PERFORMANCE METRIC COMPARISON OF AODV AND DSDV ROUTING PROTOCOLS IN MANETs USING NS-2

Sachin Kumar Gupta* & R. K. Saket

Department of Electrical Engineering
Institute of Technology, Banaras Hindu University
Varanasi – 221 005 (Uttar Pradesh), India
E-mail: sachin.gupta.eee09@itbhu.ac.in, rksaket.eee@itbhu.ac.in

ABSTRACT

Efficient routing protocols can provide significant benefits to mobile ad hoc networks in terms of both performance and reliability. Mobile Ad-hoc Network (MANET) is an infrastructure less and decentralized network which need a robust dynamic routing protocol. Many routing protocols for such networks have been proposed so far. Amongst the most popular ones are Dynamic Source Routing (DSR), Ad-hoc On-demand Distance Vector (AODV), Temporally Ordered Routing Algorithm (TORA) and Destination-Sequenced Distance Vector (DSDV) routing protocol. The performance of AODV and DSDV routing protocol have been evaluated for Mobile Ad-hoc Networks (MANETs) in terms of throughput, the average end to end delay, jitter and drop etc. The performance of the AODV is better than the performance of the DSDV routing protocol. A network simulator-2 (NS-2) called MobiREAL simulator has been designed and developed for performance evaluation of AODV and DSDV routing protocol in this paper. To compare the performance of AODV and DSDV routing protocol, the simulation results were analyzed by graphical manner and trace file based on Quality of Service (QoS) metrics: such as throughput, drop, delay and jitter. Finally, the performance differentials based on network load, mobility, and network size have been analyzed. The simulation result analysis verifies the DSDV and AODV routing protocol performances.

Keywords: *DSDV, AODV, DSR, TORA, MANET, QoS, Network Simulator-2 (NS-2)*

1. INTRODUCTION

A Mobile Ad-hoc Network (MANET) is a collection of wireless nodes that can dynamically be set up anywhere and anytime without using any pre-existing network infrastructure. It is an autonomous system in which mobile hosts connected by wireless links are free to move randomly and often act as routers at the same time. The topology of such networks is likely highly dynamic because each network node can freely move and no pre-installed base stations exist. Due to the limited wireless transmission range of each node, data packets then may be forwarded along multihops. Route construction should be done with a minimum of overhead and bandwidth consumption. Since their emergence in the 1970s, wireless networks have become increasingly popular in the computing industry. This is particularly true within the past decade, which has seen wireless networks being adapted to enable mobility.

AODV is perhaps the most well-known routing protocol for MANET [1], which is a hop-by-hop reactive (On demand) source routing protocol, combines DSR and DSDV mechanisms for routing, by using the on-demand mechanism of routing discovery and route maintenance from DSR and the hop-by-hop routing and sequence number from DSDV. For each destination, AODV creates a routing table like DSDV, while DSR uses node cache to maintain routing information [2]. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast routes to destinations within the Ad-hoc network [1].

Destination-Sequenced Distance Vector (DSDV) routing protocol is a typical routing protocol for MANETs, which is based on the Distributed Bellman-Ford algorithm [3]. In DSDV, each route is tagged with a sequence number which is originated by destination, indicating how old the route is [2]. All nodes try to find all paths to possible destinations nodes in a network and the number of hops to each destination and save them in their routing tables. New route broadcasts contain the address of destination, the number of hops to reach the destination, the sequence number of the information receive regarding the destination, as well as a new unique sequence number for the new route broadcast [2].

Wireless networking is an emerging technology that allows users to access information and services electronically, regardless of their geographic position. Wireless networks can be classified in two types:

1.1 Centralized approach Or Infrastructure Networks

Infrastructure network consists of a network with fixed and wired gateways. A mobile host communicates with a bridge in the network (called base station) within its communication radius. The mobile unit can move geographically while it is communicating. When it goes out of range of one base station, it connects with new base station and starts communicating through it. This is called handoff. In this approach the base stations are fixed.

1.2 Decentralized approach or Infrastructure less (ad-hoc) Networks

In contrast to infrastructure based wireless network, in ad-hoc networks all nodes are mobile and can be connected dynamically in an arbitrary manner. A MANET is a collection of wireless mobile nodes forming a temporary network without using any existing infrastructure or any administrative support. The wireless ad-hoc networks are self-creating, self-organizing and self-administrating. The nodes in an ad-hoc network can be a laptop, cell phone, PDA or any other device capable of communicating with those nodes located within its transmission range. The nodes can function as routers, which discover and maintain routes to other nodes. The ad-hoc network may be used in emergency search-and-rescue operations, battlefield operations and data acquisition in inhospitable terrain. In ad-hoc networks, dynamic routing protocol must be needed to keep the record of high degree of node mobility, which often changes the network topology dynamically and unpredictably.

2. ROUTING PROTOCOLS

The existing routing protocols in MANETs can be classified into three categories. Figure 1 shows the classification along with some examples of existing MANET protocols.

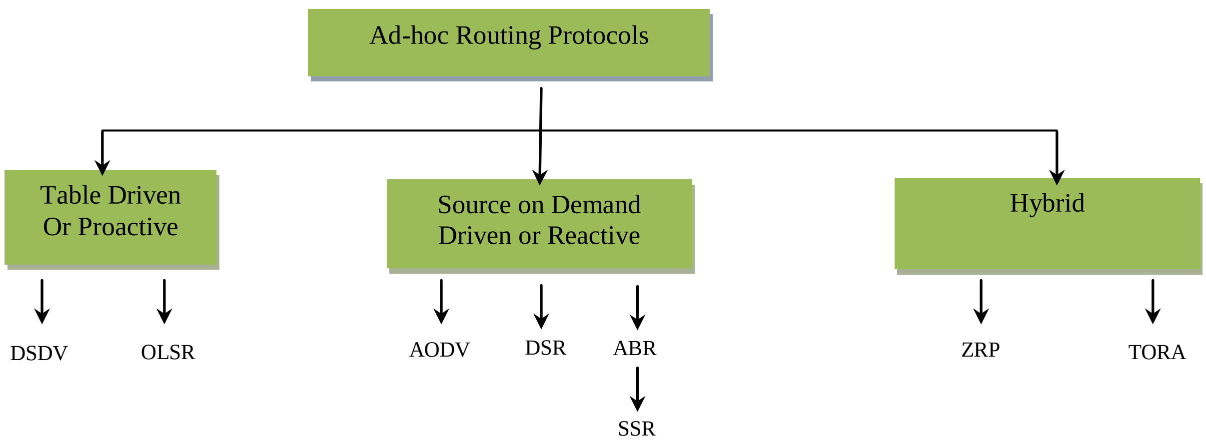

Figure 1: Classification of MANETs Routing Protocols.

2.1 Study of DSDV and AODV Routing Protocols

2.1.1 Destination-sequenced distance vector

DSDV is one of the most well known table-driven routing algorithms for MANETs. The DSDV routing algorithm is based on the classical Bellman-Ford Routing Algorithm (BFRA) with certain improvement [3]. Every mobile station maintains a routing table with all available destinations along with information like next hop, the number of hops to reach to the destination, sequence number of the destination originated by the destination node, etc. DSDV uses both periodic and triggered routing updates to maintain table consistency. Triggered routing updates are used when network topology changes are detected, so that routing information is propagated as quickly as possible. Routing table updates can be of two types – 'full dump' and 'incremental'. 'Full dump' packets carry all available routing information and may require multiple Network Protocol Data Units (NPDU); 'incremental' packets carry only information changed since the last full dump and should fit in one NPDU in order to decrease the amount of traffic generated. Mobile nodes cause broken links when they move from place to place. When a link to the next hop is broken, any route through that next hop is immediately assigned infinity metric and an updated sequence number. This is the only situation when any mobile node other than the destination node assigns the sequence number. Sequence numbers assigned by the origination nodes are even numbers, and sequence numbers assigned to indicate infinity metrics are odd numbers. When a node receives infinity metric, and it has an equal or later sequence number with a finite metric, it triggers a route update broadcast, and the route with infinity metric will be quickly replaced by the new route. When a mobile node receives a new route update packet; it compares it to the information already available in the table and the table is updated based on the following criteria:

- If the received sequence number is greater, then the information in the table is replaced with the information in the update packet

- Otherwise, the table is updated if the sequence numbers are the same and the metric in the update packet is better.

Advantages:

- DSDV was one of the early algorithms available. It is quite suitable for creating ad hoc networks with small number of nodes.

Disadvantages:

- DSDV requires a regular update of its routing tables, which uses up battery power and a small amount of bandwidth even when the network is idle.
- Whenever the topology of the network changes, a new sequence number is necessary before the network re-converges; thus, DSDV is not suitable for highly dynamic networks.

2.1.2 Ad-hoc On-demand distance vector

Reactive protocols discover routes only as needed. When a node wishes to communicate with another node, it checks with its existing information for a valid route to the destination. If one exists, the node uses that route for communication with the destination node. If not, the source node initiates a route request procedure, to which either the destination node or one of the intermediate nodes sends a reply back to the source node with a valid route [5]. A soft state is maintained for each of these routes, if the routes are not used for some period of time, the routes are considered to be no longer needed and are removed from the routing table. Example of this type algorithm is DSR and AODV.

AODV is a reactive protocol, even though it still uses characteristics of a proactive protocol [4]. AODV takes the interesting parts of DSR and DSDV in the sense that it uses the concept of route discovery and route maintenance of DSR and the concept of sequence numbers and sending of periodic hello messages from DSDV.

The protocol uses different messages to discover and maintain links:

- Route Requests (RREQs)

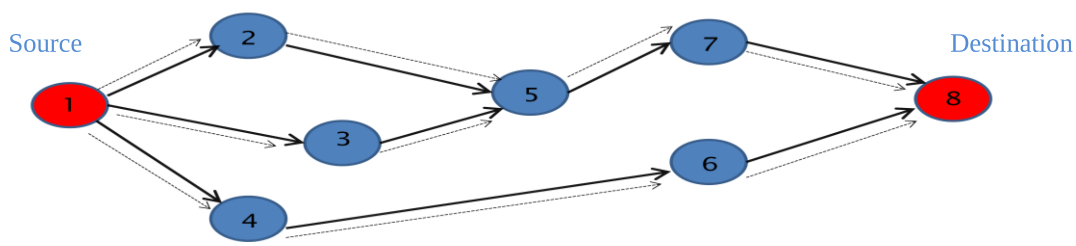

Figure 2: Propagation of Route Request (PREQ) Packet

- Route Replies(RREPs)

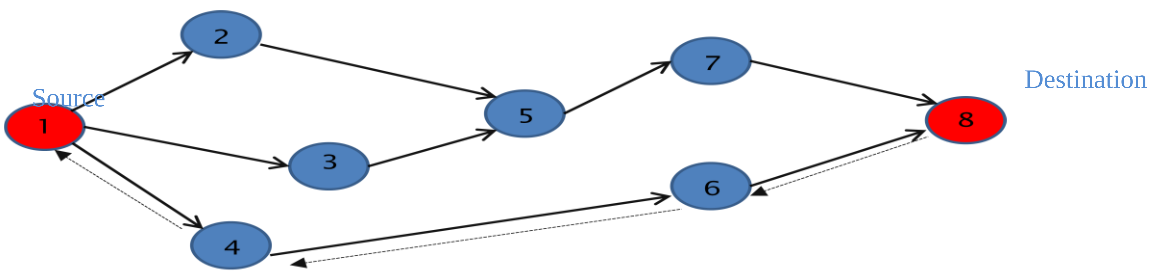

Figure 3: Propagation of Route Reply (PREP) Packet

- Route Errors(RERRs)

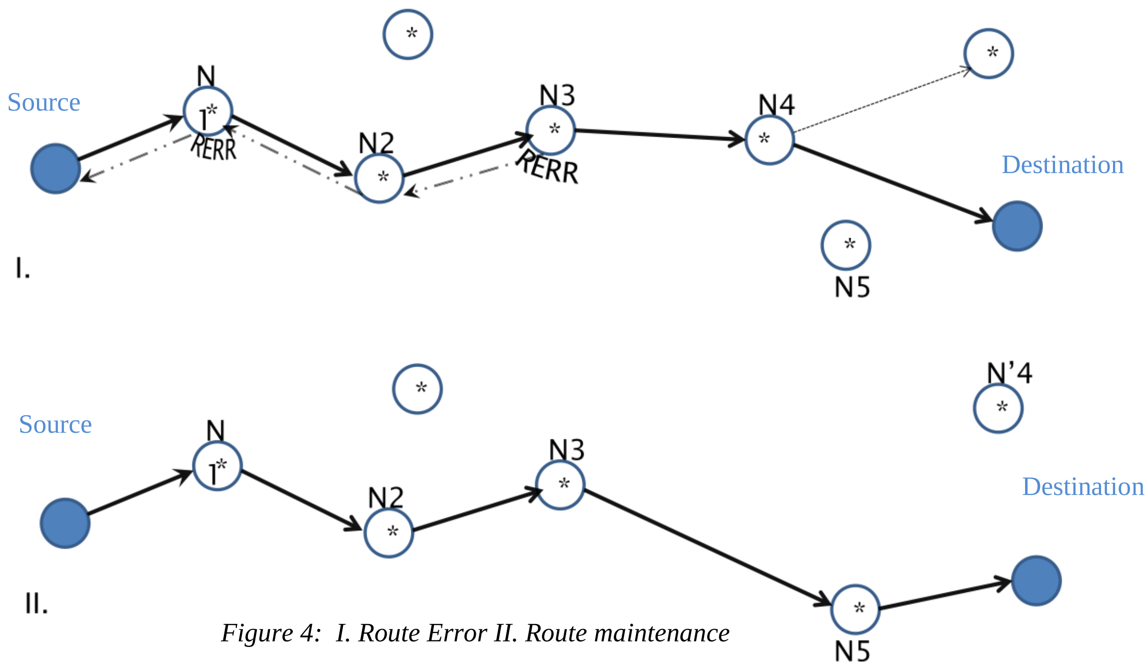

Figure 4: I. Route Error II. Route maintenance

These message types are received via 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 knowing which of the available routes is fresher and requesting node always selects the one with greatest sequence number. When a node wants to find a route, it broadcasts a RREQ to all network till either destination is reached or another node is found with a 'fresh enough' route to the destination. Then a RREP is sent back to the source and the discovered route is made available.

Note: Fresh Enough route is a valid route entry for the destination whose associated sequence number is at least as great as that contained in RREQ. Nodes that are part of an active route may offer connectivity information by broadcasting periodically local hello messages (special RREP messages) 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.

Advantages:

- Routes are established on demand and destination sequence numbers are used to find the latest route to the destination.
- Lower delay for connection setup.

Disadvantage:

- AODV doesn't allow handling unidirectional links.
- Multiple Route Reply packets in response to a single Route Request packet can lead to heavy control overhead.
- Periodic beaconing leads to unnecessary bandwidth consumption

3. SIMULATION STRATEGY

For the simulation of the developed system, latest version 2.34 of NS-2 has been used in this paper. Ns-2 is a discrete event simulator targeted at networking research [6]. It began as a part of the REAL network simulator and is evolving through an ongoing collaboration between the University of California at Berkeley and the VINT project [7].

3.1 Scenario

- Topology of 900*900 is taken for simulation.
- Nodes are being generated randomly at random position.
- Nodes are generated at random time as if few nodes are entering into the topology.
- Nodes are moving at constant random speed.
- Radio propagation model used is Two-Ray Ground.
- Antenna model used is Omni Antenna
- Movement is linear and node speed is constant for a simulation

3.2 Node Characteristics:

- Link Layer Type: Logical Link (LL) type
- MAC type: 802_11
- Queue type: Drop-Tail
- Network Interface type: wireless
- Channel type: wireless

The simulation parameters are listed in Table 1.

3.3 Performance Metrics:

The following different performance metrics are evaluated to understand the behavior of DSDV and AODV routing protocols

- Throughput
- The average end to end delay.
- Jitter

Table 1: Simulation parameters

Parameter	Value
Simulator	NS-2 (Version 2.34)
Channel type	Channel/Wireless channel
Radio-propagation model	Propagation/Two ray round wave
Network interface type	Phy/WirelessPhy
MAC Type	Mac /802.11
Interface queue Type	Queue/Drop Tail
Link Layer Type	LL
Antenna	Antenna/Omni Antenna
Maximum packet in ifq	60
Area (M*M)	900 * 900
Number of mobile node	16
Source Type	UDP, TCP
Simulation Time	350 sec
Routing Protocols	DSDV, AODV

4. SIMULATION MODEL AND RESULTS

4.1 Simulation Model

The objective of this paper is the performance evaluation of two routing protocol for mobile ad hoc networks by using an open-source network simulation tool called NS-2. Two routing protocols: DSDV and AODV have been considered for performance evaluation in this work. The simulation environment has been conducted with the LINUX operating system, because NS-2 works with Linux platform only.

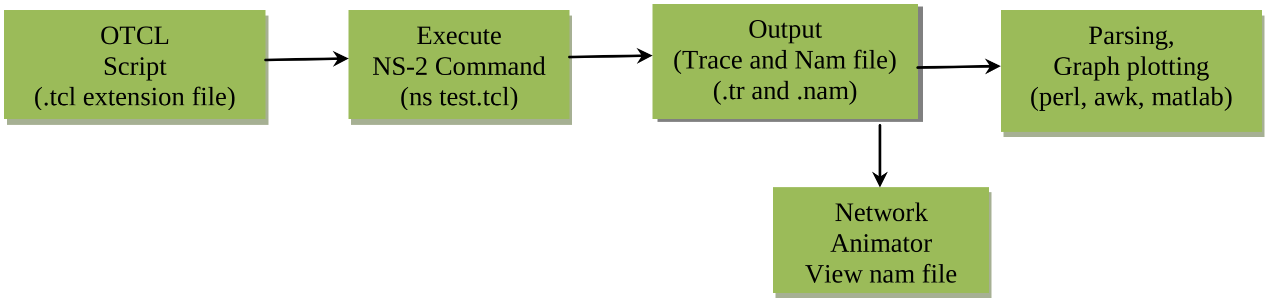

Figure 5: Simulation Overview

Whole simulation study is divided into two part one is create the node (that may be cell phone, internet or any other devices) i.e. NS-2 output. It's called NAM (Network Animator) file, which shows the nodes movement and communication occurs between various nodes in various conditions or to allow the users to visually appreciate the movement as well as the interactions of the mobile nodes. And another one is graphical analysis of trace file (.tr). Trace files contain the traces of event that can be further processed to understand the performance of the network.
Figure 5 depicts the overall process of how a network simulation is conducted under NS-2. Output files such as trace files have to be parsed to extract useful information. The parsing can be done using the *awk* command (in UNIX and LINUX, it is necessary to use gwak for the windows environment) or *perl* script. The results have been analyzed using Excel or Matlab. A software program which can shorten the process of parsing trace files (Xgraph and TraceGraph) has also been used in this paper. However, it doesn't work well when the trace file is too large. To generate trace file and nam file, we call tcl script in CYGWIN command shell. By varying the simulation parameter shown in table 1, we can see the graphical variation between various performance metrics like throughput, drop, delay, jitter etc.

4.2 Results

Generated trace file that is (.tr)

```
s 10.006348737 _1_ MAC  --- 3 ack 118 [13a 0 1 800] ------- [1:0 0:0 32 0] [0 0] 0 0
r 10.007293041 _0_ MAC  --- 3 ack 60 [13a 0 1 800] ------- [1:0 0:0 32 0] [0 0] 1 0
s 10.007303041 _0_ MAC  --- 0 ACK 38 [0 1 0 0]
r 10.007318041 _0_ AGT  --- 3 ack 60 [13a 0 1 800] ------- [1:0 0:0 32 0] [0 0] 1 0
s 10.007318041 _0_ RTR  --- 4 tcp 1560 [0 0 0 0] ------- [0:0 1:0 32 1] [1 0] 0 0
s 10.007318041 _0_ AGT  --- 5 tcp 1540 [0 0 0 0] ------- [0:0 1:0 32 0] [2 0] 0 0
r 10.007318041 _0_ RTR  --- 5 tcp 1540 [0 0 0 0] ------- [0:0 1:0 32 0] [2 0] 0 0
```

1. First field is event type; it may be r, s, f, d for 'received', 'sent', 'forwarded' and 'dropped' respectively.
2. The second field is the time.
3. The third field is the node number.
4. The fourth field is MAC to indicate, if the packet concerns a MAC layer; it is AGT to indicate the transport layer (e.g. tcp) packet, or RTR if it concerns the route packet. It can be IFQ for drop packets.
5. After the dashes comes the global sequence number of the packet (not tcp sequence number).
6. At the next field comes more information on the packet type (e.g. tcp, ack, or udp).
7. Next is the packet size in byte.
8. The 4 numbers in the first square brackets concern MAC layer information. The first hexadecimal number specifies the expected time in seconds to send this data packets over the wireless channel. The second number stand for the MAC-id of the sending node third is for receiving node. And fourth number, 800, specifies that the MAC type is ETHERTYPE_IP.
9. The next number in the second square brackets concern the IP source and destination addresses, then the ttl (time to live) of the packet (in our case 32).
10. The third brackets concern the tcp information: its sequence number and acknowledgement number.

4.2.1 Nam file output

NAM is a Tcl/TK based animation tool for viewing network simulation traces and real world packet traces. A network animator that provides packet-level animation and protocol-specific graphs to aid the design and debugging of new network protocols have been described. Taking data from network simulators (such as ns) or live networks, NAM was one of the first tools to provide general purpose, packet-level, and network animation, before starting to use NAM, a trace file needs to create [7]. This trace file is usually generated by NS. Once the trace file is generated, NAM can be used to animate it. A snapshot of the simulation topology in NAM for 16 mobile nodes is shown in figure 6, which is visualized the traces of communication or packets movements between mobile nodes [9]. And figure 7 shows the running TCL script in cygwin command shell.

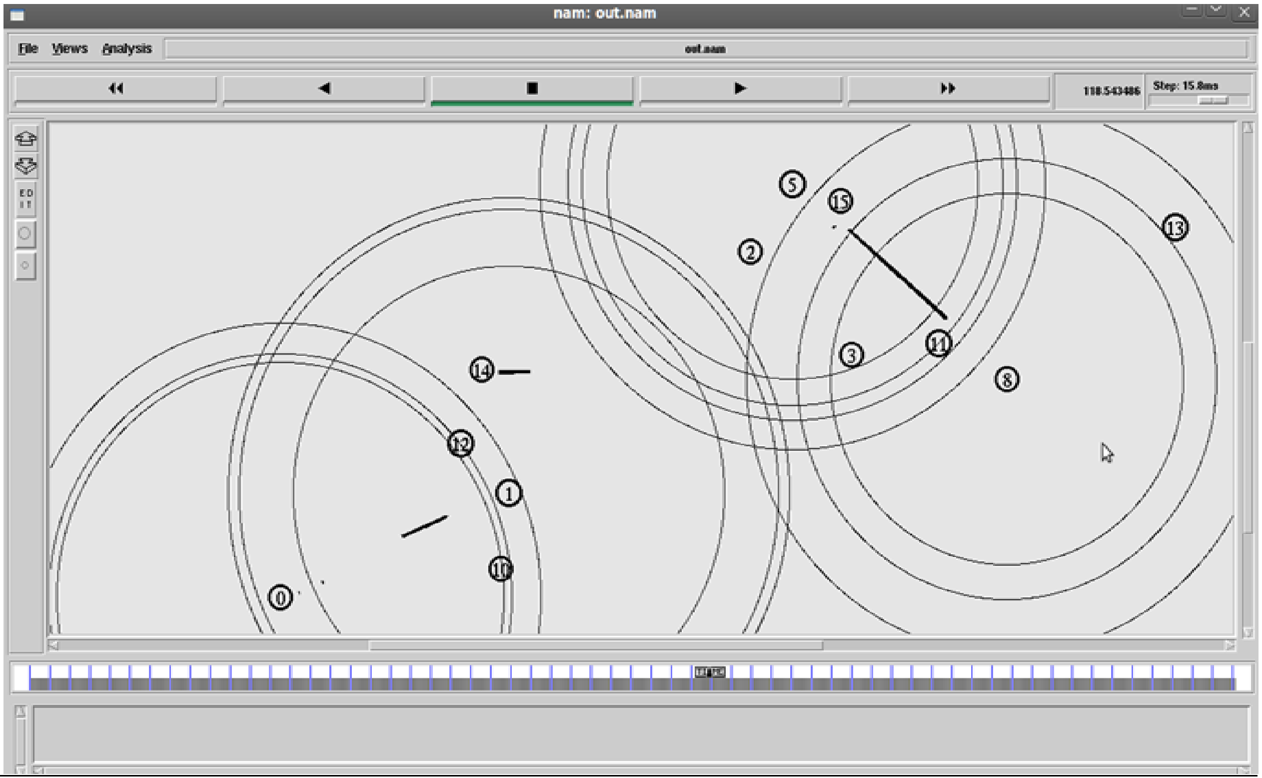

Figure 6: A snapshot of the simulation topology in NAM for 16 mobile nodes

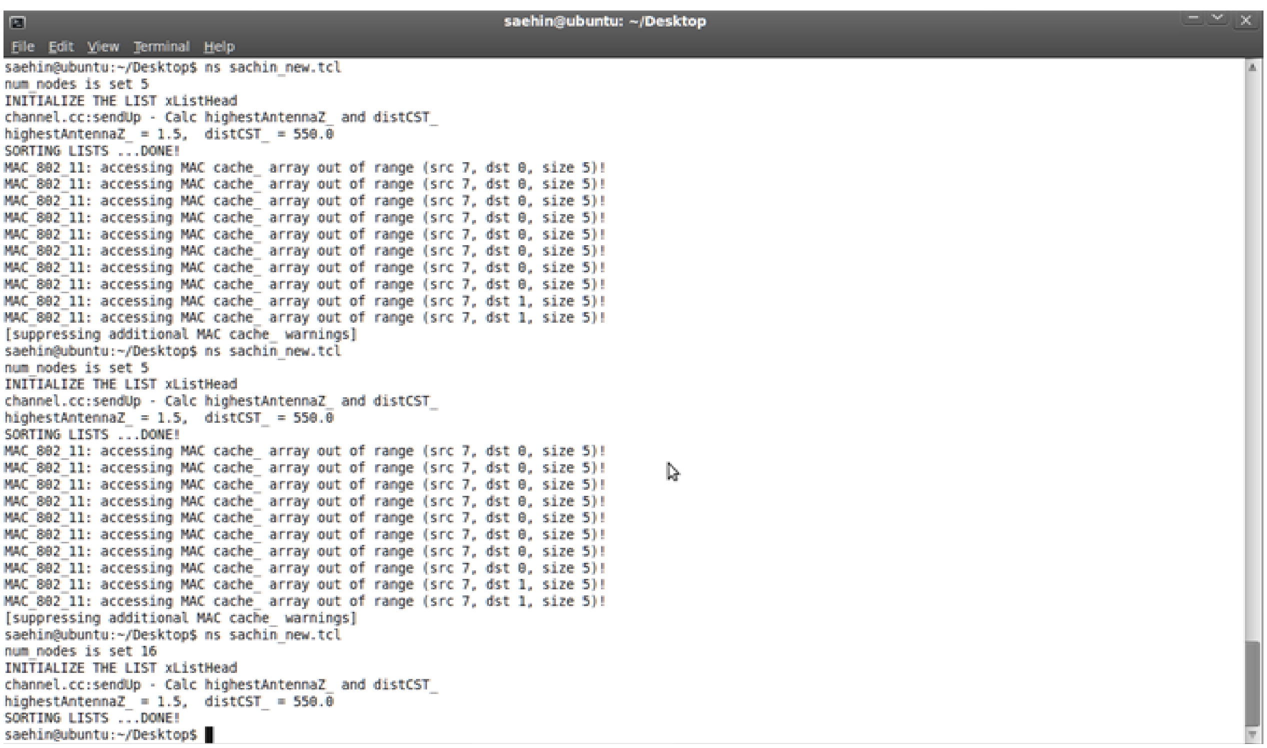

Figure 7: Running TCL script in cygwin command shell

4.2.2 Graphical Analysis

4.2.2.1 Throughput

Throughput is the number of packet that is passing through the channel in a particular unit of time. This performance metric show the total number of packets that have been successfully delivered from source node to destination node and it can be improved with increasing node density.

Figure 8 shows the sending throughput for UDP from source node. It is observe that sending throughput maximum in the time interval of 200 to 250 for both routing protocol and it is increased because of node density, less traffics and free channel. In rest of the time sending throughput is almost constant. Here, AODV performance is better than DSDV in terms of sending throughput.

Figure 9 shows the sending throughput for TCP packets. In the time interval of 200 to 250 maximum amount of TCP packets have been delivered from source to destination node in terms of DSDV because it is a proactive type routing protocols and advantage of these type of protocols is there are no delay to find out the route from source to destination nodes because path is immediately available when source need to send a packet [2].

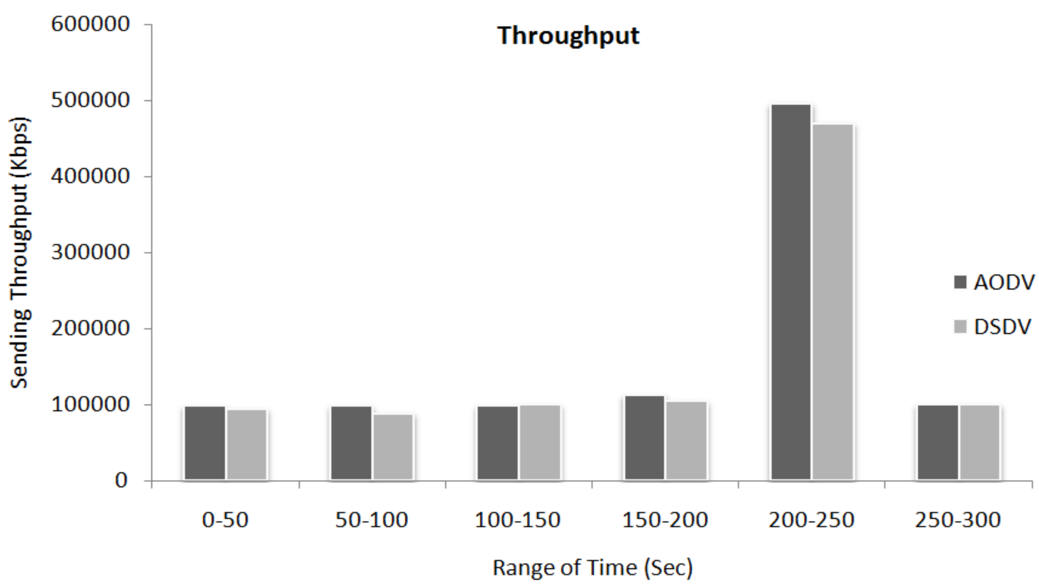

Figure 8: Transmission throughput (UDP)

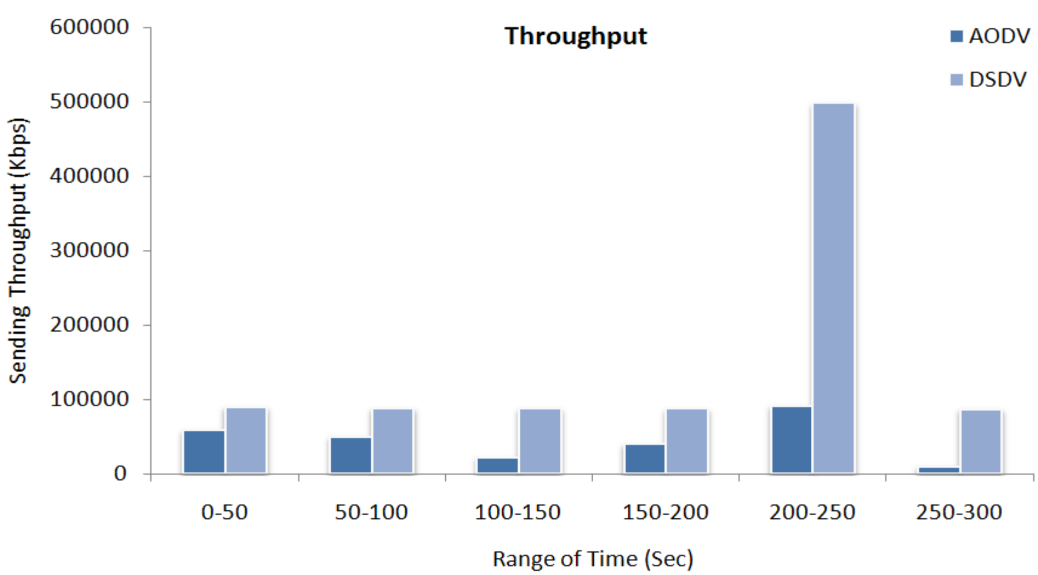

Figure 9: Transmission throughput (TCP)

Figure 10 shows the maximum receiving throughput in the time interval of 100 to 150 in terms of DSDV as well as maximum amount UDP packets actually received by the particular destination because in that particular time interval the send node and receive node distance is less, free of channel for those packets. On the other side, AODV shows better performance and the performance rate sequentially increasing.

Figure 11 shows the receiving throughput for TCP packets that is maximum in the time range of 100 to 150 for DSDV protocol because, it's maintain periodic table which broadcast routing table continuously to its neighbour for update. For the same time interval AODV decrease because of less active route.

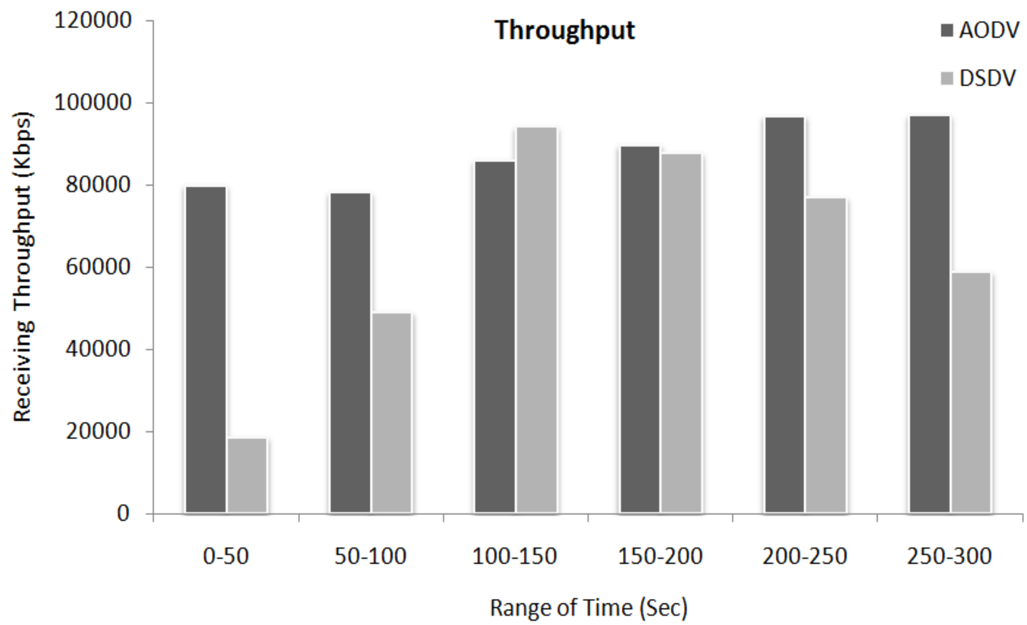

Figure 10: Receiving throughput (UDP)

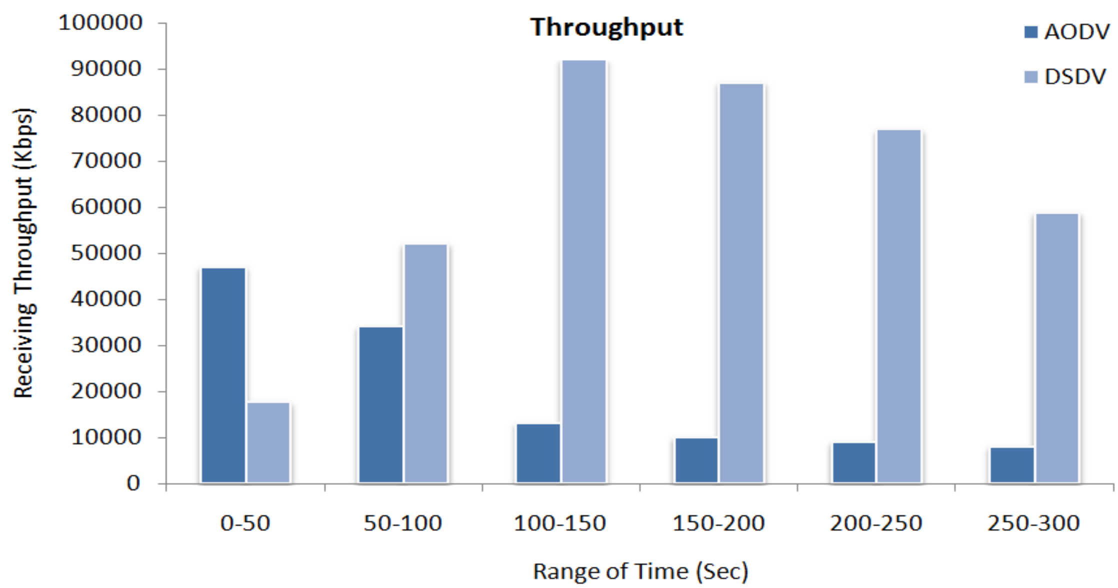

Figure 11: Receiving throughput (TCP)

4.2.2.2 The average end to end Delay

A specific packet is transmitting from source to destination node and calculates the difference between send times and received times. Delays due to route discovery, queuing, propagation and transfer time are included in the delay metric [5].

Figure 12 shows the delay graph for UDP, between sending time versus delay. It is increasing in terms of AODV because of the distance between sending and receiving nodes is high. From the Figure 12, the time range between 0-20 and 50-90 seconds, the delay was high because in that particular time interval the distance between sending node and receiving node is high due to traffic. And in the time interval 110-140 and 170 to 200 sec the delay is less because of less traffic and free channel for the UDP packets. On the other hand, initially DSDV has less delay but later on increased because DSDV performs well under low node mobility.

Figure 13 shows delay graph for TCP. Initially it is observe that delay is increases for AODV because of more traffic and busy channel. Later on approx after 50 second continuously delay is decreases when the source and destination nodes close to each other while having free channel and minimum traffic. Beside this, delay increases for DSDV because of high node mobility.

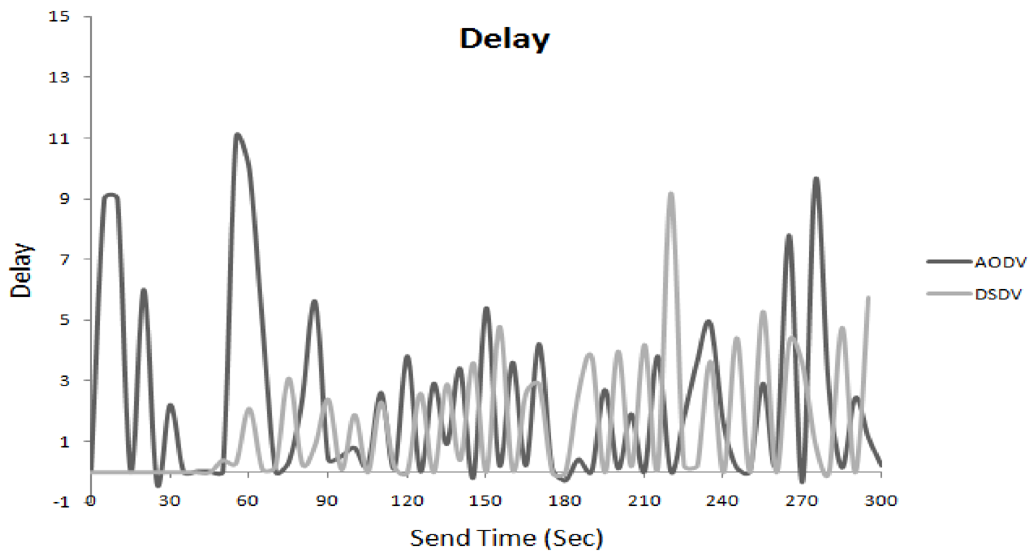

Figure 12: Delay graph between Send Time Vs Delay for UDP

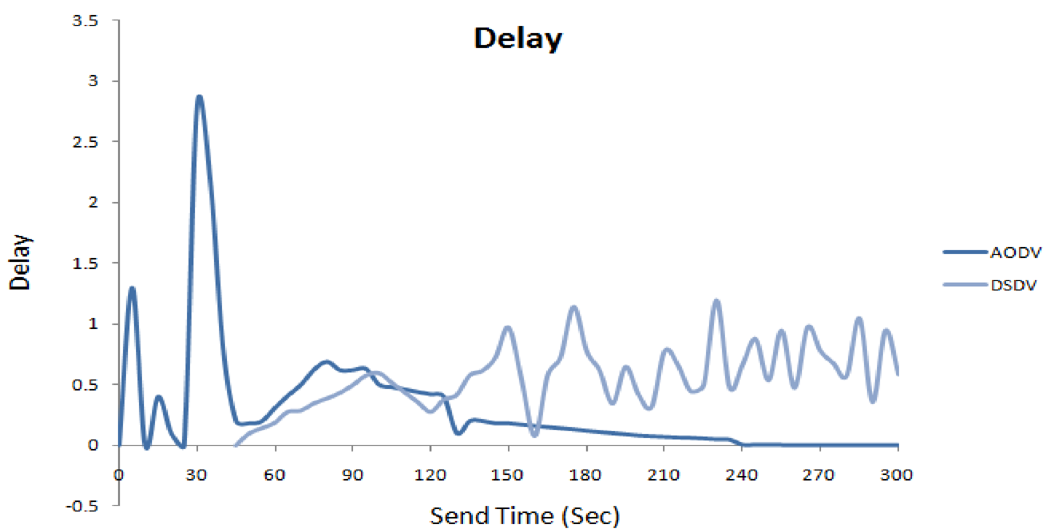

Figure 13: Delay graph between Send Time Vs Delay for TCP

4.2.2.3 Jitter

The term jitter is often used as a measure of the variability over time of the packet latency across a network. A network with constant latency has no variation (or jitter). Packet jitter is expressed as an average of the deviation from the network mean latency. However, for this use, the term is imprecise [8]. Or in other word jitter is the variation of the packet arrival time. In jitter calculation the variation in the packet arrival time is expected to minimum. The delays between the different packets need to be low if we want better performance in Mobile Ad-hoc Networks [1].

Figure 14 shows the jitter graph between sending times versus jitter for UDP packets. AODV initially shows high jitter and after a certain time interval low jitter value appears. And in the beginning DSDV has very low jitter rate approx zero within the time interval of 0-45 second and after 45 second the situation remained unstable.

Figure 15 shows the jitter graph for TCP packets. AODV shows high jitter in between 50 to 95 second but later on the performance is quite good. And here, DSDV gives better jitter performance because of low node mobility and free channel.

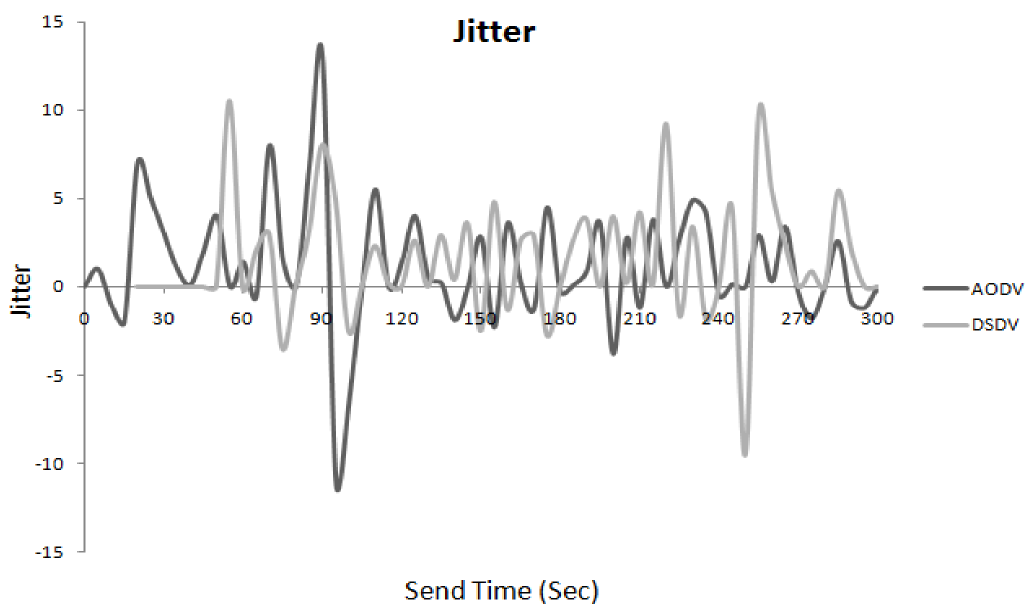

Figure 14: Graph between Send Time Vs Jitter for UDP

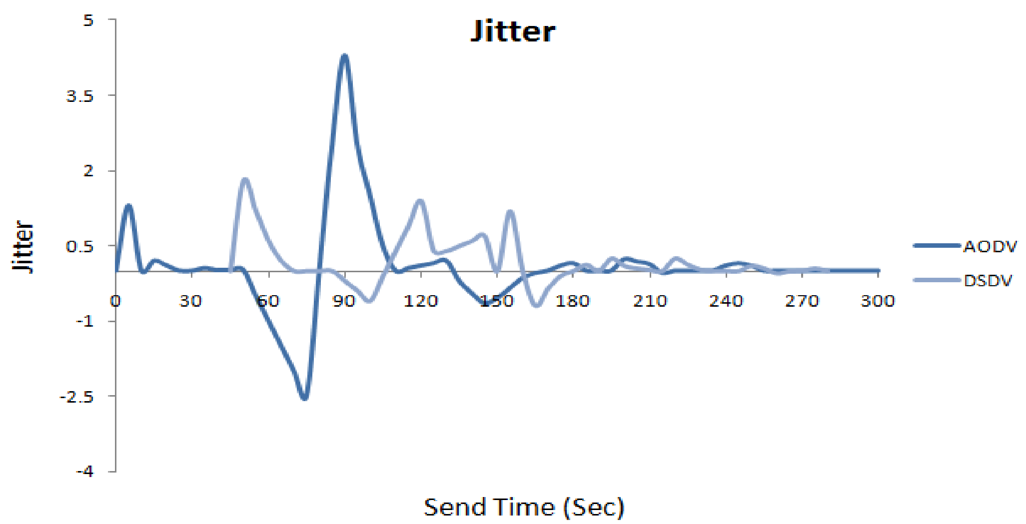

Figure 15: Graph between Send Time Vs Jitter for TCP

5. CONCLUSION

OTcl script (.tcl extension file) has been created and NS-2 command (ns test.tcl) using CYGWIN command shell has been executed successfully in this work. After execution tcl script, .tr (TRACE) files and .nam (NAM) files have been generated and simultaneously packets movement between the nodes in NAM (network animator) has been visualized. In this paper, DSDV and AODV routing protocol using different parameter of QoS metrics have been simulated and analyzed. As a reactive protocol AODV transmits network information only on-demand and DSDV maintains table driven routing mechanism as proactive routing protocol. Two types of data packet TCP and UDP have been analyzed in this paper. DSDV and AODV routing protocol, packet delivery ratio is independent of offered traffic load. AODV protocols delivering 70% to 90% of the packets in all cases, while DSDV delivering 50% to 75%. Delay is high initially in AODV but after some time it is very low. But in the case of DSDV, it is very low at starting and increased gradually specially for UDP packets. DSDV gives better jitter performance due to low node mobility and free channel, but variation of the packets arrival time or jitter is little bit high in case of AODV because of high node mobility and unavailability of free channel. So we can conclude that AODV indicating its highest efficiency and performance under high mobility than DSDV. Simulation results show the performance of TCP and UDP packets with respect to the average end to end delay, throughput, and jitter. Finally, it is concluded that the performance of AODV is better than DSDV routing protocol for real time applications.

ACKNOWLEDGEMENTS

Authors are grateful to Electrical Engineering Department, Institute of Technology, Banaras Hindu University, Varanasi (India) for providing necessary facilities and peaceful environment to complete this research work. Corresponding author presents humble thanks to his research supervisor Dr. R. K. Saket for providing excellent guidance and encouragement to complete this manuscript in its present shape. He would like to thank his Uncle Mr. Arun Kumar Gupta (Advocate) and family members for financial and moral supports throughout his technical education. Finally, he is also grateful to his senior Ph. D. research scholar Mr. Lokesh Varshney and colleagues Mr. Pankaj Kumar Verma, Vivek Kumar Gupta and Hemant Shukla who have provided technical and academic supports during this research work.

REFERENCES

[1] Md. Monzur Morshed, Md. Habibur Rahman, Md. Rezaur Rahman Mazumder, and K. A. M. Lutfullah, “Simulation and Analysis of Ad-hoc on demand Distance Vector Routing Protocol,” ICIS 2009, vol. II, pp. 610- 614, November 2009.

[2] Valid Nazari Talooki, and Jonathan Rodriguez, “Quality of Service for Flat Routing Protocols in Mobile Ad-hoc Network,” ICST, 7-9 September 2009.

[3] V. Ramesh, Dr. P. Subbaiah, N. Koteswar Rao and M. Janardhana Raju, “Performance comparison and analysis of DSDV and AODV for MANET,” International Journal on Computer Science and Engineering, vol. 02, pp. 183-188, 2010.

[4] Charles E. Perkins, Elizabeth M. Belding-Royer, and Samir R. Das, “Ad hoc On-Demand Distance Vector Routing,” IEFT MANET Draft, Charles E. Perkins, Ad Hoc Networking, ISBN 0-201-30976-9 February 2003.

[5] S H Manjula, C N Abhilash, Shaila K, K R Venugopal, and L M Patnaik, “Performance of AODV Routing Protocol using Group and entity Mobility Models in Wireless Sensor Networks,” In proceedings of the International Multi Conference of Engineers and Computer Scientists (IMECS 2008), Hong Kong, vol. 2, pp. 1212-1217, 19-21 March 2008.

[6] Network simulator-ns-2. www.isi.edu/nsnam

[7] NS Manual/Documentation-The VINT Project Collaboration between researchers at UC Berkeley, LBL, USC/ISI and Xerox. PARC.KevinFall_kfall@ee.lbl.gov, Kennan Varadhan_kannan@catarina.usc.edu

[8] Comer and Douglas E., “Computer Networks and Internet,” Prentice Hall, p.476, ISBN 9780136061274, 2008.

[9] Sachin Kumar Gupta and Dr. R. K. Saket, “Routing Protocols in Mobile Ad-hoc Networks,” In proceedings of the International Conference on Electronics, Information and Communication Engineering; Jodhpur (Rajasthan) India. pp. 1-5, 28-30 March 2011.