Simulation and Analysis of AODV routing protocol in VANETs

Tajinder Kaur, A. K. Verma

Abstract: ***Vehicular Ad Hoc Network (VANET) is a form of Mobile Ad Hoc Networks (MANET). The field of VANETs started gaining attention in 1980s and has now been an active field of research and development. VANETs provide us with the infrastructure for developing new systems to enhance drivers' and passengers' safety and comfort. There are many routing protocols that have been proposed and assessed to improve the efficiency of VANET. Simulator tool has been preferred over outdoor experiment because it is simple, easy and cheap. In this paper, simulation of one of the routing protocols i.e. AODV is done on simulators which allow users to generate real world mobility models for VANET simulations. The tools used for this purpose are SUMO, MOVE and NS2. MOVE tool is built on top of SUMO which is an open source micro-traffic simulator. Output of MOVE is a real world mobility model and can be used by network simulator NS-2. Then graphs were plotted using Tracegraph for evaluation. Based on the simulation results obtained, the performance of AODV is analyzed and compared in three different node density i.e. 4, 10 and 25 nodes with respect to various parameters like Throughput, Packet size, Packet drops, End to End delay etc.***

Keywords : AODV, MOVE, NS2, SUMO, VANET.

I. INTRODUCTION

Recent advances in wireless networks have led to the introduction of a new type of networks called Vehicular Ad Hoc Networks (VANETs).VANETs [1] is the subclass of Mobile Ad Hoc Networks (MANETs). It deploys the concept of continuously varying vehicular motion. VANETs provide us with the infrastructure for developing new systems to enhance drivers' and passengers' safety and comfort. VANETs are distributed self organizing networks formed between moving vehicles equipped with wireless communication devices. VANETs possess a few distinguishing characteristics from MANETs. These are:

- Highly dynamic topology.
- Patterned Mobility.
- Propagation Model.
- Unlimited Battery Power and Storage.
- On-board Sensors.

There are many routing protocols that have been proposed and assessed to improve the efficiency of VANET.

Manuscript received July 09, 2012.

Tajinder Kaur, Computer Science and Engineering Department, Thapar University, Patiala, India,

A. K. Verma, Computer Science and Engineering Department, Thapar University, Patiala, India, (e-mail: akverma@thapar.edu)

In this paper, we are trying to analyze the performance of one of the routing protocols AODV with respect to various parameters like Throughput, Packet size, Packet drops, End to End delay etc in three different scenarios of node density.

The performance of the proposed protocol has been studied using simulation tools mainly Network Simulator (NS) and MOVE (MObility model generator for VEhicular networks) over SUMO (Simulation of Urban Mobility).

The paper is organized in five sections. The next section describes VANET routing protocols in which AODV is described in detail. In section III we discuss research methodology used for carrying out the experiment. Section IV shows the results and analysis made and last section covers the conclusion part.

II. ROUTING PROTOCOLS

A routing protocol governs the way of exchanging information in two communication entities; it includes the procedure in establishing a route, decision in forwarding, and action in maintaining the route or recovering from routing failure. Fig. 1 illustrates the taxonomy of these VANET routing protocols which can be classified as topology-based and geographic (position-based) in VANET.

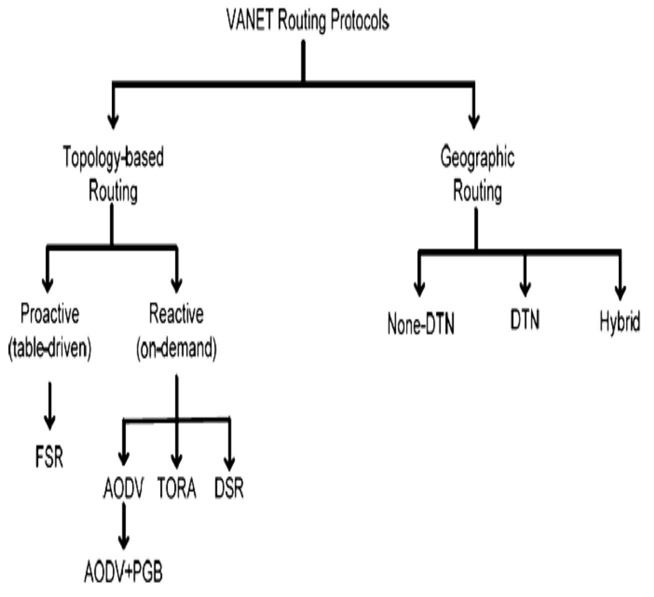


Fig. 1: Taxonomy of Various Routing Protocols in VANET

The routing protocols can be divided into topology based routing and geographic routing [2]. Topology based routing protocols use links information to forward the packet whereas geographic routing uses the information about the location of destination to forward the packet. Topology based routing can again be reactive or proactive. Proactive routing uses the routing table for propagation of message whereas reactive routing builds the route only when it is required. We have used AODV protocol for the analysis which is reactive routing protocol

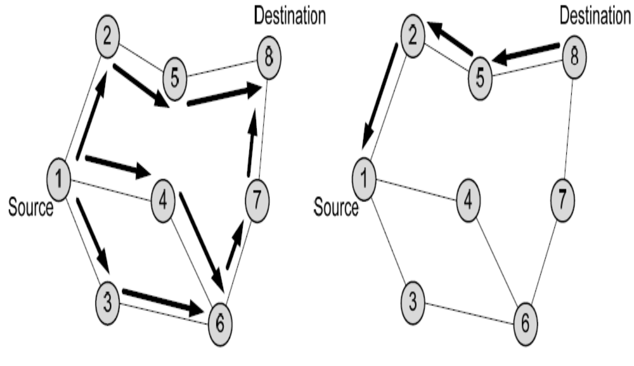
A. AODV

As in VANET, nodes (vehicles) have high mobility and moves with high speed. Proactive based routing is not suitable for it. Proactive based routing protocols may fail in VANET due to consumption of more bandwidth and large table information. AODV is a reactive routing protocol, which operates on hop-by-hop pattern.

The Ad hoc On-Demand Distance Vector (AODV) [3] algorithm enables dynamic, self-starting, multihop routing between participating mobile nodes wishing to establish and maintain an ad hoc network. AODV allows mobile nodes to obtain routes quickly for new destinations, and does not require nodes to maintain routes to destinations that are not in active communication.

Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs) are the message types defined by AODV.

In AODV routing, upon receipt of a broadcast query (RREQ), nodes record the address of the node sending the query in their routing table (Fig. 2a). This procedure of recording its previous hop is called *backward learning*. Upon arriving at the destination, a reply packet (RREP) is then sent through the complete path obtained from backward learning to the source (Fig. 2b). At each stop of the path, the node would record its previous hop, thus establishing the *forward path* from the source. The flooding of query and sending of reply establish a full duplex path. After the path has been established, it is maintained as long as the source uses it. A link failure will be reported recursively to the source and will in turn trigger another query-response procedure to find a new route.



(a) Propagation of the RREQ (b) Path of the RREP to the source

Fig. 2: AODV route discovery [4]

B. AODV Route Discovery

AODV [25] uses route discovery by broadcasting RREQ to all its neighboring nodes. The broadcasted RREQ contains addresses of source and destination, their sequence numbers, broadcast ID and a counter, which counts how many times RREQ has been generated from a specific node. When a source node broadcast a RREQ to its neighbors it acquires RREP either from its neighbors or that neighbor(s) rebroadcasts RREQ to their neighbors by increment in the hop counter. If node receives multiple route requests from same broadcast ID, it drops repeated route requests to make the communication loop free.

C. AODV Route Table Management

Routing table management in AODV is needed to avoid those entries of nodes that do not exist in the route from source to destination. In AODV Managing routing table information handled with the destination sequence numbers.

D. AODV Route Maintenance

When nodes in the network detects that a route is not valid anymore for communication it delete all the related entries from the routing table for those invalid routes. And sends the RREP to current active neighboring nodes that route is not valid anymore for communication. AODV maintains only the loop free routes.

III. RESEARCH METHODOLOGY USED

To carry out the experiments those simulations tools are used which can produce realistic mobility model.

The various tools used for simulation, simulation configuration, performance metrics used for making various comparisons are discussed in this section.

A. Simulation tools

The simulation module created using TCL makes use of two tools to simulate the implementation and evaluate its performance:

1) MOVE: MObility model generator for Vehicular networks [5], [6] tool is used to facilitate users to rapidly generate realistic mobility models for VANET simulations. MOVE is currently implemented in java and is built on top of an open source micro-traffic simulator SUMO. By providing a set of Graphical User Interfaces that automate the simulation script generation, MOVE 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. The output of MOVE is a mobility trace file that contains information about realistic vehicle movements which can be immediately used by popular simulation tools such as ns-2.

2) NS2: The Network Simulator (ns2) [7] is a discrete event driven simulator developed at UC Berkeley. We are using Network Simulator NS2 for simulations of protocols. It provides substantial support for simulation of TCP, routing and multicast protocols 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) [8]. 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 consists of two simulation tools. The network simulator (ns) contains all commonly used IP protocols. The network animator (NAM) is use to visualize the simulations.

3) SUMO: “Simulation of Urban MObility" (SUMO) [9] is an 0open source, highly portable, microscopic road traffic simulation package designed to handle large road networks. It allows the user to build a customized road topology, in addition to the import of different readymade map formats of many cities and towns of the world. Fig. 3. shows SUMO visualization.

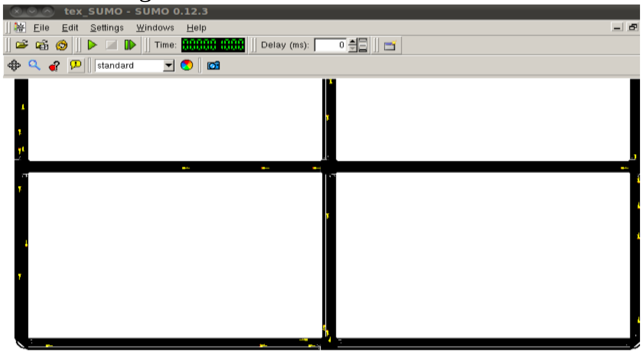


Fig. 3 SUMO Visualization

B. Simulation configuration

The following are the configurations set as per the assumed simulation context:

TABLE I : SIMULATION SETUP

Parameter	Value
Channel type	Wireless
Network Interface type	Physical wireless
Routing protocol	AODV (NS2 default)
Interface queue type	Priority queue
Queue Length	50 packets
Number of nodes in topography	4, 10, 25
X and Y Dimensions of topography	652*752 sq.m
Time of Simulation end	100 simulation seconds
Traffic Type	TCP
Number of Road Lanes	2
Speed	40 m/s
Radio Propagation Model	Two Ray Ground
MAC protocol	IEEE 802.11

C. Simulation Parameters

Various parameters used for performance evaluation are:

1) Throughput: It is the amount of data per time unit that is delivered from one node to another via a communication link. The throughput is measured in Packets per unit TIL or bits per TIL. TIL is Time Interval Length. More is the throughput of sending and receiving packets better is the performance. Lesser is the throughput of dropping packets better is the performance.

2) Average throughput: It is the average of total throughput. It is also measured in Packets per unit TIL or bits per TIL.

3) Packet Drop: It shows total number of data packets that could not reach destination successfully. The reason for packet drop may arise due to congestion, faulty hardware and queue overflow etc. Lower packet drop rate shows higher protocol performance.

4) Packet size: Size of packets in bytes.

5) Average simulation End to End delay (End2End delay): This metric gives the overall delay, from packet transmission by the application agent at the source node till packet reception by the application agent at the destination node. Lower delay shows higher protocol performance. The following equation is used to calculate the average end-to-end delay,

Average End to End Delay = (T_DataR – T_DataS), Where

T_DataR = Time data packets received at destination node

T_DataS = Time data packets sent from source node.

The end to end delay is important metrics because VANET needs a small latency to deliver quick messages. It shows the suitability of the protocol for the VANET.

6) Simulation time: Total time taken for simulation. It is measured in seconds.

IV. RESULTS AND ANALYSIS

Experiment has been carried out for three different numbers of nodes under various cases and results are drawn and evaluated. The numbers of nodes used are:

I. 4 nodes
II. 10 nodes
III. 25 nodes

Results are compared for following cases:

CASE 1: Throughput of sending packets.
CASE 2: Throughput of receiving packets.
CASE 3: Throughput of dropping packets.
CASE 4: Packet Size vs Average throughput of sending packets.
CASE 5: Packet Size vs Average throughput of receiving packets.
CASE 6: Packet Size vs Average throughput of dropping packets.
CASE 7: Throughput of sending bits vs Average simulation End2End delay.
CASE 8: Throughput of receiving bits vs Average simulation End2End delay.

A. CASE 1: Throughput of sending packets.

The graph is plotted for the throughput of sending packets against the simulation time. Throughput is the number of packets sent per unit TIL. TIL is the Time Interval Length. Simulation time is measured in seconds.

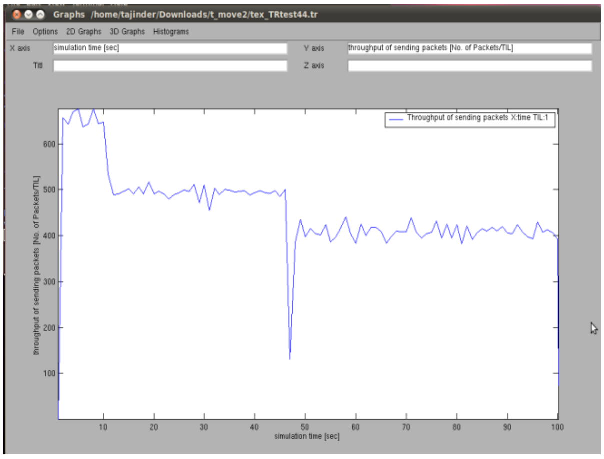


Fig. 4: Throughput of sending packets for 4 nodes

1) INFERENCE FOR FIG. 4: In this graph throughput abruptly increases to 640 packets/TIL in just 2 sec and then it continue to give thoughput of 640 packets/TIL approx. for about 10 sec , then it suudeenly goes down to 500 packets/TIL and remain there with little variation for about 35secs. Then at 47 secs, it again drops to 102 packets/TIL and rises within 1 sec to 400 packets/TIL approx.

Then it continue to give average throughput of 400 packets/TIL for the rest of the simulation time. This can be understood as the number of packets sent per unit time decreases. The packets sent are maximum in the beginning because in the initial stage of VANET, the nodes are sending beacons in order to setup the network.

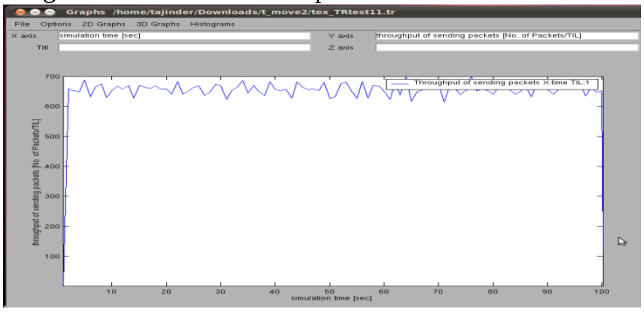


Fig. 5: Throughput of sending packets for 10 nodes

2) INFERENCE FOR FIG. 5: This graph is showing that throughput increases to 650 packets/TIL in just 1 sec in the beginning and then it keeps on giving an average throughput of 650 packets/TIL with little variation for rest of the simultaion time. Here total simulation time is 100 secs.

3) INFERENCE FOR FIG. 6: This graph is also showing that throughput increases to 650 packets/TIL in just 1 sec in the beginning and then it remains in range 600-700 packets/TIL for rest of the simultaion time.

So, graphs for 10 and 25 nodes are more uniform then for 4 nodes.

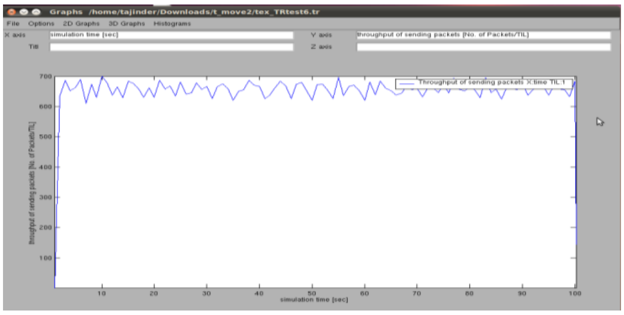


Fig. 6: Throughput of sending packets for 25 nodes

B. CASE 2: Throughput of receiving packets

The graph is plotted for the throughput of receiving packets against the simulation time.

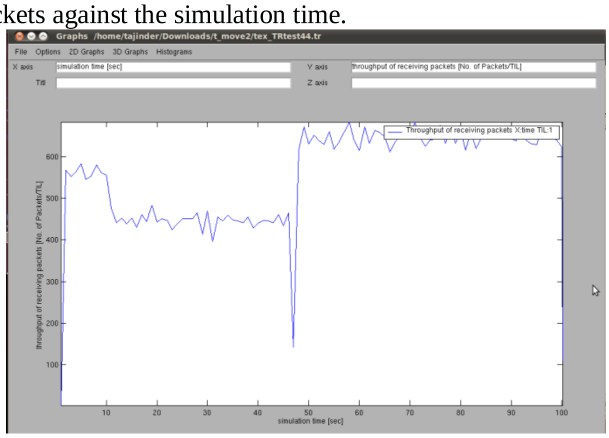


Fig. 7: Throughput of receiving packets for 4 nodes

1) INFERENCE FOR FIG. 7: This graph shows that throughput peaks to 570 packets/TIL in just 2 sec initially, then it remains there for approx. 10 secs but then drops to 450 packets/TIL and continue to give same throughput for about 35 secs. Then it drops suddenly to 150 packets/TIL and rises to 630 packets/TIL in just 2 secs.then for rest of the simulation time it gives an average throughput of 630 packets/TIL with little variations.

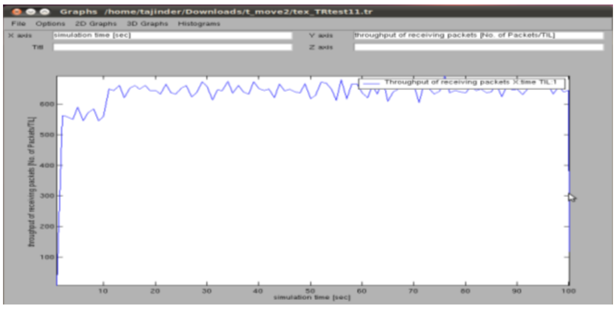


Fig. 8: Throughput of receiving packets for 10 nodes

2) INFERENCE FOR FIG. 8: This graph is showing that throughput increases to 560 packets/TIL within 2 sec in the beginning and then it remains in the range 550-600 packets/TIL for 10 secs and then it rises suddenly to 650 packets/TIL then it keeps on giving throughput in the range of 620-680 packets/TIL for rest of the simultaion time.

3) INFERENCE FOR FIG. 9: This is a more uniform graph then for 4 nodes and 10 nodes. Here throughput rises to 580 packets/TIL in 3 secs then it remains in range of 500-600 packets/TIL for 10 secs approx.and then it rises above to 630 packets/TIL in 2 secs and then it remains in the range of 610-690 packets/TIL uniformly for rest of the simulation time.

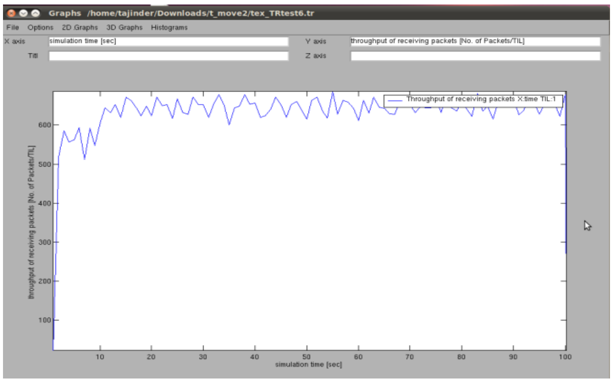


Fig. 9: Throughput of receiving packets for 25 nodes

C. CASE 3: Throughput of dropping packets

The graph is plotted for the throughput of dropping packets against the simulation time.

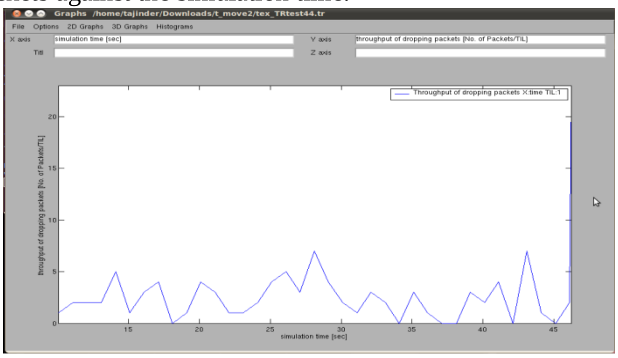


Fig. 10: Throughput of dropping packets for 4 nodes

1) INFERENCE FOR FIG. 10: This graph shows that throughput rises to only 3 packets/TIL in the beginning and rises once to 5 packets/TIL and drops to 2 packets/TIL in just 15 secs.

Then in the simulation time of 46 secs, it rises twice to above 5 packets/TIL and dropped to zero for 5 times.

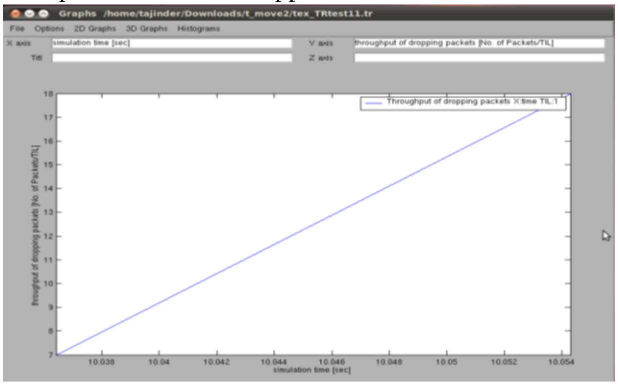


Fig. 11: Throughput of dropping packets for 10 nodes

2) INFERENCE FOR FIG. 11: Here from this graph it can be easily analyzed that number of packets dropped has increased constantly to 18 packets/TIL in just 0.016secs (10.038-10.054secs).

3) INFERENCE FOR FIG. 12: This graphs shows that throughput of drooping packets has increased to 350 packets/TIL in first 2 secs and then dropped to zero in next one sec. Then number of packets dropped per unit time remains to be at zero for about 7 secs. After which it again rises to 200 packets/TIL.

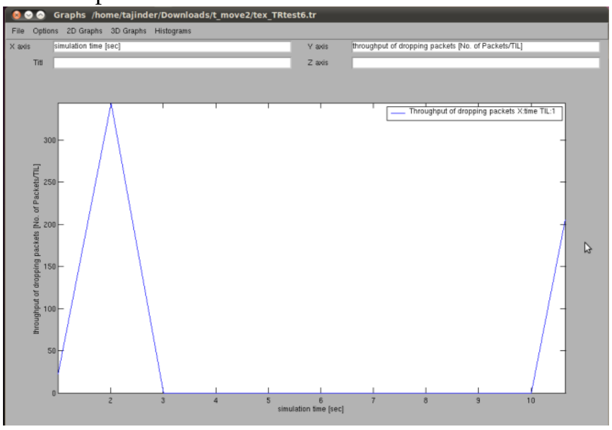


Fig. 12: Throughput of dropping packets for 25 nodes

D. CASE 4: Packet Size vs Average throughput of sending packets

The graph is plotted for the average throughput of packets sent against the size of packets. The size of packets is in bytes

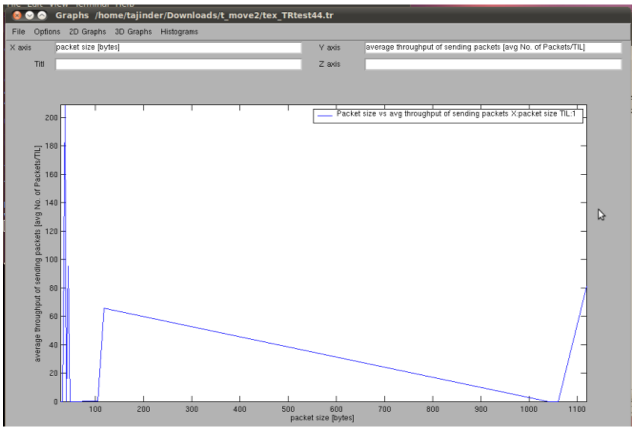


Fig. 13: Packet Size vs Average throughput of sending packets for 4 nodes

1) INFERENCE FOR FIG. 13: This graph shows that average throughput of small sized packets (1-20 bytes) is more (205 packets/TIL) than for large sized packets(1000 bytes)

2) INFERENCE FOR FIG. 14: This graph is almost similar to the graph for 4 nodes with a diference that here average throughput of small sized packets has reached to 320 packets/TIL.

3) INFERENCE FOR FIG. 15: This graph is almost similar to the graph for 4 nodes and 10 nodes.Here also throughput is maximum for small sized packets and it drops to zero for packet size of range 20-100 bytes, after which it increases to 70 packets/TIL for a packet size of 110 bytes and it drops constantly to zero as packet size increases to 1050 bytes.

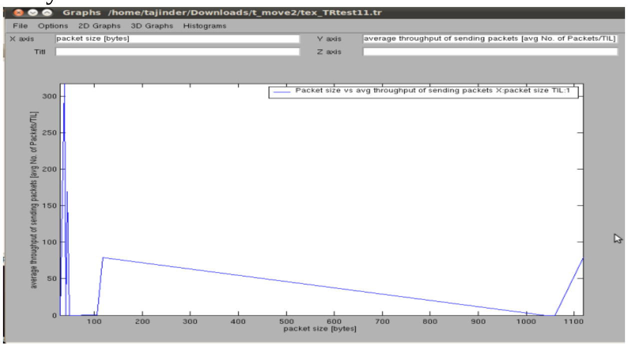


Fig. 14: Packet Size vs Average throughput of sending packets for 10 nodes

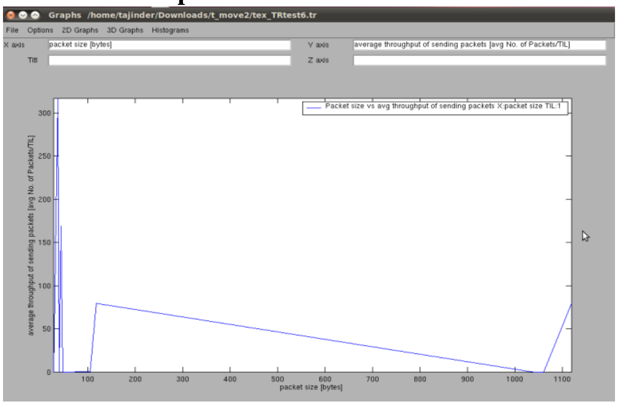


Fig. 15: Packet Size vs Average throughput of sending packets for 25 nodes.

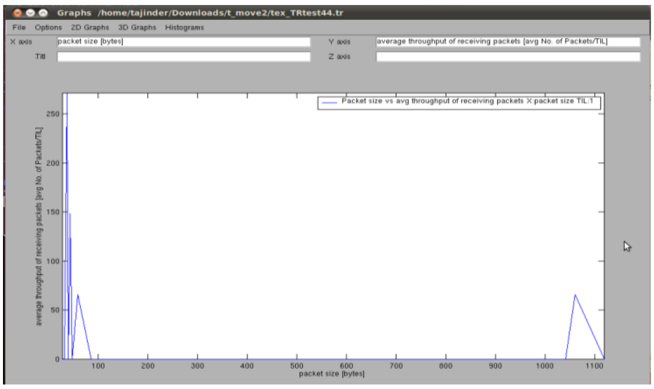


Fig. 16: Packet Size vs Average throughput of receiving packets for 4 nodes

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

E. CASE 5:Packet Size vs Average throughput of receiving packets

The graph is plotted for the average throughput of packets received against the size of packets.

1) INFERENCE FOR FIG. 16: This graph shows that throughput is maximum for small sized packets and zero for packet size of range 90-1030 bytes.

2) INFERENCE FOR FIG. 17: This graph is almost same as for 4 nodes.

3) INFERENCE FOR FIG. 18: This is also same as for 4 and 10 nodes.

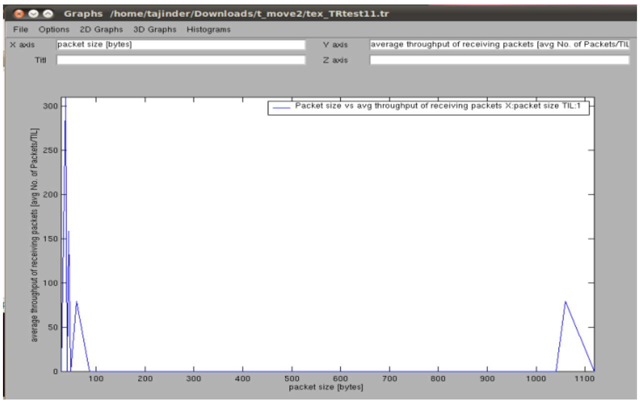


Fig. 17: Packet Size vs Average throughput of receiving packets for 10 nodes

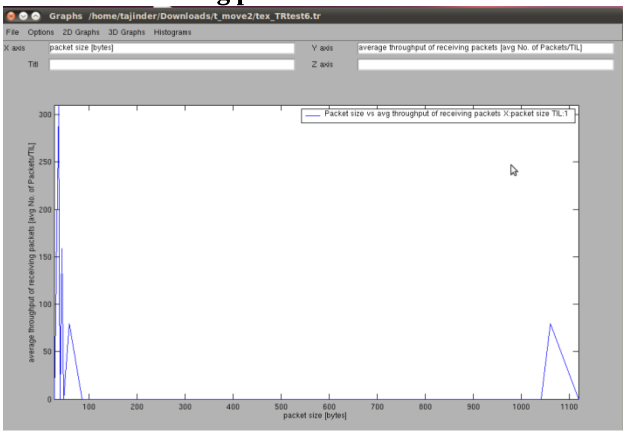


Fig. 18: Packet Size vs Average throughput of receiving packets for 25 nodes

F. CASE 6:Packet Size vs Average throughput of dropping packets

The graph is plotted for the average throughput of packets dropped against the size of packets. The size of packets is in bytes

1) INFERENCE FOR FIG. 19: Here it can be analysed that average throughput of dropping packets is maximum for vry small range of packet size i.e. 10-20 bytes. And for packet size in range 20-1040 bytes there are no packets dropped which indicate that these packet size are not used by the network . There is a small number packets dropped for a packet size of 1050-1100 bytes.

2) INFERENCE FOR FIG. 20: This graph shows a difference in the throughput interval than for 4 nodes. Here it has reached to 12 packets/TIL for packet size of 100-120 bytes. There is also one small peak where throughput is 0.5 packets/TIL for packet size 10-20 bytes.

3) INFERENCE FOR FIG. 21: This graph shows a difference in the throughput interval than for 4 and 10 nodes. Here it has reached to more than 50 nodes for packet size of 100-120 bytes.There is no other small peak.

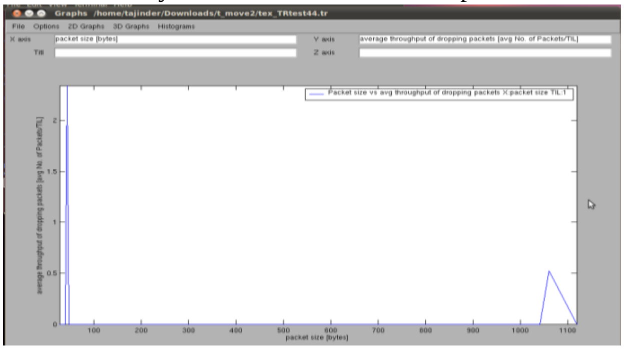


Fig. 19: Packet Size vs Average throughput of dropping packets for 4 nodes

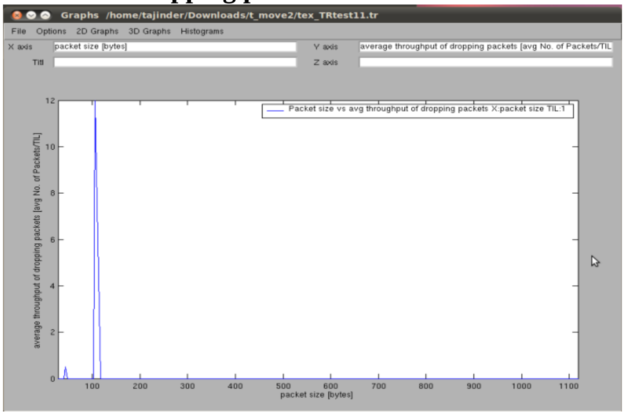


Fig. 20: Packet Size vs Average throughput of dropping packets for 10 nodes

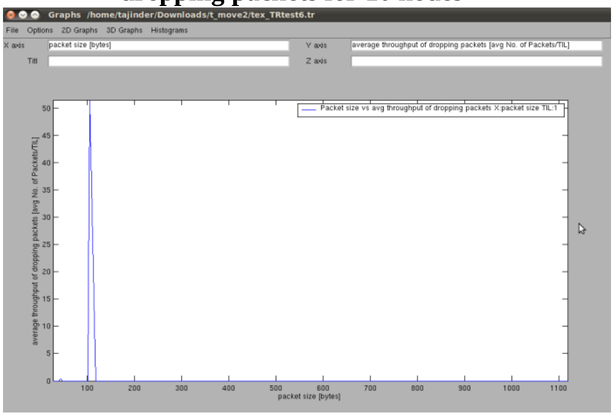


Fig. 21: Packet Size vs Average throughput of dropping packets for 25 nodes

G. CASE 7: Throughput of sending bits vs average simulation End2End delay

The graph is plotted for the average throughput of sending bits and the delay caused. The delay is measured in secs.

1) INFERENCE FOR FIG. 22: Here the delay time interval is for few ms. Within this interval, delay is maximum for a throughput range between 2.5-4.5 bits/TIL. This may be due to more congestion in the network. After this, average delay time decreases to 0.12secs at throughput of 6 bits/TIL and remains almost consistent afterwards. This can be understood as if when throughput of sending packets increases, there are more packets which get dropped which results in decrease in congestion in the network and decrease in delay time.

2) INFERENCE FOR FIG. 23: As compared to for 4 nodes here maximum delay is 0.135secs which is reached when the throughput reaches 5.8 bits/TIL and as the throughput of sending bits increases, delay time remains in the range of 0.13-0.123secs.

3) INFERENCE FOR FIG. 24: Here delay time increases constantly from 0.06secs to 0.13secs as throughput of sending bits reaches to 5.7 bits/TIL and after that it remains in the range 0.13-0.11 bits/TIL.

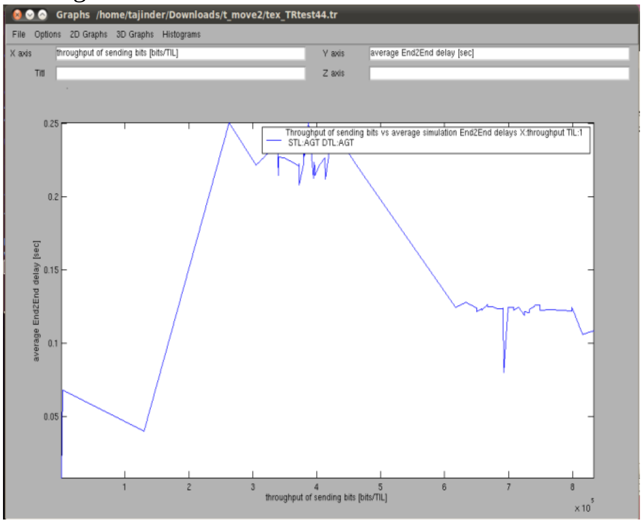


Fig. 22: Throughput of sending bits vs average simulation End2End delay for 4 nodes

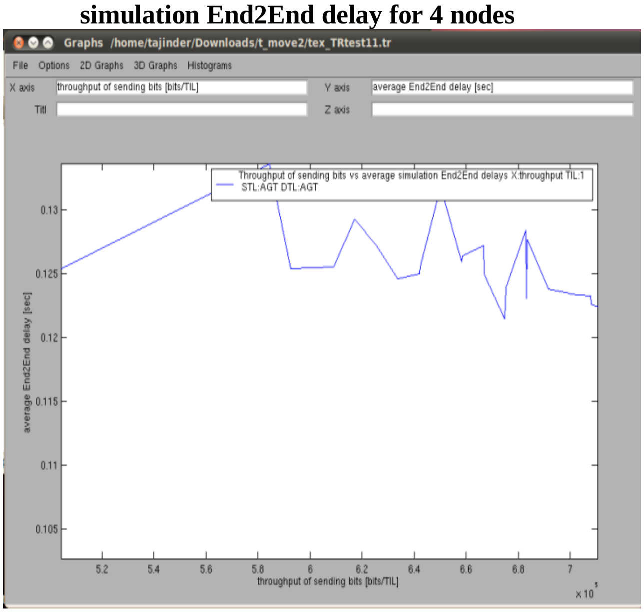


Fig. 23: Throughput of sending bits vs Average simulation End2End delay for 10 nodes

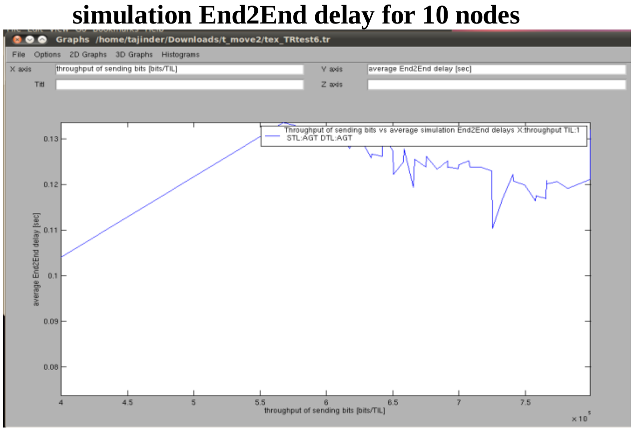


Fig. 24: Throughput of sending bits vs Average simulation End2End delay for 25 nodes

H. CASE 8: Throughput of receiving bits vs average simulation End2End delay

The graph is plotted for the average throughput of receiving bits and the simulation delay caused. The delay is measured in secs.

1) INFERENCE FOR FIG. 25: From this graph it can inferred that as number of bits received per unit TIL increases, delay time also increases concurrently till throughput reaches 325000 bits/TIL.Then for the throughput range 325000-450000 bits/TIL bits/TIL delay time varies a lot and decreases afterwards which indicates a good network has been established.

2) INFERENCE FOR FIG. 26: From this graph it can be inferred that delay time increases from 0 to 0.12secs for a throughput of 100000 bits/TIL. This can be understood as some time is required to establish a network in the starting therefore it took more time to find the route to the destination node. But after this, delay time is not much affected by the increase in number of bits received.

3) INFERENCE FOR FIG. 27: From this graph it can be inferred that as number of bits received per unit TIL increases to 1.25 bits/TIL delay time also increases rapidly till it reaches to 0.12 secs but after this the delay time increases slowly. This means delay is not affected much by the throughput.

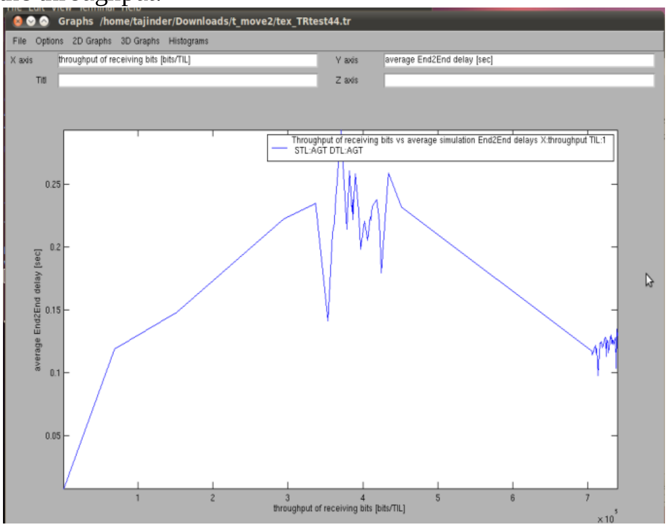


Fig. 25: Throughput of receiving bits vs Average simulation End2End delay for 4 nodes

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

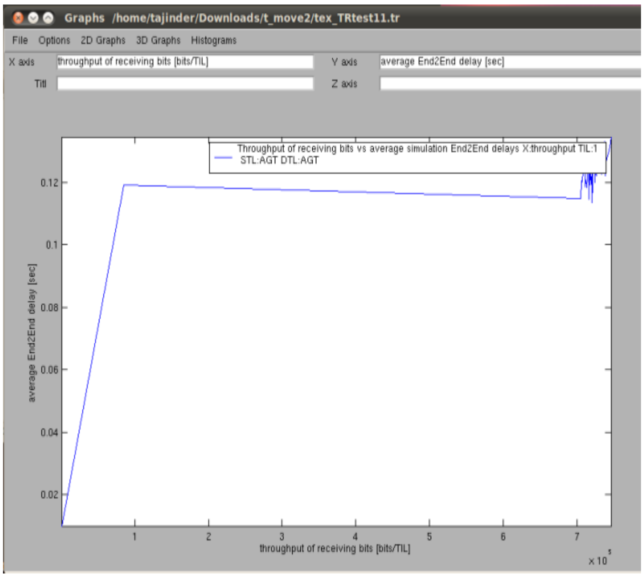


Fig. 26: Throughput of receiving bits vs Average simulation End2End delay for 10 nodes

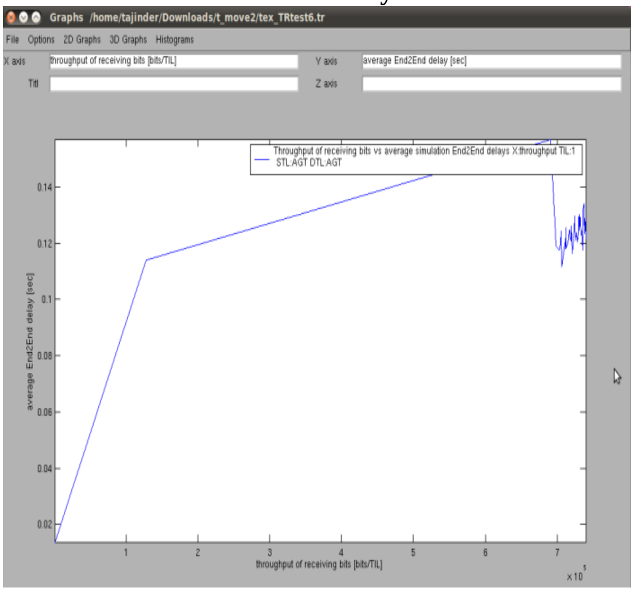


Fig. 27: Throughput of receiving bits vs Average simulation End2End delay for 25 nodes

V. CONCLUSION & FUTURE SCOPE

In this thesis, AODV is simulated with realistic mobility model. For this MOVE is used along with NS2 and SUMO. Then graphs are plotted using Tracegraph for evaluation. AODV's performance is analysed for three different number of nodes i.e. 4, 10 and 25 nodes with respect to various parameters like throughput, packet size, packet drops, delay time etc. The simulation results for various cases can be summarized as below:

CASE 1: Throughput of sending packets: Results shows that for lesser no. of nodes i.e. 4 nodes, throughout drops with time in steps, but for more nodes like 10 and 25 ,throughput of sending packets is almost uniform.

CASE 2: Throughput of receiving packets: Results shows that throughput of receiving packets becomes more uniform with increase in number of nodes.

CASE 3: Throughput of dropping packets: Results shows that number of packets dropped in intial few secs is more in a network where number of nodes are more like in case of 25 ,it has reached to 350. While for fewer nodes like 4, it is quite less (less then 5 approox.)

CASE 4: Packet Size vs Average throughput of sending packets: Average throughput of sending packets is not affected much by the change in number of nodes.

CASE 5: Packet Size vs Average throughput of receiving packets: Average throughput of receiving packets is not affected by the change in number of nodes in a network.

CASE 6: Packet Size vs Average throughput of dropping packets: Average throughput of dropping packets is affected by the change in number of nodes in a network. It increases with increase in number of nodes.This means more number of small sized packets are dropped in a network with larger number of nodes.

CASE 7: Throughput of sending bits vs average simulation End2End delay: From results it can be concluded that there is more delay in a network of lesser number of nodes like 4 nodes than the delay in 10 and 25 nodes network.

CASE 8: Throughput of receiving bits vs average simulation End2End delay: It can be concluded from the results that delay time increases in the beginning of the network establishment to a point (0.12 secs) and then it is not effected much by the throughput of receiving bits in case of 10 and 25 nodes. But in case of 4 nodes it increases till 0.25 secs. This might be because there are very few nodes in the network to communicate well with each other.

In future, it can be simulated and analyzed for higher number of nodes like 50 and 100. It would be interesting to see how AODV performs in high node density network. Here it has been implemented for single mobility model and manually generated maps. In future performance can be compared for different mobility models. And also its performance can be analyzed for random maps, spider topology and maps imported from TIGER database [10].

ACKNOWLEDGMENT

Tajinder Kaur thanks all the anonymous reviewers who have given their valuable feedback which helped her successfully complete this paper.

REFERENCES

1. Vanet Simulator, Report for the Computer Security exam at the Politecnico di Torino Walter Dal Mut, Armand Sofack.
2. Kevin C. Lee, Uichin Lee, Mario Gerla,"Survey of Routing Protocols in Vehicular Ad Hoc Networks", RoutingBookChapterKLULMario.pdf.
3. C. Perkins, E. Royer, and S. Das,"Ad hoc On-Demand Distance Vector (AODV) Routing," IETF, RFC 3561, 2003
4. Lee, K. C., Lee, U., & Gerla, M. (2010*), "Survey of Routing Protocols in Vehicular Ad Hoc Networks", Advances in Vehicular Ad-Hoc Networks: Developments and Challenge,* Watfa, M. (Ed.), (pp. 149-170), 2010.
5. MOVE http://www.cs.unsw.edu.au/klan/move/.
6. Rapid Generation of Realistic Simulation for VANET Manual Updated 11 November 2009. Available at: http://www.lens1.csie.ncku.edu.tw/MOVE/Example%20for%20step-by-step.pdf.
7. The Network Simulator ns-2. http://www.isi.edu/nsnam/ns/index.html.
8. NAM Network Animator. http://www.isi.edu/nsnam/nam/
9. SUMO http://sumo.sourceforge.net/

10. U.S. Census Bureau - Topologically Integrated Geographic Encoding and Referencing (TIGER) system, http://www.census.gov/geo/www/tiger.

AUTHORS PROFILE

Tajinder Kaur is currently pursuing her M.E. in Computer Science and Engineering from Thapar University, Patiala. She received her B.Tech in Computer Science and Engineering from Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib in 2010. She has qualified GATE 2010. She has presented paper in the conference ACNCN-2012, Andhra Pradesh, India, March 17-18, 2012. Her research interests includes wireless networks, VANETs, Routing Algorithms and Network Security.

Dr. A. K. Verma is currently a faculty in the department of Computer Science and Engineering at Thapar University, Patiala. He received his B.S., M.S. and Doctorate in 1991, 2001 and 2008, respectively, majoring in Computer science and engineering. He has worked as Lecturer at M.M.M. Engg. College, Gorakhpur from 1991 to 1996. He joined Thapar University in 1996 and is presently associated with the same Institute. He has been a visiting faculty to many institutions. He has published over 100 papers in referred journals and conferences (India and Abroad). He has chaired various sessions in the International and National Conferences. He is a MIEEE, MACM, MISCI, LMCSI, MIETE, GMAIMA. He is a certified software quality auditor by MoCIT, Govt. of India. His research interests include wireless networks, routing algorithms and cloud computing.