Performance Evaluation of some Routing Protocols using TCP Traffic Types in Mobile Ad Hoc Networks

Hassan Al-Mahdi[1], Mohamed Wahed[2], Tarek M Mahmoud[3] and Hassan Shaban[3]

[1]Faculty of Sciences &, Arts Al Jouf University, Al Quryyat, Saudi Arabia

[2]Faculty of Computer & Informatics, Suez Canal University, Egypt

[3]Faculty of Computers & Information, Minia University, Egypt

Abstract

Routing is a critical issue in a Mobile Ad-hoc Network (MANET). The most famous MANET routing protocols are AODV, DSDV, DSR and AOMDV. Although the performance evaluation of these routing protocols have been extensively simulated, most of the researchers are focusing on the User Datagram Protocol (UDP) traffic type supposing that the Transmission Control Protocol (TCP) sources offers a conforming load to the network. Despite of this, most Internet applications are carried out through TCP. In this paper, we investigate the performance of AODV, DSDV, DSR and AOMDV routing protocols using TCP traffic types such as TCP-Reno, TCP-Newreno, TCP-Vegas and TCP-Sack. The performance analysis is done in terms of packet delivery ratio, average end to end delay and average throughput using the NS2 simulator. The simulation results show that TCP-Vegas performs better compared with others in the case of end-to-end delay, and has higher packet delivery ratio. The TCP-Reno has higher throughput in the case of low data connections compared with TCP-Newreno, TCP-Vegas and TCP-Sack. In case of high data connections the TCP-Vegas have the higher throughput compared with the others.

Keywords: Routing protocols; TCP-Reno; TCP-Newreno; TCP-Vegas; TCP-Sack1; NS2 simulator.

1. Introduction

A Mobile Ad-hoc Networks (MANETs is autonomous and decentralized collection of mobile nodes forming a dynamic wireless network. In MANET, each node acts both as host and router and forwards packets for nodes that are out of transmission range [1]. In MANETs, routing protocol can be classified into three categories; Proactive, Reactive and Hybrid routing protocol [2].

The pro-active routing protocols (Table-driven) are the same as current Internet routing protocols such as the Routing Information Protocol, Distance-Vector, Open Shortest Path First and link-state. They attempt to maintain consistent, up-to-date routing information of the whole network. Some of the existing pro-active ad hoc routing protocols are: Destination Sequenced Distance-Vector (DSDV) [3], Wireless Routing Protocol, Cluster head Gateway Switch Routing, Global State Routing (GSR) [4], Fisheye State Routing (FSR) [5], Hierarchical State Routing (HSR) [6,7], Zone based Hierarchical Link State and Source Tree Adaptive Routing [8].

The Reactive routing protocols (On-demand) maintain only the routes that are currently in use, thereby trying to maintain low control overhead, reducing the load on the network when only a small subset of all available routes is in use at any time. Some of the existing reactive routing protocols are Ad hoc On-demand Distance Vector (AODV) Routing protocol [9, 10], Dynamic Source Routing Protocol (DSRP) [11], Associativity Based Routing (ABR) protocol [12], and Signal Stability Routing (SSR) protocol [13].

Since the proactive and reactive routing protocols in MANETs have relative advantage and disadvantage, then evaluating and comparing them is very critical issue. Significant works has been carried out to evaluate and compare these routing protocols under various traffic types [14]. However, very few did the simulation with TCP traffic sources for constant bit rate applications. In this paper, the performance evaluation of AODV, DSDV, DSR and AOMDV routing protocols in MANETs using TCP traffic sources will be introduced using the NS2 simulator.

The remainder of the paper is organized as follows. Section 2 shows an overview of the considered routing protocols. Section 3 shows the performance environment and metrics. The traffic patterns will be presented in section 4. Section 5 presents the simulation setup. The simulation results will be investigated in section 6. Finally, we conclude the paper in section 7.

2. Routing protocols

2.1. DSDV

DSDV [3] is a distance vector routing protocol. It is based on the Bellman-Ford routing algorithm. DSDV is a proactive routing protocol. In this protocol every node maintains a routing table that contains next-hop entry and the number of hops needed for all reachable destinations. DSDV assumes bidirectional links and thus does not have unidirectional link support. DSDV uses a concept of sequence numbers to provide loop freedom. The sequence number is originated by the destination node. To maintain routing information consistent within a network, DSDV requires nodes to broadcast periodical route advertisements. In practice updates are sent in every few seconds. The advertisement contains the routing table entries of the advertising node. These entries contain the address of destination, next hop and hop count to that destination and the last known sequence number originated by that destination. When a node receives an advertisement it updates its routing table on this basis. Routes with greater sequence numbers are always preferred. If the sequence numbers are equal, a route with lower hop count is chosen. Note that the receiving node increases the hop counts in the advertisement since the destination needs one hop more to be reached. The receiving node will then subsequently pass this new information forward within its own route advertisement. When a node detects a link failure it marks all routes through that link with hop count equal to infinity (any number beyond allowed maximum) and assigns sequence number greater than the stored sequence number for that destination, then broadcasts update information. That is why nodes detecting failures always assign odd sequence numbers to these routes. Original destination originated sequence numbers are even.

2.2. DSR

DSR is an on-demand protocol. It is composed of route discovery and route maintenance. In route discovery, a node tries to discover a route to destination if it has to send data to this destination and there is currently no known route(s). A node broadcasts a route request (RREQ) with a unique identifier and the destination address as parameters. Any node that receives RREQ does the following [18]:

1) If it has already received the request, it drops the request packet.
2) If it recognizes its own address as the destination, then the request has reached its target.
3) Otherwise, the node appends its own address to a list of traversed hops in the packet and broadcasts this updated route request. In route maintenance, a node is continuously sending packets via a route. The node has to make sure that the route is held upright. If a node detects problems with the current route, it has to find an alternative route.

2.3. AODV

AODV protocol [10] is a mixture of both DSR and DSDV protocols. It keeps the basic route-discovery and route-maintenance of DSR and uses the hop-by-hop routing sequence numbers and beacons of DSDV. When a node needs to know a route to a specific destination it creates a ROUTE EQUEST. Next the route request is forwarded by intermediate nodes which also create a reverse route for itself from the destination. When the request reaches a node with route to destination it creates again a ROUTE REPLY which contains the number of hops that are require to reach the destination. All nodes that participate in forwarding this reply to the source node create a forward route to destination. This route created from each node from source to destination is a hop by-hop state and not the entire route as in source routing.

2.4. AOMDV

The AOMDV routing protocol [17] is an extension to the AODV protocol for computing multiple loop-free and link-disjoint paths. To keep track of multiple routes, the routing entries for each destination contain a list of the next-hops along with the corresponding hop counts. All the next hops have the same sequence number. For each destination, a node maintains the advertised hop count, which is defined as the maximum hop count for all the paths. This is the hop count used for sending route advertisements of the destination. Each duplicate route advertisement received by a node defines an alternate path to the destination. To ensure loop freedom, a node only accepts an alternate path to the destination if it has a less hop count than the advertised hop count for that destination. Because the maximum hop count is used, the advertised hop count therefore does not change for the same sequence number. When a route advertisement is received for a destination with a greater sequence number, the next-hop list and advertised hop count are reinitialized. AOMDV can be used to find node-disjoint or link disjoint routes. To find node-disjoint routes, each node does not immediately reject duplicate RREQs. Each RREQ arriving via a different neighbor of the source defines a node-disjoint path. This is because nodes cannot broadcast duplicate RREQs, so any two RREQs arriving at an intermediate node via a different neighbor of the source could not have traversed the same node.
In an attempt to get multiple link-disjoint routes, the destination replies to duplicate RREQs, the destination only replies to RREQs arriving via unique neighbors. After the first hop, the RREPs follow the reverse paths, which are node-disjoint and thus link-disjoint. The trajectories of each RREP may intersect at an intermediate node, but each takes a different reverse path to the source to ensure link-disjointness.

3. Performance metrics

To evaluate the performance of the AODV, DSDV, DSR, and AOMDV routing protocols using TCP traffic types, the following performance metrics will be used.

3.1 Packet Delivery Ratio

The Packet Delivery Ratio (PDR) can be defined as the ratio of data packets that is successfully delivered to the destination to those generated by the sources. Let C the number of flow or connections. Let $\mathrm{T_i}(\mathrm{i} = 1, 2, 3, \ldots, \mathrm{N})$ and

$R_i(i = 1, 2, 3, \ldots, C)$ are the number of data packets transmitted from flow i and received from i respectively. The value of the PDR can be written as.

$$PDR = \frac{1}{C}\sum_{1}^{C}\frac{R_i}{T_i}$$

3.2 Average end–to-end delay

The average end-to-end delay D is defined as the period of time a data packet takes to propagate from source to destination [15]. Let $t_i(i = 1, 2, 3, \ldots, C)$ and $r_i(i = 1, 2, 3, \ldots, C)$are the time at which a packet transmitted from flow i and the time at which a packet received at flow i respectively. The value of the average end-to-end delay D can be expressed as.

$$D = \frac{1}{N}\sum_{i=1}^{N}(r_i - t_i).$$

Where N denotes the total number of successfully received data packets.

3.3 Average Throughput

The average throughput (AVG) is the measure of how fast we can actually send packets through network. The number of packets delivered to the receiver provides the throughput of the network. The throughput is defined as the total amount of data packets a receiver actually receives from the sender divided by the time it takes for receiver to get the last packet. Hence, the AVG can be expressed as

$$AVG = \frac{8 \times N_p}{1000 \times (stop\ time - start\ time)}.$$

Where N_p denotes the total number of data packets a receiver actually receives from the sender.

4. Traffic Patterns

Traffic Patterns describe how the data is transmitted from source to destination. The two types of traffic patterns employed in MANET are CBR and TCP.

4.1 CBR Traffic

The qualities of constant bit rate CBR traffic pattern [16] are unreliable because it is a connectionless, i.e. it has no pre-connection establishment, there is no guarantee that is the data is being transmitted to the destination, unidirectional, and there no data acknowledgment from destination. In addition, CBR is predictable, fixed packet size, fixed interval between packets, and fixed stream duration.

4.2 TCP Traffic

The qualities of transmission control protocol TCP traffic pattern are reliable and connection oriented. Connection using TCP is established prior to transmitting data. There is a guarantee that the data is being transmitted to the destination. TCP traffic is a Bi-directional, i.e. every transmitted packet is acknowledged by the destination [14]. Moreover, in TCP traffic there is a flow control of data to avoid overloading the destination and a congestion control exists to shape the traffic such that it conforms to the available network capacity. More than 90% of the internet application is carried out through TCP.

5. SIMULATION SETUP

The entire simulations were carried out using NS-2.35 network simulator which is a discrete-event driven simulator developed at UC Berkeley [16] as a part of the VINT project. A network of 50 nodes that are placed randomly within a 670m x 670m area is considered. The simulation has been carried out for 200 seconds. The simulation operational parameters are described in table 1. Multiple runs with different node speed and number of nodes are conducted for each scenario. The collected data is averaged over those runs.

Table 1: Operational parameters

Parameters	*Value*
Numberof nodes	50
Simulation Time	200sec.
Area	670x670m2
Max Speed	5,10,20,30 50 m/s
Agent	TCP
Traffic Source	Reno, newreno, vegas, sack1
PauseTime (sec)	0,25, 50, 150, 200
Packet Size	512 Bytes
Packets Rate	4 Packets/s
Max. Number of connections	10,20,30
Mobility model used	Random way point

5.1 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

"~ns/indep-utils/cmu-scen-gen/setdest/". Users need to run 'setdest' program before running the simulation program.

./setdest [-v version] [-n num_of_nodes] [-p pausetime] [-M maxspeed] [-t simtime] [-x maxx] [-y maxy] > [outdir/movement-file].

Nodes in the simulation move according to random waypoint model. The movement scenario files which used for each simulation are characterized by a pause time. Each mobile Nodes (MN) begins the simulation by remaining stationary for a certain period of time (i.e., a pause time). Once this time expires, the MN chooses a random destination in the 670 m x 670 m simulation area and moves to that destination at a speed distributed uniformly between 0 and some maximum speed [0, maximum speeds]. The MN then travels toward the newly chosen destination at the selected speed. Upon reaching the destination, the MN pauses again for pause time seconds, selects another destination, and proceeds there as previously described, repeating this behavior for the duration of the simulation. Each simulation ran for 200s of simulated time. We ran our simulations with movement patterns generated for 5 different pause times: 0, 25, 50, 150, and 200 s. A pause time of 0 seconds corresponds to continuous motion, and a pause time of 200 s (the length of the simulation) corresponds to no motion. We tested with four different maximum speeds of node movement: 5, 10, 20, 30, and 50 m/s.

5.2 Generation of Random Traffic

Random traffic connections of TCP and CBR can be setup between mobile nodes using a traffic-scenario generator script 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. These scripts are stored in the file 'cmu-scen- gen' located in the directory- ~ns/indep-utils/cmu-scen-gen. The program "cbrgen.tcl" is used according to the command

Ns cbrgen.tcl[-type cbr|tcp][-nn nodes][-seed seed][-mc connections][-rate rate] > traffic-file

6. SIMULATION RESULTS

6.1 Impact of the mobility

To analyze the effect of mobility, pause time was varied from 0 seconds (high mobility) to 200 seconds (low mobility). The number of nodes is taken as 50 and the maximum number of connection is taken as 10 and 20. In all metrics, the traffic sources are set to TCP-Reno and maximum node speed 5 and 50 seconds.

1) Packet delivery ratio

Figures 1-4, illustrate the packet delivery ratio versus the maximum movement speeds with various traffic communication scenarios. From the Figures, it is noted that the AOMDV routing protocol outperforms all other routing protocols in most cases. Mowever, at low node mobility, the performance of all protocols seems to converge at lower speeds or in very high pause time scenarios. When the number of connections is increased to 20 connections, the performance of the AOMDV is very inferior to AODV. It is the presence of multipath in the routing table of AOMDV that allow it to do better than the others.

2) Throughput

Figures 5-6, show a comparison between the routing protocols in terms of throughput as a function of pause time and using TCP-Reno traffic sources and maximum node speed 5 and 50 sec. From the Figures, it is clear that the AOMDV routing protocol has the lowest average throughput of all protocols. However, at low speed the throughput is increased compared to higher speeds.

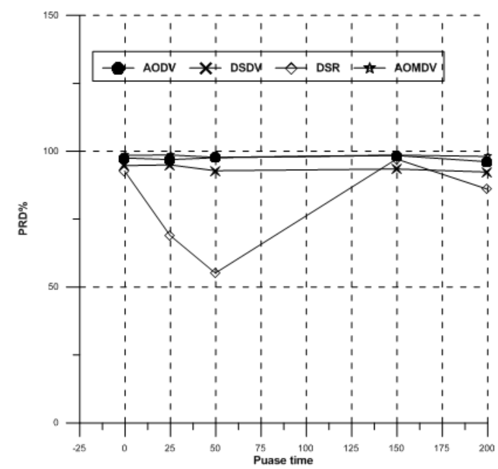


Fig. 1. PRD% for 10 data connections at the max speed of 5m/s.

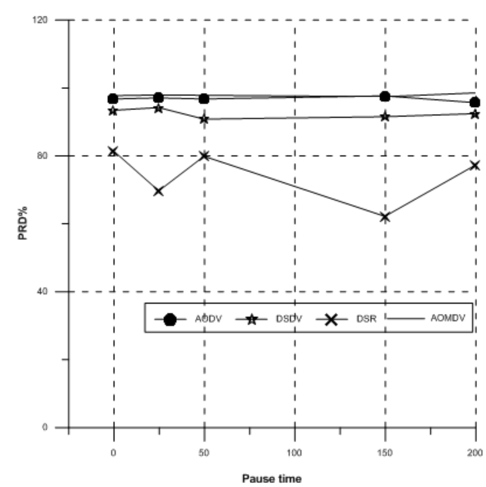


Fig. 2. PRD% for 10 data connections at the max speed of 50m/s.

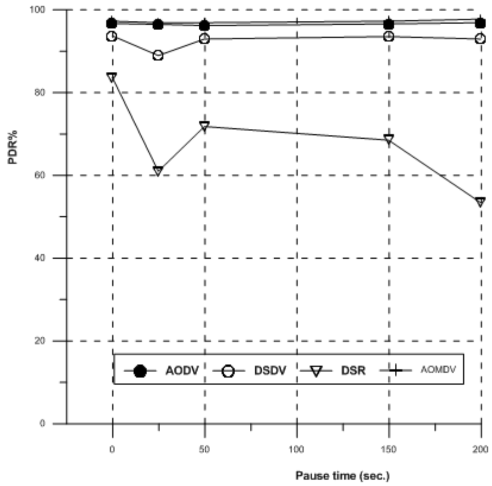


Fig. 3. PRD% for 20 data connections at the max speed of 5 m/s.

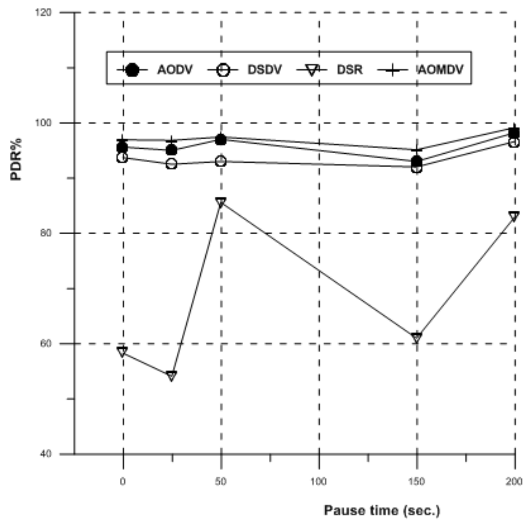


Fig. 4. PRD% for 20 data connections at the max speed of 50m/s.

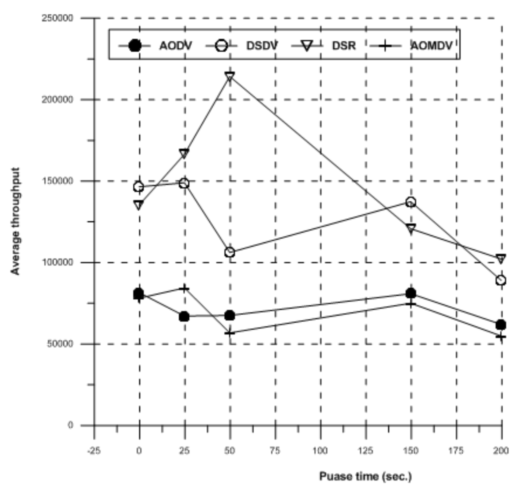


Fig. 5. Average throughput for 10 connections with node speed 5 m/s.

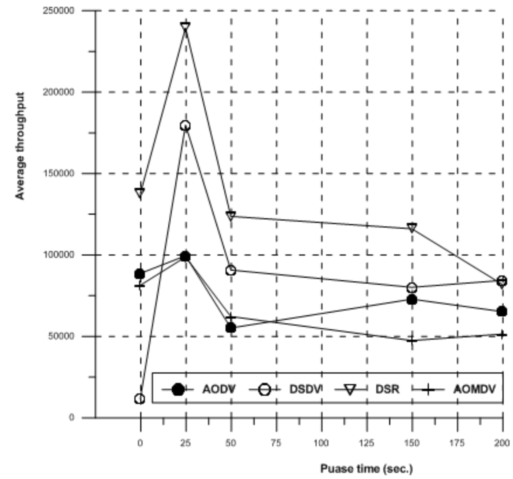


Fig. 6. Average throughput for 10 connections with node speed 50 m/s,

3) End-to-End delay

Figures 7-10 show the end to end delay versus pause time with TCP-Reno traffic source type. Different data connections (10 and 20) and different node movement speeds are used. From the Figures, it is noted that the average end to end delay increased in all protocols when increase the pause time (low mobility). Out of the four routing protocols, DSR has the longest average end-to-end delay while the DSDV outperforms all the other routing protocols when the number of connections is low and low mobility. On the other hand, when the number of connections increases and in both high and low mobility, the AODV and AOMDV close to each other.

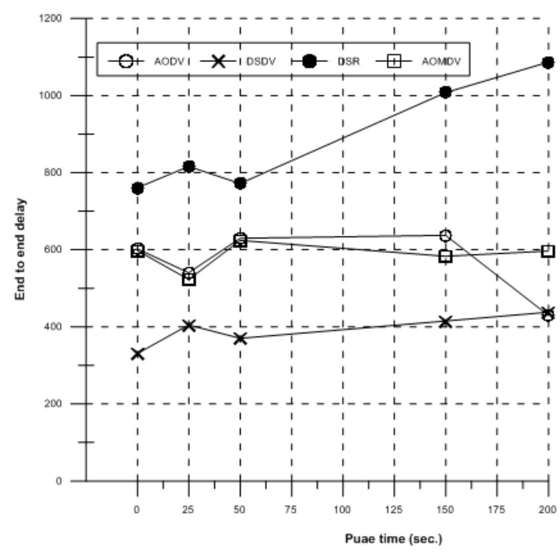


Fig. 7. End to end delay for 10 connections with speed of 5 m/s.

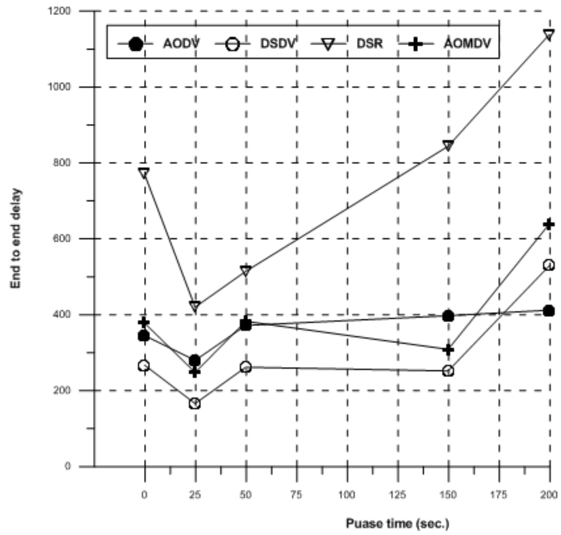


Fig. 8. End to end delay for 10 connections and speed of 50 m/s.

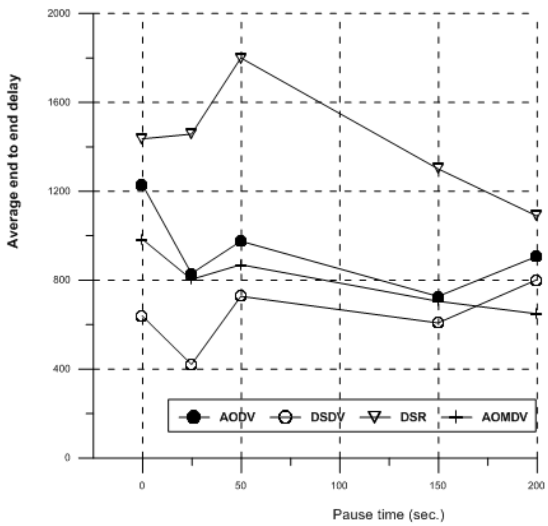


Fig. 9. End delay for 20 connections with speed of 5 m/s.

6.2 The Impact of Using Different TCP Traffic Types

Particularly, TCP has its variants, some of them namely TCP-Reno, TCP-Newreno, TCP-Vegas, and TCP-Sack. We examine the behavior of these variants over AODV, DSDV, DSR, and AOMDV routing protocols.

1) *Packet delivery ratio versus TCP traffic types.* Figures 11-12 show the PDR % for each AODV, DSDV, DSR, and AOMDV protocols versus the TCP traffic source with different maximum movement speed, and pause time 0 (high mobility). The Figures show that, when the number of data connections is increased the PDR % values decreased in all TCP types especially with DSR protocol. When the node movement speed increased the Ratio of the total number of data packets successfully delivered to total number of data packets sent deceased with all TCP variants, most likely due to route failure. Also when the number of data connections is increased the PDR % values decreased in all TCP types especially.

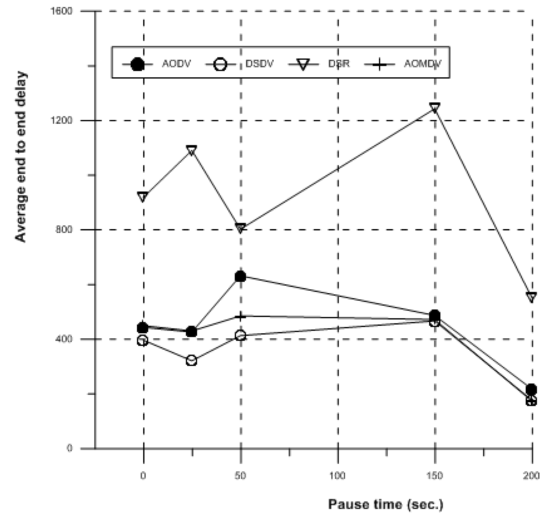
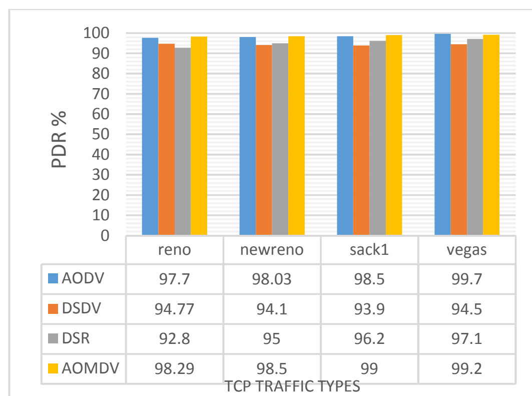
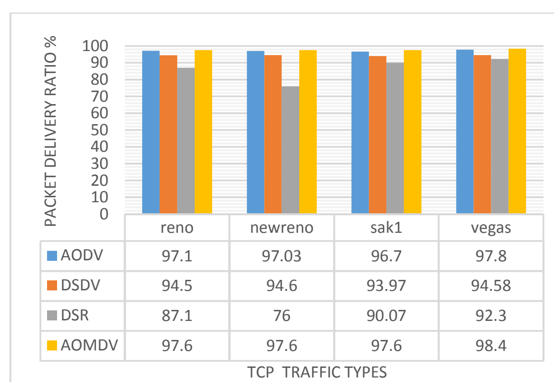


Fig. 10. Average of end to end delay for 20 connections with speed of 50 m/s.



	reno	newreno	sack1	vegas
AODV	97.7	98.03	98.5	99.7
DSDV	94.77	94.1	93.9	94.5
DSR	92.8	95	96.2	97.1
AOMDV	98.29	98.5	99	99.2

Fig. 11. Packet delivery ratio with different TCP traffic Types, for 10 data connections at maximum speed movement 5m/s.



	reno	newreno	sak1	vegas
AODV	97.1	97.03	96.7	97.8
DSDV	94.5	94.6	93.97	94.58
DSR	87.1	76	90.07	92.3
AOMDV	97.6	97.6	97.6	98.4

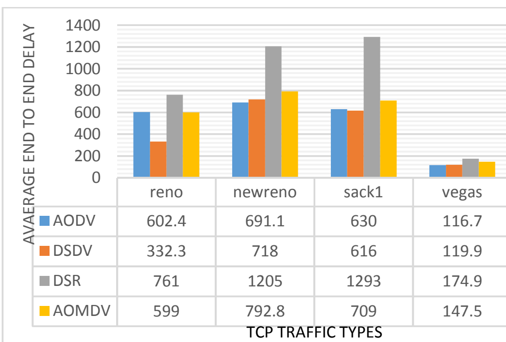
Fig. 12. Packet delivery ratio with different TCP traffic Types, for 10 data connections at maximum speed movement 30m/s.

2) Average End-to-End Delay versus TCP traffic types.

Figures 13-14 show the graphs for end-to-end delay versus TCP traffic types. AS illustrated from the graphs and data bellow, overall performance indicates that TCP Vegas is the one that has the lowest average end –to-end delay than others in different data connections 10 and 30 and with node movement speed 5 m/s. However, the average end to end delay of DSR routing protocol is the higher protocol than other protocols with different TCP types.

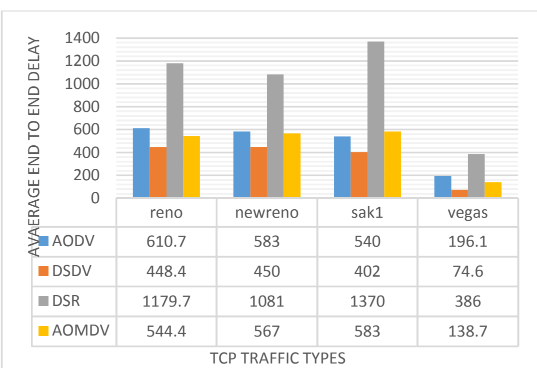
3) Average Throughput versus TCP protocol types

Figures 15-16 illustrate the throughput of the different routing protocols against the TCP traffic types. From the figures, it is clear that the DSDV protocol outperforms all other protocols using all the TCP types in different data connections and with different node movement speed. In addition, the AOMDV routing protocol has the lowest throughput in all cases.



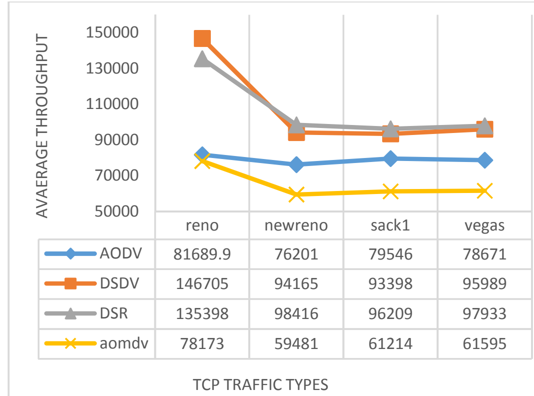
	reno	newreno	sack1	vegas
AODV	602.4	691.1	630	116.7
DSDV	332.3	718	616	119.9
DSR	761	1205	1293	174.9
AOMDV	599	792.8	709	147.5

Fig. 13. Average end to end delay with different TCP traffic types, for 10 data connections at maximum speed movement 5m/s



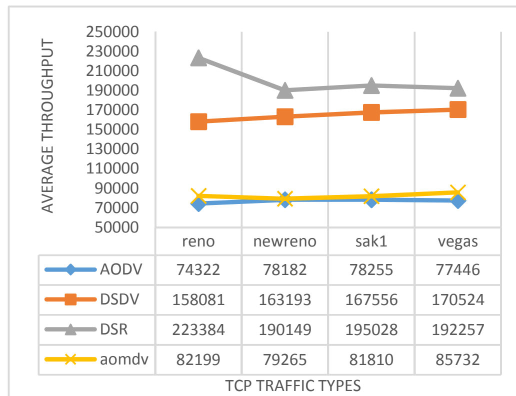
	reno	newreno	sak1	vegas
AODV	610.7	583	540	196.1
DSDV	448.4	450	402	74.6
DSR	1179.7	1081	1370	386
AOMDV	544.4	567	583	138.7

Fig. 14. Average end to end delay with different TCP traffic Types, for 30 data connections at maximum speed movement 5m/s.



	reno	newreno	sack1	vegas
AODV	81689.9	76201	79546	78671
DSDV	146705	94165	93398	95989
DSR	135398	98416	96209	97933
aomdv	78173	59481	61214	61595

Fig. 15. Average throughput with different TCP traffic Types, for 10 data connections at maximum speed movement 5m/s



	reno	newreno	sak1	vegas
AODV	74322	78182	78255	77446
DSDV	158081	163193	167556	170524
DSR	223384	190149	195028	192257
aomdv	82199	79265	81810	85732

Fig. 16. Average throughput with different TCP traffic types, for 30 data connections at maximum speed movement 5m/s

7. Conclusion

In this paper, we examined the performance differences of TCP-Reno, TCP-Vegas, TCP- Newreno and TCP-Sack1 when utilizing AODV, DSDV, DSR, and AOMDV as routing protocols in a mobile environment using NS2 simulator. The performance analysis is done in term of packet delivery ratio, average end to end delay and average throughput. A tool called ‘setdest’ developed by CMU (Carnegie Mellon University) is used to generate random movements of the nodes. A Tcl script called “cbrgen” is used to generate random traffic flow between mobile nodes.
From the viewpoint of TCP traffic types with the considered routing protocols AODV, DSDV, DSR, and AOMDV, the simulation results show that TCP-Vegas performs better compared with others (Reno, Newreno and Sack1) in the case of end to end delay, and has higher PDR%. The TCP-Reno has higher throughput in the case of low data connections compared with TCP-Newreno, TCP-Vegas and TCP-Sack1. In case of high data connections the TCP-Vegas has the higher throughput compared with the others.

REFERENCES

[1] Sapna S. Kaushik, P .R. Deshmukh, “comparison of effectiveness of AODV, DSDV, and DSR Routing protocols in Mobile Ad-hoc Networks”, International Journal of Information Technology and knowlegedge managmemt, 2009, vol. 2(2), pp.499-502.

[2] Royer, E.M., Santa Barbara, C. Toh. “A review of current routing protocols for ad hoc mobile wireless networks”, Personal Communications, Vol. 6 Issue 2, 1999.

[3] C. Perkins and P. Bhagwat, Highly dynamic destination-sequenced distance vector routing (DSDV) for mobile computers, in: Proceedings of the Conference on Communications Architectures, Protocols and Applications (ACM SIGCOMM ’94), London, United Kingdom, August–September 1994, pp. 234–244.

[4] T. Chen, and M. Gerla, Global state routing: a new routing scheme for ad hoc wireless networks, in: Proceedings of the IEEE International Conference on Communications (IEEE ICC ’98), Atlanta, GA, June 1998, pp. 171–175.

[5] G. Pei, M. Gerla, and T.-W. Chen, Fisheye state routing: a routing scheme for ad hoc wireless networks, in: Proceedings of the IEEE International Conference on Communications (ICC), New Orleans, LA, June 2000, vol. 1, pp. 70–74.

[6] G. Pei, M. Gerla, X. Hong, and C.-C. Chiang, A wireless hierarchical routing protocol with group mobility, in: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), New Orleans, LA, September 1999, pp. 1538–1542.

[7] D.L. Gu, G. Pei, H. Ly, M. Gerla, and X. Hong, Hierarchical routing for multi-layer ad hoc wireless networks with UAVs, in: Proceedings of IEEE MILCOM’00, Los Angeles, CA, vol.1, 2000, pp. 310–314.

[8] Ha Duyen Trung, Watit Beniapolakul, and Phan Minh Duc “Perfromance evalution and comparison of different ad hoc routing protocols” Scincedirect 2007, 2478-2496

[9] C.E. Perkins, E.M. Belding-Royerand, I.D. Chakeres, Ad Hoc On-Demand Distance Vector (AODV) Routing, IETF Internet Draft, draft-perkins-manet-aodvbis-01.txt, January 2004.

[10] E. Alotaibi and B. Mukherjee, “A survey on routing algorithms for wireless Ad-Hoc and mesh networks”, Computer Networks 56 (2012), pp. 940–965

[11] [D.B. Johnson, D.A. Maltz, Y. Hu, The dynamic source routing protocol for mobile ad hoc network, IETF Internet Draft, April 2003.

[12] C. Toh, Associativity-based routing for ad-hoc mobile networks, in: Journal of Wireless Personal Communications, 4, 2, March 1997, pp. 103–139.

[13] R. Dube, C.D. Rais, W. Kuang, and S.K. Tripathi, Signal stabilitybased adaptive routing (SSA) for ad hoc mobile networks, in: IEEE Personal Communications Magazine, vol. 4, no. 1, February 1997, pp. 36-45.

[14] S. Floyd, T. Henderson, and A. Gurtov, “The NewReno Modification to TCP's Fast Recovery Algorithm”, Internet Draft, April 2004.

[15] A nil Kumar and Neha Bhatia, “Behaviroal study of MANET routing protocols by using NS-2” Internal journal of computinal engineering & management, vol. 12, april 2011.

[16] Vikas Singla, Parveen Kakkar,“Traffic Pattern based performance comparison of Reactive and Proactive protocols of Mobile Ad-hoc Networks”, International Journal of Computer Applications, 2010,Vol. 5(10), pp.16-20.

Hassan Al-Mahdi He received a B.Sc. degree in Computing Science from Suez Canal University, Egypt in 1994, an M.Sc. degree in Computer Science from Suez Canal University, Egypt in 2001, a Doctoral degree in Wireless Networks in 2006 and associate professor in computer networks from Suez Canal University, Egypt in 2012. From 201 to 2005 he served as lecturer at Suez Canal University, Egypt, assistance professor of computer science at Suez Canal University, Egypt from 2005 to 20012 and associate professor of computer networks at Suez Canal University, Egypt from 2012 to 20013. Currently, he is a full associate professor at Al Jouf University, Saudi Arabia and head of the Computer Science & Information System department. He has 30 papers mostly in the area of data communication and performance evaluation. His research focuses on ad hoc networks, mobile cellular communications, and performance evaluation of networks.

Mohamed Waheed He is a Professor and dean of Faculty of Computer Science, Suez Canal University, Ismailia, Egypt. He received his First degree in Mathematics, Computer Science and Operation Research, Faculty of Science (1984), Master’s degree in Computer Science. Faculty of Science (1990) and PhD degree in Decision Support Systems, Faculty of Science, Zagazig University (1995). His research interests include neural network optimization, scheduling, heuristic search (genetic algorithms), decision support systems and image processing.

Tarek M Mahmoud He is a Professor and dean of Faculty of Computers & Information, Minia niversity, Minia, Egypt. He received his Ph.D. degree in engineering (computer science) from Bremen University (Germany) in 1997. Since 1997, he has been with the Department of computer science of Minia University. Currently, he is an associate professor there. His research interests include wired and wireless networks, pattern recognition, combinatorial optimization, metaheuristics algorithms, web and semantic mining.

Hassan Shaban He received a B.Sc. degree in Computing Science from Minia University, Egypt, an M.Sc. degree in Computer Science from Minia University, Egypt in 2009. Currently, he is associate lecturer, Faculty of Computers & Information, Minia University, Egypt. His research focuses on Mobile IP, routing, ad hoc networks, and performance evaluation of networks.