SARP: Self Acknowledge Routing protocol for Route maintenance in Mobile Ad Hoc Network

Ravindra Eklarker, Vinayadatt. V. Kohir, V. D. Mytri

Abstract— Mobile Ad hoc Networks (MANET) are characterized by multi hop wireless connectivity, infrastructure less environment and frequently changing topology. Wireless links in MANET are highly error prone and can go down frequently due to mobility of nodes. Most reactive routing protocols do not maintain routing information at the node, on link failure typically a route error message is sent to the source and source initiates another route discovery which creates a lot of routing overhead. We present a Self Acknowledge Routing Protocol (SARP) for route maintenance in mobility, where each intermediate node runs a Self Acknowledge (SA) algorithm to recover the broken link to reach the destination efficiently. Main intends of the protocol is to minimize new route discovery process through route repair and minimize the network overhead. The protocol is implemented over Dynamic Source Routing Protocol (DSR) and the experiment results shows reduction in control overhead and end-to-end delay with an improvement of the packet delivery ratio.

Keywords: DSR, MANET, multi hop, Routing Protocols, Route Maintenance, SARP, SA

---

1 INTRODUCTION

Mobile Ad hoc Network (MANET) is self-organized and multi-hop networks connected in dynamic manner. Nodes form a temporary/short-lived network without any fixed infrastructure where all nodes are free to move about arbitrarily. Since the nodes can move freely, then the node mobility pattern became an important characteristics of the MANET. One of the challenging work in MANET is link failure due to mobility. MANET nodes are equipped with wireless transmitters and receivers which at a given time depending on the nodes positions and their transmitter and receiver coverage patterns and transmission power levels, a wireless connectivity in the form of a random, multi hop graph or Ad hoc network exist between the nodes. This Ad hoc topology may change with time as the nodes move or change their transmission and reception parameters [1].

A multihop wireless network path is composed of a number of intermediate mobile nodes and wireless links connecting them. Since nodes can move at any time, wireless links are prone to be broken. Any link break along an established routing path will lead to a path failure. A shortest path may fail sooner than another path connecting a given source and destination pair. Frequent routing discovery is costly and inefficient. Moreover, shortest path routing cannot support much quality of service connection requests when the path duration is a requirement. In such a dynamic changing network, it is critical to route the packets to destinations effectively without generating excessive overhead. This presents a challenging issue for protocol design since the protocol must adapt to frequently changing network topologies in a way that is transparent to the end user.

- *Ravindra .Eklarker is currwently pursuing Ph.D in Electronics and Communication Engineering at PDACE Gulbarga in VTU belgaum, INDIA. Ph: 09448336042. E-mail reklarker@gmail.com.*
- *Vinaya Dutta .V.Kohir is currently working as professor in Electronics and Communication Engineering Department in PDACE Gulbarga. India E-mail: vvkohir@yahoo.com.*
- *V.D Mythri is currently working as Principal in Shetty Institute of Engineering and technology in Gulbarga India,. E-mail: vdmytri@yahoo.com*

Effective delivery of data packets while minimizing route maintenance overhead is crucial in ad hoc networks. Because nodes in a MANET acts as a routers for any ongoing packet communication and have limited transmission ranges, the communication links are broken and packet losses occur. This problem is amplified when a route constitutes several such links. If any of those links fails, the route breaks, which initiates series of undesirable events and outcomes. If how long a link can be operational is predicted, the routing maintenance protocol can be used to its advantage to minimize the overhead. In this paper, we present an efficient route maintenance protocol SARP for MANETs. Intend of the protocol is to handle node mobility and repair the broken link using SA Algorithm to improvise the network life time, and these optimizations can be utilized to reduce the routing overhead.To evaluate the protocol we extend the DSR routing protocol, which implements mobility prediction and self acknowledge algorithm.

The rest of the paper is organized as follows. Section 2 presents the related work on routing and route. maintenance. In section 3, we present the proposed protocol description and mechanism, Section 4 describes the experiment and results and section 5 describes the conclusion of the paper

2 RELATED WORKS

Several routing protocols have been proposed for MANETs, which differ in the approach used for discovering a new route and maintaining a known route when nodes move. In the proactive routing protocols such as DSDV [6], FSR [7], WRP [8], CGSR [9] and GSR [10] protocols compared to the on-demand routing protocols such as DSR [13, 14], AODV [15], TORA [16] and ABR [17], a constant propagation of routing information is involved, which incurs substantial routing related traffic.

Moreover such protocols require each mobile node to maintain routes to each possible target in the MANET, which most likely exceeds the requirements of any node and thus the routing overhead expended in establishing such not required routes is wasted. Since bandwidth is a scarce resource in MANETs, these limitations imposed by proactive routing protocols make them less attractive as compared to on-demand routing protocols in a bandwidth constrained MANET environment. ZRP [20] presents a hybrid behavior of proactive and reactive routing schemes which has advantages and disadvantages of both the schemes, depending on the value chosen for zone radius parameter used to limit the scope of proactive scheme.

Ramesh et al. [2] have studied the problem of link breakage prediction in the DSR routing protocol. Their idea is that during the route discovery process, the source node builds two routes which are the source route and another route can be used as a backup. The backup route can be used if the primary route (source route) was predicted to have a link breakage soon.

Zhu [4] has studied the problem of link breakage prediction by using the same equation that have been proposed by Qin & Kunz [3] which is the link breakage prediction algorithm, but she has implemented this algorithm using the AODV and MAODV routing protocols.

Choi et al. [5] has dealt with the problem of link breakage prediction in vehicular ad hoc network. They proposed an algorithm to predict a link breakage possibility using the value of the RSSI (Received Signal Strength Indicator).

Qin & Kunz [3] have dealt with the problem of link failure prediction by proposing an equation to calculate the exact time that a link breakage can occur. They named their method the link breakage prediction algorithm.

Lee et al. [11] studied three optimizations to AODV. These optimizations include an expanding ring search for route discoveries, a query localization protocol to prevent the flooding of route requests, and a local repair of link breaks on active routes.

In this paper, we present an efficient route maintenance protocol SARP for MANETs. Intend of the protocol is to handle node mobility and repair the broken link using SA Algorithm to improvise the network life time, and these optimizations can be utilized to reduce the routing overhead.

3 SELF ACKNOWLEDGE ROUTING PROTOCOL (SARP)

In routing protocols a routes are maintained at the node sequence of the whole paths, i.e., from the source to the destination. This approach works well for small networks where routes are typically short. When the network grows and mobility increases the node links break more very frequently and the maintenance overhead on the network will be very high due to the repetition of new routes discovery process.

The on-demand routing protocols suggested for MANETs basically make use of broadcast based methods for route discovery. They differ in their routing packet formats, data structures maintained by each node, various optimizations [3][11] applied in route discovery and in their approach for route maintaining. In a flooding broadcast based method, when an source node wants to send data packets to a destination node, and it does not have a valid route to this target node, it broadcasts a route request packet to its neighbors. These neighbors forward the route request packet to their neighbors and this process goes on until either the target node or an intermediate node with a valid route to target node is located. Each node receiving a particular route request packet broadcasts it only once to its neighbors, and it discards the subsequent receptions of the same route request packet, to minimize routing overhead. The drawback of Flooding Method is that it floods the entire network with the route requests even when the target node is just a few hops away from the originator node.

When a route breaks due to node mobility or node dies routing protocols like DSR and AODV typically discard the whole original route and initiate another new route discovery to establish new routes from source to the destination. In general, when a route breaks, usually only a few node links are broke, but others nodes links are still remain intact. This approach of reactive protocols wastes the valuable information of the original routes and may cause high routing maintenance and overhead over the network.

SARP propose an approach to utilize the original route and neighbouring information [18] to effectively handling the link failure and repairing the broken link based on the prediction of node life time (NLT) and link life time, and makes aware the source and neighbor node of leaving and joining with a self acknowledging mechanism, which minimize the routing overhead and improvise the communication throughput.

Like other on-demand routing protocols, we address three issues in SARP: route discovery, data forwarding and route maintenance. It defines four types of control messages are: route request (RREQ), route reply (RREP), route error (RERR), and self acknowledge (SACK). SACK message will be send by forwarding node to acknowledge source node to discard the node from its routes in case of all the links to the neighbor node are below LLT threshold limit.

3.1 Route Discovery

3.1.1 Route Request

Source node initiates route request when it requires a new route to a destination node and it broadcasts an RREQ message. A RREQ can be uniquely identified by the combination of the source address and the source's broadcast ID number. When an intermediate node receives the RREQ, it first determines whether this request is a duplicate by looking up its request seen table. Duplicate requests are discarded. If the RREQ is new and its TTL is greater than zero, the intermediate node appends its address to the path recorded in the RREQ and rebroadcasts the request to its neighbors. When the RREQ message reaches the destination, the destination node replies to the request as described next.

3.1.2 Route Reply

In DSR, for one route discovery process by a source address and a broadcast id, there is no limit on the number of RREPs that the destination can send to the source. The destination

ISSN 2229-5518

will send RREP whenever it receives a RREQ. We observe in our simulations that most of the paths discovered by RREQ messages share a large number of common nodes and only the last few hops are different. Based on this observation, we take top five routes from the source route table by the parameter MAX_ROUTES (which is set to 5 in our simulations).

The RREP is unicast to the source along the reverse path contained in the RREP message. Every node in the network maintains two tables for route information as *Route Cache Table (RCT)* and *Forwarding Neighbor Table (FW_NT)*. Route cache table is used for SARP data routing and FW_Neighbor Table is used for route repair on link failure.

An entry in the route cache table uses (*[destination]: [cache_ routes]*) structure as in the DSR protocol. The route cache stores cache routes from the current node to destination nodes, and FW_Neighbor table uses (*[destination]: [neighbor_nodes]*) structure. When an intermediate node receives the RREP it updates both its route cache and its FW_neighbor table.

For example, suppose the path from a source *A* to Destination *B* is as shown in figure-1.

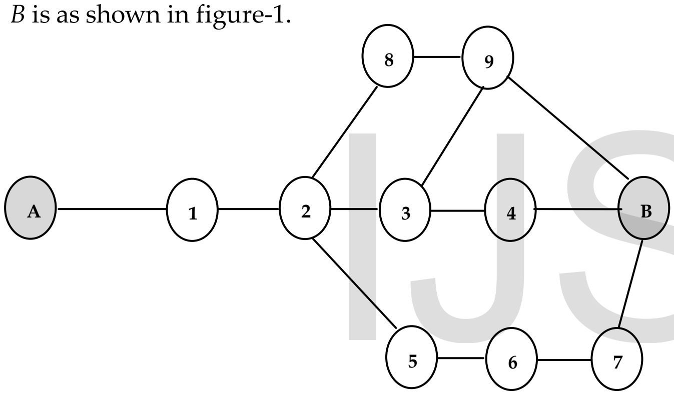

Figure-1 Routes from Source A to Destination B

On route reply from *B* to *A* intermediate node-2 receive 3 replies from *8,3* and *5* nodes. As per figure-1, to update the route cache node-2 insert route the destination as *([B]: [3,4]), ([B]: [5,6,7]), ([B]: [8,9])* and *([B]: [3,9]),* and to update the FW_neighbor table node-2 insert as *([B]: [3,8,5])* i.e., from which are the nodes it receive RREP. When the RREP reaches the source node that initiated the route discovery, the source also updates both its route cache and its FW_neighbor table as described above. Here multiple route replies are stored for finding all possible neighbors to the intermediate nodes. But for data communication only top five sort route will be used for efficient resource and overhead maintenance. Now, the source is ready to transmit data packets.

3.2 Data Forwarding

To transmit a data packet, the source node first sets the source route option in the data packet for routing. The source gets a source route to the destination from its route cache and inserts the source route into the header of the data packet, as in the DSR protocol. After setting the source route option for the data packet the source node gets the next hop to the source node of the first cache route by looking up its routing cache table and sends the packet to the next hop, which, typically, is a forwarding node. Intermediate nodes get the next hop from their route cache table and send the data packet to that next hop, this operation continue until the data packet reaches the destination.

3.3 Route Maintenance

Due to the mobility nature of MANETs, links on a route may fail. Route maintenance is the mechanism to handle link breaks. A node can confirm if a packet is correctly received by the forwarding node through any of the three types of acknowledgments: link-level, passive (listening to the forwarding by next hop node), and network-layer.
In reactive routing protocols, when a link failure occurs, typically a route error message is sent to the source and the old route is discarded and source again starts another route discovery process. Route maintenance in this manner creates a lot of route request overhead. On the other hand the advantage of the SARP is that it implements self acknowledge algorithm which calculating link life time and acknowledge self for the link life to the current next hop. A pre-prediction of the link failure makes node to find alternate route using FW_neighbor tables to reach the destination. Thus, a broken link can be repaired locally, at the level of the broken by finding an alternate path to the destination using FW_neighbor table.

3.3.1 Self Acknowledge Algorithm

Link failures are uncontrolled if the nodes are mobile, and get more severe when the mobility of nodes increases and leads to high packet loss considerably. In order to reduce packet loss due to link failures, mobility needs to be integrated in routing protocols. This integration needs to be independent of routing protocols to have an effective solution. The algorithm is based on an efficient use of the duration of connectivity known as link life time (LLT) between two neighboring nodes in a route.
The algorithm has been designed to treat mobility related problem in mobile adhoc network. As mobility increases, LLT between nodes decreases. This causes the routes to break quickly, and packet losses due to route breakages. The algorithm calculates the node Link Life Time (LLT)

A. Calculation of Link Life Time (LLT)

The method proposed in [12] for mobility prediction is utilized for the calculation of LLT in our algorithm. The calculation is described as shown in Figure-2. Two mobile nodes *A* and *B* with their radio ranges, *r* and current locations of *A* and *B* are $A(x_{a1},y_{a1})$ and $B(x_{b1},y_{b1})$, respectively. If *A* and *B* are moving with velocities v_a and v_b, and angles θ_a and θ_b, then their new locations are $A(x_{a2},y_{a2})$ and $B(x_{b2},y_{b2})$ after some time duration, *t*. We assume that nodes *A* and *B* are not changing directions within this time duration, ***t***.

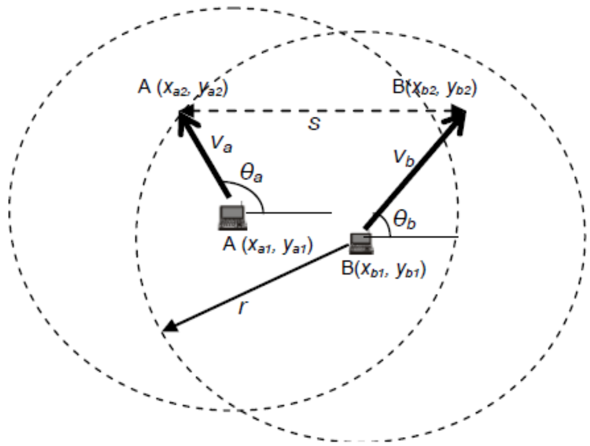

Figure-2: Node movement in MANET

If all the information related to their current locations, such as v_a, θ_a, v_b, θ_b, x_{a1}, y_{a1}, x_{b1} and y_{b1} are known, their future locations can be calculated using the provided information from known values by the following two methods.

$$A(x_{a2},y_{a2}) = f\,(t,\ v_a,\ \theta_a,\ x_{a1},y_{a1}) \qquad (1)$$

$$B(x_{b2},y_{b2}) = f\,(t,\ v_b,\ \theta_b,\ x_{b1},y_{b1}) \qquad (2)$$

If the distance between A and B after time t is s then

$$s^2 = (x_{a2} - x_{b2})^2 + (y_{a2} - y_{b2})^2 \qquad (3)$$

A and B will be able to communicate with each other as long as they will remain within their transmission range, r. So, $t = LLT$ if $s \le r$. After solving (3) with $s \le r$ and considering $t = LLT$, we get

$$LLT = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{(a^2+c^2)} \qquad (4)$$

where,

$$a = v_a \cos\theta_a - v_b \cos\theta_b\,, \quad b = x_{a1} - x_{b1}\,,$$

$$c = v_a \sin\theta_a - v_b \sin\theta_b\,,\ d = y_{a1} - y_{b1}$$

Each Node in the path from source to destination calculates its own LLT with the next hops and compares with a threshold limit of LLT. If the calculated LLT is below threshold limit it initiate self acknowledge mechanism.

B Route Repair through SA Mechanism

When a forwarding node predicts the current next hop LLT is below threshold limit using equation-4, it initiate self acknowledge mechanism to find alternate path to destination using FW_neighbor table. A Self Acknowledge mechanism monitors all neighbor nodes LLT status and update their link status in FW_neighbor table. In case of low LLT status of the current next hop it dynamically chooses next sequence hops in the FW_neighbor table and check the LLT status before forwarding. It will perform this operation till it finds a hop above LLT threshold limit. If it unable to find any hops which LLT is above threshold it sends a SACK message to the source to discard node from routes. Figure-3 provides the data flow description of the mechanism

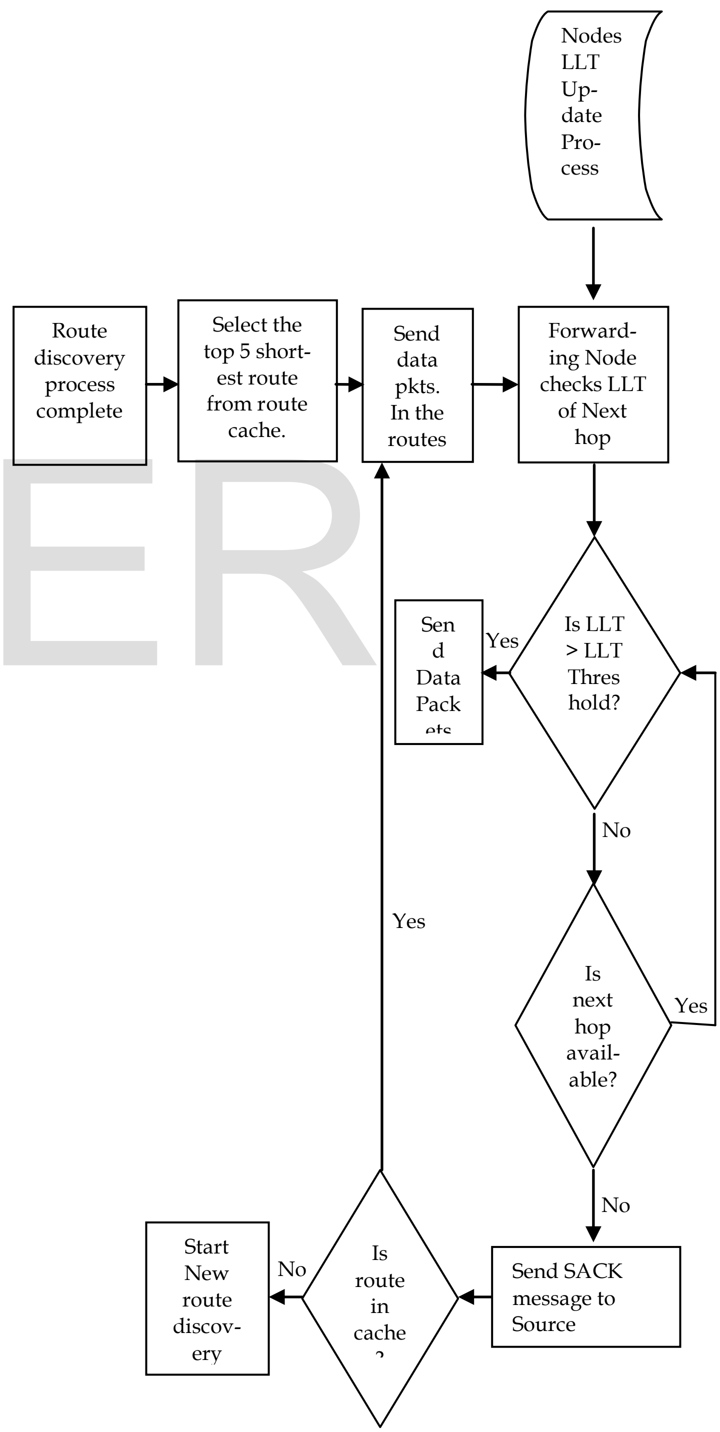

Figure 3: Self Acknowledge Algorithm Data flow Diagram

Suppose as per the above Figure-1 example, the routing path from A to B is *[1,2,3,4]*, and the FW_neighbor node of Node-2 is *[3,8,5]* and we assume, LLT threshold = 10s. Node-2 forwards the data through Node-3, as LLT is above threshold limit.

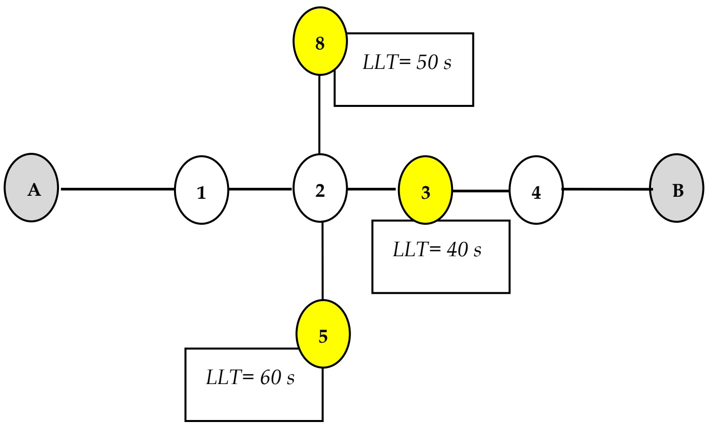

Figure-4: Node-2 neighbor nodes and its LLT

Let us assume due to the mobility nature Node-3 changed its position and its LLT goes below threshold limit as shown in Figure-4. In such case Node-2 dynamically forwards the data through Node-8 as its threshold is above the limit, the selection of Node-8 is due to next in the sequence in FW_neighbor table.

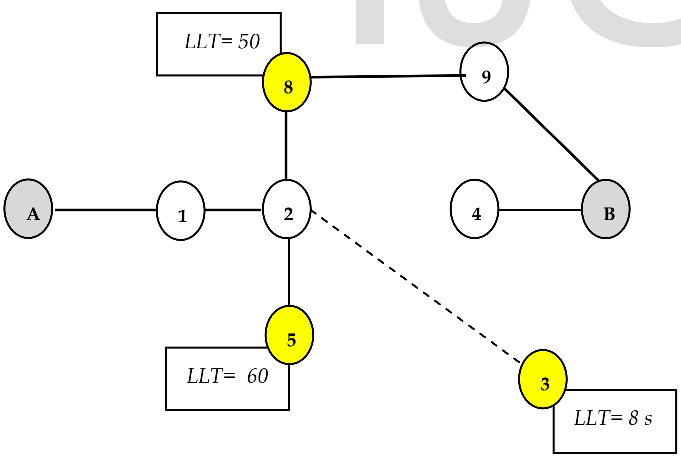

Figure-5: Alternate route to destination B as Node-3 LLT is low.

4. EXPERIMENT AND RESULTS

4.1 Simulation Setup

We implement the simulation using GloMoSim 2.03 Simulator [21]. Glomosim is a scalable network simulation environment for mobile ad-hoc networks, developed at UCLA Parallel Computing Laboratory uses the parallel discrete-event simulation capability provided by PARSEC [22]. It has the capabilities to simulate thousands of mobile nodes without disregarding the details in the lower layer protocols and allow rapid integration of developed protocols.

Extensive simulation has been conducted to evaluate the performance of SARP and compare with DSR. The traffic was constant bit rate (CBR). The source and the destination of each CBR flow were randomly selected but not identical. Each flow did not change its source and destination for the lifetime of a simulation run. Each source transmitted data packets at a rate of four 512-byte data packets per second. The mobility model was random waypoint with the speed ranging from 0 m/s to 10 m/s and a pause time of 30 seconds. We simulate the simulation with the following setup parameters as shown in Table-1.

Configuration	Parameter Values
Simulation Area	1200m X 1200m
No. of Nodes	50
Mobility Speed	0 to 20 m/s
Source-Destination Pairs	15
Packet Size	512 bytes
CBR Rates	4 pkts/sec
Mobility	RWP
Pause Time (sec)	0,30,60,120,300,600

Table-1 Simulation Parameters

We perform the experiment based on the Table-1 simulation parameter for a period of 600 seconds. The simulation runs on Random Way-point movement behavior model where each node placed randomly for a period of pause time and chooses a new location at random and moves with a speeds between 0 to 20 m/s. We run the simulation in six different pause times as mention in Table-1. For routing we have taken 15 source destination pairs at constant bit rate (CBR) flow of 4 data packets per seconds and each packet size is 512 bytes in size.
We conducted six iterations of simulations to evaluate the performance of SARP varying the pause time. In the simulations, we collected data for three metrics, namely, control overhead, packet delivery ratio (PDR) and end-to-end delay.

4.2 Results and Analysis

In this simulation, we studied the performance of SARP when the pause time varied from 0 nodes to 600 seconds. For each performance metric, we compared SARP with DSR and the results are shown below.

Control Overhead: The main purpose of the protocol is to reduce the routing overhead. Minimizing the control packets transmission in the network minimizes the routing overhead. Control overhead calculated based on the total number of control packets originated and forwarded by the protocols during entire communication process. Figure-6 shows the control overhead comparison between SARP and DSR. We observed that SARP has much less control overhead than DSR when the pause time varies. Control overhead in SARP is significantly reduced because of the use of the SA algorithm which enables the local repair of route maintenance activities.

Packet Delivery Ratio: Packet delivery ratio is measured based on the total number of data packets received against the total number of data packets originated. Figure-7 shows the comparison PDRs of SARP and DSR. We observe that SARP throughput is high compare to DSR in varying pause time. SARP maintains routes repairs of broken link locally which usually lasts longer and more data packets are delivered.

End-to-End Delay: End-To-End Delay calculated as the time between the transmissions of a data packet from a source to destination. The Figure-8 shows the comparison of average End-to-End delay of SARP and DSR. We observe that SARP exhibits the lowest end-to-end delay compare to DSR. In SARP, a smaller end-to-end delay because of the route repair mechanism recovers a broken route quickly and data packets do not have to wait for another round of route discovery before they can be transmitted.

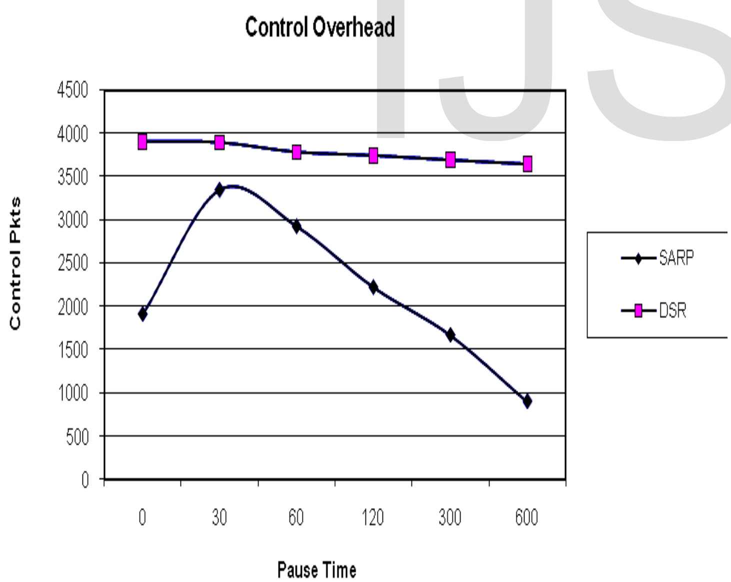

Figure 6. Control Overhead

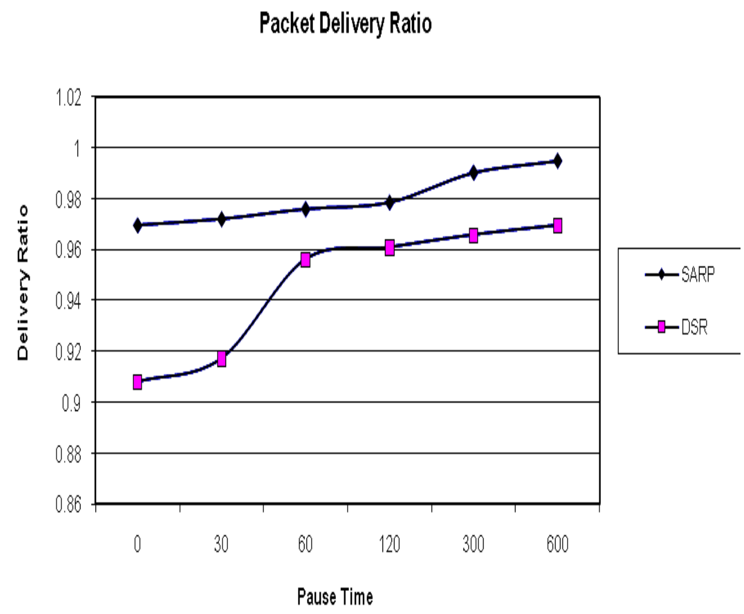

Figure 7. Packet delivery ratio

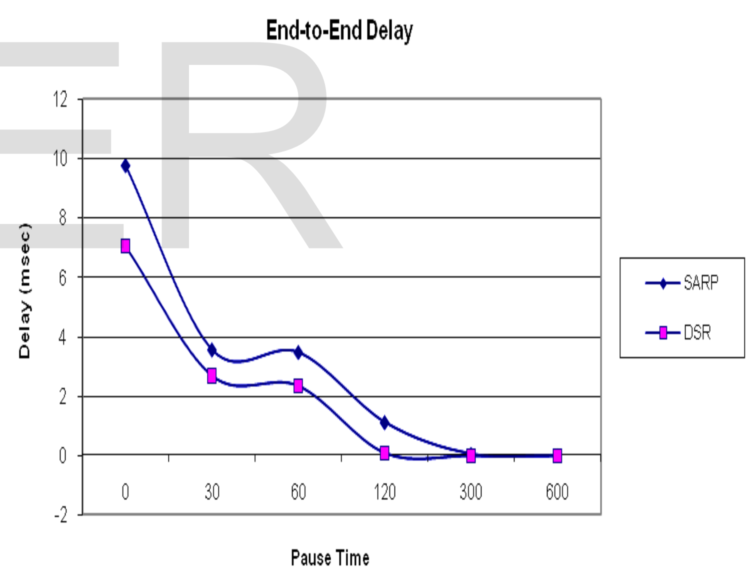

Figure 8. End to End delay

5. CONCLUSION

In this paper, we propose a new route maintenance protocol, Self Acknowledge Routing Protocol (SARP) from Mobile Ad Hoc Network. SARP maintains a route replied nodes during route discovery process in forwarding neighbor table. The proposed changes are made to the DSR route maintenance using SA Algorithm. The algorithm locally does the route maintenance based on Link Life Time Prediction. One distinct advantage of SARP is that when a route is broken because of node mobility or node failure, instead of discarding the whole route and discovering a new route from the source to the destination, the node self find the alternate path route to destination and also repair the broken link. The simulation result shows a clear improvement in packet delivery ratio and minimizing control overhead and end-to-end delay.

Our future works is to evaluate the performance of SARP on other on-demand routing protocols. Link Life Time is only the factor evaluated in SARP for calculating transmission range. We further improvise this protocol calculating Node Life Time and Route Life Time Prediction to meet the high mobility and scalability of the Mobile Ad hoc network.

REFERENCES

[1]. N. Enneya, K. Oudidi, and M. Elkoutbi, “Network Mobility in Ad hoc Networks”, Proc. of the Int. Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, 13-15 May, 2008.

[2]. Ramesh, V., Subbaiah, P., and Supriya, K, Modified DSR (preemptive) to reduce link breakage and routing overhead for MANET using proactive route maintenance (PRM). Global Journal of Computer Science and Technology. Vol. 9. Issue 5, 124-129, 2010

[3]. Qin, L., and Kunz, T, Increasing packet delivery ratio in DSR by link prediction. HICSS 03. IEEE. 2002. Hawaii.

[4]. [4] Zhu, Y., Proactive connection maintenance in AODV and MAODV. Master of Science. Carleton University, Canada., 2002.

[5]. Hoi, W., Nam, J., and Choi, S., Hop state prediction method using distance differential of RSSI on VANET. NCM 2008. IEEE. 426-431.

[6]. Charles E. Perkins and Pravin Bhagwat, .Highly Dynamic Destination- Sequenced Distance-Vector Routing (DSDV) for Mobile Computers., Computer Communications Review, pp. 234-244, October 1994.

[7]. Atsushi Iwata, Ching-Chuan Chiang, Guangyu Pei, Mario Gerla and Tsu-Wei Chen, .Scalable Routing Strategies for Ad Hoc Wireless Networks., IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, pp. 1369-79, August 1999.

[8]. Shree Murthy and J. J. Garcia-Luna-Aceves, .An Efficient Routing Protocol for Wireless Networks., ACM Mobile Networks and Applications Journal, Special Issue on Routing in Mobile Communication Networks, pp. 183-197, October 1996.

[9]. Ching-Chuan Chiang, Hsiao-Kuang Wu, Winston Liu, Mario Gerla, .Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel., Proceedings of IEEE SICON.97, pp. 197-211, April 1997.

[10]. Tsu-Wei Chen and Mario Gerla, .Global State Routing: A New Routing Scheme for Ad-Hoc Wireless Networks., Proceedings of IEEE ICC.1998.

[11]. S.-J. Lee, E.M. Belding-Royer, and C.E. Perkins, “Scalability Study of the Ad Hoc On-Demand
Distance Vector Routing Protocol,” Int’l J. Network Management, vol. 13, no. 2, pp. 97-114, 2003.

[12]. S.J. Lee, W. Su, and M. Gerla, “Wireless Ad hoc Multicast Routing with Mobility Prediction”, Int. Journal of Network Management, Pages 3-30, Vol. 11, Issue 1, Jan–Feb 2001, ISSN:1099-1190.

[13]. David B. Johnson, David A. Maltz, Yih-Chun Ho and Jorjeta G. Jetcheva, .The Dynamic Source Routing Protocol for Mobile Ad-Hoc Networks., March 2001.

[14]. David A. Maltz, Josh Broch, Jorjeta Jetcheva and David B. Johnson, .The Effects of On-Demand Behavior in Routing Protocols for Multi-Hop Wireless Ad Hoc Networks., IEEE Journal on Selected Areas in Communications, Special Issue on Mobile And Wireless Networks, August 1999.

[15]. Charles E. Perkins, Elizabeth M. Royer and Samir R. Das, .Ad Hoc On- Demand Distance Vector (AODV) Routing., IETF Internet Draft draft-ietfmanet- aodv-08.txt, March 2001.

[16]. V. Park and S. Corson, .Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification., July 2001.

[17]. Chai-Keong Toh, .A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing., Proceedings of 1996 IEEE Fifteenth Annual International Phoenix Conference on Computers and Communications, pp. 480-486, March 1996.

[18]. K.A. Rahman, K. Rahman, M. Lott, and K.E. Tepe “Characterization of the Adverse Effect of Neighborhood Capture in MANET and on the Way to a Remedy”, Proc. of IEEE EIT, Windsor, Ontario, Canada, May 2009.

[19]. S. Corson and J. Macker, .Mobile Ad-Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations., IETF RFC 2501, January 1999.

[20]. Zygmunt J. Haas, .A New Routing Protocol for The Reconfigurable Wireless Networks., Proceedings of IEEE International Conference on Universal Personal Communication (ICUPC.97).

[21]. L. Bajaj, M. Takai, R. Ahuja, R. Bagrodia, and M. Gerla, “GloMoSim: A Scalable Network Simulation Environment,“ Technical Report 990027, 1999.

[22]. R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, and H.Y. Song, “Parsec: A Parallel Simulation Environment for Complex Systems,” Computer, vol. 31, no. 10, pp. 77-85, Oct. 1998.