Simulation and Analysis of a New MANET Routing Protocol in Ns-2

Monali Sahoo[1], Ashok Verma[2], Ajay Lala[3] , Ashish Chaurasia[4]
[1] MTech Scholar Department of Computer Science & Engineering,
Gyan Ganga Institute of Technology & Sciences, Jabalpur, India
Email : monali.sahoo@yahoo.com
[2] Asst. Prof. Department of Computer Science & Engineering,
Gyan Ganga Institute of Technology & Sciences, Jabalpur, India
[3] Asst. Prof. Department of Information Technology,
Gyan Ganga Institute of Technology & Sciences, Jabalpur, India
[4] Asst. Prof. Department of Information Technology,
Gyan Ganga Institute of Technology & Sciences, Jabalpur, India
Email : ashishchaurasia@ggits.org

Abstract

A Mobile Ad hoc Network (MANET) is a dynamic wireless network that can be formed infrastructure less connections in which each node can act as a router. The nodes in MANET themselves are responsible for dynamically discovering other nodes to communicate. The Efficient routing protocols can provide significant benefits to mobile ad hoc networks, in terms of both performance and reliability. Many routing protocols for such networks have been proposed so far. Simulation is an efficient way of evaluating performance of MANETs . The Network Simulator -2 (Ns-2) has gained extreme popularity in past years. Network Simulator is a discrete event driven simulator. The goal of ns-2 is to support networking, research, and education. In this paper we add a New Routing Protocol called MyRTR step by step in Ns-2. We test this new implementation over Mobile Ad –hoc nodes and compare the results with other protocols.

1. Introduction

MANET provides a possibility of creating a network in situations where creating the infrastructure would be impossible or prohibitively expensive. Unlike a network with fixed infrastructure, mobile nodes in ad hoc networks do not communicate through the fixed structures. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. The ongoing trend is to adopt ad hoc networks for commercial uses due to their certain properties; hence these have been under maximum study and research. These are best evaluated using Network Simulators.

In this paper we give detailed steps to implement a New Protocol to Ns-2 and also present a comparative analysis w.r.t OLSR Routing Protocol.

2. Mobile Ad-hoc Network

MANET is a kind of wireless ad-hoc network and it is a self-configuring network of mobile routers (and associated hosts) connected by wireless links, the union of which forms an arbitrary topology. The routers, the participating nodes act as router, are free to move randomly and manage themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably. Such a network may operate in a standalone fashion, or may be connected to the larger Internet

Wireless devices communicate directly with devices inside their radio range in a peer-to-peer nature. If they wish to communicate with a device outside their range, they can use an intermediate device or devices within their radio range to relay or forward communications. Each mobile node acts as a host when requesting/providing information from/to other nodes in the network, and acts as router when discovering and maintaining routes for other nodes in the network.

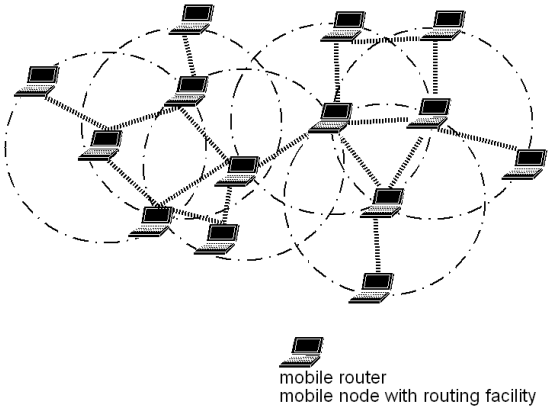

Figure 1.1 Mobile Adhoc Network

The routing protocols for MANETs can be broadly classified as on-demand/reactive and periodic/proactive protocols. Reactive routing protocols propagate route updates only when a route to destination is required. There are several reactive routing protocols available for ad hoc networks, including DSR, AODV, etc. On the other hand, proactive routing protocols such as DSDV maintain an active route to every neighbor.

3. OLSR Protocol

The Optimized Link State Routing protocol is a variation of the pure Link State Routing (LSR) protocol. OLSR produces the different links between the nodes as given by the source to destination. It is a proactive link state routing protocol based on the Open Shortest Path First (OSPF) protocol. First, it sends the HELLO message to check its neighbor. It makes the changes in the route table after every node, when the packet transmission is possible. For the distribution of signalling traffic OLSR adopts a flooding mechanism where every node forwards a flooded message that it has not forwarded previously [4]. The routing table is managed by the information of Topology control (TC) Packets. Control packet is sent to the network, by the special nodes called multipoint relays. Due to this control traffic is reduced; the path is selected by using the shortest path algorithm. Only MPR nodes forward messages for those neighbour nodes that selected them as an MPR node. Each node selects its MPR set of nodes in a way that, through them, it can reach all of its two-hop neighbours. A Topology Control (TC) message is sent to the whole MANET periodically by each MPR in the network to respectively declare its MPR selector set and is used in the construction of routing tables in every MANET node.

4. Network Simulator-2

Ns-2 is a discrete event simulator targeted at networking research and is widely utilized among academic researchers. It is an object oriented open source simulator written in OTcl and C++ [1] [2]. Ns2 provides substantial support for simulations of TCP, UDP, IP routing, and multicast protocols over wired and wireless networks, and it is supported by several research organizations. Helps to debug problems in a controlled environment. Ns2 also helps in performing Analysis of hypothetical changes.

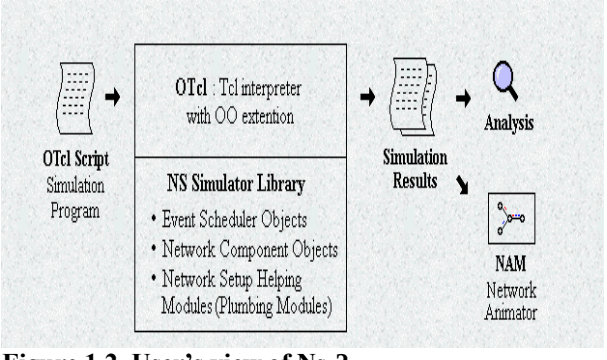

Figure 1.2 User's view of Ns-2

5. Steps to Add New Protocol to Ns-2

Network Simulator -2 is an object oriented open source simulator written in OTcl and C++ . Because it is open source ,new functions and new algorithms can be added by modifying the source files . We implement the new routing protocol by modifying codes from OLSR source code in ns-2. It has an implementation partly in OTcl and partly in C++.

We take the following steps as reference to allocate our code.

- Create a new directory called MyRTR inside NS2 base directory. Here include two files: MyRTR_agent.cc andMyRTR_agent.h These files are same as those for OLSR [DSDV can also be taken instead] . We have changed all classes, functions, structs, variables and constants names and added MyRTR implementation in the files.

- Compile it by editing makefile.in and defining all our .cc files as .o files under OBJ_CC = \ here to be compiled into ns2.

- Declare the new packet type constant " PT_MyRTR"to *~ns2source / Common / packet.h*

- Provide a name to the packet type under p_ifo() as for other protocols is done.

- We define protocol name MyRTR to use in tcl file by writing it under "for each prot " function present in ~/ns-2source/ns-2.29/tcl/lib/*ns-packet.tcl*

- Edit file ~ns/tcl/lib/ns-lib.tcl. see a code section having switch-case control "switch -exact $routingAgent_" Copy the same as other protocols and modify it for our new protocol. In same file write the function to instantiate the MyRTR and signals to start functioning .

- Add trace function to *~ns2source/trace/cmu-trace.cc* , so that NS2 trace our simulation and write it to a tr file and trace routines present in cmu-trace.cc are able to identify the packet of type PT_MyRTR.

- To add trace function we add code to *~/~ns2source /trace/cmu-trace.h* .

- Now after all settings done, run the MAKE command.

5. Simulation and Results

All extensive simulations are conducted in NS-2 . The New Protocol is created in Ns2.29.It is first validated by running a validate_My.tcl which can be a simply for Mobile Adhoc Network of 4 nodes. We have compared the results obtained by MYRTR with the OLSR protocol results. For this the simulated network consist of 15, 45 and 80 MANET nodes in radio range 60, 125 and 150 respectively.

5.1. Average End to End Delay (seconds)

It refers to the time taken for a packet to be transmitted across a network from source to destination. The average end to end delay of packet delivery was higher in OLSR as compared to MyRTR as shown in Fig 1.3. The New Protocol gives less E2E delay than OLSR.

5.2. Packet Delivery Ratio

This is the number of packets sent from the source to the number of received at the destination. The Optimized Link State Routing protocol OLSR performed particularly well MyRTR depicts gives better performance than OLSR (Fig. 1.4) .

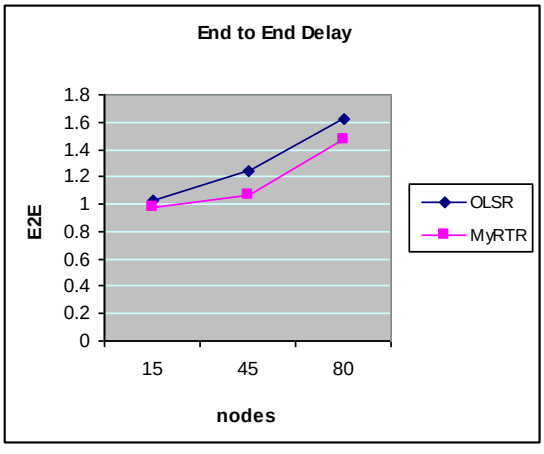

Figure 1.3 End to END Delay

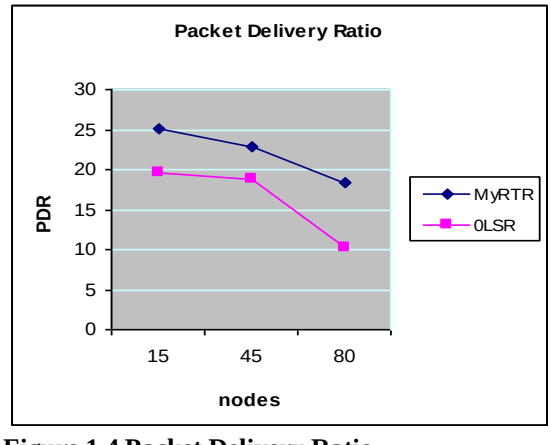

Figure 1.4 Packet Delivery Ratio

5.3. Routing Overhead

Defined as the average amount of routing protocol control packets in the network. For 15 , 45 ,80 nodes it is very clear that the routing overhead is less in case of our New protocol MyRTR than for OLSR (fig. 1.5) .

5.4. Hop Count

It is equal to the number of intermediate links in a path, hence it is equal to the length of a path with each link having a uniform weight of one. The Hop Cont for MyRTR (Fig. 1.6) is lesser than that for OLSR whereas the hop count increase with number of nodes for both MyRTR and OLSR.

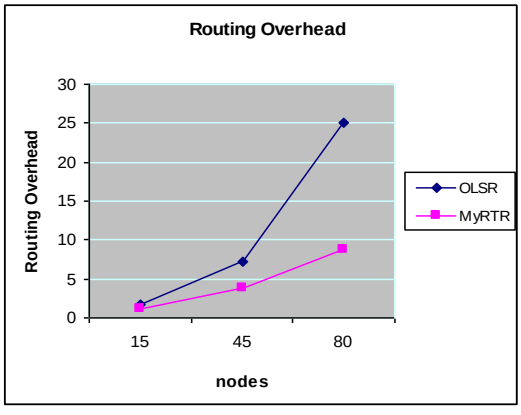

Figure 1.5 Routing Overhead

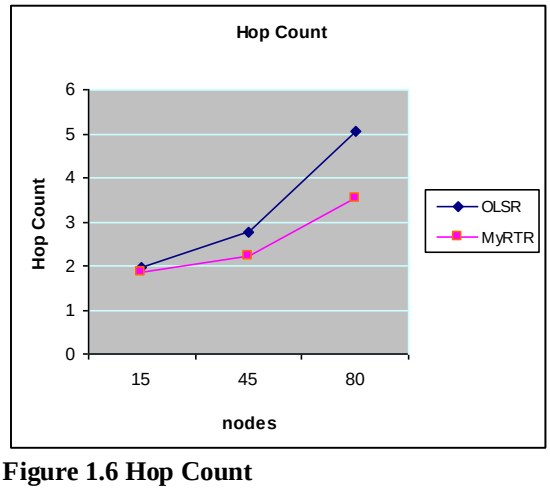

Figure 1.6 Hop Count

6. Conclusion and Future Work

In this paper , we give detailed step by step procedure to add a New MANET Routing Protocol to Ns-2 . Since Ns-2 is open source, new algorithms and function codes can easily be added to source files. We have also given a performance analysis of The New Protocol MyRTR with OLSR Routing Protocol. The Mobile Adhoc Network of 15, 45 and 80 nodes is considered. The New Protocol gives lesser End to End delay and Hop Count than OLSR. On increasing the no. of nodes Routing Overhead increases but it is comparatively lesser than OLSR . We can conclude that MyRTR gives satisfactory and better results than OLSR .

In the future, extensive complex simulations could be carried out using other existing performance metrics by increasing no. of nodes in order to gain a more in-depth performance analysis of the ad hoc routing protocols. We can compare it with other routing protocols as well.

7. References

[1] J. Chung and M. Claypool, "NS by Example". Available at http://nile.wpi.edu/NS/.

[2] NS-2, The ns Manual (formally known as NS Documentation) available at http: //www. isi.edu/nsnam/ ns/doc.

[3]Kiess W. and Mauve M. (2007) *Ad Hoc Networks*, vol. 5, no. 3, pp. 324–339, 2007.

[4] T. Clausen and P. Jacquet (2003), "Optimized Link State Routing Protocol (OLSR)", *RFC 3262, IETF Network Working Group*, October 2003.

[5]H. Ehsan and Z. A. Uzmi (2004), "Performance Comparison of Ad HocWireless Network Routing Protocols", *IEEE INMIC* 2004.

[6]Aleksandr Huhtonen, Comparing AODV and OLSR, Routing Protocols , Seminar on Internetworking, HUT T-110.551, Helsinki University of Technology, Sjökulla ,2004.

[7] Thammakit Sriporamanont and gu Liming, Wireless Sensor Network Simulator, Technical report, IDE0602,January 2006

[8]V. Nazari, K. Ziarati (2006), "Performance Comparison of Routing Protocols for Mobile Ad hoc Networks", *IEEE* 2006

[9] Hilal Karatoy , wordpress.com Available at http://hilalkaratoy.wordpress.com/ns-2/add-new-protocol-to-the-ns-2/ August 2009

[10]Jiazi YI, Asmaa ADNANE, Sylvain DAVID, Benoˆıt PARREIN Multipath Optimized Link State Routing for Mobile ad hoc Networks *Preprint submitted to Ad Hoc Networks September 28, 2010*

[11] Zhibin Wu , Network Simulator 2 for Wireless ,http://www.winlab.rutgers.edu/~zhibinwu/html/Routing_Agent.html#Agent

[12] Routing Protocols to Enhance Security in MANETS Rakesh Vanaparthiα, Pragati.G Global Journal of Computer Science and Technology Volume 11 Issue 13 Version 1.0 August 2011