

Reactive Protocol in VANET

P.Geetharani, Research Scholar, RVCE, Bangalore

Abstract—The Dynamic Source Routing protocol (DSR) is a simple and efficient Routing protocol designed specifically for use in multi-hop wireless mobile nodes networks called Ad hoc networks. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. This paper describe the design of our own implementation of DSR protocol using DSR VANET draft in NS2 simulator environment and provide a summary of some the simulation and tested implementation results for the protocol.

Keywords— Ad Hoc networks, DSR, node, VANET,NS2.

I. Introduction

The routing protocols meant for wired networks cannot be used for mobile ad hoc networks because of the mobility of networks. In contrast to infrastructure based networks, in ad hoc networks all nodes are mobile and can be connected dynamically in an arbitrary manner. All nodes of these networks behave as routers and take part in discovery and maintenance of routes to other nodes in the network. Ad hoc networks are very useful in emergency search-and-rescue operations, meetings or conventions in which persons wish to quickly share information, and data acquisition operations in hospitable terrain. Using DSR, the network is completely self-organizing and self-configuring, requiring no existing network infrastructure or administration. Network nodes cooperate to forward packets for each other to allow communication over multiple "hops" between nodes not directly within wireless transmission range of one another. As nodes in the network move about or join or leave the network, and as wireless transmission conditions such as sources of interference change, all routing protocol is automatically determined and maintained by the DSR routing protocol. Since the number or sequence of intermediate hops needed to reach any destination may change at any time, the resulting network topology may be quite rich and rapidly changing. NS-2 is an object-oriented simulator developed as part of the VINT project at the University of California in Berkeley. Using which new protocol (Agent) implementations can be tested efficiently..

II. BASIC TECHNOLOGY

The basic features of these type of non-administrative network, and the information about how this "source-routed on-demand routing protocol"(DSR)

works and under what This work was done for the laboratory class "Mobile communication and wireless networking lab" University of Bonn , Informatics Department IV "Communication Systems Operating Systems and Stochastic modeling" assumptions this protocol is implemented is explained below 1.1 Mobil AD Hoc networks Mobile Ad-hoc Networks(MANETs) are future wireless networks consisting entirely of mobile nodes that communicate on-the-move without base stations. Nodes in these networks will both generate user and application traffic and carry out network control and routing protocols. Rapidly changing connectivity, network partitions, higher error rates, collision interference, and bandwidth and power constraints together pose new problems in network control—particularly in the design of higher level protocols such as routing and in implementing applications with Quality of Service requirements.

A. NS2 Simulator

NS-2 is extensively used by the networking research community. It provides substantial support for simulation of TCP, routing, multicast protocols over wired and wireless (local and satellite) networks, etc. The simulator is event driven and runs in a non-real-time fashion. It consists of C++ core methods and uses Tcl and Object Tcl shell as interface allowing the input file (simulation script) to describe the

model to simulate. Users can define arbitrary network topologies composed of nodes, routers, links and shared media. A rich set of protocol objects can then be attached to nodes, usually as agents. The simulator suite also includes a graphical visualizer called network animator (nam) to assist the users get more insights about their simulation by visualizing packet trace data.

B. DSR Protocol

The Dynamic Source Routing Protocol is a source-routed on-demand routing protocol. A node maintains route caches containing the source routes that it is aware of. The node updates entries in the route cache as and when it learns about new routes. DSR does not rely on functions like periodic routing advertisement, link status sensing or neighbor detection packets and because of the entirely on demand behavior, the number of overhead packets caused by DSR scales down to zero.

C. Basic mechanism

Route Discovery and Route Maintenance, which are the main mechanisms of the DSR protocol, allows the discovery and maintenance of source routes in the ad hoc network's works entirely on an on-demand basis. As DSR works entirely on demand and as nodes begin to move continuously, the Routing packet overhead automatically scales to only that needed to react to changes in the route currently in use. In response to a single Route Discovery if a node learns and caches multiple routes to a destination, it can try another route if the one it uses fails. The overhead incurred by performing anew Route Discovery can be avoided when the caching of multiple routes to a destination occurs. and Mobile IP routing and supports internetworking between different types of wireless networks. [1]

D. DSR Route Discovery

The header of the packet, which originates from a source node S to a destination node D, contains the source route, which gives the sequence of hops that the packet should traverse. A suitable source route is found normally when searching the Route Cache of routes obtained previously but if no route is found then the Route Discovery protocol is initiated to find a new route to D. Here S is the initiator and D the target. [1] Node A transmits a ROUTE REQUEST



Fig.1 Node A is the initiator and Node E is the target

message, which is received by all the nodes in the transmission range of A. Each ROUTE REQUEST message identifies the initiator and target of the Route Discovery and also contains a unique request ID, determined by the initiator of the REQUEST. Each ROUTE REQUEST also contains a record listing the address of each intermediate node through which this particular copy of the ROUTE REQUEST message

has been forwarded. The initiator of the Route Discovery initializes the route record to an empty list.

[1] When the target node receives the ROUTE REQUEST message, it returns a ROUTE REPLY message to the ROUTE Discovery initiator with a copy of the accumulated route record from the ROUTE REQUEST. This route is cached in the Route Cache when the initiator receives the ROUTE REPLY and is used in sending subsequent packets to this destination. When the target node finds a ROUTE REQUEST message from the same initiator bearing the same request ID or if it finds its own address is already listed in the route record of the ROUTE REQUEST message, it discards the REQUEST. If the target node does not find the ROUTE REQUEST message from the initiator, then it appends its address to the route record in the ROUTE REQUEST message and propagates it by transmitting it as a local broadcast packet. When Route Discovery is initiated the copy of the original packet is saved in a local buffer called Send Buffer. The Send Buffer contains copies of each packet that cannot be transmitted by the sending node. The packets are kept until a source route is available or a timeout or Send Buffer overflow occurs. As long as a packet is in the Send Buffer, the node should initiate new Route Discovery until time out occurs or overflow of Buffer occurs. An exponential Back off algorithm is designed to limit the rate at which new ROUTE Discoveries may be initiated by any node for the same target. [1] 1.4.2 DSR Route Maintenance When a packet with a source route is forwarded, each node in the source route makes sure that the packet has been received by the next hop in the source route. The confirmation of receipt will be received only by re-transmitting the packet for a number of times. [1] Node A is the originator of a packet to the desired destination E. The packet has a source route through intermediate nodes B, C and D. Node A is responsible for



Fig.2 Node C is unable to forward a packet from A to E over the next node D

receipt of the packet at B, node B at C, node C at D and node D at E. Node B confirms receipt of packet at C by overhearing C transmit the packet to forward it to D. The confirmation of acknowledgement is done by passive acknowledgements or as link-layer mechanisms such as option in MAC protocol. The node receiving the packet can return a DSR specific software acknowledgement if neither of the acknowledgements

is available. This is done by setting up a bit in the packet's header and then requesting DSR specific software acknowledgement by the node transmitting the packet. When a node is unable to deliver a packet to the next node then the node sends a ROUTE ERROR message to the original sender

of the packet. The broken link is then removed from the cache by the originator of the packet and retransmissions to the same destination are done by upper layer protocols like TCP. [1] Route maintenance is also carried out also by both ROUTE REQUEST and ROUTE REPLY packets, when they traverse from each node the data from the option header of these packets which contain the link information of the nodes are updated in the nodes route cache 2 Setup and metrics The simulation consist of five nodes, the source is *Node* Which generates a cbr traffic using udp data packet every .2 seconds. The sink is the *Node4*, which constantly has to receive the packets generated from the source. *Node4* moves in a linier manner from the range of *node0* to range of *node3* as shown in the figure 3. So every time when a data packet arrive without the route to destination the ROUTE REQUEST packet is created and broadcasted and when the ROUTE REPLY arrives ,if the route is broken in between, then a ROUTE ERROR is generated to the source node that generated send the data packet.

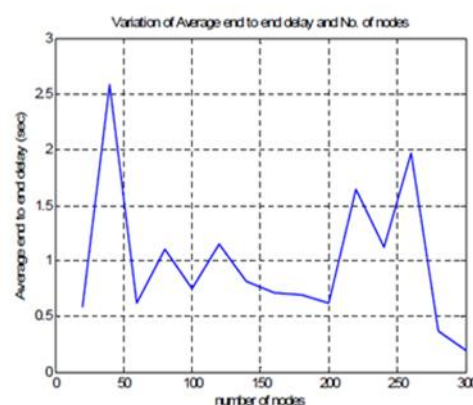
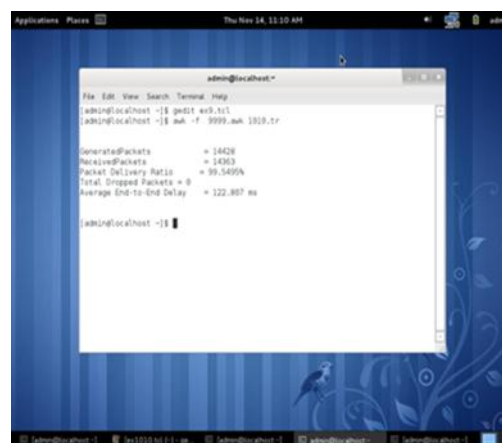
I. Simulation Setup

In this simulation study following network parameters and tools are selected:

1. IEEE 802.11 (ad-hoc mode) standard is used for each vehicular node.
2. 1400 bytes of UDP packets used for communication.
3. 15dbm Transmission power used for node operation.

Communication Model

Propagation Model	Two Ray Ground
Node Configuration	AdHoc Routing
Communication System	MAC IEEE 802.11
Total Number of Nodes	20
Type of Traffic	UDP



II. Future development

This implementation of DSR Source routing protocol includes most of the basic facilities described in DSR–VANET IEEE draft, but some optimization measures like cached route request, and flow control are not implemented in this implementation. Packet salvaging, automatic route shortening (Gratuitous

Route Reply), caching negative information, caching over head routing information and increased spreading of route error packets are the other options available in draft which can be implemented in this implementation The DSR protocol allows multiple routes to any destination and allows each sender to select and control the routes used in routing its packets (in this implementation the best route is determined by Dijkstra's shortest path algorithm) for example for use in load balancing or for increased robustness. Other advantages of the DSR protocol include easily guaranteed loop-free routing, support for use in networks containing unidirectional links, use of only "soft state" in routing, and very rapid recovery

when routes in the network change. The DSR protocol is designed mainly for mobile ad hoc networks of up to about two hundred nodes, and is designed to work well with even very high rates of mobility.

III. Conclusion


DSR protocol for ad-hoc networks works fine, but even it runs under some assumptions. This implementation works fine with less number of nodes, when the number of nodes increase then due to no much functionality in send buffer we face lot of problems like upper layers start to retransmit the packet again when it does not get an acknowledgement for the packet. In future implementations if these optional features are implemented then this implementation can be used for real ad hoc network in the unnum-bered footnote on the first page.

References

- [1] Charles.E.Perkins. AdHoc Networking.. Addison- Wesley, 2001.
- [2] draft-ietf-manet-dsr-09.txt IETF MANET Working Group David B. Johnson, Rice University INTERNET-DRAFT David A. Maltz, Carnegie Mellon University 15 April 2003 Yih-Chun Hu, Rice University.
- [3] S.A.Ade and P.A.Tijare, “Performance Comparison of AODV, DSDV, OLSR, DSR Routing Protocols in Mobile Ad Hoc Networks”, International Journal of Information Technology and Knowledge Management, July- December 2010, Volume 2, Number 2, pp:545-548.
- [4] Ravinder Ahuja, “Simulation Based Performance Evaluation and Comparison of Reactive, Proactive and Hybrid Routing Protocols
- [5] Based on Random Waypoint Mobility Model”, International Journal of Computer Applications, October 2010, Volume 7, Number11, pp: 20-24

- [6] S.P. Setti, K. N. Raju and K. N. Kumar, “Performance Evaluation of AODV in Different Environments”, International Journal of
- [7] Engineering Science and Technology, 2010, Volume 2, Number 7, pp: 2976-2981.
- [8] A.K.Sharma and N Bhatia, “Behavioral Study of MANET Routing Protocols by using NS-2”, International Journal of Computational
- [9] Engineering and Management, April 2011, Volume 12.
- [10] S.A. Gupta and R. K. Saket, “Performance Metric Comparison of AODV and DSDV Routing Protocols in MANETs using NS-2”,
- [11] June 2011, Volume 7, Number 3, pp: 339-350.
- [12] “AODV Group Presentation”, Available at: www.ece.iupui.edu, (Accessed on: 18 June 2012).

About Author (s)

	P.Geetharani completed B.E degree in the year 1995 &Mtech in the year 2006. At present persuing phd in wireless communication.
---	--