International Journal of Advance Engineering and Research Development (IJAERD)
Volume 1,Issue 6,June 2014, e-ISSN: 2348 - 4470 , print-ISSN:2348-6406

A Novel Approach to Reduce Routing Traffic in Vechicular Ad-hoc Network

Amit B. Parmar[1],Rahul N. Vaza[2], Trupti M. Kodinariya[3]

[1]*PG Student, AITS,Rajkot, er.amitparmar@gmail.com*
[2]*PG Student,AITS,Rajkot, rahulatmiya.rahul9@gmail.com*
[3]*Asst. Prof., AITS,Rajkot, tmkodinariya@aits.edu.in*

Abstract : Vehicular ad hoc networks (VANETs) differ from usual mobile ad-hoc networks (MANETs) in many different aspects. First, VANETs consist of mostly highly mobile nodes moving in the same or opposite directions. Second, the network shape can be best described by a one-dimensional line (for a single-lane road) or a strip (for a multilane road) rather than a square shape.

One of the most important routing protocols used in Ad hoc networks is AODV. It always exchanges control packets between neighbor nodes for routing. For reduction of control overheads and bandwidth consumption in AODV. We make an improvement on AODV and propose a novel approach. In proposed novel approach, we are reducing traffic control packets i.e. routing traffic and improve network efficiency. A novel approach is also analyzed in terms of routing overhead, end-to-end delay and throughput for varying node density i.e. 25, 50, 75, and 100 no. of nodes and compared with simple AODV routing protocol.

Keywords--Vehicular Ad-Hoc Networks (VANETs), Network simulation, Node Mobility, Traffic Control Interface(Traci) , Routing Protocol

I . INTRODUCTION

Vehicular Ad Hoc Network (VANET) is a new challenging network environmentthat pursues the concept of ubiquitous computing for future. Vehicles equipped withwireless communication technologies and acting like computer nodes will be on theroad soon and this will revolutionize the concept of travelling. VANETs bring lots ofpossibilities for new range of applications which will not only make the travel saferbut fun as well. Reaching to a destination or getting help would be much easier. Theconcept of VANETs is quite simple[7]: by incorporating the wireless communicationand data sharing capabilities,the vehicles can be turned into a network providingsimilar services like the ones with which we are used to in our offices or homes.For the wide spread and ubiquitous use of VANETs, a number of technical challenges exist.

Most applications targeting VANETs rely heavily on broadcast transmission. When a vehicle rebroadcasts a message, it is highly likely that the neighboring vehicles have already received it and these results in a very large number of redundant messages. This affects inter-vehicle communications, since redundant rebroadcasts, contention and collisions can be largely increased as the no. of vehicles increases. Broadcasting packets may lead to frequent contention and collisions in transmission among neighboring vehicles this problem is known as the broadcast storm problem.

II. BROADCAST STORM PROBLEM

The broadcast storm problem is a caused generated due to flooding. For example, figure 2.4 depicts a sample network with five nodes, where if node A broadcasts a packet, nodes B, C and D will receive that packet. Nodes B, C and D will then forward the packet and lastly E will also broadcast the packet. In fact, this case clearly shows the broadcast redundancy inherent with flooding. Forwarding the broadcast packet by nodes A and D is sufficient for the broadcasting to cover all the five nodes.

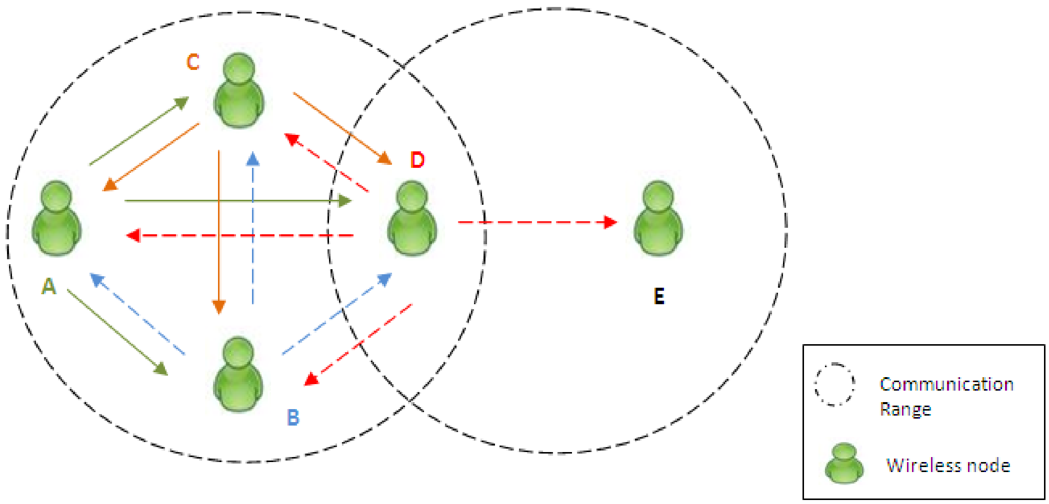


Figure A sample ad-hoc network with 5 nodes

However, when the size of the network increases and the network becomesdense, more transmission redundancy will be introduced and these transmissionsare likely to cause serious drawbacks (i.e. redundant rebroadcast, contentionandcollision) which can lead to a total collapse in the operation of the network.These drawbacks are collectively referred to as the broadcast storm problem[3][16].

III. ROUTING PROTOCOL

The routing protocols are classified as follows on the basis of the way the network information is obtained in these routing protocols.

- **Proactive (or Table-driven) routing protocol**

The proactive protocols maintain routing information about each node in the network. The information is updated throughout the network periodically or when topology changes. Each node requires to store their routing information. For example: Destination sequenced Distance vector routing (DSDV),Source Tree Adaptive Routing (STAR).

- **Reactive or On-demand routing protocol**

The reactive routing protocols look for the routes and are created as and when required. When a source wants to send to a destination, it invokes the route discovery mechanisms to find the path to the destination. For example: Ad-Hoc On-demand Distance Vector (AODV), Dynamic Source Routing (DSR).

- **Hybrid Protocols**

These protocols are using the best features of both the on-demand and table driven routing protocols. **For example:**Temporally ordered routing algorithm (TORA),Zone Routing Protocol (ZRP).

These classes of routing protocols are reported but choosing best out of them is very difficult as one may be performing well in one type of scenario the other may work in other type of scenario.

3.1 Ad-hoc On-demand Distance Vector (AODV) Routing Protocol

AODV is the best-known and most studied MANET routing protocol. It is reactive in nature, requesting and establishing routes only when needed and maintaining only those that remain active. The AODV routing mechanism consists of two phases; route discovery and route maintenance.

3.1.1 Route Discovery

When a source node wants to send data to a destination and does not already have a valid route to the destination, it initiates a route discovery process in order to locate the destination. A route request (RREQ) packet is broadcast throughout the network via simple flooding and in a well managed fashion using expanding ring search technique. The RREQ packet contains the following main fields: source identifier, source sequence number, broadcast identifier, destination identifier, destination sequence number (created by the destination to be included along with any route information it sends to requesting node), and time-to-live. To prevent excessive transmission of the RREQ packets, the source node optimizes its search by using an expanding ring search. A time-to-live field (TTL) field in the header of the RREQ packet control the search. The destination sequence number is used by AODV to ensure loop-free routes which also contain most recent route information.

Each intermediate node that forwards an RREQ packet creates a reverse route back to the source node, for that it append the next hop information in its routing table. Once the RREQ packet reaches the destination or an intermediate node with a valid route, the destination or intermediate node responds by sending a unicast route reply (RREP) packet to the source using reverse route. The validity of a route at the intermediate is determined by comparing packet's sequence number field with the destination sequence number. Each node that participates in forwarding the RREP packet back to the source creates a forward route to the destination and append the next hop information in the routing table. However, nodes along the path from source to destination are not required to have knowledge of which nodes are forming the path.

Figure 3.1 depicts an example of route discovery process. It shows how the path is determined from the source node 2 , to the destination node 9. Node 2 i.e source node, propagates a route request packet to its neighbours, nodes 1, node 3, and node 4. These nodes, in turn, disseminate the route request i.e RREQ to their neighbours while collecting route data. The route request, along with the path to the source node 2, is eventually received by the destination node 9. Base on the route data that has been collected during the route discovery process, the destination node is able to send its reply message back along the shortest route.

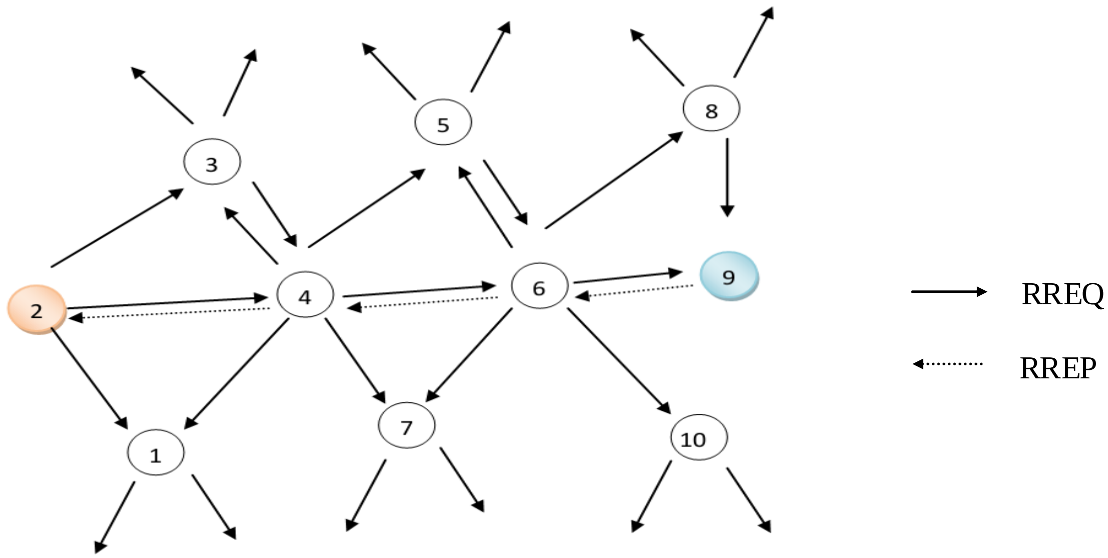


Figure 3.1 Illustration of route discovery process in AODV

3.1.2 Route Maintenance

Route maintenance is second phase of AODV routing mechanism. Route maintenance is the process of responding to changes in topology that happen after a route has initially been created. Once route discovery process completed and as long as a discovered routes are used, they have to be maintained. To maintain paths, intermediate nodes along the path continuously monitor the active links and maintain an up-to-date list of their 1-hop neighbours. The routing table entries include a destination, sequence number and next hop toward the destination. Routes are only updated if the sequence number of the incoming message is larger than the existing sequence number. The route expiration time is also maintained in routing table. Each time that route is used to forward data packet, the expiration time is updated to the current time + ACTIVE_ROUTE_TIMEOUT. When a broken link occurs, or a node receives a data packet for a destination for whch it has no forwarding route, it must respond with creation of a Route Error (RERR) message. The RERR message gives information for all of the unreachable nodes. The source node can either try to find a new route by initiating a new route discovery for the destination if there is no intermediate node with an alternative path to destination, or the intermediate node may try to repair the route locally.

IV. PROPOSED ARCHICTURE

Phase 1: Generate Route Request:

In this first phase if any nodes wants to communicate with other node in network then first it checks whether the path from source node to destination is available or not. If source node does not have valid path then it starts route discovery process and generate RREQ packet and broadcast it.

Phase 1: Generate Route Request

If node has data to send
    If source node has no valid entry in route table for destination
        Generate RREQ and broadcast it
    Else If source has an invalid destination entry in route table
        Generate RREQ and broadcast it.

Phase 2: Processing or Forwarding Route Requests:

When node receive RREQ packet, if node itself a destination node or it has a valid path to destination then it generaters a RREP packetand send back to sourse node. If intermediate node has no valid path to destination then we will performe phase 3.

Phase 2:Processing or Forwarding Route Requests

If Node listens (receive) a RREQ Packet first time
    If Node is Destination for this RREQ **OR** Node has route to destination
        Send RREP
        Discard RREQ
    Else
        Call PHASE 3: Find Number of Neighbors
Else
    Drop RREQ packet.

Phase 3: Find Number of Neighbors

In this phase first of all each node find out number of neighbors based on Hello Packet. Each node in network broadcast Hello packet periodically after HELLO TIME INTERVAL. From this Hello packet each node calucalte no. of neighbors at that perticular node. Base on that we will calculate avg no of neighbors in the network.

Phase 3: Find Number of Neighbors

In this phase each node finds no of neighbors using Hello Packet.

Each node sends **Hello message** after every **Hello_Time_Interval**

On receiving **Hello message** from neighbor update the routing table for neighbors

Count total no of neighbors at each node. i.e $N_{X.}$

Count average no. of neighbors in network. i.e avg

Example:

Total nodes: 25

Neighbors at **node 1** i.e. N_1=3

Neighbors at **node 2** i.e. N_2=5

Neighbors at **node 3** i.e. N_3=8

Neighbors at **node 4** i.e. N_4=5

.

.

Neighbors at **node 25** i.e. N_{25}=2

Avg $=\sum_{x=1}^{n} \frac{N_x}{n}$

Phase 4 : Set Rebroadcast Probability

In this phase we will set rebroadcast probabiliy base on no of neighbors at that node. If the node has no. of neighbors less then average values then assign rebroadcast probability high i.e. p=0.8.If the node has no. of neighbors equal to average values then assign rebroadcast probability medium i.e. p=0.5.If the node has no. of neighbors greater then average values then assign rebroadcast probability low i.e. p=0.4.

Phase 4: Set Rebroadcast Probability

Node x has N_x neighbors

If N_X< avg then

X has less no of neighbors i.e X is in sparse network then avg so set rebroadcast probability **p = 0.8**

Else if N_X = avg then

X has avg no of neighbors i.e X is in Regular Network so set rebroadcast probability **p = 0.5**

Else

X has highno of neighbors then avg i.e X is in Dense Network so set

Phase 5 : Rebroadcast RREQ

In this phase now we will re-broadcast the received RREQ packet with different

Phase 5: Rebroadcast RREQ

If p >= 0.5 then
Rebroadcast the RREQ packet
Else
Drop RREQ packet.

probability. We will rebroadcast the RREQ packet if rebroadcast probability is greater or equal to 0.5.

So, using this approach we are reducing control traffic i.e. RREQ packets in network..

V. SIMULATION AND RESULT ANALYSIS

Simulation Parameters

Simulation Parameter	**Values**
Protocol	AODV,M_AODV
Simulation time	1000 sec
Mobility model	Random Way Point
Type of traffic	TCP
No. of connection	20,40,60,80
No. of vehicles	25,50,75,100
Ns2 version	2.34
Mac layer protocol	IEEE 802.11
Packet size	512 bytes
Hello packet size	64 bytes

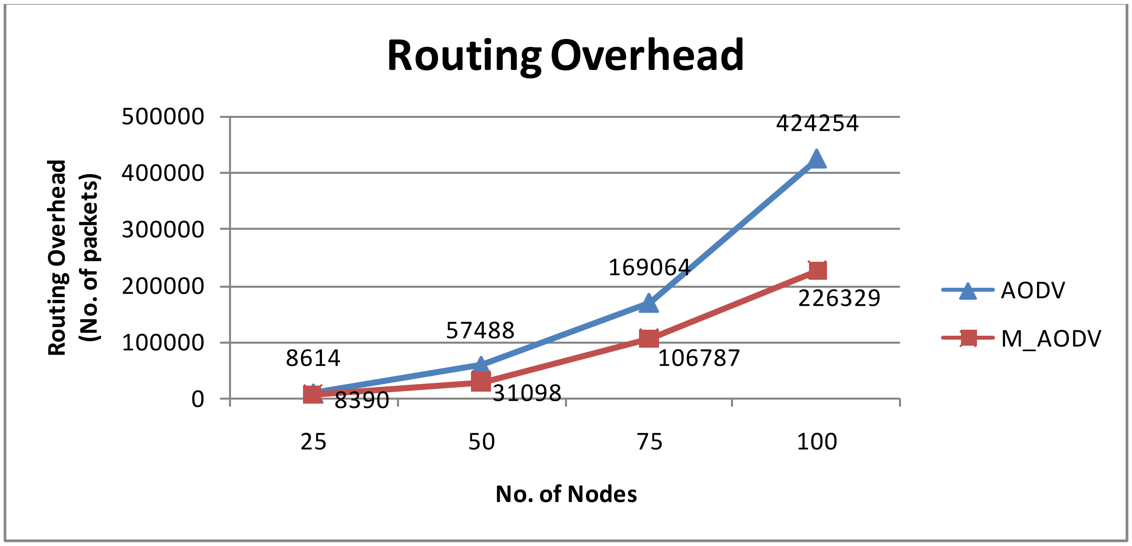
We will evaluate the performance based on end-to-end delay, routing overhead, throughput as the matrics. These matrices describe nature of Ad-hoc networks and formulate boundary conditions of Ad-hoc networks but these properties do not directly related to performance. To measure external performance of a protocol, we consider end-to-end delay as matrics and to measure internal effectiveness of a protocol we consider routing overhead and throughput as the matrics. All these matrics are most widely used for representing performance of routing protocols because higher data delivery, lower control overhead and lower delay are always desirable.

5.1 Routing Overhead:

All the routing packets no matter broadcasting or unicasting per-hop should be count once.Routing overhead means the total number of routing packets i.e the number of RREQ packets transmitted for the purpose of routing data packets during the simulation time.

$$R.O = \frac{\text{Total no.of routing packets}}{\text{data packets}}$$

Routing Overhead(No. of Packets)				
No. of Nodes	25	50	75	100
AODV	8614	57488	169064	424254
M_AODV	8390	31098	106787	226329



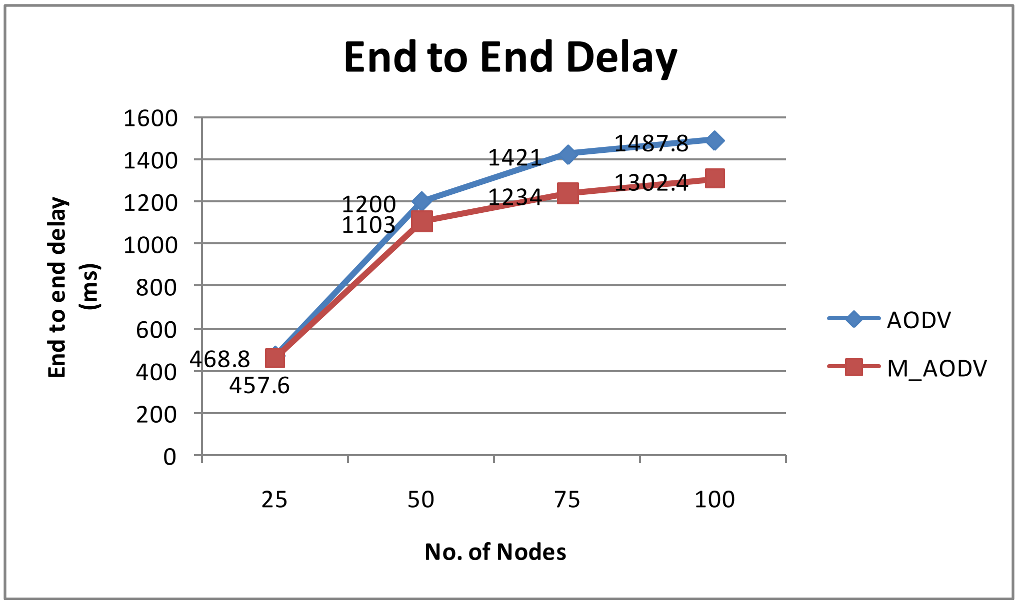
The above graph shows the performance of the AODV and M_AODV in terms of routing overhead versus no. of nodes i.e. network density. The RREQ Packets increased as the number of nodes is increase. From above graph it is very clear that routing overhead generated by M_AODV is lower compared by normal AODV. The improved performance of M_AODV is due to the significant reduction in the number of redundant RREQ packets.

5.2 Average End-to-end Delay:

This delay can be caused by many reasons, like, latching during route discovery latency, queuing at interface queue, andretransmission delays at the MAC. End-to-end delay can be calculated by dividing the time difference between every TCP packets sent and received, to the total number of TCP packets received. For the betterperformance of the protocol end to end delay must be aslower as possible.

End to End Delay = (Time data packets received at destination node – Time datapackets sent from source node).

End to end delay (ms)				
No. of Nodes	25	50	75	100
AODV	468.794	1200.14	1420.68	1487.84
M_AODV	457.58	1102.61	1234.1	1302.35



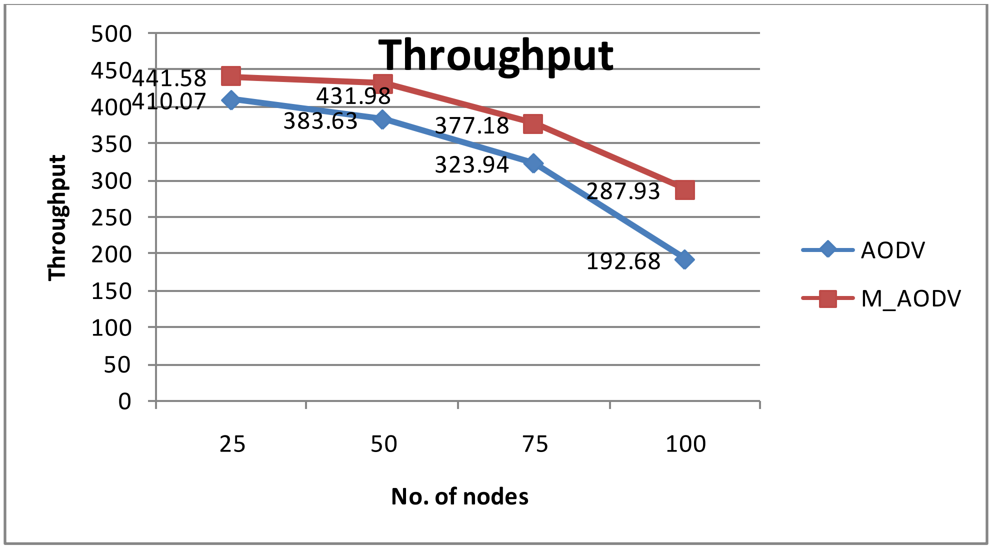
The above graph shows the performance of the AODV and M_AODV in terms of average end-to-end delay versus no. of nodes i.e. network density. When network density increase, the number of duplicated RREQ packets which generated by nodes is also increased, and this is increased the number of dropped packets. As a result, packets experience high latencies in the interface queues. From the above graph it is very clear that end-to-end delay generated by M_AODV is lower compared by simple AODV.

5.3 Throughput:

It is the amount of data per time unit that isdelivered from one node to another via a communication link. The throughput is measured in packets per time interval length. It is also expressed in bits per second or packets per second. Factors that affect throughput include frequent topology changes, unreliable communication, limited bandwidth and limited energy. A high throughput network is desirable.

$$\text{Throughput} = \frac{\text{Total no.of packet}}{\text{second}}$$

Throughput(Data packets/sec)				
No. of Nodes	25	50	75	100
AODV	410.07	383.63	323.94	192.68
M_AODV	441.58	431.98	377.18	286.93



The above graph shows the performance of the AODV and M_AODV in terms of throughput versus no. of nodes. In a network where excessive redundant retransmissions of control packets (e.g. RREQ packets) are predominant, channel contention and packet collisions increase thereby lowering the bandwidth available for data transmission. Therefore, if we control the redundant retransmissions of RREQ packets in network, the degradation of the throughput can be reduced. As shown in above graph, M_AODV outperformance well compared to AODV. The improved performance of M_AODVis due to the significant reduction in the number of retransmissions of RREQ packets.

VI. CONCLUSION

Broadcasting of messages can result in broadcast storm problem due to which efficiency of network is decreases. We have performed simulation for 25, 50, 75, 100 nodes and analyzed performance of AODV in terms of routing overhead, throughput as well as end-to-end delay.

The proposed approach considers the message rebroadcast process by selecting a limited number of vehicles, acting as forwarders so there is less contention of messages in network due which less probability of collision of messages in network. In this approach there are less redundant messages in network. Thus with the help of this scheme broadcast storm problem can be reduced.

We have implemented this proposed approach with AODV protocol and analyze the results for 25,50,75 and 100 no. of nodes. As compared to AODV this modified AODV (M_AODV) have better results in terms of routing overhead, throughput as well as end-to-end delay

FUTURE WORK

This research has presented an extensive performance analysis of proposed algorithm for pure broadcast and application scenarios (e.g. route discovery) based on the reactive AODV routing protocols. It would interested to investigate the impact of these broadcasting algorithms when used as a route discovery mechanism in other reactive routing protocols, such as DSR

REFERENCES

[1]C. Perkins, E. Belding-Royer, and S. Das, "*Ad hoc On-Demand Distance Vector (AODV) Routing*" IETF Mobile Ad Hoc Networking Working Group INTERNET DRAFT, RFC 3561, July 2003,http://www.ietf.org/rfc/rfc3561.txt

[2] D. Johnson, Y. Hu, and D. Maltz, "*The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks,*" IETF Mobile Ad Hoc Networking Working Group INTERNET DRAFT, http://www.ietf.org/rfc/rfc4728.txt

[3] D. Krajzewicz and C. Rossel. Simulation of Urban MObility (SUMO). German Aerospace Centre, 2007.http://sumo.sourceforge.net/index.html.

[4]Fan Li and Yu Wang, "*Routing in Vehicular Ad Hoc Networks: A Survey*", IEEE VehicularTechnology Magazine, vol. 10, no. 3, pp. 12-22, JUNE 2007.

[5] F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni. "*City-mob: a mobility model pattern generator for VANETs"*, IEEE Vehicular Networks and Applications Workshop (Vehical-Mobi, held with ICC), Beijing, China, May 2008.

[6]FreeSim, 2008 http://www.freewaysimulator.com/.

[7] H. Mori, H. Kitaoka, and E. Teramoto. "*Traffic simulation forpredicting traffic situations",*R&D Review of Toyota CRDL, vol. 41 no. 4, pp. 45-51, 2006.

[8] J. Haerri, M. Fiore, F. Fethi, and C. Bonnet. VanetMobiSim: generating realistic mobility patterns for VANETs. Institute Eurécom and Politecnico Di Torino, 2006http://vanet.eurecom.fr/index.php

[9] J. Jakubiak and Y. Koucheryavy, "*State of the art and research challenges for VANETs*", Consumer Communications and Net-working Conference, 2008. CCNC 2008., pp. 912-916, Jan. 2008.

[10] J. Martin. GloMoSim. Global Mobile Information Systems Simula-tion Library. UCLA Parallel Computing Laboratory, 2001.
http://pcl.cs.ucla.edu/projects/glomosim/

[11] JiST/SWANS: Java in Simulation Time/Scalable Wireless Ad hocNetwork Simulator, 2004.http://jist.ece.cornell.edu/.

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

[13] K. Suriyapaiboonwattana and C. Pomavalai. "*An effective safety alert broadcast algorithm for VANET*", International Symposium on Communications and Information Technologies. ISCIT'08, pp. 247-250, Oct. 2008.

[14] K. Walsh and E. G. Sirer. SNS. A staged network simulator. Com-puter Science Department. Cornell University, 2003. http://www.cs.cornell.edu/people/egs/sns/.

[15] MobiREAL, 2008. http://www.mobireal.net/

[16] MohammedAminu,Ph.D.Thesis, "*Hybrid Probabilistic Broadcast Schemesfor Mobile Ad hoc Networks*"University of Glasgow, July 2007.

[17] MOVE (MObility model generator for VEhicular networks): Rapid Generation of RealisticSimulation for VANET, 2007. http://lens1.csie.ncku.edu.tw/MOVE/index.htm

[18] M. Piorkowski, M. Raya, A. Lezama, P. Papadimitratos, M. Grossglauser,and J.-P. Hubaux. TraNS (Traffic and Network SimulationEnvironment). EcolePolytechnique Fédérale de Lausanne, EPFL,Switzerland, 2007. http://trans.ep_.ch/

[19] NCTUns 5.0, 2008. http://nsl10.csie.nctu.edu.tw/.

[20] N. Wisitpongphan, O.K. Tonguz, J.S. Parikh, P. Mudalige, F. Bai, and V. Sadekar. "*Broadcast storm mitigation techniques in vehicular ad hoc networks*", Wireless Communications, IEEE, vol. 14, no. 6, pp. 84-94, December 2007.

[21] P. Kyasanur, R. R. Choudhury, and I. Gupta. Smart gossip: "*An adaptive gossip-based broadcasting service for sensor networks",* in Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference , pp. 91-100, Oct. 2006.

[22] R. Mangharam, D. Weller, R. Rajkumar, P. Mudalige, andF Bai. GrooveNet: A Hybrid Simulator for Vehicle-to-VehicleNetworks. Carnegie Mellon University,2006. http://www.seas.upenn.edu/_ rahulm/Research/GrooveNet/

[23]STRAW - STreet RAndom Waypoint - vehiclar mobility model for networ simulations (e.g., car networks), 2008. http://www.aqualab.cs.northwestern.edu/projects/STRAW/index.php.

