Volume: 5 Issue: IX Month of publication: September 2017
DOI:

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue IX, September 2017- Available at www.ijraset.com

Simulators for VANET

Venkatamangarao Nampally[1], M. Raghavender sharma[2], K. R. Balaji[3]
[1]Department of Computer Science, University College of Science, Osmania University, Hyderabad, Telangana,India [2]Department of Statistics, University College of Science, Saifabad, Osmania University, Hyderabad, Telangana,India
[3]Department of Network Systems and Information Technology, University of Madras, Guindy Campus, Chennai, Tamil Nadu, India

Abstract*: VANET system would behave under different conditions.It is known that networking and mobility component are required to form a VANET. Continuous increased of on-road vehicles has also increased the possibility of accidents, most of it caused by human errors. Hence, a solution has been proposed, which is to make vehicles more intelligent through research on vehicular ad hoc networks (VANETs). Researchers and developers have built VANET simulation software to allow the study and evaluation of various media access, routing, and emergency warning protocols. MATLAB and Simulation of Urban Mobility (SUMO) are further used for VANET simulation respectively. At the same time, TraCI4Matlab, an Application Programming Interface (API) developed in MATLAB that allows communication between any applications written in MATLAB and SUMO simulator is also discussed. Absence of road traffic safety takes a toll of precious human lives and poses a dire threat to our environment as well. Moreover, we present results of our proposed algorithm secure key exchange in NS2 simulator.
***Keywords:* VANET, Simulator, Taxonomy, Attraction point, Lane Changing, Topology and GUI.**

I. INTRODUCTION

Vehicular Ad-hoc Networks (VANETs) represent a rapidly emerging, particularly challenging class of Mobile Ad Hoc networks (MANETs). On a highway a vehicle cannot currently predict the speed of other vehicles. However, with use of sensor, computer and wireless communication equipment, speed could be predicted and a warning message sent every 0.5 seconds could limit the risk of potential accidents. An automated tool called simulation can imitate the protocol and yield a similar result to that of the real world. Simulator tool has been preferred over outdoor experiment because it simple, easy and cheap. VANET requires that a traffic and network simulator should be used together to perform this test. Prior to the implementation of VANETs on the roads, realistic simulations of VANETs using simulators are necessary. It is a technique whereby a software program models the behavior of a network either by calculating the interaction between the different network entities (routers, switches, nodes, access points, links etc.). Users customize the simulator to fulfill their specific analysis needs. Most simulators use discrete event simulation i.e. the modelling of systems in which state variables change at discrete points in time. The behavior of the network and the various applications and services it supports can then be observed in a test lab; various attributes of the environment can also be modified in a controlled manner to assess how the network / protocols would behave under different conditions. We have classified existing VANET simulation software into three different categories. They are (a) vehicular mobility generators, (b) network simulators, and (c) VANET simulators. This paper presents a comprehensive study and comparisons of the various publicly available VANET simulation software and their components and we see secure key exchange results when applying in VANET system in NS2 simulator.

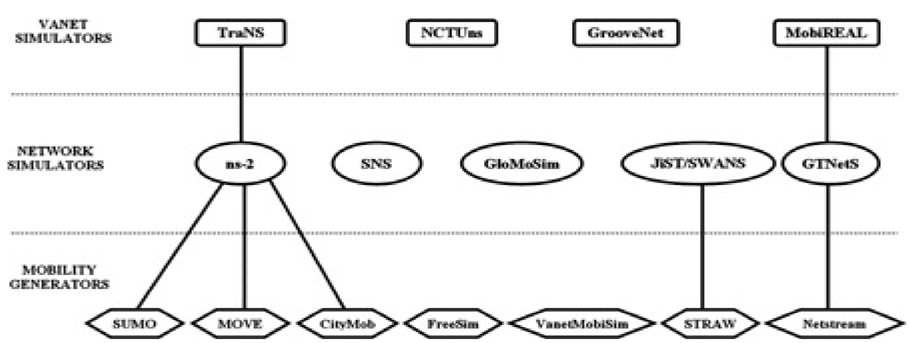


Fig. 1 Categorization of simulation software in VANET

Various approaches that has been undergone by the research community in order to develop realistic mobility models adapted to vehicular traffic are; 1. Mobility models: 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 inputs of the mobility generator include the road model, scenario parameters (i.e., maximum vehicular speed, rates of vehicle arrivals and departures, etc.). The output of the trace details the location of each vehicle at every time instant for the entire simulation time and their mobility profiles. Globally, the development of modern vehicular mobility models may be classified in four different classes: Synthetic Models wrapping all models based on mathematical models, Traffic Simulators based Models, where the vehicular mobility models are extracted from a detailed traffic simulator, *Survey-based Models* extracting mobility patterns from surveys, and finally Trace based Models, which generate mobility patterns from real mobility traces. The taxonomy of classification is illustrated in Fig.

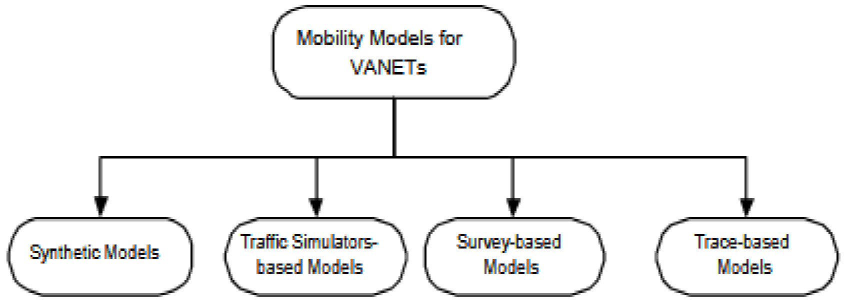


Fig. 2 Taxonomy of mobility models for VANET

A. Mobility Models in VANET

Vehicular mobility models are usually classified as either microscopic or macroscopic. When focusing on a macroscopic point of view, motion constraints such as roads, streets, crossroads, and traffic lights are considered. The microscopic approach, instead, focuses on the movement of each individual vehicle and on the vehicle behavior with respect to others. Yet, in order to be used by the networking community, those models need to be made available to network simulators. Initially, mobility was seen by network simulators as random perturbations from optimum static configurations. Then, in order to give some control to the user on the mobility patterns, network simulators became able to load mobility scenarios. Network simulators perform detailed packet-level simulation of source, destinations, data traffic transmission, reception, background load, route, links, and channels. There is virtually no limitation to the complexity of the models handled by those simulators, loading scenarios extracted from traffic simulators, or complex synthetic models for instance. VANET simulators provide both traffic flow simulation and network simulation. A sound simulation can produce strikingly similar results to that of real world. Construction of a simulation therefore seems inevitable for VANET.

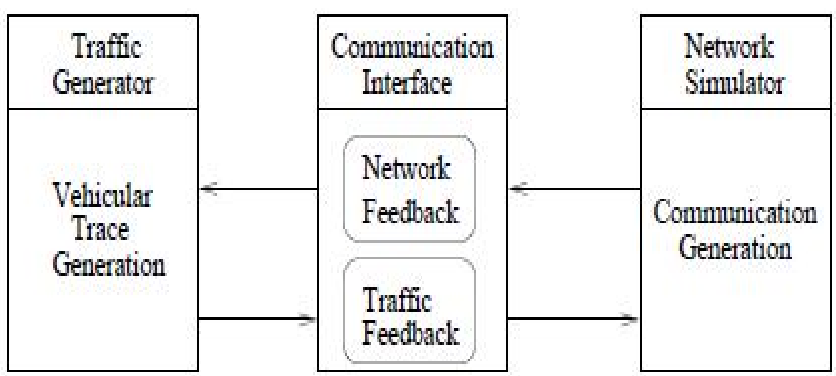


Fig. 3 Interaction between traffic generator and network simulator

B. Various Simulators

In order to get best performance of a network, available various network simulators are

1) VanetMobiSim
2) SUMO
3) MOVE
4) STRAW

5) FreeSim
6) CityMob
7) SNS
8) NCTUn
9) MobiREAL
10) iST/SWANS
11) GrooveNet
12) NHTSA (National Highway Traffic Safety Application) simulator
13) CAN
14) NS2
15) .GLOMOsim
16) . QualNet

Besides, these mainly used simulators are TRANSIM, SIS-CORSIM, Paramics, Carisma, VISSIM, OPNET, Netstream, GTNetS, TraNS and. AIMSUN.

1) *VanetMobiSim* : It is an extension of the CANU Mobility Simulation Environment (CanuMobiSim) which focuses on vehicular mobility, and features realistic automotive motion models at bothmacroscopic and microscopic levels. At the macroscopic level, VanetMobiSim can import maps fromthe US Census Bureau topologically integrated geographic encoding and referencing (TIGER) database,or randomly generate them using Voronoi tessellation. The TIGER/Line files constitute a digital database of geographic features, such as roads, railroads, rivers, lakes, and legal boundaries, covering the entire United States. VanetMobiSim adds support for multi-lane roads, separate directional flows, differentiated speed constraints and traffic signs at intersections. At the microscopic level, it supports mobility models such as Intelligent Driving Model with Intersection Management (IDM/IM), Intelligent Driving Model with Lane Changing (IDM/LC) and an overtaking model (MOBIL), which interacts with IDM/IM to manage lane changes and vehicle accelerations and decelerations, providing realistic car-to-car and car-to-infrastructure interactions. VanetMobiSim is based on JAVA and can generate movement traces in different formats.

2) *SUMO (Simulation of Urban MObility):* It is an open source, highly portable, microscopic road traffic simulation package designed to handle large road networks. Its main features include collision free vehicle movement, different vehicle types, single-vehicle routing, multi-lane streets with lane changing, junction-based right-of-way rules, hierarchy of junction types, an openGL graphical user interface (GUI), and dynamic routing. SUMO can manage large environments, i.e., 10 000 streets, and it can import many network formats such as Visum, Vissim, ArcView, or XMLDescriptions. Thus, by combining SUMO and openstreetmap.org, we can simulate traffic in different locations of the globe.

3) *MOVE (MObility model generator for Vehicular networks)* : t rapidly generates realistic mobility models for VANET simulations. MOVE is built on top of SUMO. The output of MOVE is a mobility trace file that contains information of realistic vehicle movements which can be immediately used by popular network simulation tools such as ns-2 or GloMoSim. In addition, 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.

4) *STRAW (STreet RAndom Waypoint) :* It provides accurate simulation results by using a vehicular mobility model on real US cities, based on the operation of real vehicular traffic. STRAW's implementation is written for the JiST/SWANS discrete-event simulator, and its mobility traces cannot be directly used by other network simulators, such as ns-2. STRAW is part of the C3 (Car-to-Car Cooperation) project. A more realistic mobility model with the appropriate level of detail for vehicular networks is critical for accurate network simulation. The STRAW mobility model constrains node movement to streets defined by map data for real US cities and limits their mobility according to vehicular congestion and simplified traffic control mechanisms.

5) *FreeSim* : is a fully customizable macroscopic and microscopic free-flow traffic simulator that allows for multiple freeway systems to be easily represented and loaded into the simulator as a graph data structure with edge weights determined by the current speeds. Traffic and graph algorithms can be created and executed for the entire network or for individual vehicles or nodes, and the traffic data used by the simulator can be user generated or be converted from real-time data gathered by a transportation organization. Vehicles in FreeSim can communicate with the system monitoring the traffic on the freeways, which makes FreeSim ideal for *Intelligent Transportation System* (ITS) simulation. FreeSim is licensed under the GNU General Public License, and the source code is available freely for download.

6) *CityMob v.2* : CityMob is a ns-2 compatible mobility model generator proposed for use in VANETs. Citymob implements three different mobility models: (a) *Simple Model* (SM), (b) *Manhattan Model* (MM), and (c) realistic *Downtown Model* (DM). In

DM model, streets are arranged in a Manhattan style grid, with a uniform block size across the simulation area. All streets are two-way, with lanes in both directions. Car movements are constrained by these lanes. Vehicles will move with a random speed, within a user-defined range of values. DMmodel also simulates semaphores at random positions (not only at crossings), and with different delays. DM adds traffic density in a way similar to a real town, where traffic is not uniformly distributed.

7) *SNS (Staged Network Simulator)* : The staged simulation technique proposes to eliminate redundant computations through function caching and reuse. The central idea behind staging is to cache the results of expensive operations and reuse them whenever possible. SNS is a staged simulator based on ns-2. On a commonly used *ad hoc* network simulation setup with 1500 nodes, SNS executes approximately 50 times faster than regular ns-2 and 30% of this improvement is due to staging, and the rest to engineering. This level of performance enables SNS to simulate large networks.
8) *NCTUns (National Chiao Tung University Network Simulator) :* It is a high-fidelity and extensible network simulator and emulator capable of simulating various protocols used in both wired and wireless IP networks. Its core technology is based on a novel kernel re-entering methodology. Due to this novel methodology, NCTUns provides many unique advantages that cannot be easily achieved by traditional network simulators such as ns-2 and OPNET. It can be easily used as an emulator since it supports seamless integration of emulation and simulation. It uses Linux TCP/IP protocol stack to generate high-fidelity simulation results. It can run any real-life UNIX application program on a simulated node without any modifications. Supported networks include Ethernet-based fixed Internet, IEEE 802.11b wireless LANs, IEEE 802.11e QoS wireless LANs, IEEE 802.16d WiMAX wireless networks, DVBRCS satellite networks, wireless vehicular networks for Intelligent Transportation Systems (including V2V and V2I), multi-interface mobile nodes for heterogeneous wireless networks, IEEE 802.16e mobile WiMAX networks, IEEE 802.11p/1609WAVEwireless vehicular networks, etc. NCTUns supports parallel simulations on multicore machines. By using an innovative parallel simulation approach, it supports parallel simulations for fixed networks on multi-core machines. It also provides a highly integrated and professional GUI environment that can help a user too quickly: (1) draw network topologies, (2) configure the protocol modules used inside a node, (3) specify the moving paths of mobile nodes, (4) plot network performance graphs, (5) play back the animation of a logged packet transfer trace, etc. All of these operations can be easily, intuitively, and quickly done with the GUI. Its main drawback is that NCTUns requires Fedora 9 Linux distribution to be installed.
9) *MobiREAL* : It provides a new methodology to model and simulate realistic mobility of nodes and evaluate MANET applications. It is a network simulator that can simulate realistic mobility of humans and vehicles, and allow the changing of their behavior depending on a given application context. MobiREAL can easily describe mobility of nodes using C++. It adopts a probabilistic rule based model to describe the behavior of mobile nodes, which is often used in cognitive modeling of human behavior. MobiREAL Animator dynamically visualizes node movement, connectivity states, and packet transmission. This enhances the understanding of simulation results intuitively. Mobility of nodes is simulated in the Behavior Simulator. Also an algorithm for collision avoidance among pedestrians is implemented. Traffic congestion of vehicles can also be modeled. Using MobiReal, it is possible to simulate a mixture of various mobility models concurrently. MobiREAL cause other traffic simulators to provide vehicular mobility.
10) *JiST/SWANS* : JiST is a high performance discrete event simulation engine that runs over a standard Java virtual machine. It is a prototype of a new general purpose approach to building discrete event simulators that unifies the traditional systems and language-based simulator designs. It outperforms existing highly optimized simulation engines both in time and memory consumption. Simulation code that runs on JiST need not be written in a domain specific language invented specifically for writing simulations, nor must it be littered with special purpose system calls and 'call backs' to support runtime simulation. Instead, JiST converts an existing virtual machine into a simulation platform, by embedding simulation time semantics at the byte-code level. SWANS is a scalable wireless network simulator built on top of the JiST platform. It was created primarily because existing network simulation tools are not sufficient for current research needs. SWANS contains independent software components that can be composed to form complete a wireless network or sensor network. Its capabilities are similar to ns-2 and GloMoSim, but SWANS is able of simulating much larger networks. SWANS leverages the JiST design to achieve higher simulation throughput, lower memory requirements, and run standard Java network applications over simulated networks. SWANS can simulate networks that are one or two orders of magnitude larger than what is possible with GloMoSim and ns-2, respectively, using the same amount of time and memory, and with a same level of detail.
11) *GrooveSim :* It was the first tool created for evaluation of VANET performance mainly motivated by vehicular traffic flow and forecasting. GrooveSim was coded in C++ and Matlab provides GUI for drawing structures and graphs. GrooveSim could operate in five different modes: predetermined, on road, simulation, hybrid, research.

12) *NHTSA (National Highway Traffic Safety Application) :* VANET estimation and focused on a global perception of VANET performance. The NHTSA simulator was designed with networking research in mind and was built on the top of NS-2 simulator. The simulator is platform-independent and is capable to run on both Win32 as well as Linux. It has strong GUI support implemented by C++. The platform is very scalable and flexible for researchers to alter the configuration according to the requirements. A platform based on simulation results from simulation tools and a software prototype called FleetNet Demonstrator FND. The development of this software was aimed to state the problems found in inter-vehicular communication and realistic evaluation of VANET. The focus of this project was primarily on how mobility is achieved with position based routing protocols.
13) *CanuMobiSim CANU (Communication in Ad Hoc Networks for Ubiquitous Computing):* This mobility simulator is a java-based application with a graphical user interface (GUI). The CanuMobiSim project started in Germany at the University of Stuttgart. This tool can generate many mobility models like smooth mobility model, pedestrian, graph walk, fluid traffic, activity based mobility models. CanuMobiSim extracts topology files from Geographical Data Files (GDF) or from user-defined graphs. During simulation, CanuMobiSim takes micro-mobility into consideration and generates traces that are used by NS-2 and GlomoSim. CanuMobiSim is a complex traffic simulator, in which path calculation is based on the basis of Dijkstra's algorithm, also known as the shortest Path First (SPF) algorithm, and generates trips based on how users create different motion patterns. CanuMobiSim provides us with a very good quality reliable solution for generating mobility traces for network simulator.
14) *NS2:* In 1989, NS-2 appeared as a network simulator that provides significant simulation of transport, routing, and multicast over-wired and wireless networks. Ns-2 code is written either in C++ and OTCL and is kept in a separate file that is executed by OTCL interpreter, thus generating an output file for NAM (Network animator). It then plots the nodes in a position defined by the code script and exhibits the output of the nodes communicating with each other. It is packaged with a bundle of rich libraries for simulating wireless networks. All the mobile nodes in NS-2 quickly assume that they are the part of Ad hoc network and the simulation mobile nodes connected with infrastructure networks are not really possible. For simulating a wireless node the physical layer, the link layer and MAC (media access control) protocol are all included at the same time.
15) *GlomoSim :* GlomoSim (Global Mobile Information System Simulator) is a second most popular network simulator after NS-2. It was developed in California, USA mainly targeted towards wireless network simulation. GlomoSim was coded in Parsec and all new protocols must be defined in Parsec as well. it includes a java based front end as well. GlomoSim has the ability to run on SMP (shared-memory symmetric processor: memory simultaneously accessible by all programs) and helps to divide the network into separate modules each running as a distinct process. Hence, reduce the load on CPU by dividing its workload. Because of this extraordinary feature of multi-tasking, GlomoSim is able to simulate tens of thousands of nodes in single simulation. GlomoSim follows OSI layer model and support different protocols and models at each layer. GlomoSim has two ray and free-space radio propagation models. GloMoSim was designed to support millions of nodes just as a single simulation is due to parallelism technique.
16) *Qualnet:* QualNet (Quality Networking) is network evaluation software and is entirely modelled as a finite state machine. It is written purely in C++ and can run on a variety of operating systems like UNIX, Windows, MAC and Linux. QualNet is equipped with an extensive range of libraries for simulating a variety of networks like Wi-Fi, Sensor networks, MANET and WiMAX etc. The simulation could be performed with a powerful 3D visualization tool along with a QualNet Analyzer. It is a powerful simulation tool that can support simulation of 500 to 20,000 nodes. Similar to TCP/IP, QualNet is engineered on a layered architecture comprising a one can use an evaluation version of QualNet for testing purposes.

C. Strengths of Among Three Models

We can see the strengths of three simulation software models in a pictorial form as below:

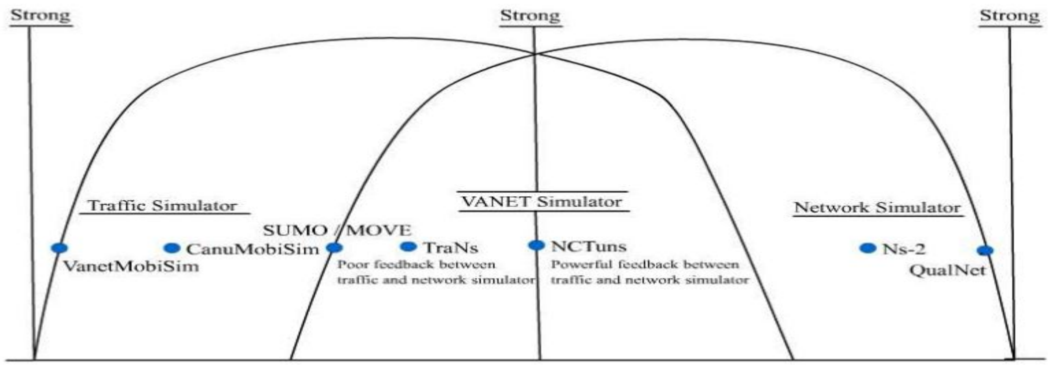


Fig. 4 Strengths among simulation software models

II. RELATED WORK

Fiore wrote a complete survey of models falling into this category [1].MoVes [2] is an embedded system generating vehicular mobility traces and also containing a basic network simulator. The major asset of this project is its ability to partition the geographical area into clusters and parallelize and distribute the processing of the tasks from them, which improves the simulation performance. Although the mobility model reaches a sufficient level of detail. The vehicular traffic simulator is a synthetic model integrating basic microscopic motions where drivers may be in one of the following four modes: *free driving*, *approaching*, *following*, *braking*. MOVE [3] contains a single graphical user interface for the configuration the mobility modeling and network simulation. However, MOVE does not itself include a network simulator, but simply parses realistic mobility traces extracted from a micro motion model SUMO [4] into a network simulator dependent input trace format, then generates the appropriate scripts to be loaded by the network simulator. No interaction is therefore possible between the network simulator and the mobility model. A different approach, taken by Prof. Fujimoto and his group in Georgia Tech [5] is to generate a simulation infrastructure composed of two independent commercial simulation packages running in a distributed fashion over multiple networked computers. They federated a validated traffic simulator, CORSIM, with a state of the art network simulator, QualNet, using a distributed simulation software package called the Federated Simulations Development Kit (FDK) [6] that provides services to exchange data and synchronize computations. In order to allow direct interaction between the two simulators, a common message format has been defined between CORSIM and QualNet for vehicle status and position information. During initialization, the transportation road network topology is transmitted to QualNet. Once the distributed simulation begins, vehicle position updates are sent to QualNet and are mapped to mobile nodes in the wireless simulation. A similar solution has been taken by a team from UC Davis [7]. They developed a simulation tool federating the network simulator Swans and a synthetic traffic model. The complex vehicular flows are based on the Nagel and Schreckenberg model, extended to include lane changing in highway scenarios. The network simulator and the traffic simulator interacts with each other's by means of specific input and output messages. Authors in [8] proposed *AutoMesh*, a realistic simulation framework for VANET. It is composed of a set of modules controlling all parts of a realistic simulation. It includes a *Driving Simulator Module*, a *Radio Propagation Module*, and *a Network Simulator Module*, all interlinked with feedback in order that any alteration made in one module influences the other modules. Another promising approach is called *TraNS* [9] and also aims at federating a traffic simulator SUMO and a network simulator ns2. Using an interface called *Interpreter*, traces extracted from SUMO are transmitted to ns-2, and conversely, instructions from ns-2 are sent to SUMO. TraNS will be extended to handle other network simulators such as Swans or Nab in the future. A similar project called MSIE [10] has been developed but using VISSIM instead of SUMO. This project is also more complete, as it proposes to interlink different simulators for traffic simulation, network simulation and application simulation. The major actual limitation is the communication latency between the different simulators, and the expensive price of VISSIM. Besides, the interlinking interface itself is also not freely available at this time. We point out that many realistic traffic simulation tools, such as PARAMICS [11], CORSIM [12], VISSIM [13] or TRANSIMS [14] have been developed to analyze vehicular mobility at both microscopic and macroscopic level with a very high degree of detail. When mobility was first taken into account in simulation of wireless networks, several models to generate nodes mobility patterns were proposed. The Random Waypoint model, the Random Walk model, the Reference Point Group (or Platoon) model, the Node Following model, the Gauss-Markov model, just to cite the most known ones, all involved generation of random linear speed-constant movements within the topology boundaries. Further works added pause times, reflection on boundaries, acceleration and deceleration of nodes. Simplicity of use conferred success to the Random Waypoint model in particular, however, the intrinsic nature of such mobility models may produce unrealistic movement patterns when compared to some real world behavior. Despites, random models are still widely used in the study of Mobile Ad-hoc Networks (MANETs). As far as Vehicular Ad-hoc Networks (VANETs) are concerned, it soon became clear that using any of the aforementioned models would produce completely useless results. Consequently, the research community started seeking more realistic models. Recently, new open-source tools became available for the generation of vehicular mobility patterns. Most of them are capable of producing traces for network simulators such as *ns-2* [15], *GloMoSim* [16], *QualNet* [17], or *OpNet* [18]. The BonnMotion tool [19] and the IMPORTANT tool [20], and implement mostof the random mobility models presented in [21], including the Manhattan model. This model restrict nodes macro-mobility on a grid, while the micro-mobility contains a *Car Following Model*. The BonnMotion does not consider any micromobility.When related to our proposed framework, we can easily see that the structure of both tools is definitely too simple to represent realistic motions, as they only model basic motion constraints and hardly no micro-mobility. The GEMM tool [63] is an extension to BonnMotion's and improves its traffic generator by introducing the concepts of human mobility dynamics, such as *Attraction Points (AP)*, *Activity*, or *Roles*. Attraction points reflect a destination interest to multiple people, such as grocery stores or restaurants. Activities are the process of moving to an attraction point and staying there, while roles characterize the mobility tendencies intrinsic to different classes of people. While the basic concept is interesting, its implementation in the tool is limited to a simple enhanced RWM between APs. It however represents an initial attempt to improve the realism of mobility models by considering human mobility dynamics. The MONARCH project [22] proposed a tool to extract road topologies from real road maps obtained from the TIGER [23] database. Another important microscopic mobility simulator is the SHIFT Traffic Simulator [24]. It has been developed by the PATH Project at the UC Berkley, and is now a well-established micro-mobility simulator that generates the trajectories of vehicles driving according to validated models on realistic road networks. More specifically, SHIFT is a new programming language with simulation semantics and was used in *SmartAHS* as means of specification, simulation and evaluation framework for modeling, control and evaluation of Automated Highway Systems (AHS). The major limitation of this simulator is its limitation to the modeling of segments of highways and its lack of complete topology modeling. The CARISMA traffic simulator [25] is a realistic simulator containing microscopic and macroscopic features. Recently, new approaches appears in realistic scalable simulations of vehicular mobility. In [26], the authors created MoVes, a complex mobility generator on top of Artis [27], a scalable distributed simulation middleware. MoVes features cars following models, drivers' characterization, intersection management and includes a parser module to include GPS maps using the GPS TrackMaker program [28].

III.METHODS

A. Taxonomy Criteria

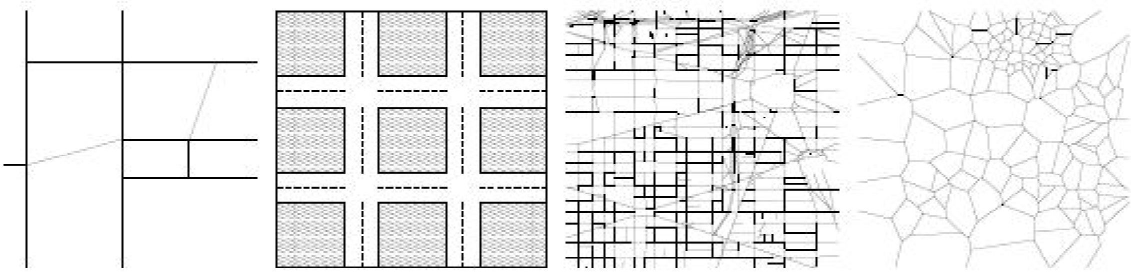
We defined the criteria based on which to generate the taxonomy. The criteria fall in three categories: Macro-mobility, Micro-mobility, and Simulator Related.

1) *Macro-mobility Criteria:* When considering macro-mobility, we do not only take into account the road topology, but also include trip and path generation, or even the effects of points of interests, which all influence vehicles movement patterns on the road topology. We therefore define the following criteria:
   Graph – The macro-motion is restricted to move on a graph. *Initial and Destination Position* – The positions may be either random, random restricted on a graph or based on a set of attraction or repulsion points.
   Trip Generation – A trip may be randomly generated between the initial and destination points, or set according to an activity sequence.
   Path Computation – Provides the algorithms used to generate the path between the points contained on the trip.
   Velocity – The simulated velocity may be uniform, smooth or road dependent.
2) *Graphs:* The selection of the road topology is a key factor for obtaining realistic results when simulating vehicular movements. Indeed, the length of the streets, the frequency of intersections, or the density of buildings can greatly affect important mobility metrics such as the minimum, maximum and average speed of cars, or their density over the simulated map. We categorize the graphs by the following criteria:
   User defined – The road topology is specified by listing the vertices of the graph and their interconnecting edges.
   Random – A random graph is generated, which are often *Manhattan-grid*, *Spider*, or *Voronoi* graphs.

Maps – The road topology is extracted from real maps obtained from different topological standards, such as *GDF*, *TIGER*, or *Arcview*.

Multi-lane – The topology includes multi-lanes, potentially allowing lane changes, or not.

We show examples of the possible topologies in Fig.



(a) User-defi ned topology (b) Manhattan-grid map topology (c) TIGER map topology (d) Clustered Voronoi

Fig. 5 Graph criteria for VANET

3) *Attraction Points:* Attraction or Repulsion points are particular source or destination points that have a potentially attractive or repulsive feature. For instance, for a weekly morning, residential areas are repulsion points and office builds are attraction points, as a large majority of vehicles are moving from the former and to the latter. We depict the use of attraction points on a user-defined graph in Fig. 8, where a round is for the entry/exit points of high-speed roads (thick lines), and a square for the entry/exit points of normal-speed roads (thin lines).

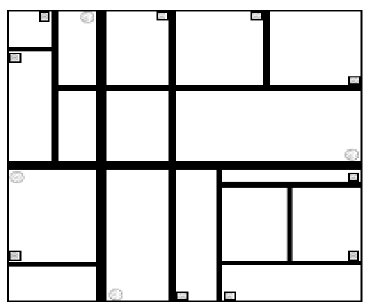


Fig. 6 Attraction points on a user defined graph for VANET

4) *Activity-based Trips:* Activity sequences generation is a further restriction in vehicles spatial and temporal distributions. A set of start and stop points are explicitly provided in the road topology description, and cars are forced to move among them. In particular, multiple sets of points of interest can be specified, along with the probability matrix of a vehicle switching from one set to another. Fig. shows an activity sequence generated from a first order Markov chain between two categories of attractions points.

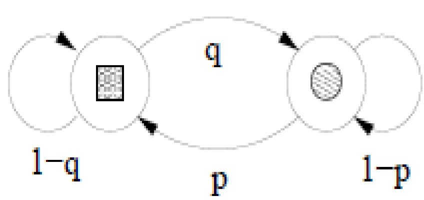


Fig. 7 Activity-based sequence between the attraction points for VANET

B. *Micro-mobility Criteria*

In the proposed taxonomy, the micro-mobility aspect uses the following criteria: *Human Mobility Patterns* – The car's internal motion and its interactions with other cars may be inspired from human motions described by mathematical models such as Car Following, or not. *Lane Changing* – Describes the kind of overtaking model implemented by the model, if any. *Intersections* – Describes the kind of intersection management implemented by the model, if any. In VANET for a larger coverage of the different microscopic mobility models are:

1) *Car Following Models:* The car following models is a class of microscopic models that adapts a following car's mobility according to a set of rules in order to avoid contact with the lead vehicle. A general schema is illustrated in Fig.

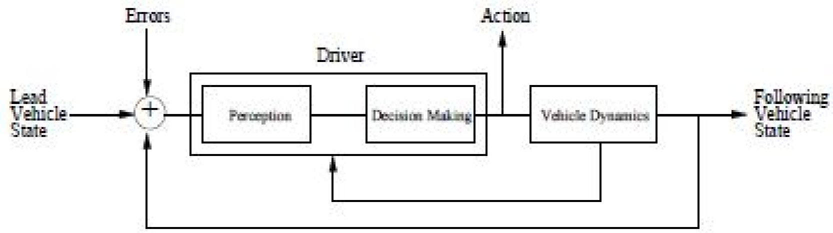
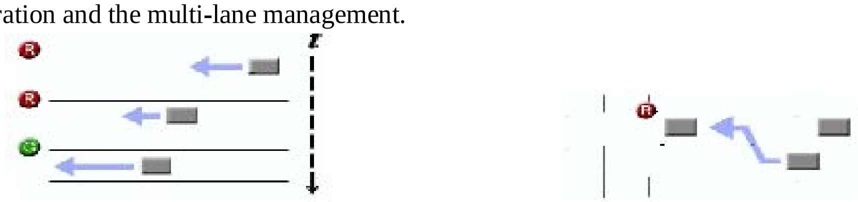


Fig. 8 General schema for car following models for VANET

Car Following Models in five classes: GHR Models, Psycho-Physical Models, Linear Models, Cellular Automata, Fuzzy Logic Models. A description of the differences between those models is out of scope of this paper. We only list here the widely used models in traffic simulations; Krauss Model (KM), Nagel and Schreckenberg Model (NSHR), Wiedeman Psycho-Physical Model (Psycho), General Motors Model (GM) , Gipps Model (GP) and Intelligent Driver Model (IDM)

2) *Lane Changing Models:* Despite the large attention given to the driving tasks in general (such as Car Following Models), much less attention has been directed to lane changing. Modeling lane changing behavior is a more complex task. Indeed, it actually includes three parts: the need of lane changing, the possibility of lane changing, and the trajectory for lane changing. Each part is important to generate realistic lane changing models. And unlike car following models, it also needs to consider nearby cars and traffic flow information. Most of the models are based on a *Gap Acceptance* threshold or a set of rules. But recent approaches, also considered forced merging, behavior aspects or game theory. Lane changing is not widely considered in open vehicular mobility models.
3) *Intersection Management:* Intersection management adds handling capabilities to the behavior of vehicles approaching a crossing. In most cases, two different intersection scenarios are considered: a crossroad regulated by stop signs, or a road junction ruled by traffic lights. Nevertheless, all intersection management technics only act on the first vehicle on each road, as the car following model automatically adapts the behavior of cars following the leading one. The most basic ones consider intersections as obstacles and abruptly stop, yet more complex ones, such as the IDM IM and IDM LC, smoothly stop at stop-based crossing, or acquire the state of the semaphore in a traffic light controlled intersection. If the color is green, passage is granted and the car maintains its current speed through the intersection. If the color is red, crossing is denied and the car is forced to decelerate and stop at the road junction. Fig. illustrates the IDM IM behavior when approaching an intersection with respect to the deceleration and the multi-lane management.



(a) Acceleration management

(b) Multi-lane management

Fig. 9 Insertion management for VANET

C. Taxonomy of Synthetic Vehicular Models

We cannot include the Trace-based nor the Survey-based models as they have been obtained from real mobility and do not fall in the taxonomy. We include some Traffic Simulator-based models if they are based on freely available traffic simulators. However, all the aforementioned softwares are distributed under commercial licenses, a major impediment to adoption by the academic research community. With the exception of few teams that developed parsers or federated a realistic traffic simulation tool with a network simulator, these tools have been originally designed for traffic analysis and not for generation of movement traces usable by networking simulators. Furthermore, the presence of copyrights impedes the modification/extension of the sources when particular conditions, not planned by the original software, have to be simulated. For such reasons, we will not consider these tools in the following, their scope being very different from VANET mobility simulators which are intended.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue IX, September 2017- Available at www.ijraset.com

IV. SIMULATION RESULTS

A. Parameters

We implemented secure key exchange algorithm in NS2 simulator. In order to implement this algorithm we used some parameters. These parameters are given in below table. And some output screenshots are given below.

Table 1: Simulation parameters

Parameters	Values
Network Size	4000m x 4000m
Number of Vehicle Nodes	112
Packet_ Size	1000bytes
Simulation Time	10 sec.
MAC protocol	IEEE 802_11
Number of AS	1
Number of LEs	16
Number of OBUs	86

B. Simulation Output

1) *Network Simulation in NS2 simulator*

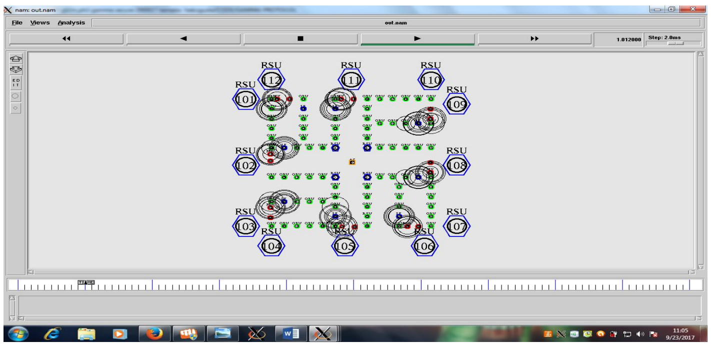


Fig. 10 Simulation output for VANET

2) *Performance Result:* It is defined as ratio of performance vehicles taken into consideration and non-performance vehicles consideration in a channel over the simulation time. Mathematically it can be written as:

Performance of vs. Trust vs. Non-Trust = Trust Vehicles / Non-Trust Vehicles

Where N is the number of bits bought by all destinations.

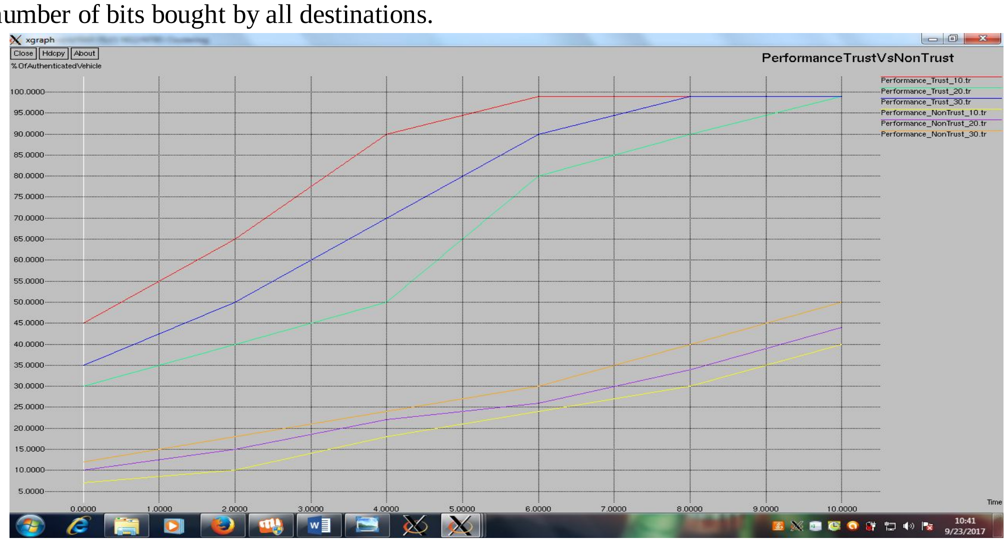


Fig. 11 Performance result for VANET

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue IX, September 2017- Available at www.ijraset.com

3) *Throughput***:** is defined as rate of successful message delivery over a channel or aggregate number of packets delivered over the simulation time. Mathematically it can be written as:
Throughput=N/100
Where N is the number of bits bought by all destinations.

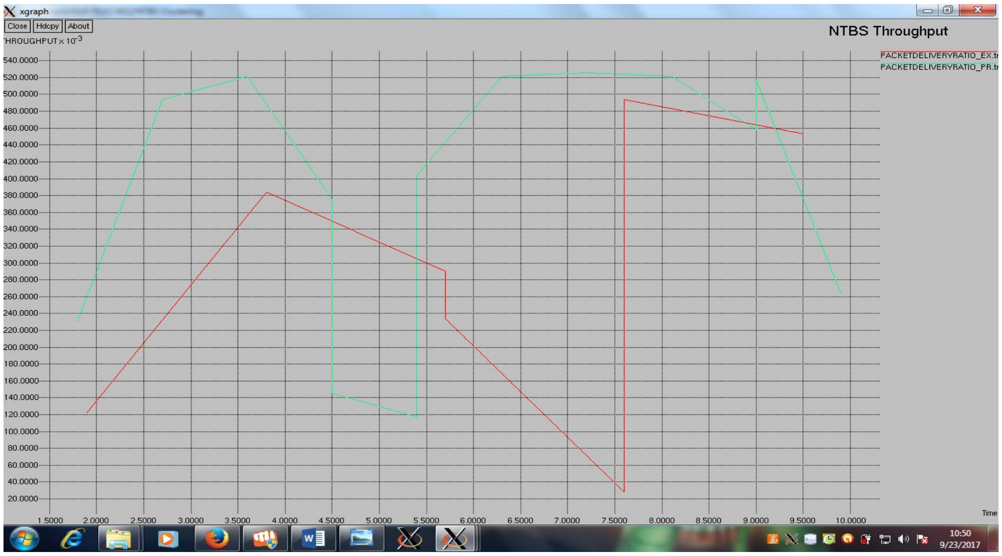


Figure 19 NTBS throughput in VANET system

4) *End-to-End Delay:* It is defined as time taken for a packet to be transmitted successfully across a network from source to destination. Mathematically it is defined as:

$$AED = \frac{\sum_{i=0}^{n}(t_i(r) - t_i(s))}{n_{pr}}$$

Where AED is average end to end delay $t_i(r)$ is the receiving time of packet I by the destination node, $t_i(s)$ is the sending time of packet i by the source node and n_{pr} is the total number of packets received.

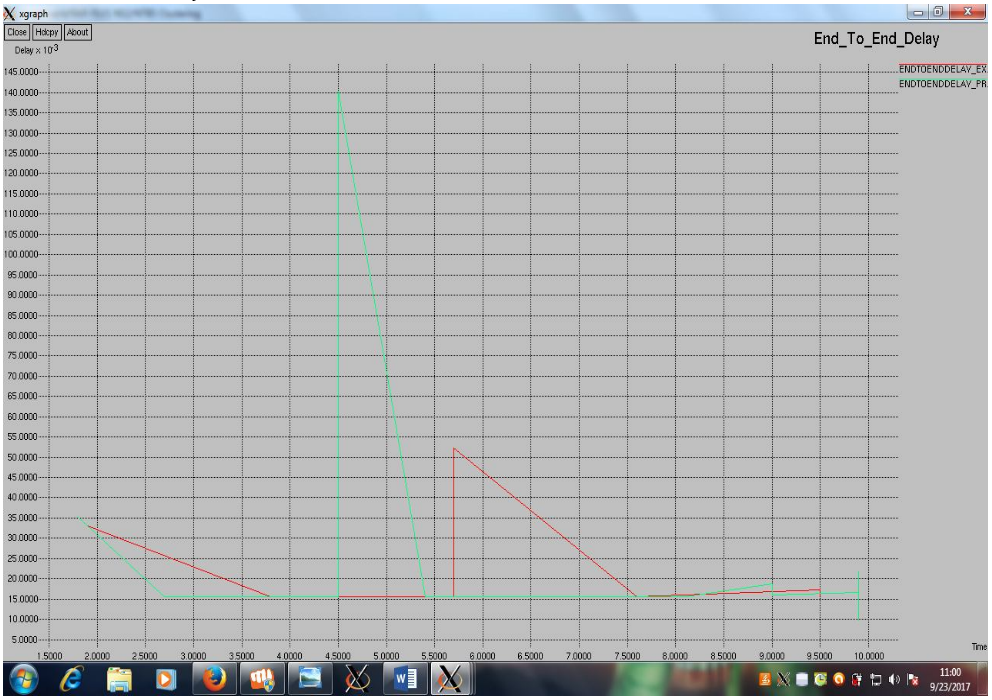


Figure 20. NTBS end-to-end delay in VANET system

V. CONCLUSION

As a prospective technology, Vehicular Ad Hoc Networks (VANETs) have recently been attracting an increasing attention from both research and industry communities. One of the fastest growing field of interest in VANETs is safety, where communications are exchanged in order to improve the driver's responsiveness and safety in case of road incidents. VANETs characteristics are a higher mobility and a limited degree of freedom in the mobility patterns. Such particular features make standard networking protocols inefficient or unusable in VANETs. In this paper, we first presented a framework which should be followed for the generation of realistic vehicular mobility patterns, then we disserted on the different approaches in vehicular mobility modeling and proposed a classification of vehicular mobility models according to the technics used for their generation. We finally described the most popular models available to the research community at this time, and provided their detailed taxonomy according to criteria based on natural building blocks required for realistic vehicular mobility modeling. We also described the new trend to interlink traffic and network simulators in order to create a cross layer collaboration between routing and mobility schemes. The aim of this paper is to facilitate the comprehensive understanding of the emerging development of realistic vehicular traffic generators, the different methods, their justifications, and the interlinking with network simulators. This could be a good guideline for people interested in understanding the unique relationship between traffic models and network protocols in vehicular networks. This article also provided a large coverage of the most popular mobility models for vehicular networks, and could thus be a good starting point for people starting in this field or desiring to increase their knowledge in Vehicular Ad Hoc Network simulators.

VI. ACKNOWLEDGMENT

I will be thankful forever to the LORD BAJARANGBALI for his boundless blessings showered on me. I am very grateful and express my heartfelt countless Namaste to most respectable and my M.Phil. guide and supervisor who are Dr. S. Anathi madam ji, B.E.,M.Tech.(IISC),Ph.D., Associate Professor, Department of Network Systems and Information Technology, University of Madras, Guindy Campus, Chennai for their constant support, invaluable and inspiring guidance to the progress of my paper work. Without madam ji, and Sir K. Padmanabhan inspiration, definitely this paper work would not have been possible. I would like to express my heartfelt special thanks to most respectable, emeritus and senior Prof. (ret.) K.Padmanabhan, Former Head, CISL and Emeritus Professor in AC Technology College, Anna University for their kind support to me for carrying out this paper work. I would like to express special thanks to Prof. (ret.) Ramana Murthy M. V., Department of Mathematics & Computer Science, University College of Science, Osmania University, Hyderabad, Telangana, Prof. (ret.) Shankar B., Department of Mathematics, University College of Science, Osmania University, Hyderabad, Telangana and R.K.sir for clarifying my doubts On NS2 software to run on windows 7. And I take this opportunity to express my heartfelt thanks to Dr. K.R.Balaji (Department of NS & IT, University of Madras, Guindy Campus, Chennai) M.Sc., M.Phil., Ph.D., and Arun Ananthanarayanan, UGC research scholar, Department of NS & IT, University of Madras, Guindy Campus, Chennai for giving their constant support and valuable help.
Finally, thank you very much Dept. of NS & IT Scholars!

REFERENCES

[1] Marco Fiore, "Mobility Models in Inter-Vehicle Communications Literature", Technical Report, November 2006.
[2] L.Bononi,M.DiFelice,M.Bertini,E.Croci,"Parallel and Distributed Simulation of Wireless Vehicular Ad Hoc Networks",in proc.of the ACM/IEEE International Symposium on Modeling , Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Torresmolinos, Spain,2006.
[3] F. Karnadi, Z.Mo,K.-C.Lan,"Rapid Generation of Realistic Mobility Models for VANET", in Proc. of the IEEE Wireless Communication and Networking Conference(WCNC'07), March 2007.
[4] SUMO Simulation of Urban Mobility , http://sumo.sourcefor ge.net
[5] H.Wu,J. Lee,M.Hunter, R.M.Fujimoto,R.L.Guensler, and J. Ko,"Simulated Vehicle-to-Vehicle Message Propagation Efficiency on Atlanta' s I-75 Corridor ," in Transportation Research Board Conference Proceedings, Washington D.C., 2005.
[6] The Federated Simulations Development Kit (FDK), http://www.static.cc.gatech.edu/computing/pads/fdk.html
[7] Andrew Chen,et al., "Impact of Transmission Power on TCP Performance in Vehicular Ad Hoc Networks" in Proc.of the 4th IEEE/IFIP Wireless On demand Networks and Services , pp.65-71,Austrian 2007.
[8] Rama Vuyyuru,Kentaro Oguchi,"Vehicle-to-Vehicle Ad Hoc Communication Protocol Evaluation using Simulation Frame work",in Proc. of the 4th IEEE/IFIP Wireless On demand Networks and Services , pp.100-106, Austria 2007.
[9] Traffic and Network Simulation Environment http://wiki.epfl.ch/trans/
[10] Multiple Simulator Interlinking Environment for C2CC in VANETs, http://www.cn.uni-duesseldorf.de/projects/MSIE
[11] Paramics: Microscopic TrafficSimulation,http://www.paramics-online.com/
[12] CORSIM:Microscopic Traffic Simulation Model, http://www-mctrans.ce.ufl.edu/featured/TSIS/Version5/corsim.htm
[13] ptv simulation VISSIM,http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl
[14] TRANSIMS,transims.tsasa.lanl.gov
[15] The network simulator ns-2,http://www.isi.edu/nsnam/ns

[16] Global Mobile Information Systems Simulation Library, http://pcl.cs.ucla.edu/projects/glomosim.
[17] QualNet Developer, http://www.scalable-networks.com
[18] The OPNET Modeler, http://www.opnet.com/products/modeler/home.html.
[19] BonnMotion,http://web.informatik.uni-bonn.de/IV/BonnMotion
[20] F. Bai,N.Sadagopan,A.Helmy, "The IMPORTANT Framework for Analyzing the Impact of Mobility on Performance of Routing for Ad Hoc Networks",Ad Hoc Networks Journal Elsevier Science, Vol.1,Issue 4,pp.383-403,November 2003.
[21] T. Camp,J. Boleng,V. Davies,"A Survey of Mobility Models for Ad Hoc Network Research", Wireless Communications and Mobile Computing, Vol. 2, Issue 5, pp.483-502,September 2002.
[22] A.K.Saha,D.Johnson,"Modeling Mobility for Vehicular Ad Hoc Networks", Poster Session, 1st ACM Workshop on VehicularAd Hoc Networks (VANET 2004) , October 2004.
[23] U.S.Census Bureau Topologically Integrated Geographic Encoding and Referencing (TIGER) system, http://www.census.gov/geo/www/tiger.
[24] The SHIFT Traffic Simulator,http://path.berkeley.edu
[25] C.Schrothet al., "Simulating the traffic effects of vehicle-to-vehicle messaging systems", in Proc. of the 5th International Conference on ITS Telecommunications, Brest,France,2005.
[26] L.Bononi,M.DiFelice,M.Bertini,E.Croci,"Parallel and Distributed Simulation of Wireless Vehicular Ad Hoc Networks", in proc.of the ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Torresmolinos,Spain,2006.
[27] ARTIS:Advanced Runtime Infrastructure,http://pads.cs.unibo.it/dokuwiki/doku.php?id=pads:artis-doc
[28] GPS TrackMak er, http://www.gpstm.com

AUTHORS

[1] Mr. Venkatamangarao Nampally (n.venkat018@gmail.com) pursed Bachelor of Science in Computer Science, Master of Science in Computer Science and Master of Technology in Computer Science & Engineering from Osmania University, Hyderabad, Telangana, India and pursed Master of Philosophy from University of Madras, Guindy Campus, Chennai, Tamil Nadu, India. He is currently persuing Ph.D. in Computer Science from Osmania University, Hyderabad, and Telangana, India. His main research work focuses on Cryptography Algorithms, Network Security, and Privacy. He has 7 years of teaching experience and 2 year of Research Experience

[2] Dr. Raghavendra Sharma (drmrsstatou@gmail.com) got his Ph.D. from Telangana State, India and currently he is an Assistant Professor and Head of the Department, Department of Statistics at University College of Science, Saifabad, Osmania University, Hyderabad, and Telangana, India. He is the author of several papers. Now, he is supervising many Ph. D.'s. He has excellent teaching track record.

[3] Dr. K. R. Balaji (balajicisl@yahoo.com) has completed M.sc, M.Phil., Ph.D. currently he is working at University of Madras,Guindy Campus, Chennai,Tamil Nadu, India in the department of network systems & information technology, Guindy Campus. He is the author of several papers.