IJCSMS International Journal of Computer Science & Management Studies, Vol. 12, Issue 03, September 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

Simulation Study and Implementation on Routing Protocols in MANET

Anju Yadav
M.Tech Scholar, Shekhawati Engineering College, Jhunjhunu, Rajasthan
anjuyadav02@yahoo.co.in

Abstract

Ad-hoc networks are characterized by a lack of infrastructure, and by a random and quickly changing network topology; thus the need for a robust dynamic routing protocol that can accommodate such an environment. Consequently, many routing algorithms have come in to existence to satisfy the needs of communications in such networks. This article presents a simulation study and comparison the performance between two categories of routing protocols, table-driven (Proactive) and on-demand (Reactive) routing protocols, this two categories were illustrated by using two different examples of routing protocols, first example is DSDV (Destination Sequenced Distance-Vector) from the Proactive family and the second example is AODV (Ad Hoc On-Demand Distance Vector) and DSR (Dynamic Source Routing Protocol) from the Reactive family. Both protocols were simulated by using NS-2 (network simulator-2) package. Both routing protocols were compared in terms of average throughput (packets delivery ratio) while varying data rate, TCP types ,maximum packets in queue and number of packet drop while varying data rate by using the Trace file. For the same queue length DSDV performs better than AODV. AODV is preferred due to its more efficient use of bandwidth.

Keywords: *Ad-hoc Networks, NS, routing protocol, AODV, DSDV.*

1. Introduction

An ad-hoc network is a collection of wireless mobile hosts forming a temporary network without the assistance of any stand-alone infrastructure or centralized administration . Mobile Ad-hoc networks are self-organizing and self-configuring multi-hop wireless networks where, the structure of the network changes dynamically. This is mainly due to the mobility of the nodes. Nodes in these networks utilize the same random access wireless channel, cooperating in a friendly manner to engaging themselves in multi-hop forwarding. The nodes in the network not only act as hosts but also as routers that route data to/from other nodes in network.A mobile ad-hoc network (MANET) is a kind of wireless ad-hoc network, and is a self-configuring network of mobile routers (and associated hosts) connected by wireless links the union of which form an arbitrary topology. The routers are free to move randomly and organize 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.

Mobile ad hoc networks became a popular subject for research as laptops and 802.11/Wi-Fi wireless networking became widespread in the mid to late 1990s. Many of the academic papers evaluate protocols and abilities assuming varying degrees of mobility within a bounded space, usually with all nodes within a few hops of each other, and usually with nodes sending data at a constant rate. Different protocols are then evaluated based on the packet drop rate, the overhead introduced by the routing protocol, and other measures.

The Children's Machine One Laptop per Child program has developed a cheap laptop for mass distribution (>1 million at a time) to developing countries for education. The laptops will use IEEE 802.11 based ad hoc wireless mesh networking to develop their own communications network out of the box. Vehicular Ad Hoc Networks (VANET) are a form of MANETs used for communication among vehicles and between vehicles and roadside equipment.

1.1 Characteristics of Mobile Ad Hoc Networks

I. Does not rely on a fixed infrastructure for its operation autonomous transitory association of mobile nodes.
II. It can be rapidly deployed with user intervention.
III. Need not to operate in a standalone fashion but can be attached to the Internet or Cellular networks.
IV. Devices are free to join or leave the network and they may randomly, possibly resulting in rapid and unpredictable changes.

1.1.1 Simple MANET Architecture

The specific MANET issues and constraints described above pose significant challenges in ad hoc network design. A large body of research has been accumulated to address these specific issues, and constraints.

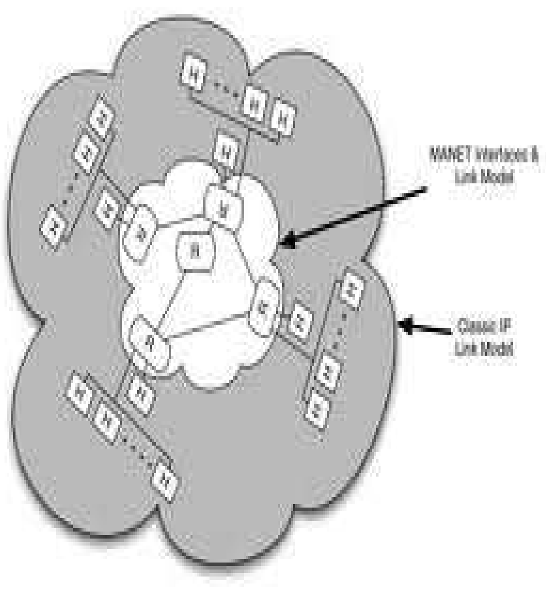

Figure 1.1

A simplified architecture is shown in Fig. the research activities are grouped, according to layered approach into three main areas:
1. Enabling Technologies
2. Networking
3. Middleware and applications
In addition, several issues (Energy management, security and cooperation, quality of service, Network simulation) span all areas.

1.1.2 Applications

With the increase of portable devices as well as progress in wireless communication, ad hoc networking is gaining importance with the increasing number of widespread applications. Ad hoc networking can be applied anywhere where there is little or no communication infrastructure or the existing infrastructure is expensive or inconvenient to use. Ad hoc networking allows the devices to maintain connections to the network as well as easily adding and removing devices to and from the network. The set of applications for MANETs is diverse, ranging from large-scale, mobile, highly dynamic networks, to small, static networks that are constrained by power sources. Besides the legacy applications that move from traditional infrastructure environment into the ad hoc context, a great deal of new services can and will be generated for the new environment. Typical applications include:

I. Military battlefield. Military equipment now routinely contains some sort of Computer equipment. Ad hoc networking would allow the military to take advantage of common place network technology to maintain an information network between the soldiers, vehicles, and military information head quarters. The basic techniques of ad hoc network came from this field.

II. Commercial sector. Ad hoc can be used in emergency/rescue operations for disaster relief efforts, e.g. in fire, flood, or earthquake. Emergency rescue operations must take place where non-existing or damaged communications infrastructure and rapid deployment of a communication network is needed. Information is relayed from one rescue team member to another over a small handheld. Other commercial scenarios include e.g. ship-to-ship ad hoc mobile communication, law enforcement, etc.

1.2 Working of MANET

In mobile ad-hoc networks where there is no infrastructure support and since a destination node might be out of range of a source node transmitting packets, a routing procedure is always needed to find a path so as to forward the packets appropriately between the source and the destination. A base station can reach all mobile nodes without routing via broadcast in common wireless networks. In the case of ad-hoc networks, each node must be able to forward data for other nodes. The following flow chart shows the working of any general ad hoc network.

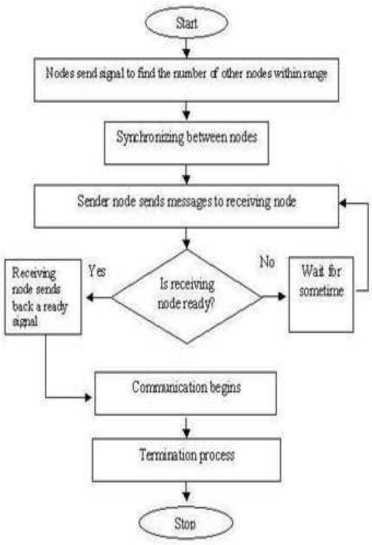

Figure 1.2: Flow chart of working of MANET

1.3 Advantages of Mobile Ad-hoc Networks

There are several advantages of using mobile ad hoc network.

1. Setting up a wireless system is easy and fast and it eliminates the need for pulling out the cables through walls and ceilings.
2. Network can be extended to places, which cannot be wired.
3. Multiple paths increase reliability.
4. Wireless network offers more flexibility and adapt easily to changes in the configuration of the network.

1.4 Limitations of Mobile Ad hoc Networks

There are certain constraints found in MANET, described below Asymmetric links: Most of the wired networks rely on the symmetric links, which are always fixed. But this is not a case with ad-hoc networks as the nodes are mobile and constantly changing their position within network. For example consider a MANET (Mobile Ad-hoc Network) where node B sends a signal to node A but this does not tell anything about the quality of the connection in the reverse direction.

Routing Overhead: In wireless ad hoc networks, nodes often change their location within network. So, some out-of-date routes are generated in the routing table, which leads to unnecessary routing overhead.

Interference: This is the major problem with mobile ad-hoc networks as links come and go depending on the transmission characteristics, one transmission might interfere with another one and node might overhear transmissions of other nodes and can corrupt the total transmission.

Dynamic Topology: This is also the major problem with ad-hoc routing since the topology is not constant. The mobile node might move or medium characteristics might change. In ad-hoc networks routing tables must somehow reflect these changes in topology and routing algorithms have to be adapted. For example in a fixed network routing table updating takes place for every 30sec. This updating frequency might be very low for ad-hoc networks.

2. IMPLEMENTATION OF MANET ROUTING PROTOCOL IN NS2

NS started as a variant of the original network simulator made in 1989 and many modifications are made during the past years. In 1995, the development of NS became supported by The Defense Advanced Research Projects Agency (DARPA) through the Virtual Inter Network Tested (VINT) project at Xerox Palo Alto research Center (PARC), and at the Information Sciences Institute (USC/ISI) of the University of Southern California, etc. Currently, NS development is supported through DARPA with Simulation Augmented by Measurement and Analysis for Networks (SAMAN) and National Science Foundation (NSF) with Collaborative Simulation for Education and Research (CONSER), in collaboration with other researchers including The ICSI (International Computer Science Institute) Center for Internet Research (ICIR).

2.1 Introduction to NS2

The implementation part is an important time of the project. We have chosen to work with NS2.NS2 is available under Linux, with a GPL license. Some standard algorithms are already implemented in this simulator, and DSR, AODV and DSDV is one of these.

NS2 is a network simulator; built with C++ and TCL. As every simulator, the main purpose is to simulate different networks, to test different protocols, and to find the limitations of each. It has been developed in the California University, by LBL, Xerox PARC, UCB, and USC/ISI through the VINT project supported by DARPA.

First, this simulator was build for fixed network: all links among nodes were wired. That means that the neighbor had no direct neighbor: if two nodes were very close, they don't communicate each other if they don't have a cable between each other. So, later, an extension for wireless network was developed by UCB Deedless, CMU Monarch projects and Sun Microsystems.

Now a days, this simulator is used around the world, because of the GPL license, and because it is a powerful simulator. It can be download form Internet, at this URL: http://www.isi.edu/nsnam/ns/. There are some tutorials to help beginners, and there is a lot of documentation (as the manual of ns2 - actually 400 pages).

The simulator is composed of two parts:

1. The TCL code: it is used to communicate with the simulator, and permits to define different simulation parameters.
2. The C++ code: it is the main part of the project, because it defines how the simulator has to behave

2.1.1. TCL: The Scenario Interface

The main reason for using a TCL language in ns2 is because it is not useful to use only C++ code. In fact, by this mean, the user does not need to compile the simulator every time he wants to do a new simulation. The TCL language is interpreted by the C++ code in ns2, without being compiled.

To use a network simulator, you have to define two things:

1. How does the protocol behave? It is done by the C++ code, because it does not change for every simulation
2. What are the simulation parameters? It is done by the TCL code, because every simulation is different (number of nodes, positions, speeds, protocol used)

The TCL code allows the user to choose between fixed or wireless network, and among the different implemented protocols: DSDV, AODV, DSR (for wireless networks).

The TCL file contains also information about nodes like position and speed, or information's about source and destination, the transmission rate, and a lot of other parameters. The syntax of this language is defined in the ns2 manual. Some tools have been developed to build these scenarios. For example, if we want to have a random model with several nodes, it is possible to use 'setdest'. It is a tool that generates random positions and random speeds for a number of nodes. By doing this, it is easy to use different random models and to test a protocol.

2.1.2. The C++ Code

Like C++, TCL is an object oriented language. So, there are parallels between C++ objects, the TCL objects. A C++ object can be used in the TCL language.

If the user needs to share a C++ object with the TCL code, he needs to use the Tcl Object. This class is eveloped in the tcltl package, independently to ns2. It allows defining the TCL name of the C++ object. Then the C++ object is used in the TCL file, using this name.

The TCL object communicates with its corresponding C++ object by using some basic commands. These commands are defined in the commands () function of the C++ commands, which takes the known parameters argc and argv. Thus, the TCL object can initialize the C++ object. The main purpose of the C++ code is to define how the simulator works, independently from the simulation parameters, and the results depend only on different initial conditions.
For instance, some C++ objects represent the different layers of the different nodes in the simulation. So, when a packet is sent from one node to another one, the packet goes through different C++ objects.
In the wireless layer, these packets are received in the recv() function if the node is the destination, and in the tap function if the node is a neighbor and can listen the packet even if it is not the destination.
The simulation runs with a specific simulation time, not the real one. Then, by sending packets, C++ objects creates some events: they want that one C++ object receives one packet at a time, by introducing a delay for example. But because ns2 is a mono process program, there is a table, containing all the events and sorting them according to the time they occur.

3. MOBILE NETWORKING IN NS2

The wireless model essentially consists of the Mobile Node at the core, with additional supporting features that allows simulations of multi-hop ad-hoc networks, wireless LANs etc.
To create mobile nodes, the user will firstly set the value of parameters which will be used when configuring mobile nodes .The OTcl code for setup of the mobile nodes part.

3.1. Trace file formats in wireless networks

Trace file is one of the text based results that the user gets from a simulation. It records the actions and relevant information of every discrete event in the simulation. There are a variety of forms for trace files. Simulations using different simulation networks or using different routing protocols could get trace files having different trace file formats. For example, a wired network and a wireless network have absolutely different format for recording each event. In the same network, for example in wireless networks, each routing protocol has its own format of the routing record. In the trace file, actions of different layers in the network can be traced. It includes agent trace, router trace, MAC trace and movement trace. All of
these traced events can be written to a file in a predefined format. When the user simulates large events, the trace file can be very large. It will not only require time to generate the trace file during simulation but also need space to store it. As a result, user should always choose part of the choices to trace. For example, in simulation of Section 5, the author would always take the agent trace and the router trace on, and MAC trace and movement trace off, since what the author interested in is the actions of nodes in the routing layer.

3.2. Tools used in NS2

3.2.1. Generation of node movement

A tool called setdest is developed by CMU (Carnegie Mellon University) for generating random movements of nodes in the wireless network. It defines node movements with specific moving speed toward a random or specified location within a fixed area. When the node arrives to the movement location, it could be set to stop for a period of time. After that, the node keeps on moving towards the next location. The location setdest is at the directory of ~ns/indep-utils/cmu-scen-gen/setdest/.

3.2.2 Traffic generation

To generate random flows of traffic, a Tcl script called .cbrgen. can be used. This script helps to generate the traffic load. The load can be either TCP or CBR. This script locates in the directory of ~ns/indep-utils/cmu-scen-gen. The file name is cbrgen.tcl. This tool is used according to the following command line ns cbrgen.tcl [-type cbr|tcp] [-nn nodes] [-seed seed] [-mc connections] [-rate rate] .

3.2.3. GAWK Tools used for data analysis

AWK is a computer program that is designed to process text-based data. GAWK is AWK developed by GNU (GNU is a recursive acronym for .GNU’s Not UNIX.). Using AWK, a

command file and an input file should be given. A command file can be a file or a command line input. The command file would tell AWK how to deal with the input file. It is composed of patterns and actions. For an input file, every line of the input file will be examined to judge whether this line matches the pattern. If this is the case, this line will be processed by the corresponding action. During the processing of the input file, GAWK will first separate the input file into
pieces of records. The record separator is .\n. by default. That is why AWK normally parses the file line by line. Each record is composed of several fields; different fields are separated by white space by default. In the command file,$1.represents the first field of a record. In actions, GAWK uses printf to print out the processing result.
BEGIN and END are two special patterns in GAWK. Their corresponding actions are executed only at the beginning and the ending of the execution of a command file.

4. DETAILS OF THE MANE ROUTING PROTOCOL IN NS2

There are two approaches for wireless communication between two hosts. The first is the centralized cellular network in which each mobile is connected to one or more fixed base stations (each base station is responsible for another cell), so that a communication between two mobile stations require to involve one or more base stations. A second decentralized approach consists based of an ad-hoc network between users that wish to communicate between each other. Due to the more limited range of a mobile terminal (with respect to a fixed base station), the approach requires mobile nodes not only to be sources or destination of packets but also to forward packets between other mobiles. Cellular station has a much larger range than ad-hoc networks. However, ad-hoc networks have the advantage of being quickly deployable as they do not require an existing infrastructure. In cellular networks, the wireless part is restricted only to the access to a network, and within the network classical routing protocols can be used. Ad-hoc network in contrast rely on special routing protocols that have to be adapted to frequent topology changes. To model well cellular networks, we will use the simulation tools of NS-2 (Network Simulator) with Linux operating system. In ad-hoc networks, the routing protocols are central. Na allows simulating the main existing routing as well as transport and applications that use them. Moreover, it allows taking in o account the MAC and link layer, the mobility, and some basic features of the physical layer.
The current routing protocol which we implemented by using NS-2 are:
1.DSDV - Destination Sequenced Distance Vector.
2. DSR - Dynamic Source Routing
3. AODV - Ad-hoc On Demand Distance Vector

4.1 The Routing Algorithms

There are several approaches in conventional algorithms in traditional wire line networks, and some ideas from these are also used in ad-hoc networks. Among the traditional approaches we shall mention the following:

I. Link State: Each node maintain a view of the complete topology with a cost per each link, each node periodically broadcasts the link costs of its outgoing links to all other nodes using flooding. Each node updates its view of the network and applies a shortest path algorithm for choosing the next-hop for each destination.
II. Distance Vector: Each node only monitors the cost of its outgoing links. Instead of broadcasting the information to all nodes, it periodically broadcasts to each of its neighbors an estimate of the shortest distance to every other node in the network. The receiving nodes use this information to recalculate routing tables using path algorithm. This method is more computation efficient, easier to implement and requires less storage space than link state routing.
III. Source Routing: Routing decisions are taken at the source and packets carry along the complete path they should take.

IV. Flooding: The source sends the information to all neighbors who continue to sending it to their neighbors etc. by using sequence numbers for the packets a node is able to relay a packet only once.

4.2. MANET routing protocol

4.2.1. Distance sequenced distance vector (DSDV)

DSDV is a distance vector routing protocol. Each node has a routing table that indicates for each destination, which is the next hop and number of hops to the destination. Each node periodically broadcasts routing updates. A sequence number is used to tag each route. It shows the freshness of the route, a route with higher sequence number is more favorable. In addition, among two routes with the same sequence number, the one with fewer hops is more favorable. If a node detects that a route to a destination has broken, then its hop number is set to infinity and its sequence number updated (increased) but assigned an odd number, even numbers correspond to sequence numbers of connected paths.

4.2.2 Ad-hoc On-Demand Distance Vector (AODV)

AODV is a distance vector type routing. It does not require nodes to maintain routes to destinations that are not actively used. As long as the endpoints of a comm. unication connection have valid routes to each other, AODV does not play a role. The protocol uses different to discover and maintain links, Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs). These message types are received via TCP, UDP, and normal IP header processing applies. AODV uses a destination sequence number for each route entry. The destination sequence number is created by the destination for any route information, it sends to requesting nodes. Using destination sequence numbers ensures loop freedom and allows which of several routes is more "fresh". Given the choice between two routes to a destination, a requesting node always selects the one with the greatest sequence number. When a node wants to find a route to another one, it broadcasts a RREQ to all the network till either the destination is reached or another node is found with a "fresh enough" route to the destination (a "fresh enough" route is a valid route entry for the destination whose associated sequence number is at least as great as that contained in the RREG). Then a RREP is sent back to the source and the discovered route is made available.

Nodes that are part of an active route may offer connectivity information by broadcasting periodically local Hello messages (special RREP message) to its immediate neighbors. If Hello messages stop arriving from a neighbor beyond some given time threshold, the connection is assumed to be lost.

When a node detects that a route to a neighbor node is not valid it removes the routing entry and sends a RERR message to neighbors that are active and use the route, this is possible by maintaining active neighbor lists. This procedure is repeated at nodes that receive RERR messages. A source that receives an RERR can reinitiate a RREQ message; AODV does not allow handling unidirectional links.

4.2.3. Dynamic Source Routing Protocol (DSR)

The Dynamic Source Routing protocol (DSR) was specifically designed for use in multi-hop wireless mobile ad hoc networks [2]. The DSR protocol does not require any existing network infrastructure or central administration and is completely self-organizing and self-configuring. This protocol basically consists of two mechanisms: Route Discovery and Route Maintenance, where the route discovery mechanism handles the establishments of routes and the route maintenance mechanism keeps update the route information. DSR is an on demand routing protocol, which means that no data is sent periodically and therefore it scales routing traffic and avoid the overhead package. The entire route in this routing protocol is known before the beginning of packet transmission; and it stores the route information in a Route Cache.

5. SIMULATION RESULTS OF THE PERFORMANCE METRICS

The performance measures which have been used for evaluating the performance of the three routing protocols by using the Trace file and compare the results with different nodes number.

1.1 Packet Delivery Ratio Vs Data Rate

Packet Delivery Ratio: it is the ratio between the number of packets received and the number of packets sent.

Packet Delivery Ratio= Total Received Packets/ Total Sent Packets

In our work, we have done the throughput evaluation through two different ad-hoc wireless networks just to confirm our results. The first ad-hoc wireless network contains 4-nodes only (see the diagrams below)

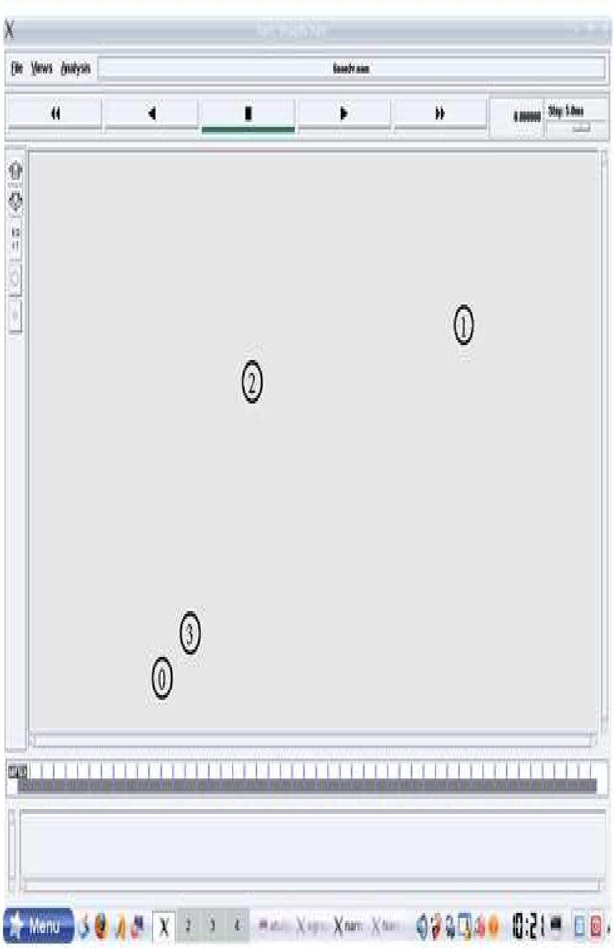

Figure 5.1: The topology of the network at time 0

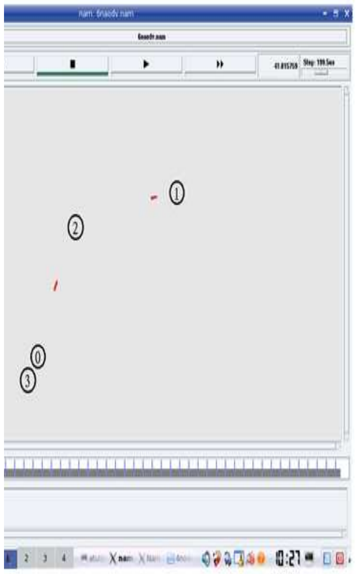

Figure 5.2: The topology of the network at time 41.01

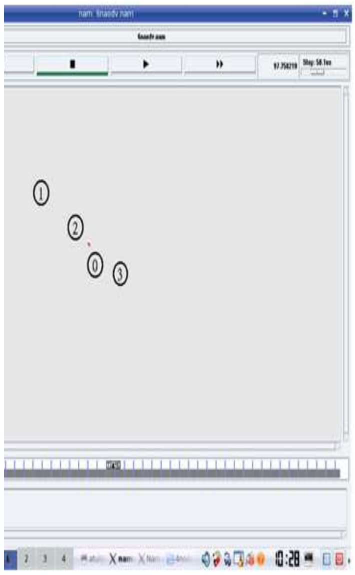

Figure 5.3: The topology of the network at time 97.75

6. CONCLUSION AND FUTURE SCOPE

In this article I have conducted a performance study of DSDV, DSR, and AODV routing protocols in Mobile Ad-hoc Networks (MANETs). I have evaluated the effect of speed, TCP types, and maximum queue length on throughput and the effect of speed on number of packet drops.

I. Data Rate: by examining that the throughput increases as the node speed increases in the beginning till it reaches a steady state between speed of 5 m/s and 15 m/s. As the speed increases above 50 m/s, there is a decrease in the throughput. Although DSR shows a slight better performance than AODV and DSDV in the steady state case between 5 m/s and 15 m/s, it performs worse speeds higher than 15 m/s.

II. TCP Types: In my analysis I have examined four types of TCP protocols; Newreno, Reno, Sack1, and Vegas. By examining noticed that DSDV has performed worse in all the four types of TCP protocol when compared to DSR and AODV. AODV has performed better than DSR except in the case of Vegas.

III. Maximum number of packets: By analyzing that the throughput for the two routing protocols is almost constant for a maximum queue length greater than 40. As the maximum queue length decreases, the throughput decreases. For the same queue length DSDV performs better than AODV.

IV. Packet drop: : By analyzing that the number of packets drop starts at a data rate of 350 m/s in case of DSR and AODV routing protocol and the number of packet drop is 249 from 50m/s to 700 m/s in case of DSDV routing protocol and the number of packets drops at a data rate of 700m/s in DSDV is greater than that of DSR and AODV routing protocol.

I have compared the performance of DSDV (Destination Sequenced Distance-Vector) from the Proactive family with the second type is AODV (Ad-hoc On-Demand Distance Vector) from the Reactive family. I used a detailed simulation model to demonstrate the performance characteristics of these protocols. By simulating we can argue that if delay is main criteria than DSDV can be my best choice but if reliability and throughput are main parameters for selection then AODV gives better results compared to others because its throughput and packet delivery ratio is best among others. While there are many other issues that need to be considered in analyzing the performance of ad-hoc networks, I believe that our work could provide intuition for future protocol selection and analysis in ad-hoc networks. While focusing only on the network throughput would be interesting to consider other metrics like power consumption, the number of hops to route the packet, fault tolerance, minimizing the number of control packets etc.

In the future, extensive complex simulations could be carried out to gain a more in depth performance analysis of the ad-hoc wireless networks and enhancing the performance and also for proposing new protocols and new algorithms to solve some of ad-hoc routing protocol problems.

7. References

[1]. http://en.wikipedia.org/wiki/Ns2

[2].David B Johnson, David A Maltz, Josh Broch, DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks, Mobile Computing, edited by Tomasz Emilienski and Hank Korth, Kluwer Academic Publishers, 1996.

[3].Marc Geris, NS Tutorial, available at: http://www.isi.edu/nsnam/ns/tutorial.

[4].David B Johnson, David A Maltz, Yih-Chun Hu, The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks (DSR), RFC Internet Draft, 2003.

[5].Network Simulation 2 official website: http://www.isi.edu/nsnam/ns/, visited 2006-07-22.

[6].NS manual http://www.isi.edu/nsnam/ns/ns-documentation.html, visited 2006-07-22.

[7].Marc Geris, NS Tutorial, available at: http://www.isi.edu/nsnam/ns/tutorial.

[8].Trace file record format http://wcc.iiita.ac.in/ns/ns2-trace-formats.html, visited 2006-07-22.

[9]. Elizabeth M.Royer and C.K.Toh, "A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks", IEEE Personal Communication Magazine, pp 46-55, April 1999.

[10].David B.Johnson and Dravid A. Maltz, “Dynamic Source routing in ad hoc wireless networks”, Technical report, Carneigie Mellon University, 1996.

[11].M. Abolhasan, T. Wysocki, and Eryk Dutkiewicz, “A review of routing protocol for mobile ad hoc networks”, Technical report, University of Wollongong, 2003.

[12].X. Hong, Kaixin Xu, and Mario Gerla, “Scalable routing protocols for mobile ad hoc networks”, IEEE Network Magazine, pp11-21, July-Aug 2002.

[13]. NS2 installation instruction under windows operating system: http://140.116.72.80/~smallko/ns2/setup_en.htm, visited 2006-07-22.

[14]. Mobile ad hoc network described in wiki http://en.wikipedia.org/wiki/Ad_Hoc_Routing_Protocol, visited 2006-07-22.

[15].Explanation of the term: data rate, http://en.wikipedia.org/wiki/Data rate, visited 2006-07-22.

[16].http://en.wikipedia.org/wiki/Ad_Hoc_Routing_Protocol#Pro-active_.28table-driven.29_routing

[17].Chakeres, I., "IANA Allocations for Mobile Ad Hoc Network (MANET) Protocols", RFC 5498, March 2009