RESEARCH ARTICLE OPEN ACCESS

A Simulated Behavioral Study of DSR Routing Protocol Using NS-2

Ali El-Desoky*,Amany Sarhan**, Reham Arnous***
*Department of computers and system engineering, Faculty of Engineering, Mansoura University, Egypt
** Department of Computer and system engineering, Faculty of Engineering, Tanta University.
**Department of computers and system engineering, Faculty of Engineering, Mansoura University, Egypt

Abstract
Mobile ad hoc network (MANETS) consists of wireless nodes communicating without any centralized administration .thus each node acts as a host and a router. There are many routing protocols designed for MANETS. The Dynamic Source Routing protocol (DSR) is a simple and efficient Routing protocol designed specifically for use in multi-hop wireless mobile nodes. This paper describes the design the implementation of DSR protocol in NS-2 simulator environment. We also analyze the performance of the DSR for various performance matrixes such as average end to- end delay, throughput and packet delivery ratio. The analysis was made for different number of nodes using the NS-2 simulator.
***Keywords* –** MANETs, DSR, routing protocol, performance evaluation, NS-2

I. Introduction

Wireless networks are emerging new technology where there is no physical wired connection between sender and receiver, but rather the network is connected by radio waves and/or microwaves to maintain communications. Wireless networks are rapidly evolving, and is playing an increasing role in the lives of people throughout the world as they provide connection flexibility between users in different places. Wireless networks are classified into two categories; Infrastructure networks and Ad Hoc networks. In Infrastructure networks devices are connected using a central device, namely a wireless access point. An access point (AP) represents a central coordinator for all nodes. AP acts as a master with all the other stations associating to it. All communications pass through the AP. A wireless ad hoc network is a decentralized type of wireless network. Ad hoc networks do not rely on a preexisting infrastructure or any centralized access points. Mobile ad hoc networks (MANETs) are self-organized and self-configuring infrastructurless networks of mobile devices communicating through wireless links. In MANETs the nodes which are in the transmission range of each other communicate directly, otherwise communication is done through intermediate nodes which forward packets; hence the node acts as a router and a node, these networks are also called as multi-hop networks [1, 2]. Routing in MANETs is considered a challenging task due to the mobility of nodes and limited bandwidth and energy.
There are several types of routing protocol developed for efficient routing in MANETs [3]. Theycan be categorized into two different approaches; position-based routing and topology-based routing [4-7].Position-based or geographic routing approaches rely on knowing the nodes' physical location information. Thus, it is necessary for nodes to obtain their coordinates either by using a location service such as GPS or other types of positioning services. In Topology-based routing protocols the paths are maintained and routes are established based on the information about the links that exist in the network. These protocols can be further divided into proactive (table driven), reactive (on demand) and hybrid approaches.

Table-driven (proactive) routing protocol: Proactive protocols discover routes and maintain them in routing tables, each node maintains up-to-date routing table containing a list of all the destinations, next hop, number of hops to each destination. Hello packets are exchanged periodically in order to inform nodes about changes in the topology. Destination sequenced distance vector (DSDV), wireless routing protocol (WRP) and global state routing (GSR) are examples of proactive routing protocols.

On demand (Reactive) routing protocol: The main idea in reactive routing (on-demand routing) is to find and maintain routes 'only when there is needs (on demand) which by turn minimize routing overhead but it may result in initial delays until the routes are set up. There are different types of on demand routing protocols such as Ad hoc On Demand Distance Vector (AODV), Dynamic Source routing protocol (DSR), temporally ordered routing algorithm (TORA), Ad-hoc On-demand Multipath Distance Vector Routing (AOMDV). Our discussion is limited to DSR protocol.

Hybrid routing protocols area combination of both reactive and proactive approaches. Zone routing protocol (ZRP) is an example [8-11].

II. DSR protocol description

The DSR is a pure reactive routing protocol based on the concept of source routing. it employs route discovery and route maintenance procedures working together in a way that that can enable nodes to discover and maintain routes to arbitrary destinations in thenetwork [12-15]. Each node maintains its route cache which is kept in the memory to save the discovered route for future use. The node updates entries in the route cache when it learns about new routes.The original DSR keeps multiple routes to a destination ordered by hop count. It selects a route having the minimal number of hops between the available routes. DSR uses the route selection based on the shortest path algorithm; it selects a route having the minimal number of hops between the available routes. When a source node wants to send a data packet to destination node, it first checks its route cache for a route to the desired destination; If a route is found the node uses it to send the packet. The route is inserted in the packet header and then the packet is forwarded to its destination through this route. In case there is no route to the desired destination in the node's route cache, it applies a route discovery process by broadcasting a route request packet (RREQ) to all its neighbors. The source node appends source ID, destination ID and a unique request ID to it. The packet also contains a route record listing the address of each intermediate node it passes through. The source route initializes this record as an empty list. When an intermediate node receives a RREQ packet it appends its id to the RREQ packet's route- record then it rebroadcasts the RREQ again, unless it is the destination node, or it has a route to the intended destination in its cache. When initiating a Route Discovery, the sending node saves the original packet in a local buffer called the Send Buffer. Each packet in the Send Buffer is stamped with the time that it was placed into the Buffer the packet is saved in the send buffer until a route found or it reaches time out period. The packet is discarded in case of timeout. Timeout period is necessary for preventing the Send Buffer from overflowing. When the route request reaches either the destination itself or a node that has a route to the destination**,** a route reply (RREP) is generated**.** The RREP is then sent back (through the reverse path found in the route record of RREQ) to the source node. The source node then appends this path to the original packet saved in the send buffer and forwards it to destination through it. It also updates its route cache with this new learned route [16-20]. For route maintenance procedure; DSR uses two types of packets; route error packets and acknowledgment. When a node detects a transmission problem on forwarding a packet, it generates a route error packet while acknowledgment packets are used to confirm the correct operation of the route links [21-25].

III. DSR implementation in NS-2 simulator

NS-2 is a an object oriented ,open source network simulation tool which can be used by researchers working on wired or wireless networks as it supplies numerous models and source codes of routing protocols. There are two programming languages used in NS-2 otcl and C++ language. C++ is used for detailed protocol implementation where otcl language is used to write script files (simulation script) where users describe the model to be simulated then NS-2 interpret the script and output the simulation process and result to an output file (trace file). The simulation process and result can be shown by Nam and Xgraph files [26, 27]. There are two main types of NS-2 agents routing agents and transport layer agents. A routing agent creates, transmits and receives control packets and routing protocol commands to act accordingly, where as Connecting an application to a low level network, a transport layer agent controls the congestion and reliability of a data flow based on an underling transport layer protocol .NS-2 implements agents in C++ [28, 29]. The DSR routing protocol is implemented in NS-2as Agent (DSR Agent). This section analyses the DSR source code in NS-2. The chart in figure 1 shows the basic functions of the DSR agent. When a DSR agent receives a packet it first arrives at recv method (basic function) which first checks the packet header to see if it has a valid source route (srh) then calls different functions to handle the packet as shown in flow chart described in figure 1. If no source route found in packet header and it is not a broadcast packet, it will be handled through the method named handlePktWithoutSR which is described later in the flow chart shown in fig 2. If there is a valid source route then the packet is handled through methods handlePacketReceipt described in figure 3. If the current node is the destination (P.dest= net_id) else it is handled either through the method handleRouteRequest (described in flowchart shown in figure 4) if it is a RREQ packet, or through handleForwarding method if it is not a RREQ packet.

IV. Simulation

In this paper the DSR is analyzed using the NS-2 simulator. Three performance metrics are used to evaluate the performance Average End-To-End

Delay, Throughput, and packet delivery ratio for various numbers of nodes considering TCP as the transport protocol and FTP as the traffic generator.

4.1. Simulation tool and parameter

The simulations tool used is NS-2. Before starting the simulation a tcl script is created that describes the simulated model which defines the network topology composed of nodes, routers, links, and shared media. Otcl language is used to write the script file**.**Table 1 shows a part of the tcl script containing the tcl coding which specify the simulation parameters.

Table 1 tcl code for simulation setup parameters

```
Set val(chan)       channel /wireless channel      ;# channel type
Set val(prop)       propagation /two rayground     ;# radio-propagation model
Set val(netif)      phy/wirelessphy                ;# network interface type
Set val(mac)        mac\802_11                     ;# mac type
Set val(ifq)        cmupriqueue
Set val(ll)         ll
Set val(ant)        Antena\omniAntena
Set val(ifqlen)     50                             ;# max packet in ifq
Set  val(nn)        120                            ;# number of mobile nodes
Set val(rp)         dsr                            ;# routing protocol
Set val(x)          500
Set val(y)          400
Set val(stop)       150                            ;# time of simulation end
```

4.2. Simulation analysis and Performance metrics

After running the tcl script at the end of each simulation, a NAM trace file and a tr file which is a trace file will be created by NS-2 automatically. NAM trace file is used as an input to network animator (NAM) which is used to graphically visualize the simulator while the trace file is used for simulations analysis. AWK Scripts are used for processing the data from the output (trace files). The performance of the simulation is evaluated according to the following performance metrics.

Average end to end delay: Average End-to-End delay is the average time it takes a data packet to reach the destination. Low average end-to end delay indicating good performance. it is calculated using awk script (delay.awk) which processes the trace file and produces the result.Table3 represents the average end-to-end delay for all packets received for (20,30,40,50,60,70,80,90,100,110,120,130,140,150) nodes.

Table 2. Average End-to-End Delay of DSR at different numbers of nodes.

No of nodes	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**	**110**	**120**	**130**	**140**
Delay (ms)	**199.470**	**191.878**	**179.747**	**199.385**	**205.325**	**147.317**	**148.673**	**223.399**	**179.043**	**176.976**	**209.569**	**171.902**	**210.424**

DSR end to end delay decreases with increase the number of nodes till 80 nodes. So DSR protocol good for nodes less than 80. Increasing number of nodes Increases the number of hops in routes and in its turn increases the time it take the packet to reach the destination.

Throughput: The throughput is defined as the total amount of received data packets divided by the simulation time. The throughput is measured in bits per second (bps). Table3 represents the throughput for different numbers of nodes.

Table3. Throughput at different numbers of nodes

Number of nodes	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**	**110**	**120**	**130**	**140**
Throughput (kbps)	**521.33**	**536.66**	**542.38**	**541.64**	**522.02**	**577.38**	**558.77**	**521.43**	**520.64**	**556.38**	**522.92**	**557.45**	**532.46**

DSR throughput increases with increase the number of nodes till 80 nodes. With the increase of number of nodes than 80 the throughput decreases. This may result due to the increase of routing overhead by increasing the number of nodes.

Packet Delivery Ratio: Packet delivery ratio is the ratio of number of packets received at the destination to the number of packets sent from the source. The performance is better when packet delivery ratio is high .The reason for having better packet delivery ratio of DSR is that allow packets to stay in the send buffer until a valid route found then data packets are sent on that route to be delivered at the destination.

Table 4. Packet delivery ratio at different numbers of nodes.

No of nodes	20	30	40	50	60	70	80	90	100	110	120	130	140
No of packets sent	1337	1378	1258	1256	1339	1201	1296	1340	1210	1294	1341	1293	1366
No of received packets	1334	1376	1255	1253	1336	1195	1294	1338	1204	1292	1339	1291	1364
Packet delivery	.998	.998	.998	.997	.998	.995	.998	.998	.995	.998	.998	.998	.998

V. Conclusion

In this paper, a detailed description of DSR routing algorithm and its implementation as arouting agent (DSR agent) in NS-2 is presented. The performance of DSR is evaluated in terms of throughput,average end-to-end delay, and packet delivery ratio for various numbersof nodes using the NS-2 simulator. From the above simulation result analysis, we can conclude that DSR provides good performance for routing in wireless ad hoc networks.

VI. Future work

DSR protocol for routing in ad-hoc networks works fine where nodes are not equipped with GPS or any other device to be aware of its own positions.Networks are assumed to be heterogeneous; some nodes may be aware of their position through GPS while some others are not. We are currently working to develop a routing algorithm which is able to take the advantages of position information of nodes when available but also to be able to continue the routing through a reactive on demand fashion like DSR scenario.

References

[1.] SS. Dhenakaran, and A Parvathavarthini. An overview of routing protocols in Mobile ad-hoc network. *International journal of advanced research in computer science and software engineering*, 3(2), 2013, 251-259.
[2.] M Kumar, N Geethanjali, and N. Ramesh Babu. Performance evaluation of AODV and FSR routing protocols in MANETS. *Global Journal of computer science and technology*, 13(4), 2013, 29-34.
[3.] A. Odeh, E. AbdelFattah, and M. Alshwokan. Performance evaluation of AODV and DSR routing protocols in MANET networks. *International journal of distributed and parallel systems*, 3(4), 2012, 13-17
[4.] K. Pandey K, and A. Swaroop. A comprehensive performance analysis of proactive, reactive and hybrid MANETs routing protocols. *International journal of computer science,* 8(6), 2011, 432-436
[5.] K. Vanaja, and R. Umaram. A simulated behavioural study of prominent reactive routing protocol AODV of MANET using NS-2. International journal of advanced research in commuter and communication engineering, 1(3), 2012, 161-166
[6.] H. Soni, and P. Verma. Performance based survey of routing protocol in MANET. International Journal of Advanced Research in Computer Engineering & Technology, 2(1), 2013, 145-154
[7.] K. Patel, R. Kumar, andK. Daniel. Performance analysis and behaviouralstudy of proactive and reactive routing protocols in MANET. *International journal of advanced research in computer and communication engineering*, 2(4), 2013, 1789-1796
[8.] KR. Reddy, S. Venkata Raju, and N. Venkatadri. Reactive, proactive MANET routing protocol comparison. *International journal of Video&image processing and network security*, 12(5), 2012, 22-27
[9.] G. Balamurugan, P. Sakthivel, and C. Jayanavithraa. The comparative study of adhoc network routing protocols. *International Journal of Advanced Research in Computer and Communication Engineering*, 3(2), 2014, 5337-5340.
[10.] A. Hinds, M. Ngulube, S. Zhu, and H. Al-Aqrabi. A review of routing protocols for mobile ad-hoc NETworks (MANET). *International journal of information and education technology*,3(1), 2013, 1-5
[11.] TP. Singh, S. Dua, and V. Das. Energy-efficient routing protocols in mobile ad-hoc networks. *International journal of advanced research in computer science and software engineering*, 2(1), 2012, 132-137.
[12.] R. Saleh, M Saman, and N. Rahman N. A new policy for path selection in dynamic source routing protocol for mobile ad-hoc networks. *Journal of computer science*, 9(5), 2013, 551-557

[13.] K. Manoj, R. Kumar, and K. Arti. Analysis of delay and throughput in MANET for DSR protocol. *world academy of science, engineering and technology*, 5(2), 2011, 653-656
[14.] S. sofat, and A Gupta. A Survey on Throughput Analysis of DSR Routing Protocol in MANETs.*International Journal of Multidisciplinary in Cryptology and Information Security*, 2(2), 2013, 56-63
[15.] N. Khan, and N. Agarwal. Adaptive routing in mobile adhoc networks and comparison between routing protocols aodv and dsr for apu strategy. *Journal of computer science* 2014 (ICAET-2014): 58-65
[16.] R. Jagannadha, K. Sreenu, and P. Kalpana. A study on dynamic source routing protocol for wireless ad-hoc networks. *International Journal of Advanced Research in Computer and Communication Engineering*, 1(8), 2012, 522-529
[17.] Y. Saravanan, and D Vijayakumar. Performance of reactive and proactive MANET routing protocols with trajectories. *International Journal of Engineering Research and Technology,* 1(8), 2012, 24-29
[18.] D. NareshVadhwani, D. Kulhare, and M. Singh. Behavior analysis of DSR MANET protocol with http traffic using opNET simulator, *The International Journal of Innovative Research in Computer and Communication Engineering*, 1(3), 2013, 680-686.
[19.] S. Ali and A. Ali.*Performance analysis of AODV, DSR and OLSR in MANET* (LAP LAMBERT Academic Publishing, 2011).
[20.] R. Sharma, and S. Sabharwal. Dynamic source routing protocol (DSR). *International journal of advanced research in computer science and software engineering*, 3(4), 2013, 293-241
[21.] S.V. Gumaste, M.U. Kharat, and V.M. Thakare. Performance analysis of DSR protocol. *International journal of scientific and engineering research*, 1(3), 2013, 62-68
[22.] R. Saleh, M. Saman, and M. Rahman. Impact of route selection metrics on the performance of DSR protocols for MANETs. *International Journal of Computer Science Issues*, 10(2), 2013, 123-126
[23.] D. Goel. A new approach of route selection in DSR using time routing metric in MANET. *International Journal of Advanced Research in Computer Science and Software Engineering,* 4(4), 2014, 360-363
[24.] P Rohal, R. Dahiya, and P. Dahiya. Study and analysis of throughput, delay and packet delivery ratio in MANET for topology based routing protocols (AODV, DSR, and DSDV). *International Journal of Advanced Research in Engineering and Technology*, 1(2), 2013, 54-58
[25.] D.N. Vadhwani, M. Singh, D. Kulhare. Simulation of wireless dynamic source routing protocol with IP traffic flow analysis, memory efficiency and increased throughput. *IJCom*, 3(4), 2013, 136-141
[26.] S. Upadhyay, P. Joshi, N. Gandotra, and A. Kumari. Comparison and performance analysis of reactive type DSR, AODV and proactive type DSDV routing protocol for wireless mobile ad-hoc network, using NS-2 simulator. *Journal of engineering and computer innovations,* 2(10), 2012, 36-47
[27.] M. Chandrika, and N. Papanna. Performance comparison of AOMDV and POR routing protocols in MANETs. *International Journal of Engineering Research and Applications*, 3(4), 2013, 37-41
[28.] T. Issariyakul, and E. Hossain, *Introduction to network simulator NS2* (New York: Springer, 2012
[29.] A. Gani, Q. Han, N. B. adrulAnuar, and O. Zakaria. Enhancing DSR protocol performance in mobile ad hoc network using ACK reply. *WSEAS TRANSACTIONS On Communications,* 2(2), 2009, 227-236

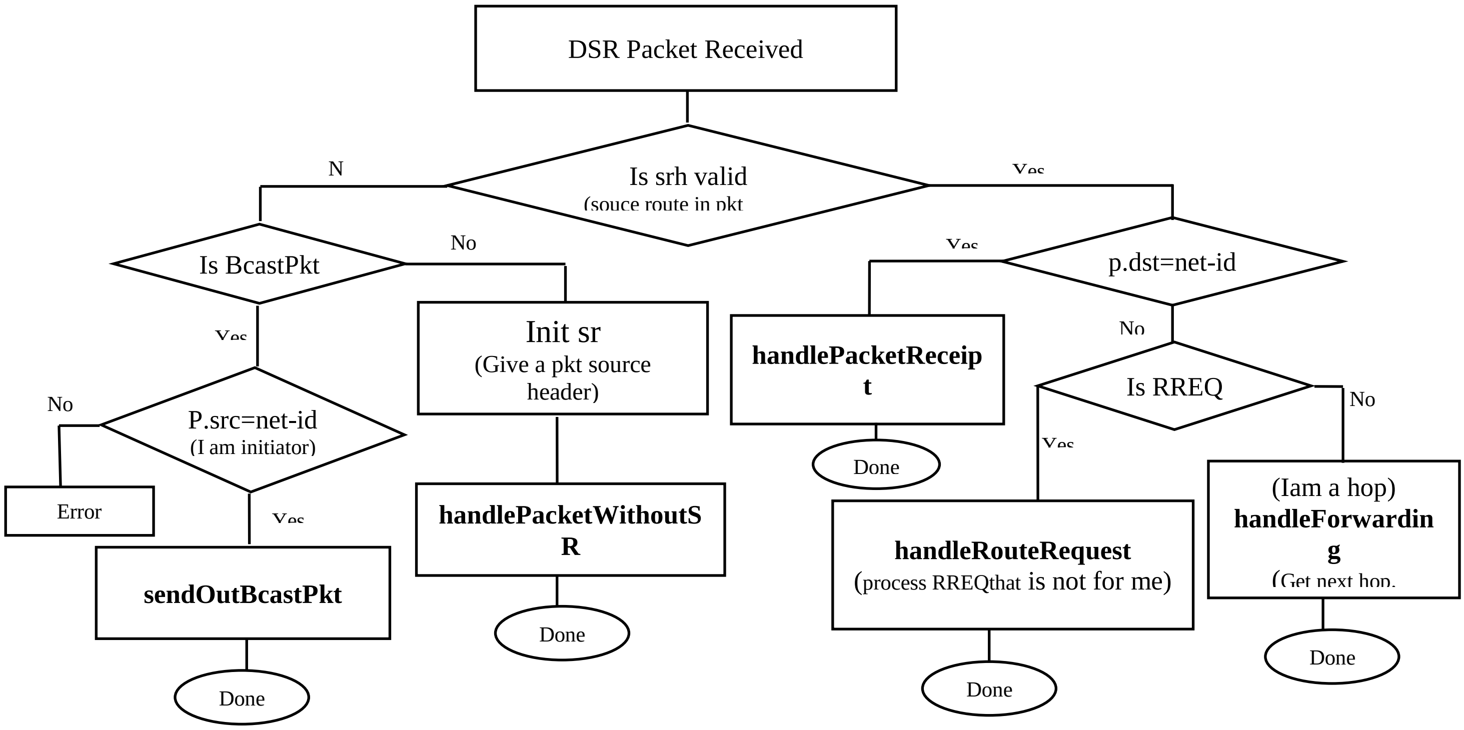

Fig1. Handling packet arrival in DSR agent implemented in NS2

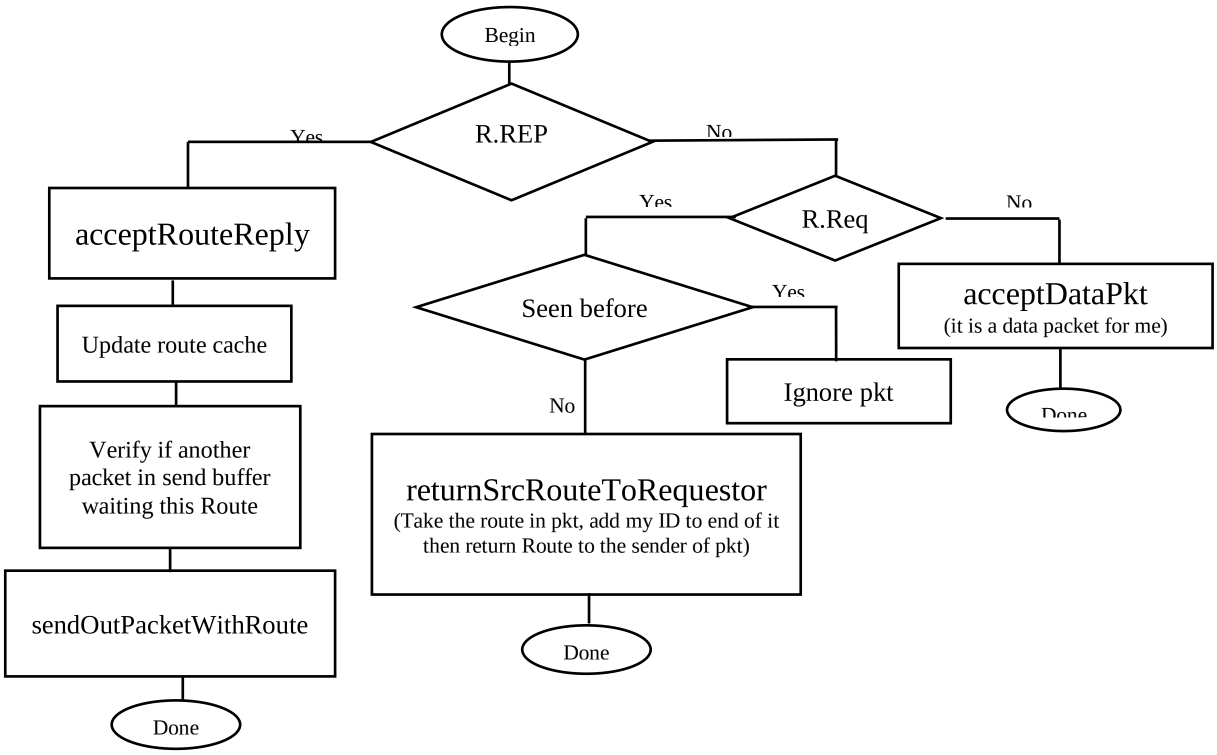

Fig3. handlePacketReceipt method

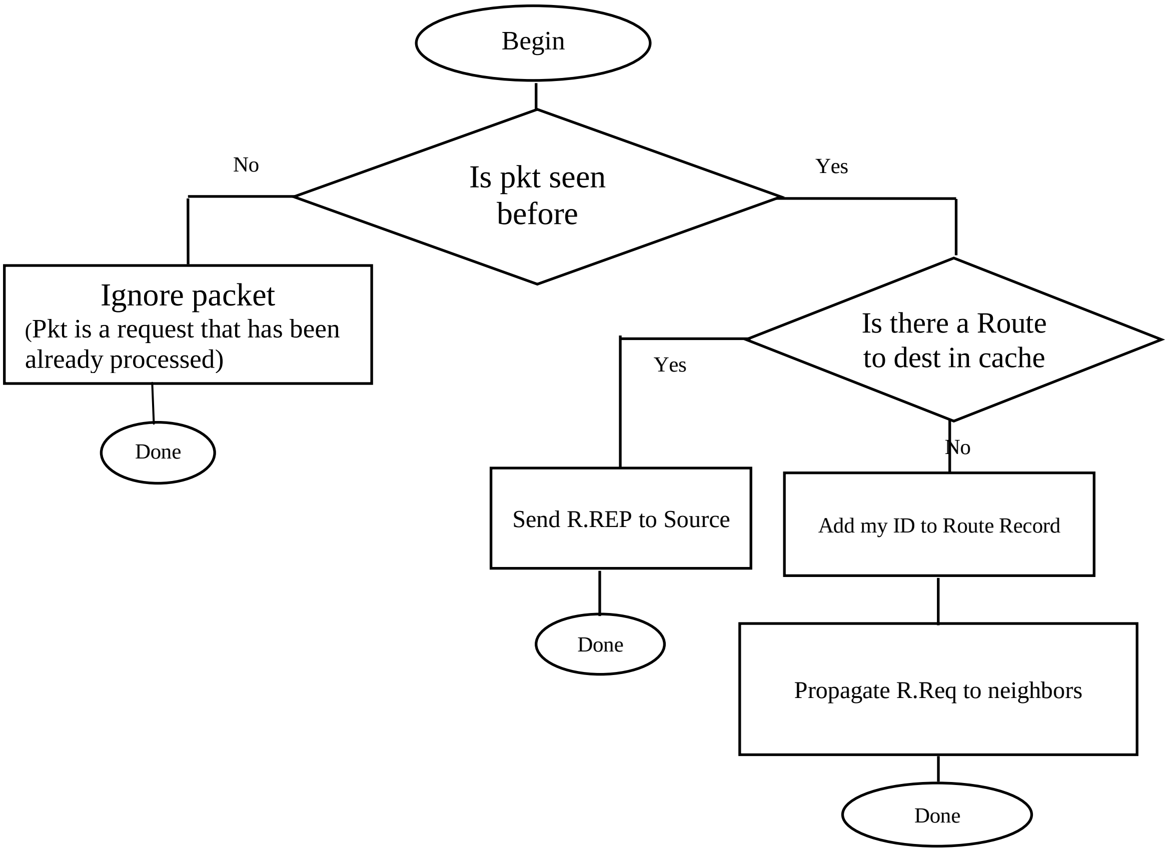

Fig 4. handleRouteRequest method

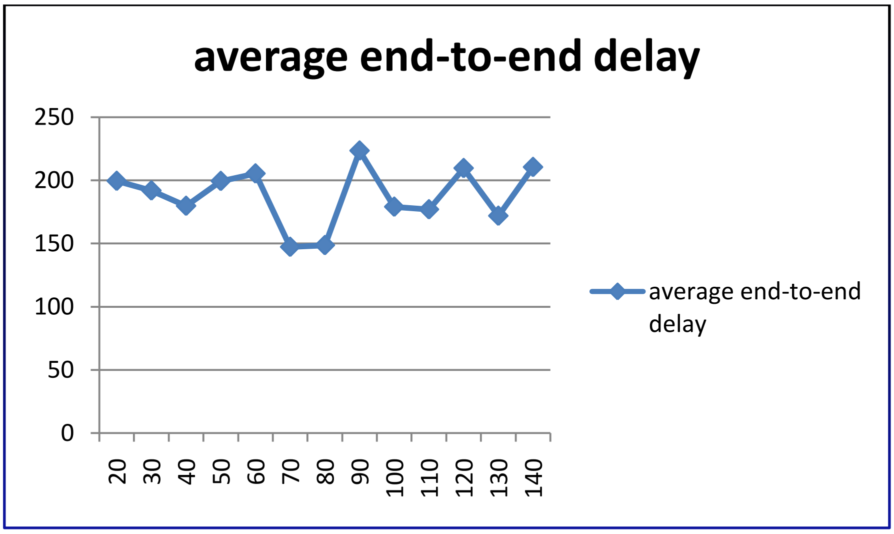

Fig 5. Average-end-to-end delay for number of nodes.

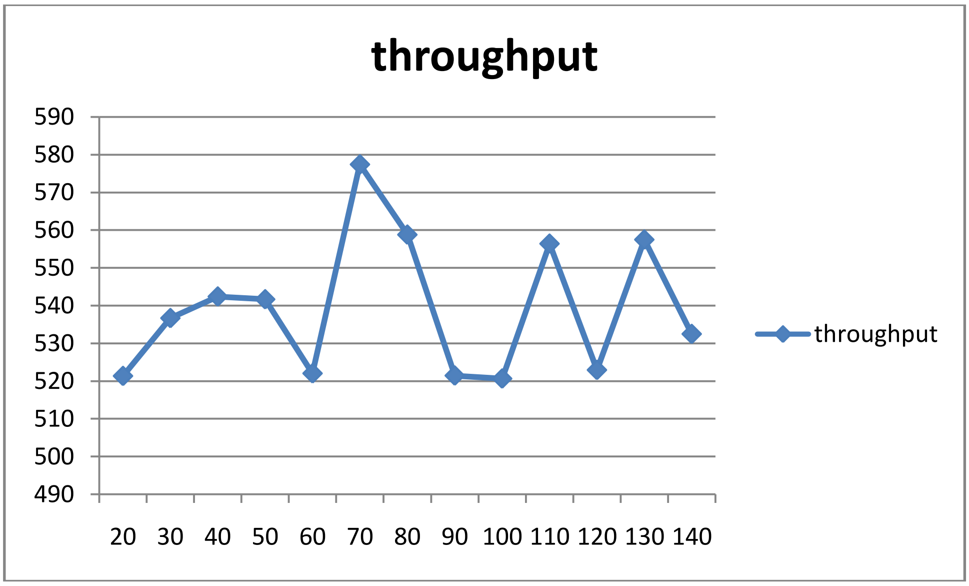

Fig 6. Throughput over number of nodes.

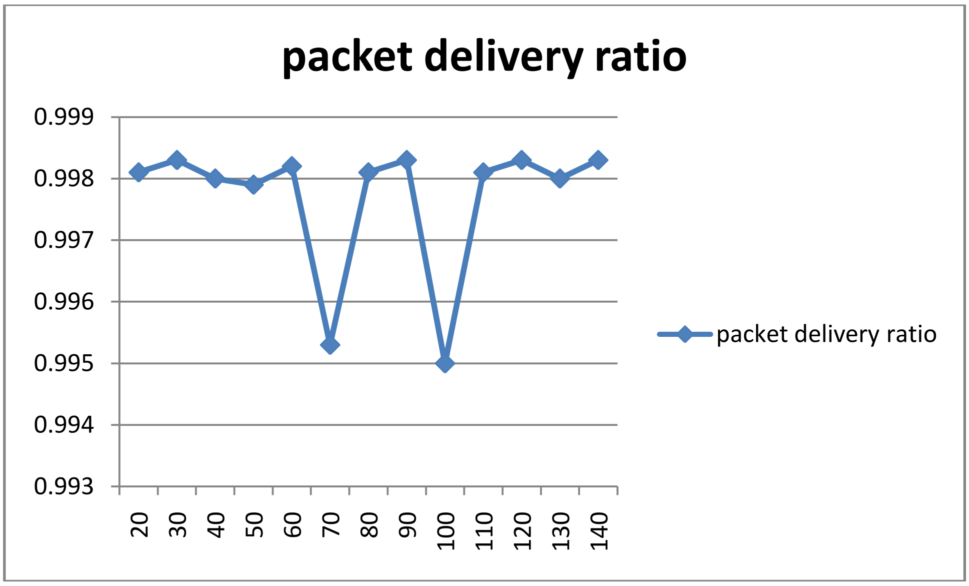

Fig 7. Packet delivery ratio at different nodes.