Impact of Route Selection Metrics on the Performance of DSR Protocol for MANETs

Ragb O. M. Saleh[1], Md Yazid Mohd Saman[1] and M Nordin A Rahman[2]

**[1] Faculty of Sciences and Technology, University Malaysia Terengganu
Kuala Terengganu, Malaysia**

**[2] Faculty of Informatics, Universiti Sultan Zainal Abidin,
Kuala Terengganu, Malaysia**

Abstract

Mobile Ad-hoc Networks (MANETs) are characterized by connectivity through a group of wireless nodes. MANETs are deployed in circumstances where no base station is available. Accordingly, MANETs need routing protocols which can adopt with dynamic topologies. To achieve this, several routing protocols are proposed and deployed. The route selection is one of the most important method design optimization criterion in routing protocols. Most conventional route selection methods do not consider both the freshness and shortness of the route while selecting routes. In this paper, we present two route selection methods for DSR protocol, which consider the freshness of the route as the primary route selection metric. In addition, we describe the proposed methods and explain their effectiveness via presenting and comparing simulation results with popular DSR protocol. The parameters used for evaluation are packet delivery ratio, routing overhead and dropped packets. The simulations are carried out using GloMoSim simulator.

Keywords: *MANET, DSR, Source Routing, Minimum Hop, Route Selection*.

1. Introduction

Mobile Ad-hoc Networks (MANETs) [11][18] consist of wireless mobile nodes communicating without the aid of any centralized administration or established infrastructure. MANET is deployed in circumstances where no access point is available, and a network has to be built unplanned. The hosts in MANET move arbitrarily, self organized and decentralized nodes. Furthermore, all hosts can be mobile and the topology of the network change continually. Main challenges in MANETs are routing of messages with frequently free nodes movement. Therefore, routing protocols are an important issue in MANET communications. One of most popular routing protocols in MANETs is reactive routing protocols. They also named on-demand routing protocols. These kinds of protocols build paths only when a source node wants to send data packets to some destination node. The source checks for path availability. If there is no path exists, it calls the path discovery process to discover a path to the intended destination. The route discovery mechanism typically consists of the network-wide flooding of a route request packet. When a route between the source and the destination has been established; this route maintained by the mechanism of route maintenance during the transmission session.

Most of the existing on-demand protocols [11, 4, 20] utilize hop-count as routing metric for route selection process. However, One of the significant issues in minimum hop-count routing, the routes selected based on hop count only possibility of old construction, and ignoring the possibility that a recent/fresh path might offer higher throughput [10]. Examples of on-demand routing protocols are Dynamic Source Routing (DSR) [6], and Ad hoc On Demand Distance Vector (AODV) [4].

In MANET, one of the most important concerns is how to find a recent short route from source to the destination in presence of mobile nodes. In this paper, we determine the recentness and shortness of route based on the source of route reply (R.RP) by adding a flag to the structure of the R.RP packet. This flag determines who the source of R.RP is; a destination or an intermediate node. Accordingly, the route will be marked as a recent route if the source of the R.RP is the destination node. Otherwise, it will be un-recent route. In the earlier works, the freshness or the recentness of route was not considered as important for performance comparison of MANET protocols, but importance of the freshness of route was highlighted in this work.

2. Related Work

Work in [17] presents a performance comparison of four proposed multi-hop routing metrics: Expected Transmission Count (ETX) [8], Per-hop Round Trip Time (per-hop RTT) [1], Per-hop Packet Pair (per-hop RPP) [19], and the Minimum Hop-Count. It studies these metrics using a DSR-based routing protocol running. Furthermore, describes a routing protocol that incorporates the notion of link quality metrics. Subsequently, presents detailed experimental results to show that in scenarios with stationary nodes, the ETX metric out-performs hop-count although it uses longer paths. The experimental evaluation shows that the one-hop RTT and one-hop packet-pair metrics perform poorly, because their load-sensitivity leads to self-interference. In addition, a comparison study has been done between the link-quality metrics and the minimum hop-count metric; it shows that the hop-count metric better all of the link-quality metrics in a situation where the sender node is mobile; the reason for that is because the minimum hop-count metric respond sufficiently rapidly.

Work in [3] has explained that the minimum hop-count metric (shortest path) is likely to find routes with long slow links. Accordingly, low effective throughput and high network congestion has exposed due to the selected routes. Moreover, the selected routes are tending to contain long links. To improve the route selection; the Medium Time Metric (MTM) technique has presented. MTM selects the route which has the highest effective capacity. The experimental results have shown that MTM technique gives higher throughput than other metrics. Furthermore, explained the importance of using MTM for improve the route selection process.

Work in[12] has evaluated and compared the performance of three protocols, two of them based on on-demand multicast routing protocol (ODMRP [20]) with different routing metrics [16]: (i) ODMRP_DSR: ODMRP has modified to choose routes based on hop-count metric, as chosen by the Dynamic Source Routing (DSR) protocol. (ii) ODMRP_FORP: ODMRP has modified to predicted link lifetime, as chosen by the Flow-Oriented Routing Protocol (FORP). (iii) OptMeshTrans: algorithm has proposed to determine the sequence of stable multicast meshes which connect a set of sources to a set of destinations. The protocols have simulated under different environments of node mobility, network density, number of sources and destinations. As result, there is no important impact of the route selection metrics on the energy consumption per node for ODMRP_DSR and ODMRP_FORP with respect to hop-count metric per source-destination route; meshes formed using these two protocols have relatively lower hop count. In contrast, and compared to the meshes determined using the two ODMRP-based protocols, the OptMeshTrans provides the most stable meshes with less number of links per mesh, and incur lower energy consumption per node.

3. Dynamic Source Routing (DSR) Protocol

3.1. Overview of the DSR Protocol

DSR [11] is a popular on-demand routing protocol to implement in MANETs. For routing process, DSR utilizes a routing technique called Source Routing (SR). SR means that the source node determines the complete sequence of nodes through which the packet has to pass. That is, the data packets carry the complete hop-by-hop route to the destination in the packet header. DSR nodes use a route cache to maintain multiple routes for future use.

Basically, The DSR protocol depends on two on-demand mechanisms: Route Discovery (RD) and Route Maintenance (RM). DSR applies RD scheme only when a source node wants to send data packets to a destination node, and no route in the cache is available. RD starts by flooding the network with route request (R.RQ) messages. Each intermediate node receiving a R.RQ, if it is the destination or it has a route to the intended destination, route reply R.RP message is sent back to the source. Otherwise, the received node will rebroadcast R.RQ again to its neighbours, unless it has received the same R.RQ. When the R.RP is sent back to the source, the source and all intermediate nodes will cache the route for future use. On the other hand, DSR invokes RM scheme as soon as a node fails to forward a data packet to the next hop on the source route. In that case, a route error (R.ER) message will send back to notify the source node. Then, the node that detects that failure link will try to salvage the packet by checking its own cache to find alternate route to the intended destination. However, when a node receives or overhears the R.ER message, all the routes that use the broken link will erase from its route cache. When the source receives the notification of failure route, and this route still desired. The RD process must be restart by the source node.

Additional DSR features, it allows mobile nodes to overhear or listen to messages using the promiscuous mode, thus nodes know useful information about the network (e.g. R.RP, R.ER, etc.). One more; in the source route when an intermediate hop becomes no longer needed, the source route will be automatically shortened by using automatically shortening approach. Also, DSR nodes have sleep mode to save their energy and bandwidth, also it is loop-free.

3.2. Weakness of Route Selection Policy in DSR

In standard DSR, during the route discovery process, it applies the route selection approach based on the shortest-route metric. 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. Basically, mobile nodes can cache a route when overhear it or relay a packet unless it has cached the same route. Also, the node can cache multiple routes to a destination ordered by hop count [13]. Cached routes are continually updated and reordered when a new route is learned according to shortest path policy [25][15][22][24]. The original DSR keeps multiple routes to a destination ordered by hop count.

Despite those advantages, existing route selection approach of DSR [6][5] has some weaknesses. It will not be an efficient method without an effective route selection metric. Our concern is that routes which maintain in caches might be into an inefficient order, which have a significant negative impact on the route selection. In particular; if the cache contains multiple routes to the same destination with an equal number of hop-counts, the source will apply the shortest path policy to select the shortest route to the destination from its cache, regardless of the source of route and its construction time. Also, cached routes might be in disorder despite equality of hop-counts and their destination. The problem with this approach is that, while the source is still using the primary shortest route, the primary route might be fail, and the source would remain unaware of that its cache contains a recent/fresh route with same number of hop count to the same destination. Therefore, the selected source route might not be the best shortest route, and the route which having the minimal number of hops does not frequently mean the best route. Consequently, we need a routing metric which help to select better source routes by explicitly taking into account the shortness and freshness of the selected route.

In this paper, we study the performance of two route selection metrics, and compare them against minimum hop-count "shortest path" routing. The first route selection metric is based on "freshest-short path", whereas the second one is based on "freshest path".

4. Proposed Techniques for DSR Route Selection

In this paper, we consider two route selection methods based on different routing metrics for DSR protocol. The first one based on "Fresh-Shortest Route", whereas the second based on "Freshest Route". We also support minimum hop-count routing by defining a "Shortest Route" metric for standard DSR. Each of these routing metrics represents a different concept of DSR route selection. The shortest path doesn't always describe the best available path from the source node to the destination node in MANET's environment. In the updated DSR, the metric of route selection has been changed from hop count (or the shortest path) to: (a) the freshest-short path (the selected path depends on two operators; the number of hop-count, and the source of R.RP), and (b) the freshest path (the selected path depends on two operators; the source of R.RP, and the construction time of path).

4.1. Route Selection based on "Fresh-Shortest Route First"

For DSR protocol, we present a new route selection scheme, the key of improvement in our scheme is that the performance of DSR can be achieved by selecting the fresh-shortest route to the intended destination. We call it FSDSR algorithm. FSDSR applies the fresh-shortest route as routing metric. FSDSR estimates the freshest of the route using a new policy. Where, the source node gives the priority to R.RP packets which answered back by the original destination of data packet rather than intermediate nodes. In standard DSR protocol, R.RP packets reply back to the source node by intermediate nodes or the intended destination node. Wherever, nodes employ their caches to send R.RP packets to the source node. Mostly, due to the high mobility of nodes; cached routes are likely to be disjointed. As result, if the selected cached route fails frequently, the path selection scheme will be a time consuming method.

In this work, for estimate a fresh-short route; the proposed FSDRS mechanism gives the priority for shortest routes that answer back by the intended destination rather than intermediate nodes. The source labels each a source route as a fresh route, if it was replied by the destination itself. If there is more than one source route labeled as a fresh route for a destination, the fresh-shortest route among the fresh routes will be selected by the source node. Whereas, if there is no source route labeled as a fresh route, the shortest route in the route cache will be selected as a source route regardless of the source of R.RP. This criteria will ensure a better route is selected and not simply the shortest route. A trade-off can be made between the freshness of the route and the hop count from source to destination.

Essentially, the proposed FSDSR method has five cases:

Case 1: If a source (S) desires to send information to a destination (D):

- If S has one or more routes in its route cache: select the shortest route (the priority gives for routes which replied by the destination).
- If S has not a route to D: propagate a route request packet (R.RQ), and wait for R.RP.

Case 2: If an intermediate node received a R.RQ:

- If it has a route to D, send R.RP with Flag=0 to S.
- If has not; re-propagates R.RQ.

Case 3: If D received a new R.RQ:

Stop the propagating of R.RQ.

Send R.RP with Flag=1 towards S.

Case 4: If an intermediate node received a new R.RP:

- Caches the route with its flag status (0/1).
- Forward the R.RP to next intend intermediate node towards S.

Case 5: When S receives a new R.RP, caches the route, then:

- If S has one or more than cached route with Flag=1 (fresh route), select the shortest-fresh one as the candidate source route.
- If there is no R.RP with Flag=1, select the shortest route as source route.
- Stop the propagating of R.RP.

4.2. Route Selection based on "Freshest Route First"

The key of improvement in this approach is that the performance of DSR can be achieved by selecting a recent source route. We call it as FDSR (Freshest Route Selection for DSR). As the response to solve route selection problems in DSR protocol, FDSR introduces a new route selection strategy that utilized the freshness of the source route as route selection metric. FDSR tries to select the freshest source route based on two operators: the source of the route reply and the time of construction of the source route. It allows mobile nodes to reorder the cached routes as soon as a new route has learned; the reordering will do according to "Freshest Route First (FRF) policy" by giving the priority to the route whose reply by D and has the recent time of construction (the recent time of building the route compared to the cached routes).

As result, in small MANET's environment, FDSR gives some advantages; nodes can save its resources (i.e., bandwidth and power consumption) by reducing recall the route discovery process, which is costly. Also, some performance objectives can be achieved by FDSR such as high delivery ratio, low overhead and fewer dropped packets.

Basically, the proposed FDSR method has three cases as the following:

Case 1: If a source (S) desires to send information to a destination (D):

- If S has one or more routes to D: give the priority to the route whose reply by D (Flag =1), and has the recent build time "D applies FRF policy".
- If S has not; propagate route request packet (R.RQ), and wait for route reply packet (R.RP).

Case 2: If S receives a new R.REP:

- Reorder routes cache according to FRF policy.
- Stop the forwarding of R.REP.

Case 3: If an intermediate node receives a new R.REP:

- Reorder cached routes according to FRF policy.
- Forward the R.REP to next intend node.

The shortest path doesn't always describe the best available path from the source node to the destination node in MANET's environment. In the updated DSR, the metric of route selection has been changed from hop count (or the shortest path) to the freshness of path (the path selection strategy depends on two operators; the build time of path, and the source of R.RP).

4.3. Pseudo-Code of FSDSR and FDSR

This section presents a pseudo-code to describe the details of FDSR and FSDSR. Table (1) lists the common variables and functions that are used in the pseudo-code of both FSDSR and FDSR.

Table 1: Common pseudo-code variables and functions

Variable	***Description***
S	The source node
D	The destination node.
SR	The source route which selected by the source to send information to a specific destination.
SRD	Set of source routes have same hop-count & same destination.
RC	Route cache
FR	Freshest Route to the destination
T	A period of time for route discovery phase.
Seek _RC()	Function to check if there is a source route in the node's cache.
FDSR()	Function for find the freshest route to the destination.
R..RP	Route Reply Packet.
R.RP→SR	The Source Route which sent back to the source using Route Reply Packet.
R..RP→D	The destination of the route replay packet.
node	The current node which received the route reply packet.
Send_RQ()	Function utilizes to send route request for the intended destination.
Send_D ()	Function utilizes the source route to send data to the intended destination.
Ins_RC()	Function for save the new Source Route in the route cache.
SRt	The construct time of the source route.
SRnew	A new Source Route.
Relay()	Function for forward the route reply to the next intended node.
Stop()	Function for end forwarding the route reply packets.
Crnt_node_adr	The address of the current node.
Intnd_nod_adr	The address of the intend node of route reply packet.
n	Number of source routes in the route cache.

Note: The pseudo code of our methods is presented below (*).

5. Simulation Results

In this work, we have used Global Mobile Information System Simulator (GloMoSim) [9] to simulate the proposed methods (FDSR and FSDSR). Also, we have evaluated and compared the performance of SDSR and FSDSR with the standard DSR in MANETs environment.

5.1. Simulation Environment

Table (2) shows the common parameter values in our simulation:

Table (2): Simulation Parameter Values

Parameter	***Value***
Simulation-Time	900 seconds
Simulation Area	2200m x 600m
Number-of-Nodes	50 nodes
Mobility Model	Random Way-point Model
Pause Time	0/300/600/ 900 seconds
Bandwidth	2 Mbps
Mac-Protocol	IEEE802.11
Network-Protocol	TCP - UDP
Routing-Protocol	Standard DSR/FDSR/FSDSR
Data traffic - CBR	4 packets/sec
Packet Size	512 bytes
Nodes speed	0 to 10 & 0 to 15 m/s

5.2. Performance metrics

The following performance metrics are evaluated [2][23][14]:

1. *Packet Delivery Fraction (PDF):* PDF is defined as the average of the number of data packets successfully transmitted to the destinations and number of data packets originated by sources.

$$PDF = \left[\left(\sum R_{packets}\right) \Big/ \left(\sum G_{packets}\right)\right] \times 100$$

Where, $R_{packets}$ is the number of received data packets by destinations, and $G_{packets}$ is the number of sent data packets by sources.

2. *Routing Overhead (RO):* RO is defined as the average of the number of routing control packets and the total number of received data packets.

$$RO = \left(\sum C_{packets}\right) \Big/ \left(\sum R_{packets}\right)$$

Where, $R_{packets}$ is the number of received data packets by destinations, and $C_{packets}$ is the number of sent control packets by nodes.

3. *Number of Dropped Packets (NDP):* The data packets that dropped during transmission session due to the link breaks and collisions.

$$NDP = \sum D_{packets}$$

Where, $D_{packets}$ is the number of dropped data packets in the network.

4. Number of Broken Links (NBL): The links that fail during transmit data packets.

$$NBL = \sum B_{Links}$$

Where, B_{Links} is the number of failure links in the network.

5.3. Effect on Packet Delivery Fraction (PDF)

In order to evaluate the improvement in packet delivery fraction (PDF), we examine the standard DSR (using the shortest route as routing metric) and DSR enhanced with FDSR (using the freshest route as routing metric) and FSDSR (using the freshest-short route as routing metric) with varying pause time of mobile nodes. Since we require route discovery to use the proposed route selection methods (FDSR or FSDSR), we disable the shortest path strategy in FDSR route selection. While in FSDSR, we combine the proposed freshness path strategy with the shortest path strategy for find out the freshest-short route. As observed in Fig. 2. PDF for all methods (DSR, FDSR and FSDSR) increases when nodes move less rapidly because low mobility is less prone to broken links, while PDF decreases with high mobility of nodes, which may force the routing selection process to select a stale route.

The results of the PDF experiments are shown in Fig. 2. It shows the fraction of PDF successfully against decreasing the pause time of nodes (increasing mobility of nodes). According to the observed PDF results, FSDSR consistently outperforms both FDSR and standard DSR. The gains of PDF with FSDSR are higher, indicating the fact that freshest-short routes are being exploited. This is expected, since increase in mobility means the shortest route is often not the best one (especially when the source of shortest route is not the destination node). However, the gains with FDSR are lower than both FSDSR and DSR for the following reason. There is no appreciable length of the selected fresh route; so the selected fresh route is often a long route. This increases the possibility of failure links due to the mobility of nodes, which lead to decrease the PDR in the network.

5.4. Effect on Routing Overhead (RO)

In DSR, the main reason of routing overhead (RO) is the frequent route discovery process, which affects the performance of DSR protocol. For that reason, a significant benefit of selecting a stable route is the reduced routing overhead. Since, we require route discovery mechanism to use a route selection method which leads to a choice of routes that are more efficient. As observed in Fig.3. the RO for all methods (DSR, FDSR and FSDSR) decreases when nodes move less rapidly because low mobility is less prone to broken links, while it increases with high mobility of nodes, which may force the routing selection process to select a stale route.

The results of the RO experiments are shown in Fig. 3. It shows that the fraction of RO successfully against decreasing the pause time of nodes (increasing mobility of nodes). According to the observed RO results, FSDSR consistently outperforms both FDSR and standard DSR. The RO of FSDSR is the lowest, indicating the fact that freshest-short routes are being exploited. This is expected, since increase in mobility means the shortest route is often not the best one, and the protocol needs to recall route discovery process again (especially when the source of shortest route is not the destination node). However, the RO with FDSR are higher than both FSDSR and DSR. The most important reason is no appreciable length of the selected fresh route by FDSR; so the selected fresh route is often a long route. In FDSR, despite the freshness of selected route, often the long route has higher possibility of failure links due to the mobility of nodes, which lead to frequent discovery process for an alternative route, and then decrease the RO. The results presented in this section demonstrate that the route selection based on freshest-short route (FSDSR) achieves the lowest RO.

5.5. Effect on Number of Broken Links (NBL)

The traditional route selection of DSR protocol are mainly based on the shortest path as routing metric[7], which in turn depends on the caching policy. It cannot effectively improve the efficiency of cached routes in cache nodes. The efficiency of cached routes decreases when the number of broken links of the cached routes is increased. So, we need a route selection method that selects the route which does not contain a broken link.

As observed in Fig.4. the number of broken links (NBL) of all methods (DSR, FDSR and FSDSR) decreases when nodes move less rapidly because low mobility is less prone to broken links, while it increases with high mobility of nodes, which may force the routing selection process to select a broken route.

In both low and moderate mobility (pause time =900; 600; 300 sec); FSDSR presents the lowest NBL compared to standard DSR and FDSR. The reduced NBL is due to the successful route selection policy. However, standard DSR presents the lowest NBL in the high mobility environment (pause time =0 sec), while FDSR introduces the highest NBL in low and high mobility environments. As result, FSDSR in the simulation result shows the overall best result in almost situations.

5.6. Effect on Number of Dropped Packets

The lower number of dropped packet (NDP) of the standard DSR, FDSR and FSDSR protocols contributes to their higher forwarding packets rate, where intermediate nodes successfully forward data packets to their destinations according to the routing protocol. However, as the broken links increase with mobility, the NDP of all methods (DSR, FDSR and FSDSR) drops with the decrease in pause time of mobile nodes. Accordingly, we need a route selection method that selects the route which does not contain a broken link. As observed in Fig.5. The NDP of all methods (DSR, FDSR and FSDSR) increases when nodes move more rapidly because high mobility is more prone to broken links, whereas the ratio of NDP decreases with low mobility of nodes. In most mobility environments (pause time =900; 600; 300 sec); FSDSR presents the lowest NDP compared to standard DSR and FDSR. The reduced NDP is due to the successful route selection policy. At zero mobility, standard DSR presents the lowest NDP in the high mobility environment (pause time =0 sec), while FDSR introduces the highest NDP in low and high mobility environments. As result, FSDSR in the simulation result shows the overall best result in almost situations.

6. Conclusion

We present two route selection methods for DSR protocol. We called them FDSR and FSDSR. FDSR based on the freshest route as route selection metric. While FSDSR based on the freshest-short route. In addition, our contributions in this paper are the investigation about the impact of route selection metrics on the performance of the DSR protocol, and a comparison of the performance of FDSR and FSDSR with standard DSR. This paper evaluated the performance of DSR, FDSR and FSDSR using GloMoSim. Comparison was based on the packet delivery ratio, routing overhead, number of drop packets and number of link breaks. We concluded that in most simulation scenarios FSDSR gives better performance as compared to FDSR and DSR in terms of packet delivery ratio, routing overhead, number of drop packets and number of link breaks. Also we have seen that DSR protocol is best in terms of routing overhead, number of drop packets and number of link breaks in individually high dynamic network (only when pause time zero sec).

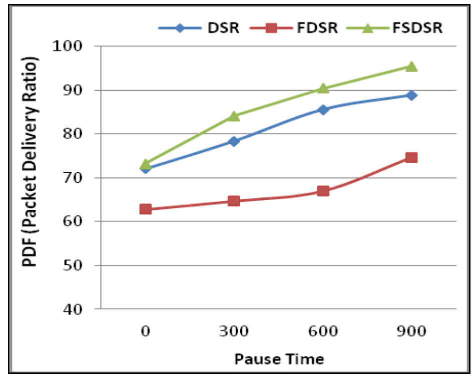

Fig. 2. PDF .vs. Pause Time

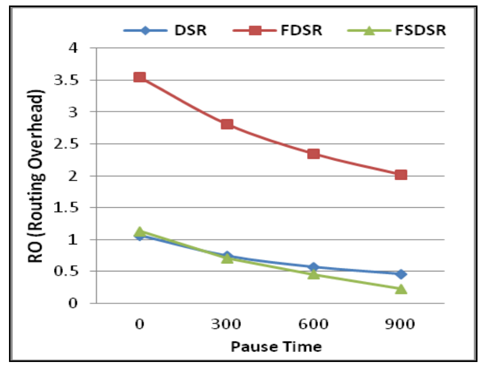

Fig. 3. RO .vs. Pause Time

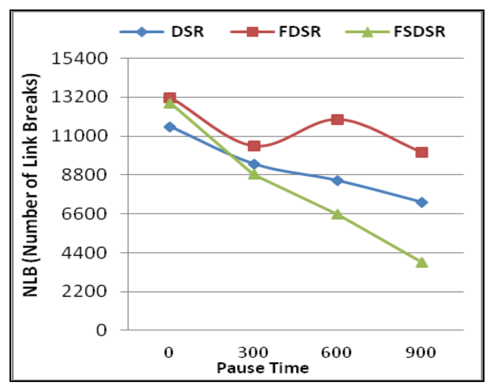

Fig. 4. NBL .vs. Pause Time

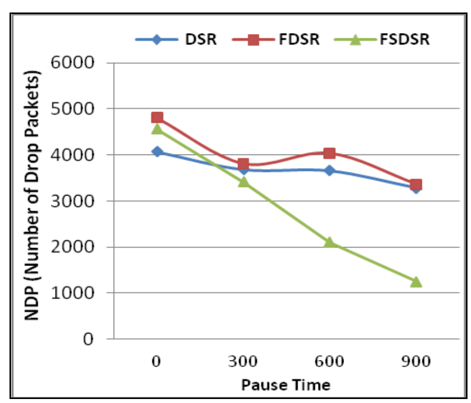

Fig. 5. NDP .vs. Pause Time

References

[1] Adya, P. Bahl, J. Padhye, A. Wolman and L. Zhou, "A multi-radio unification protocol for ieee 802.11 wireless networks", *in Proceedings of BroadNets conference*, 2004.

[2] El Gamal, J. Mammen, B. Prabhakar, D. Shah, "Throughput-delay trade-off in wireless networks", *in Proceedings of IEEE INFOCOM Conference on Computer Communications,* 23 (1):464-475, March 2004.

[3] Awerbuch, D. Holmer, H. Rubens, "The medium time metric: High throughput route selection in multirate ad hoc wireless networks", MONET, *Spec. Iss. on Internet Wireless Access: 802.11 and Beyond*, 2005.

[4] C.E.Perkins and E.M.Royer, "Ad hoc on demand distance vector routing", *Proceedings of IEEE Workshop on Mobile computing systems and Applications*, pp. 90-100, February 1999.

[5] D. B. Johnson, D. A. Maltz, Y.C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)", *IETF Draft*, 2003.

[6] D. Johnson, D. B. Maltz, D. A. and Broch, J., *The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks*, Ad Hoc Networking, Perkins, C. E., Addison-Wesley Publisher, 2001.

[7] D.A. Maltz, J. Broch, J. Jetcheva, "The effects of on-demand behavior in routing protocols for multihop wireless Ad Hoc networks", *IEEE Journal on Selected Areas in Communications*, 1999, 17(8), pp1439-1453.

[8] D.S.J. De Couto, D. Aguayo, J. Bicket and R. Morris, "A highthroughput path metric for multi-hop wireless networks", *Proceedings of 9th Annual International Conference on Mobile Computing and Networking (MobiCom 03),* Sep. 2003.

[9] Global Mobile Information Systems Simulation Library, *http://pcl.cs.ucla.edu/projects/glomosim/*.

[10] M. H. Mamoun, "A Proposed Route Selection Technique in DSR Routing Protocol for MANET", *International Journal of Engineering & Technology IJET-IJENS*, April 2011, Vol: 11 No: 02 10118702-0404.

[11] M. Barbeau, E. Kranakis, *Principles of Ad-hoc Networking*, Wiley, 2007.

[12] M. Natarajan, "Impact of Route Selection Metrics on the Performance of On-Demand Mesh-based Multicast Ad hoc Routing", *Computer and Information Science*, Jackson State University, Vol. 3, No. 2; May 2010.

[13] M. Neelakantappa, B. Satyanarayana, A. Damodharam, "Performance Improvement Techniques for Dynamic Source Routing Protocol in Mobile Ad Hoc Networks'', *International Journal of Recent Trends in Engineering*, November 2009, Vol 2, No. 2.

[14] M. Shama, R. L. Dua, "Investigation of Performance Matrics of Dynamic Source Routing with Different Terrain Areas and Pause Time for Wireless Sensor Network", *International Journal of Advanced Research in Computer Science and Software Engineering*, July 2012, Vol. 2, Issue 7, ISSN: 2277 128X.

[15] M.K. Marina, S.R. Das, "Performance of Route Caching Strategies in Dynamic Source Routing". *Proceedings of the 2nd Wireless Networking and Mobile Computing (WNMC)*, Mesa, AZ, USA, pp. 425-432, April, 2001.

[16] N. Meghanathan, "Survey and Taxonomy of Unicast Routing Protocols for Mobile Ad hoc Networks", *The International Journal on Applications of Graph Theory in Wireless Ad hoc Networks and Sensor Networks*, Dec 2009 ,Vol.1, No.1.

[17] R. Draves, J. Padhye and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks", *in Proceedings of ACM SIGCOMM*, Portland, Oregon, USA, 2004.

[18] S. Basagni, m. Conti, s. Giordano, and i. Stojmenovic, *Mobile Ad Hoc Networking*, Wiley IEEE Press, 2004.

[19] S. Keshav, "A control-theoretic approach to flow control", *in Proceedings of SIGCOMM*, 1991.

[20] S.J. Lee, M. Gerla, , C. Chaing, "On-Demand Multicast Routing Protocol", *in Proceedings of IEEE Wireless Communications and Networking Conference*, USA, 1999.

[21] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", *Proceedings of the 9th IEEE International Conference on Computer Communications and Networking (INFOCOM'97)*, Kobe, Japan, pp. 1405-1413, Apr. 1997.

[22] W. Lou, F. Yuguang, "Predicative Caching Strategy for On-Demand Routing Protocols in Wireless Ad Hoc Networks", *Journal Wireless Networks*, 2002, Vol. 8, pp. 671-679.

[23] W. Gibson," An Investigation of the Impact of Routing Protocols on MANETs using Simulation Modeling", *Master Thesis, Auckland University of Technology*, New Zealand, 2008.

[24] X. Li, W. Wei, X. Sun, Y. Gao, "A DSR Protocol Based on the Performance Value", *In Proceedings of the Third International Symposium on Electronic Commerce and Security Workshops(ISECS '10)*, Guangzhou, P. R. China, 29-31, pp.332-335, ISBN 978-952-5726-11-4, July, 2010.

[25] Y. Hu, D.B. Johnson, "Caching Strategies in On-Demand Routing Protocols for Wireless Ad Hoc Networks", *In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom)*, Boston, MA, USA, August 6–11, 2000.

Ragb O. M. Saleh
He has completed his Bachelor's degree in Computer Science from Naser University (Libya) in 1998, and the M.S. degree from Al Al Bayt University at Mafraq (Jordan) in 2007. He is currently working towards his Doctorate degree at University Malaysia Terengganu. His research interests are in networking, including wireless ad-hoc netwrks.

Md Yazid Mohd Saman
He is a Professor of Computer Science in the field of Parallel and Distributed Systems. He has been in the academia for over 20 years, starting in UPM Serdang as a Tutor and Lecturer, then in KUSTEM since 2001, and until now in UMT. He has published over journal and conference papers in the areas of parallel & distributed systems, multimedia technologies and simulation. His research interests are in Distributed, Parallel Computing, Computer Networks, Simulation and Performance Modelling.

M Nordin A Rahman
He is working as Associate Prof., and Dean of Faculty of Informatics, Universiti Sultan ZainalAbidin, Malaysia. He has more than 15 years of teaching and haspublished numbers of papers in referred international journals.

(*) ***The pseudo code of the updated DSR with our methods (FDSR / FSDSR)***

Pseudo-Code of the proposed methods (FSDSR and FDSR)

```
Input: S: the source; D: the destination;
       SR_D: Set of SR for a D, RC=Set of SR_D.
Output: Freshest Route to D (FR)
Initialization: T= n sec; FR = Ø;

Begin
//when S wants to send a data to D
Seek _RC():
{
If   (∄ (SR∈RC & SR∈D)) Do
 Route_Discovery(S->D):
 { //steeps at source node
  Send_R.RQ:
  { //current node sends
   //a R.RQ to its neighbours
    if(current_node.adrs!=Dest.Adrs)Do
     { if it has route to D :
          R.RP.Flag=0;
          send_RRP(R.RP);
      }
     Else
     {
        R.RP.Flag=1;
        send_RRP(R.RP);
     }
  Receive_R.RP:
  For i=1 to T  Do
   // S waits T sec for R.RPs.
   while (∃(R.RP)) Do
    If (node=( R.RP→D))
    { //if the node is D
     SR_new=R.RP→SR;
     Stop(R.RP);//don't forward R.RP
     Call R.Selection(D,SR_new);
  // call a route selection function
  // (FDSR or FSDSR) to return SR
    }
   else  //the node isn't D
   {
     Ins_RC(R.RP→SR);
     SR_new=R.RP→SR;
        Relay(R.RP);
      }
   }; //end_while
}; //end of route discovery phase.
//Using the SR to send data
If (∃(FR∈ RC & FR∈ D)) then
  Send_Data(FR);
 else
  Call Route_Discovery(D);
End if ;
```

FDSR Algorithm

```
// Update the route cache and find
// the Freshest path from S to D.
FDSR():
{
Foreach(SR_new∈D & SR_new∈ SR_D)  Do
{ FR= SR_new ;
for (SR_D=SR_1 to SR_n ) Do
{ if((FR->S_Time)< (SR_D->S_Time))
                  && (FR ->Flag=1))
  FR=SR_D ;
}
Return (FR);
}; // end FDSR function.
```

FSDSR Algorithm

```
FSDSR():
{
Foreach(SR_new∈D & SR_new∈ SR_D)  Do
{ FR= SR_new ;
for (SR_D=SR_1 to SR_n ) Do
{ if((SR_D->Hop_count)< (FR->
Hop_count))&& (FR ->Flag=1))
  FR=SR_D ;
}
Return (FR);
}; // end FDSR function.
```