© December 2021| IJIRT | Volume 8 Issue 7 | ISSN: 2349-6002

Simulation Based Analysis of Delay & Throughput in MANET(IEEE 802.11b) for TORA Protocol

Kumari Arti[1], Dr Kumar Manoj[2]
[1,2] *Department of Electronics and Communication, Himalayan University*

***Abstract* - A wireless Ad-hoc network consists of wireless nodes communicating without the need for a centralized administration, in which all nodes potentially contribute to the routing process. In this paper, we report the simulation results of four different scenarios for wireless ad hoc networks having thirty nodes. The performances of proposed networks are evaluated in terms of number of hops per route, delay and throughput with the help of OPNET simulator. Channel speed 5.5 Mbps and simulation time 600 sim-seconds were taken for all scenarios. For the above analysis DSDV routing protocols has been used. The throughput obtained from the above analysis (four scenario) are compared as shown in Figure 3. The average media access delay at node_20 for two routes and at node_18 for four different scenario are compared as shown in Figures 4 and 5. It is observed that the throughput will degrade when it will follow different hops for same source to destination (i.e. it has dropped from 1.85Mbps to 1.72 Mbps which is around 12.9%, and then dropped to 0.88Mbps which is around 37%).**

***Index Terms* - Throughput, Delay, DSDV, OPNET, MANET, DSSS.**

1.INTRODUCTION

A wireless Ad-hoc network consists of wireless nodes communicating without the need for a centralized administration. A collection of autonomous nodes or terminals that communicate with each other by forming a multihop radio network and maintaining connectivity in a decentralized manner is called an ad hoc network. There is no static infrastructure for the network, such as a server or a base station. The idea of such networking is to support robust and efficient operation in mobile wireless networks by incorporating routing functionality into mobile nodes. Figure1 shows an example of an ad hoc network, where there are numerous combinations of transmission areas for different nodes. From the source node to the destination node, there can be different paths of connection at a given point of time. But each node usually has a limited area of transmission as shown in Figure 1 by the oval circle around each node. A source can only transmit data to node B but B can transmit data either to C or D. It is a challenging task to choose a really good route to establish the connection between a source and a destination so that they can roam around and transmit robust communication.

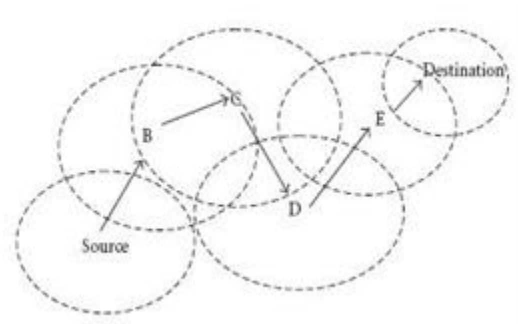

Figure 1. Ad hoc networking example

In this paper, OPNET simulator [3] has been used to simulate the network as proposed in Figure2 in which 30 nodes have been taken for the analysis with four mode of operation as shown in Table1.

Table 1: Different Scenario

Scenario	No. of Hopes	Route
First	Three	Between node_18 to node_1
Second	Five	Between node_18 to node_1
Third	Six	Between node_18 to node_1
Fourth	Seven	Between node_18 to node_1

The following system parameters are taken for the simulation of all of the above scenario at channel speed 5.5 Mbps and simulation time 600 sim-seconds. The comparative studies of the simulation results for these parameters are also reported.

1. Number of hops per route,
2. Delay,
3. Throughput,

DSDV routing protocols proposed by [4] has been above analysis. We evaluated all available metrics supported by OPNET for this protocol. To the best of our knowledge, very few papers are reported in the literature, which compares the simulation results with different scenario reported in this paper. This work is the one of the major comprehensive performance evaluation of ad hoc routing protocols using OPNET Modeler [3]. We also simulated this protocol under different loads (number of nodes in a network) and showed their corresponding performance differences. In Section-2, a summary of the Ad hoc routing protocols have been reported. The simulation software and the network simulation setup are described in section 3. In this paper the simulation results & conclusion has been reported in section 4 & 5 respectively.

2. AD HOC ROUTING PROTOCOLS

Most widely used routing protocols for wireless ad hoc networks used in OPNET simulator available till today are DSR [7], AODV[8], destination sequence distance vector (DSDV)[6], and TORA[9] ad hoc routing protocols. All these protocols are constantly being improved by IETF
[1]. Since these protocols have different characteristics, the comparison of all performance differentials is not always possible. DSDV routing protocols has been used for the present analysis.

2.1 Destination-Sequenced Distance Vector (DSDV)

DSDV [6] is a hop-by-hop distance vector routing protocol requiring each node to periodically broadcast routing updates. The key advantage of DSDV over traditional distance vector protocols is that it guarantees loop-freedom.

2.1.1 Basic Mechanisms: Each DSDV node maintains a routing table listing the “next hop” foreach reachable destination. DSDV tags each route with a sequence number and considers a route R more favorable than R’ if R has a greater sequence number, or if the two routes have equal sequence numbers but R has a lower metric. Each node in the network advertises a monotonically increasing even sequence number for itself. When a node B decides that its route to a destination D has broken, it advertises the route to D with an infinite metric and a sequence number one greater than its sequence number for the route that has broken (making an odd sequence number). This causes any node A routing packets through B to incorporate the infinite-metric route into its routing table until node A hears a route to D with a higher sequence number.

2.1.2 Implementation Decisions: We did not use link layer breakage detection from the 802.11 MAC protocol in obtaining the DSDV data presented in this paper, because after implementing the protocol both with and without it, we found the performance significantly worse with the link layer breakage detection. The reason is that if a neighbor N of a node A detects that its link to A is broken, it will broadcast a triggered route update containing an infinite metric for A. The sequence number in this triggered update will be one greater than the last sequence number broadcast by A, and therefore is the highest sequence number existing anywhere in the network for A. Each node that hears this update will record an infinite metric for destination A and will propagate the information further. This renders node A unreachable from all nodes in the network until A broadcasts a newer sequence number in a periodic update. A will send this update as soon as it learns of the infinite metric being propagated for it, but large numbers of packets can be dropped in the meantime.

Table 2: Constants used in the DSDV-SQ simulation

Periodic route update interval	15 s
Periodic updates missed before link declared broken	3
Initial triggered update weighted settling time	6 s
Weighted settling time weighting factor	7/8
Route advertisement aggregation time	1s
Maximum packets buffered per node per destination	5

Our implementation uses both full and incremental updates as required by the protocol’s description. However, the published description of DSDV [6] is ambiguous about specifying when triggered updates should be sent. One interpretation is that the receipt of a new sequence number for a destination should cause a triggered update. We call this approach DSDV-SQ (sequence number). The advantage of this approach is that broken links will be detected and routed around as new sequence numbers propagate around the broken link and create alternate routes. The second interpretation, which we call simply DSDV, is that only the receipt of a new metric should cause a triggered update, and that the receipt of a new sequence number is not sufficiently important to incur the overhead of propagating a triggered update.

We implemented both DSDV-SQ and DSDV and found that while DSDV-SQ is much more expensive in terms of overhead, it provides a much better packet delivery ratio in most cases. The second scheme (DSDV) is much more conservative in terms of routing overhead, but because link breakages are not detected as quickly, more data packets are dropped.

3. SIMULATION SETUP

OPNET simulator is used to construct models for two different purposes: to study system behavior and performance. A network model may contain any number of communicating entities called nodes as shown in Figure 2. Node models consist of modules and connections. OPNET supports predefined statistics that are typically of interest in simulation studies [2,5].

3.1 Network Model Overview

In the present work the network model as proposed in Figure2, consists of thirty nodes which includes an Application and a Profile Definition. The proposed network model and DSR protocol is taken for validation and comparison of throughput and delay.

The channel speed of the WLAN is set to 5.5 Mbps and simulation time is taken 600 sim-sec. The Application and Profile Definition are used to define the type of traffic sent between the nodes. In this work, these are configured to send TCP traffic. The throughput between two nodes is measured by generating TCP packets from the one node and sending them to the another node. The throughput is calculated based on the time (sim-sec) and delay is calculated based on the distance. The simulation study consists of four scenarios as shown in Table 1.

3.2 Network Environment

The network environment parameters like area, Physical Characteristics, Packet Reception Power Threshold etc. are given in Table 3 The TCP parameter like Maxm ACK Delay(sec), Slow start initial count (MSS), Duplicate ACK Threshold etc. are given in Table 4.

4. SIMULATION RESULTS AND ANALYSIS

The simulations are carried out for throughput and delay for all the scenario as reported above in the Table 3-4 above. The variation in throughput in all the scenario are shown in Fig 3. All simulations run for 600 sim-seconds. The throughput obtained from the above analysis (four scenario) are compared as shown in Figure 3.

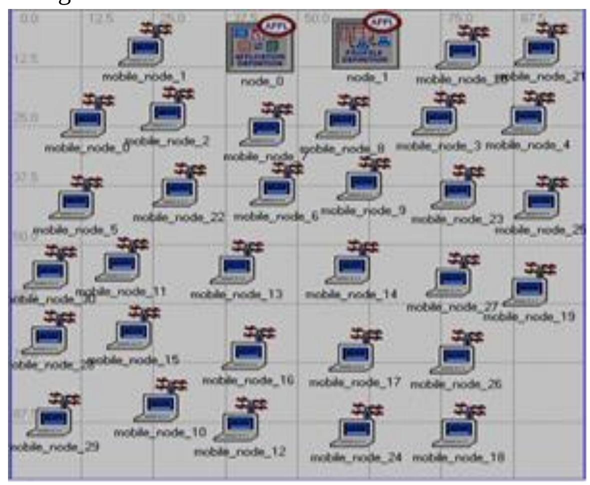

Figure 2. Simulation setup for the proposed network

Table 3: Simulation Environment

Area	300m * 300m
Physical Characteristics	DSSS
Packet Reception Power Threshold	7.33 E-14
Buffer Size	25600
Fragmentation Threshold	1024
Data Rate	2 Mbps
Node Speed	10m/s

Table 4: TCP Parameter

Max^m ACK Delay(sec)	0.200
Slow start initial count (MSS)	4
Duplicate ACK Threshold	3
Fast Recovery	Reno
RTT Gain	0.125
Deviation gain	0.25
RTT Deviation Coefficient	4.0

The average media access delay at node_20 for two routes and at node_18 for four different scenario are compared as shown in Figs. 4 and 5. The scale up network model consists of thirty nodes distributed randomly in a space of 300m x 300m. The channel speed of the wireless LAN is also set to 5.5Mbps. In this network some of the nodes are fixed and some are moving with the speed of 10m/s. Figure 2 is a snapshot of the proposed network model consider for simulation. The first set of scenarios deals with adding more relay nodes between source and destination. The simulation results are obtained for the four-scenario mentioned in Table 1. Figure 3 shows a comparison of the average of the throughput the simulation time for comparison of four different scenarios as reported in Table 1. The top curve corresponds to the first scenario, where three hops exist between node_18 and node_1. The second curve relates to the second scenario. In this scenario five hops exist between node_18 and node_1. The third curve shows the results for the third scenario where six hops exist between node_18 and node_1. The bottom curve corresponds to the fourth scenario, where seven hops exist between node_18 and node_1. The graph in Figure 4 shows the average throughput at node_20 for the two routes. It is clear from the graph in Figure 3 that even though the number of hops in the second route (five hops) is less then the number of hops in the first route (six hops), the average throughput is smaller. The second set of scenarios deals with increasing the number of nodes trying to communicate simultaneously with one node. The graph in Figure 3 shows the average throughput at node_18 for four different scenarios. The top curve corresponds to the scenario where only node_18 is communicating with node_1 (the average value is about 1.85Mbps).

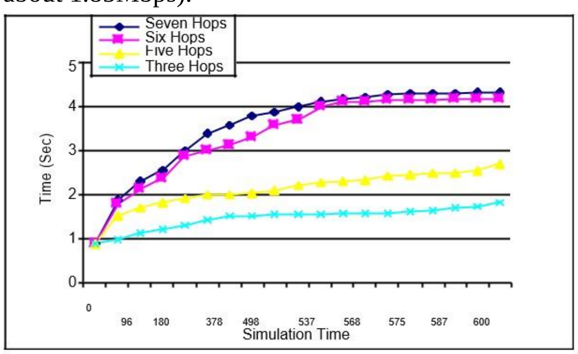

Figure 3. Average throughput comparison between the four scenarios

The second curve corresponds to the scenario where four nodes are trying to communicate simultaneously with node_18 (the average value is around 1.72Mbps). The third curve represents the average throughput for the scenario where five nodes are sending traffic simultaneously to node_18 (the value here is around 0.88Mbps). The bottom curve corresponds to the scenario where seven nodes are communicating simultaneously with node_18 (the average value here is around 0.46Mbps). It is clear from the graph in Figure3 that the more nodes are trying to communicate simultaneously with the same node the less the throughput will be. Also, it is noticeable that the drop is linear (i.e. it has dropped from 1.85Mbps to 1.32 Mbps which is around 12.9%, and then dropped to 0.88Mbps which is around 37%).

The graph in Figure 4 shows the average delay at node_30 for the two routes The top curve indicates that average delay between source to destination corresponds to the five hops route, the bottom curve corresponds for the four hops route. To check the effect of the signal strength on the throughput, node_20 is going to move away from node_29 and try to connect to node_1 through node_27.

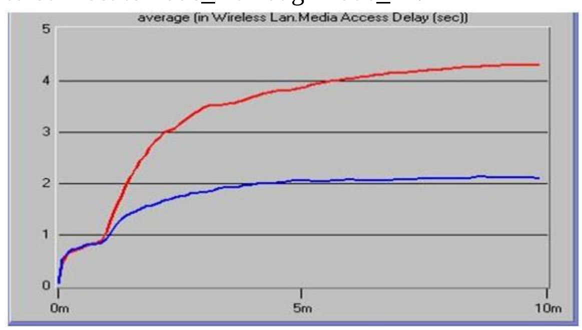

Figure 4. Average Media Access Delay at node_20 for the two routes

It is clear from the graph in Figure 4 that the number of hops in the second route (four hops) is less then the number of hops in the first route (five hops), the average delay is smaller. This is due to the fact that the signal quality between node_20 and node_29 is stronger than the one between node_20 and node_27. The second set (Figure5) of scenarios deals with increasing the number of nodes trying to communicate simultaneously with one node.

The graph in Figure 5 shows the average delay at node_18 for four different scenarios. The top curve corresponds to the scenario where only node_18 is communicating with node_1 (the value here is around 4.2s). The second curve corresponds to the scenario where four nodes are trying to communicate simultaneously with node_1 (the value here is around 4.1s). The third curve represents the average throughput for the scenario where five hopes are sending traffic simultaneously to node_1 (the value here is around 2s). The bottom curve corresponds to the scenario where seven hopes are communicating simultaneously with node_1 (the average value here is around 1.8s). It is clear from the graph in Figure 5 that the more nodes are trying to communicate simultaneously with the same node which results less the delay. Figure 6 shows the average and current simulation speed with total simulation time during running process.

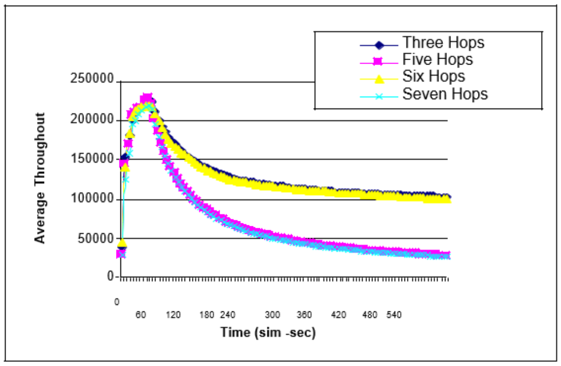

Figure 5. Average delay at node_18 for four different scenarios

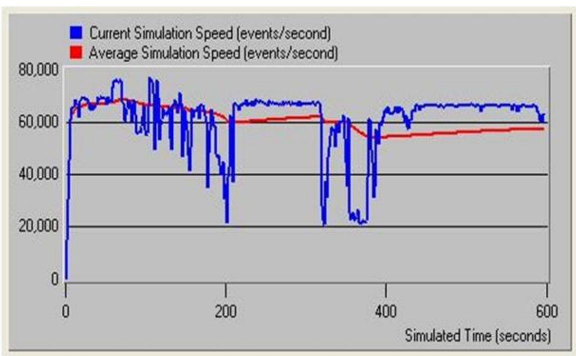

Figure 6. Average & Total Simulation Time during running Process

5. CONCLUSION

In this paper, the performance of wireless Ad-hoc networks has been studied through OPNET simulator and the results are compared. The results reported in this paper suggests a linear drop in the throughput at node_18 while moving towards node_27, with regard to the number of nodes that are trying to simultaneously connect to the same destination. This study has also shown that when the signal strength between source and destination is not strong enough, routing the traffic through an intermediary node can lead to higher values of throughput. In such a situation, the increased latency introduced by an intermediary node, needs to be noted.

It has been observed that after the simulation throughput decreases as the no. of hopes increases as shown in Figure 3. Communication between source to destination through 5 hops will decrease the throughput by 25.23%. Communication between source to destination through 6 hops will decrease the throughput by 40.45%. Communication between source to destination through 7 hops will decrease the throughput by 76.32%. Whereas delay increases when the no. of hops increases as shown in Figure 5. Communication between source to destination through 5 hops will increases the delay by 22.14%. Communication between source to destination through 6 hops will increases the delay by 31.44%.Communication between source to destination through 7 hops will increases the delay by 36.24%.

REFERENCES

[1] E. Celebi, "Performance evaluation of wireless multi-hop adhoc network routing protocols," http://cis.poly.edu/~ecelebi/esim.pdf.

[2] Kumar Manoj, S.C. Sharma, Sandip Vijay and Amit Dixit "Performances Analysis of Wireless Ad-hoc Network Using OPNET Simulator " International Conference on "Intelligent Systems and Networks" ISN-2008 ISTK Haryana, 22-24th 2008, 267-270 Feb. 2008.

[3] OPNET Modeler, http://www.opnet.com

[4] Josh Broch, David B. Johnson, and David A. Maltz. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet-Draft, draft-ietf-manet-dsr-00.txt, March 1998. Work in progress.

[5] A.K. Jain, S.C. Sharma, Kumar Manoj, R. Jha "Effect of Congestion on the Global Performance of IEEE 802.11 Network" International Conference on "Intelligent Systems and Networks" ISN-2008 ISTK Haryana, 22-24th 2008, 231-235 Feb. 2008.

[6] Charles E. Perkins and Pravin Bhagwat. Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers. In Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications, pages 234–244, August 1994. A revised version of the paper is available from http://www.cs.umd.edu/projects/mcml/papers/Sigcomm94.ps.

[7] David B. Johnson and David A. Maltz. Dynamic source routing inad hoc wireless networks. In

Mobile Computing, edited by Tomasz Imielinski and Hank Korth, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.

[8] Charles Perkins. Ad Hoc On Demand Distance Vector (AODV) routing. Internet-Draft, draft-ietf-manet-aodv-00.txt, November 1997. Work in progress.

[9] Vincent D. Park and M. Scott Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In Proceedings of INFOCOM'97, pages 1405–1413, April 1997.