiST architecture has four basic components: a compiler, a language run time environment or virtual machine, re-writer and simulation run time kernel. A simulation is first compiled then dynamically rewritten at application load time and finally executed by the virtual machine with support from the simulationtime kernel.Compiler or run time environment of JiST system can be any standard JAVA compiler or JAVA virtual machine.The re-writer component of JiST is a dynamic class loader.intercepts all class load requests, and subsequently verifies and modifies the requested classes.The program transformations occur

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 5, May 2015

once,load time, and do not incur rewriting overhead duringexecution [19]. At run time, the mod ified classes interact closely with the simulation time kernel through the various injected or modi-fied operations. The kernel is responsible for all the runtimeaspects of the simulation time abstraction.

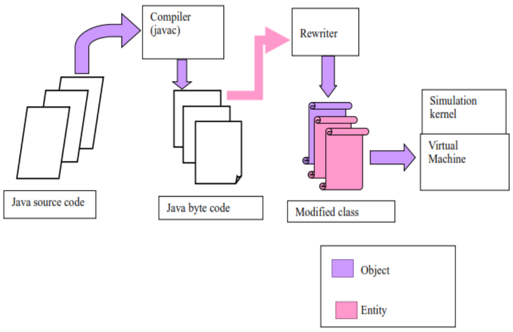

FIG 9: Architecture of JiST

5).Optimized network engineering tool (OPNet) [19]: This is commercial based event driven network simulator basically work on C, C++ language. OPNet operate at packet level and originally build for wired networks.OPNet contain a huge library which contain commercially available fixed wired networks and protocol.The major disadvantages of OPNet is there is a loss of wireless network, but a lot of research has been done to full-fill this requirement.Its modeling is define into three main domain: first is network domain second is node domain and third is process domain [20].

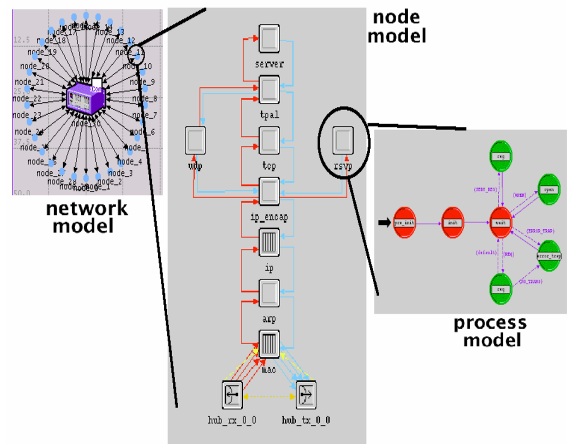

FIG 10: Architecture of OPNet

Network domain specify the overall scope of system to be simulated.

Network model specifies the object in the system as well as their physical location, interconnection and configuration.Node model specify the internal structure of network domain.Nodes include workstations packet switches satellite terminal and remote sensor.Node can be fixed ,mobile or satellite terminal.Processor domain specify the behavior of processor or queue modules which exist in node domain.

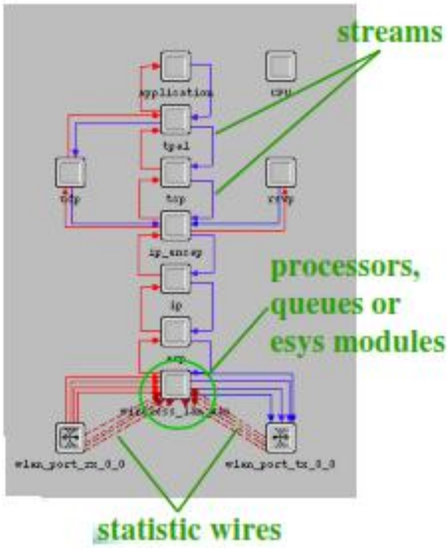

FIG 10: Node model and queue process

Comparison of network simulator

Comparison of network simulator					
	NS-2	NS-3	OPNet	OMNet++	JiST
Source	Open	Open	Commercial	Open	Open
Developing year	1995	2005	1987	1997	2005
Language	C++	C++	C,C++	C++	JAVA
Simulation type	Discrete event	Discrete event	Discrete event	Discrete event	Discrete event
GUI	Yes	Yes	Yes	Yes	Yes
Scalability	Poor	Moderate	Moderate	Moderate	High
Set up and ease of use	Moderate	Moderate	Difficult	Moderate	Difficult
protocol	802.11 MAC,PHY	802.11 PHY	802.15.4/Zig bee	802.11OFDM PHY,ethernet	802.11 MAC
Network model	parallel	Parallel	Queuing	Queuing and parallel	Parallel
Internet stack	TCP,UDP, IPV4,IPV6	TCP,UDP, IPV4,IPV 6	UDP, ARP	TCP/IP,IPV6	TCP,UDP
Platform	Linux	Linux	Window, linux	Winow, Linux	Window

IV. CONCLUSION

We have define different type of traffic and network simulators. When we define VANET scenario we require both type of simulation. Traffic simulation define nodes and edges configuration i.e road topology, and traffic light,obstacles and vehicles.network simulators define the communication environment between the VANET nodes(vehicles).hence, two study both type of simulator is necessary for VANET.In this paper we define different type of network and traffic simulator to define the ease of compatibility to each other so that a good simulation result is find out for VANET.

REFERENCES

1). Francisco J. Martinez 1* Chai Keong Toh2 Juan-Carlos Cano 3 "A survey and comparative study of simulators forvehicular *ad hoc* networks (VANETs)" in wireless computing and mobile computing(2011).

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 5, May 2015

2). Toh C-K. *Ad Hoc Mobile Wireless Networks: Protocolsand Systems*. Prentice Hall: Upper Saddle River, NJ,USA, 2001.

3).Cavin D, Sasson Y, Schiper A. On the accuracy of MANET simulators. In *Proceedings of the 2nd ACM International Workshop on Principles of Mobile Computing*. ACM: New York, NY, USA, 2002; 38–43.

4). Boangoat Jarupan, Eylem Ekici, ―A survey of cross-layer design for VANETs‖, Ad Hoc Networks 9 (2011) 966–983. Journal homepage: www.elsevier.com/locate/adhoc

5). STRAW - STreet RAndom Waypoint - vehiclarbmobility model for network simulations (e.g., car networks), 2008. Available at: http://www.aqualab.cs. northwestern.edu/projects/STRAW/index.php.

6) .Krajzewicz D, Rossel C. Simulation of Urban MObility (SUMO). German Aerospace Centre, 2007. Available at: http://sumo.sourceforge.net/index.html

7). Michael Behrisch, Laura Bieker, Jakob Erdmann, Daniel Krajzewicz Institute of Transportation Systems SUMO – Simulation of Urban MObility (2011)

8). MOVE (MObility model generator for VEhicular networks): Rapid Generation of Realistic Simulation for VANET, 2007. Available at: http://lens1.csie.ncku. edu.tw/MOVE/index.htm**i**Jérôme Härr

9). University of Karlsruhe, Institute of Telematics Karlsruhe, Germany *haerri@kit.edu*Marco Fiore Politecnico di Torino, Corso Duca degli Abruzzi *fiore@tlc.polito.it*Vehicular Mobility Simulation withVanetMobiSim(2009)

*10)*Trupti G.Nimje S.S.Dorle *A Survey onVarious Mobility Models to improve Realistic Simulation and Accuracy of IVC Protocols (2013) in IEEE conference.*

11).David R. Choffnes Fabiàn E. Bustamante STRAW - An Integrated Mobility and Traffic Model forVANETs

12). FreeSim, 2008. Available at: http://www. freewaysimulator.com/

13).Jeffrey Miller Ellis Horowitz FreeSim – A Free Real-Time Freeway Traffic Simulator.

14). .Mrs. Saba Siraj , Mr. Ajay Kumar Gupta , Mrs Rinku-Badgujar Network Simulation Tools Survey *International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 4, June 2012*

15).Fall K, Varadhan K. ns notes and documents. The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, February 2000. Available at: http://www.isi.edu/ nsnam/ns/ns-documentation.html

16) Ricardo Fernandes and Michel Ferreira Instituto de Telecomunicac¸ oes, DCC/FC - University of PortoRua Campo Alegre, 1021/1055, 4169-007 Porto, PortugalScalable VANET Simulations with NS-3(2012).

17).András Varga Rudolf Hornig OpenSim Ltd. Budapest, HungaryAN OVERVIEW OF THE OMNeT++ SIMULATION ENVIRONMENT(2008).

18). JiST/SWANS: Java in Simulation Time/Scalable Wireless Ad hoc Network Simulator, 2004. Available at: http://jist.ece.cornell.edu/.

19)Rimon Barr Zygmunt J. Haa Robbert van Renesse Cornell University, Ithaca NY JiST: Embedding Simulation Time into a Virtual Machine.

20).OPNET Technologies, 2008. Available at: http://www.opnet. COM. Digests 9th Annual Conf. Magnetics Japan,