Journal of Computer Science, 9 (5): 551-557, 2013
ISSN 1549-3636

doi:10.3844/jcssp.2013.551.557 Published Online 9 (5) 2013 (http://www.thescipub.com/jcs.toc)

A New Policy for Path Selection in Dynamic Source Routing Protocol for Mobile Ad-Hoc Networks

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

[1]Department of Computer Science, Faculty of Sciences and Technology,
University Malaysia Terengganu, Terengganu, Malaysia
[2]Department of Computer Science, Faculty of Informatics, Universiti Sultan Zainal Abidin, Terengganu, Malaysia

Received 2012-12-29, Revised 2013-05-16; Accepted 2013-05-17

ABSTRACT

In standard DSR protocol, nodes automatically discover and maintain the routes in the network by storing source routes, discovered dynamically on-demand. They maintain route caches that contain the source routes which the node is aware of. Entries are continually updated when new routes are learned and reorder the cached routes according to the traditional shortest path policy. The problem with this approach is that, while the source is still using the primary shortest route, the primary route might fail and the source would remain unaware of that its cache contains a recent/fresh route, which has the same number of hop count and to the same destination. Thus, we proposed a method to improve the performance of DSR protocol by increasing packet delivery ratio and reducing routing overhead by adopting a path selection mechanism. In this study, we present a new strategy for route selection in DSR protocol that uses the recent-short path as routing metric. It calls RSRS. Basically, RSRS depends on the construction time and the hop-count of the cached route for route selection. Since, it utilizes the recent -short path in the route cache as routing metric. The proposed method has been implemented in GloMoSim simulator and its performance compared to the conventional DSR protocol and we have reached more important and comparable simulation results. The simulation results confirmed that the SRSR method improved packet delivery ratio, reduction of routing overhead, number of dropped packets and number of failure links.

Keywords: GloMoSim Simulator, DSR Protocol, Routing Protocols, Mobile Ad-Hoc Networks

1. INTRODUCTION

Mobile Ad-hoc Networks (MANETs) are unlike the traditional wireless networks that need a costly infrastructure to support mobility. MANET is completely infrastructure-less and does not need the expensive wired infrastructure (Labiod, 2010). Each node is connected to other nodes through wireless links. Nodes act as routers by forwarding data with each other to complete the data transmission process. Because of the limited transmission range of mobile nodes, more than one hop may be needed to send and receive data. In order to facilitate communication within the network, several classes of routing protocols have been proposed. They can be classified into two categories (Alotaibi and Mukherjee, 2012): reactive and proactive routing protocols. Some of the existing proactive routing protocols (Soujanya and Selvam, 2011) are: FSR, ZPR and WPR. Reactive routing protocols are also called "On-demand" routing protocols. Some of the existing reactive routing protocols (Taneja and Kush, 2010) are: AODV and TORA.

In this study, we present a new strategy for route selection in DSR protocol. It calls SRSR. For selecting a source route from the source node to the destination node, it utilizes the recent-short path in the cache as

Corresponding Author: Ragb O.M. Saleh, Department of Computer Science, Faculty of Sciences and Technology, University Malaysia Terengganu, Terengganu, Malaysia

routing metric. Basically, SRSR depends on the construction time and the hop-count of the cached route. SRSR estimates the best source route in the cache according to two processes: firstly, the node reorders its cached routes as soon as a new route is discovered, secondly, select the shortest recent route.

The rest of study is organized in five sections as followed. The current section, the Introduction, gives a brief description in relation to MANETs and their routing protocols. The next section, Materials and Methods, presents the weakness of DSR route selection and describes the proposed method. Then, the Results-section provide the simulation results of the proposed method and the standard DSR protocol. After that, the Discussion-section, discusses and evaluate the observed results of the study. The last section, Conclusion, concludes the study.

1.1. DSR Routing Protocol

The DSR protocol is a reactive routing protocol for MANETs. It is designed for multi-hop wireless Ad-Hoc networks. DSR allows the network to be completely self-organizing and self-configuring. DSR is based on the concept of source routing. Moreover, it applies on-demand schemes for both route discovery and route maintenance. The DSR protocol has two major techniques, which work together to allow the discovery and maintenance of source routes in the MANET. When a source node in the MANET requires sending information to the destination and there is no path in its cache; it applies a route discovery process. Since; it starts by flooding the network with Route Request (R.RQ) messages. Each intermediate node receiving an R.RQ rebroadcasts it again, unless it is the destination node, or it has a path to the intended destination. Route Replay (R.RP) message is sent back to the source node either by destination or an intermediate node that has a path to the intended destination. While in the transmission session, if the selected path is broken, the source is notified using a Route Error (R.ER) message. When the source node receives the R.ER message, it will remove all paths that contain the failure link from the cache. In such a situation, the node that discovered the failure link can utilize other paths to the intended destination to salvaging the data packets (if it has an alternative path). This is called Route Maintenance process (Johnson *et al*., 2003).

DSR protocol uses the route cache to store the routes that have discovered via the source for possible future use. In addition, it is feasible to cache each overheard route without causing loops. The route cache itself possesses the multi-path capability by allowing the storage of more than one route to a destination. This comes in very practical when a route fails. However, DSR protocol needs a technique to determine the relative freshness among cached routes in the route cache. During the route discovery process, DSR uses the route selection based on the shortest path algorithm. It selects a route having the minimal number of hops between the available routes. Furthermore, it is one of the main techniques to reduce the number of route discovery processes to select more than a stable path per the destination. However, the shortest path is always complicated to meet the need of the changing topology of the MANET. As a result, it may raise the number of the unreliable route; therefore, the route which having the minimal number of hops does not frequently mean the optimal route (Li *et al*., 2010).

1.2. Related Work

A new technique for route selection proposed in (Mamoun, 2011a). It named NTRS. It applies to find the most stable route for minimizing the network overhead. NTRS utilizes the received signal strength to calculate the stability of the route. However, an indirect measurement of the distance has used for represent the fragility of the route. When couples of sequential nodes in a route are unconnected via a distance similar to the communication range, the link between those nodes is expected to be broken shortly, which caused due to the high sensitivity of the link to the mobility. A path may become a failure immediately when one of its links is broken. The fragility of a path is dependent on the more unstable links. As result,NTRS demonstrates that the estimation of the remaining path lifetime could benefit in the performance of the MANET if this information is considered in the path selection strategy.

Method proposed in (Mamoun, 2011b) called Route Selection Technique (RST). The method has proposed for optimizing the efficiency of the route discovery process in DSR protocol. RST method aims to reduce the number of route request packets. RST is based on a decision algorithm; it weighs individual links as a route to the intended destination node, which is being constructed if this link is considered suitable by the fuzzy logic system. This route fuzzy logic system also generates Route Reply (R.RP) packets just for selected routes and not for all available source routes as is common with DSR.

In general, the route selection algorithm depends on the route cache to select the source route. Therefore, we will introduce some of the previous works for solving route cache problems in DSR.

Work in (Husieen *et al*., 2011) has presented different approaches of the route cache in DSR protocol and explains the drawbacks of the earlier methods for updating the cached routes of DSR protocol. As a case in point, with high mobility environments and high load traffic MANET broken routes will be cached in the route cache for future use, which is serious issue in DSR protocol. As result, these cached broken routes can cause increase packet loss, long delay and reduce the efficiency of the performance of DSR protocol. As well, this study demonstrates that an efficient method for updating the cached routes in the cache of DSR protocol is desired.

Work in (Chen *et al*., 2010) has proposed Tiding Active Packets (TAP) mechanism to improve the Active Packet Method (He *et al*., 2002). TAP is composed of the Active Packets and Route Error (RERR) flooding to update the stale routes in the cache of DSR protocol. As stated in (Johnson *et al*., 2001), TAP has three cases: first case is the Topology Collection, in which; Active Packet makes two visits to all the nodes. Firstly, Active Packet travels to each node to collect the information of the neighbour nodes. The second visit, through which, all the nodes get all the topology information from Active Packet and use it in the route discovery process. Second case is the Path Calculation; it represents a node which estimates its route from the source to the destination early before it needs to send a data packet. Third case is the Topology Maintenance, in which a node has responsibility to success the transmission of data packet to the next hop depends on a local information cache. TAP mechanism has tested with low mobility 0 m/s to 1 m/s only and low load traffic network. However, with high mobility environments and high load network traffic, a route cache may contain stale routes that can affect the performance of DSR protocol.

2. MATERIALS AND METHODS

2.1. DSR Route Selection Weaknesses

In standard DSR protocol, nodes automatically discover and maintain the routes in the network by storing source routes, discovered dynamically on-demand. They maintain route caches that contain the source routes which the node is aware of. In the route cache, entries are continually updated when new routes are learned and reorder the cached routes according to shortest path policy. In addition, the node caches the route when overhear it or forward a packet for future use. It can store multiple routes and check its route cache for a suitable route before initiating a new route discovery process. Cached routes prevent the flooding early, also reducing the overhead and delay. Despite those advantages, route selection will not be an efficient method without an effective caching strategy. Our concern is that routes which maintain in caches might be into disorder. For instance, if the route cache contains multiple routes per destination and have the same number of hop counts. These cached routes might be disorder despite equality of the number of hop counts and their destination. In this case, the source node uses the shortest path policy to select the source route to the destination from its route cache, regardless of the time of construction the route. The problem with this approach is that, while the source is still using the primary shortest route, the primary route might fail and the source would remain unaware of that its cache contains a recent/fresh route, which has the same number of hop count and to the same destination. To solve this problem, we present a new strategy for route selection in DSR protocol by choosing the recent-short route between the source and destination in the route cache.

2.2. The Proposed Route Selection Scheme

The key of enhancement in our approach is that the performance of DSR protocol can be achieved by selecting a stable route. We call it as The Recent-Short Route Selection (RSRS) method. RSRS tries to select the recent-short route to the destination based on construction time of the source route. Where RSRS allows to MANET's node to reorder its route cache as soon as a new route has learned; the reordering will do according to the recent-short route policy. In addition, RSRS presents some advantages; nodes can save its resources (i.e., bandwidth and power consumption) by reducing recall the route discovery process, which is costly. Other advantages can be achieved by using this approach is some performance objectives such as high delivery ratio, low overhead and fewer dropped packets. As the response to solve route selection problems in DSR protocol, we introduce a new route selection strategy that utilized two different concepts; Freshness and Shortness of the source route to select an efficient source route.

2.3. SRSR Strategy

SRSR is a method to help DSR protocol to organize the route cache and select an efficient source route from the source to the destination.

We assume that the source node has more than a Source Route (SR) in its route cache.

SR has three operators:

- The construction time of SR
- Number of hop-count of SR
- The source of SR (a destination/an intermediate node)

We classify the SR to three types:

- SR_D: (one or more SR reply by the destination node)
- SR_{MHC}: (one or more SR have the minimum number of hop-count)
- SR_{RCT}: (a SR has the recent construct time)

According to previous assumptions, SRSR extracts the Recent Source Route (RSR) to the destination as follows:

$$RSR = SR_{RCT}\left(SR_{MHC}\left(SR_D\left(SR_n\right)\right)\right)$$

where, SR_n is set of SR per destination.

Put the RSR on the top of cached source routes for future use.

Essentially, the principle of RSRS strategy is finding out the best source route which has: the recent construction time and minimum hop-count of the cached routes, which sent back to the source node.

Basically, RSRS has three phases during routing process, which will be described as the following.

Phase 1:

- If the source node wants to send data packets
- If the node's cache has one or more routes; it selects the recent-short route in its route cache according to RSRS strategy
- If the node's cache has not a route to the destination; call the route discovery process, where the source node propagates route request packet (R.REQ) and wait for route reply packet (R.REP)

Phase 2:

- If the source node receives a new R.REP
- Reorder its cached routes as soon as a new route has received according to RSRS strategy
- Stop the propagating of R.REP

Phase 3:

- If an intermediate node receives a new R.REP
- Reorder its cached routes as soon as a new route has received according to RSRS strategy.
- Forward the R.REP to next intend node

2.4. Pseudo-Code of the Proposed Algorithm

we present a pseudo-code to describe the details of RSRS method. "**Table 1**" lists the common variables and functions that are used in the pseudo-code.

Table 1. Common pseudo-code variables and functions

Variable	Description
S	The source node which wants to send data packets to a specific destination node
D	The destination node of data
SR	The source route which selected by the source node to send data packets to a specific destination.
SRD	Set of source routes have same hop-count and same destination
RC	Rote cache (it is provided to store source routes)
RSR	The recent-short route between two nodes.
T	A specific time for route discovery process
Check_RC()	Function to find a specific route in the node's cache.
RSRS()	Function to select the recent-short route to the destination
R.REP	A route reply packet which replied to the source route via an intended destination or an intermediate node
R.REP→SR	The source route which come back with route reply packet
R.REP→D	The intended destination for route replay
Crnt_node	The current node which received the route reply packet
Send_RREQ()	Function uses to send route request packets for the destination node
Send_Data()	Function uses the selected source route to send data packets to the destination
Insrt_RC()	Function to save the new source route and its construct time in the route cache
SR_t	The construct time of the source route
SR_{new}	A new source route
Forward ()	Function to forward R.REP to the next intended node
Free ()	Function to stop forwarding the R.REP packets
Reorder_RC()	Function to insert the new source route in the suitable order in the cache according to The recent-short Route First policy
Crnt_node_adr	The current node address
Intnd_nod_adr	The intend intermediate node address of R.REP
n	Number of SR in RC

In this updated DSR, nodes call RSRS as soon as receive or overhear a new source route during simulation time. RSRS tries to select the recent-short route to the intend destination using routing information such as the number of hop counts and the construction time of the source route.

Employ of RSRS will be illustrated via an abbreviated pseudo-code as the following.

2.5. Pseudo-Code of RSRS Algorithm

```
Input: S: Source node; D: destination node;
SRD: Set of source routes for same
Destination, RC = Set of DRD.
Output: The Recent-Short Route (RSR)
Initialization: T = N sec; RSR = ϕ;
Begin
//when a source wants to send a data
Check_RC():
Begin
If ( ∄ (SR∈RC and SR∈ D)) Do
Run Route_Discovery (D):
Begin //process at source node
Send_ R REQ(D);
For t = 1 T Do
//waiting T sec for R.REPs to D.
while (∃(R.REP)) Do
If (Cmt_node = (R.REP→D))Do
{//if the current node is D
SR_new = R.REP→SR;
Free (R.REP)://don't forward R.REP
Call RSRS (D, SR_new, SRt); //return RSR
}
Else //if the current node isn't D
{
insert RC(R.REP→SR);
SR_new = R.REP→SR;
Call R.STS (D, SR_new, SRt);
Forward (R.REP);
}
End While;
End;//end of route discovery.
//Using the selected RSR to send data
If (∃(RSR∈RC and RSR∈ D)) Do
Send_Data(RSR);
else
Recall Route_Discovery (D);
End if;
//Update the route cache and retum the recent-short Route
//Insert the new SR in the suitable order in the route cache and
//reorder cached routes according to the recent-short route first.
RSRS():
Begin
Foreach (SR_new∈ D and SR_new∈ SRD) Do
Insert_RC(R.REP→SR);
Reorder_RC(SR_new, SRt);
Retum (RSR);
End;//end RSRS.
```

3. RESULTS

3.1. Simulation

A simulator called GloMoSim (Global Mobile Information System Simulator [http://pcl.cs.ucla.edu/projects/glomosim/]) has been employed to simulate our proposed model. We also evaluate the performance of the proposed DSR scheme and compare its performance with the standard DSR.

3.2. Simulation Environment

"**Table 2**" lists the common parameters that are used in the simulation.

3.3. Performance Metrics

The following metrics have been used to evaluate the performance (Li *et al*., 2010; Hassan *et al*., 2010):

- Packet Delivery Ratio: the average rate of successful data packets received over a communication channel
- Routing overhead: the total number of routing packets transmitted during the simulation time
- Dropped packets: the data packets that are dropped during the link breaks and collision
- Broken Links: The total numbers of links that break down during transmit data packets

3.4. The Simulation Results

In this study, we report the results of the simulation experiments for the standard DSR and the improved DSR (DSR_RSRS). The simulation has been averaged for five sessions of the runs (each data point represents an average of five runs).

Figure 1 shows the Packet Delivery Ratio for the data packets by varying the pause time. **Figure 2** presents the Routing Overhead in the network for for various pause time. **Figure 3** shows the number of broken links in the network by varying the pause time. **Figure 4** presents the number of dropped packets in the network for various pause time.

Table 2. Simulation parameters

Parameter	Value
Simulation-time	900 sec
Terrain-dimensions	2200×600m
Number-of-nodes	50 mobile nodes
Mobility model	Random Way-point Model
Pause time	0, 300,600 and 900 sec
Bandwidth	2Mbps
Mac-protocol	IEEE 802.11
Network-protocol	TCP - UDP
Routing-protocol	DSR
Data traffic - CBR	4 packets/sec
Packet size	512 bytes
Nodes speed	0 to 10 m/s

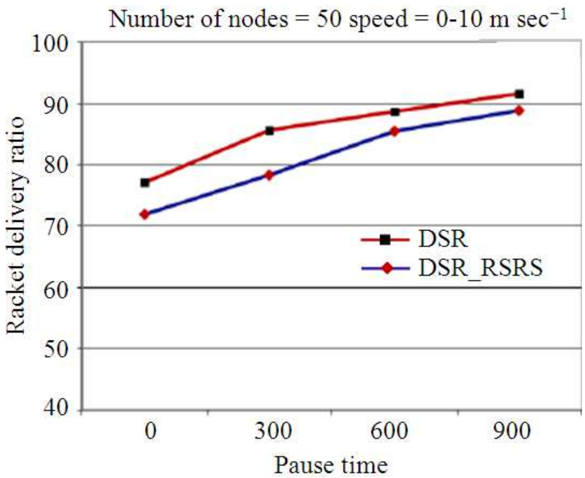

Fig. 1. The packet delivery ratio

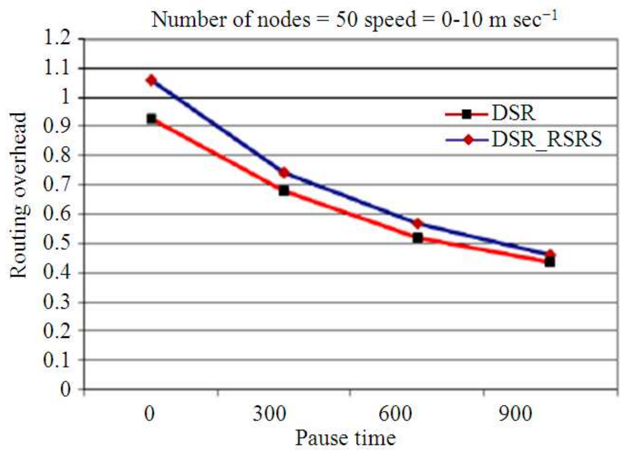

Fig. 2. The routing overhead

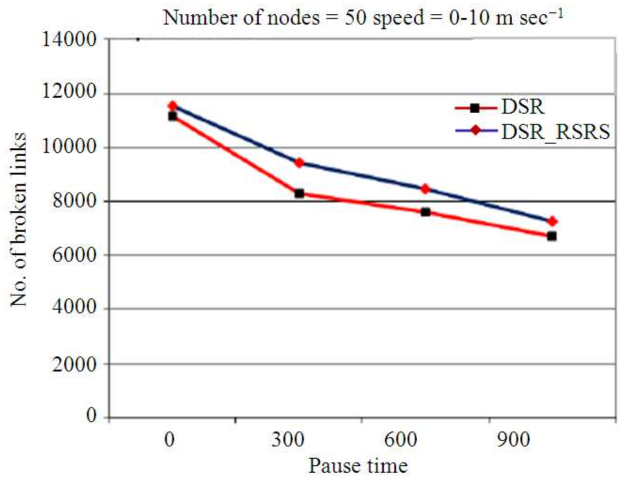

Fig. 3. The number of broken links

4. DISCUSSION

Figure 1 graph shows the packet delivery ratio with respect to the pause time. The packet delivery ratio for both schemes reduces when the mobile nodes move more rapidly because high mobility is more prone to link failures, which may force the protocol to select a broken route. In both low and high mobility environment, DSR_RSRS introduces more packet delivery ratios compared to standard DSR. This is due to the successful selection to the recent-short route to the intend destination.

Figure 2 graph shows the routing overhead with respect to the pause time. The routing overhead for both schemes; increases when the mobile nodes move more rapidly because high mobility is more prone to link failures, which may force the protocol to invoke new route discovery processes; subsequently, cause more and more overhead. In both low and high mobility environment DSR_RSRS introduces less routing overhead compared to standard DSR. The reduced overhead due to the successful selection to the recent-short route to the intended destination. In this manner, those nodes may do not need to recall a new route discovery for more than a needed route.

Figure 3 graph shows the number of broken links with respect to the pause time. The number of broken links increases when nodes move more rapidly because high mobility is more prone to link failures. In both low and high mobility; DSR_RSRS introduces fewer broken links compared to standard DSR. The reduced broken links is due to the successful route selection method during routing process.

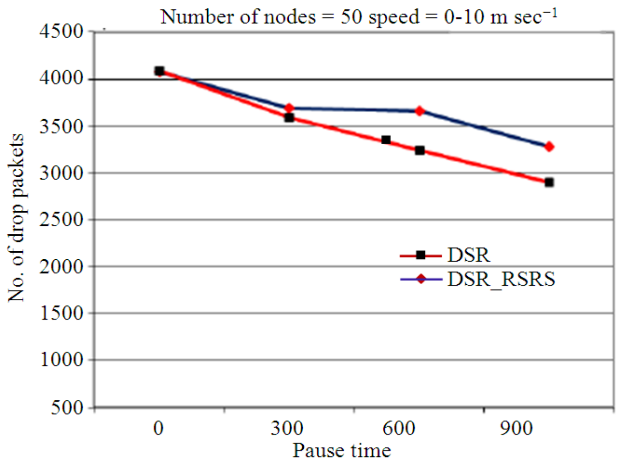

Fig. 4. Number of dropped packets

Figure 4 graph shows the dropped packet ratio with respect to the pause time. The dropped packet ratio increases when the mobile nodes move more rapidly because high mobility is more prone to link failures; furthermore, when the node fails to select a suitable route to send its data. In both low and high mobility environment DSR_RSRS introduces less dropped packet ratio compared to standard DSR. The reduced dropped packet ratio is due to the successful route selection process.

5. CONCLUSION

In this study, we have discussed an improved DSR routing protocol for ad-hoc networks. We have added the proposed route selection method (RSRS) to DSR protocol. It developed to select the recent-short route from the source to the destination. The performance of RSRS method for DSR Protocol has been evaluated and compared with standard DSR using detailed simulations. Several common performance metrics are considered. The simulation results show that the improved DSR performs well; it can overall generate lower overhead; higher delivery ratio; fewer dropped packets and fewer broken links.

6. REFERENCES

Chen, J., Y. Tang, D. Fu and H. Chang, 2010. On the improving strategies upon the route cache of DSR in MANET. Proceedings of the 7th International Conference on Ubiquitous Intelligence and Computing, Oct. 26-29, Springer-Verlag Berlin, China, pp: 445-458. DOI: 10.1007/978-3-642-16355-5_35

Alotaibi, E. and B. Mukherjee, 2012. A survey on routing algorithms for wireless Ad-Hoc and mesh networks. Comput. Netw., 56: 940-965. DOI: 10.1016/j.comnet.2011.10.011

Hassan, Y.K. M.H.A. El-Aziz and A.S.A. El-Radi, 2010. Performance evaluation of mobility speed over MANET routing protocols. Int. J. Netw. Sec., 11: 128-138.

He, Y., S. Berson, C.S. Raghavendra and B. Braden, 2002. Active packets improve dynamic source routing for ad-hoc networks. University of Southern California.

Husieen, N.A., O.B Ghazali, S. Hassan and M.M. Kadhum, 2011. Route cache update mechanisms in dsr protocol-a survey. Proceedings of International Conference on Information and Network Technology, (INT' 11), IPCSIT Press, Singapore, pp: 136-141.

Johnson, D.B., D.A. Maltz and J. Broch, 2001. DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. In: Ad Hoc Networking, Perkins, C.E., (Eds.), Addison-Wesley, Boston, ISBN-10: 0201309769, pp: 139-172.

Johnson, D.B., D.A. Maltz and Y.C. Hu, 2003. DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks. Carnegie Mellon University.

Labiod, H., 2010. Wireless Ad Hoc and Sensor Networks. 1st Edn., John Wiley and Sons, London, ISBN-10:0470393483, pp: 352.

Li, X., K. Wei, X. Sun and Y. Gao, 2010. A DSR protocol based on the performance value. Proceedings of the 3rd International Symposium on Electronic Commerce and Security Workshops, Jul. 29-31, Guangzhou, P.R. China, pp: 332-335.

Mamoun, M.H., 2011a. A novel technique for the route selection in DSR routing protocol. Int. J. Video Image Proc. Netw. Secu., 11: 1-4.

Mamoun, M.H., 2011b. A proposed route selection technique in DSR routing protocol for MANET. Int. J. Eng. Technol., 11: 10-14.

Soujanya, A.J. and P.A. Selvam, 2011. Study of routing protocols in mobile ad hoc networks. Int. J. Eng. Sci. Technol.

Taneja, S. and A. Kush, 2010. A survey of routing protocols in mobile ad hoc networks. Int. J. Innovat., Manage. Technol., 1: 279-285.