

# Multipath Power Sensitive Routing Protocol for Mobile Ad Hoc Networks

Anand Prabhu Subramanian, A.J. Anto, Janani Vasudevan, and P. Narayanasamy

Department of Computer Science and Engineering  
Anna University, Chennai – 600 025, India.  
{anand\_ps2000, jesusanto, jananivasu}@yahoo.com  
sam@annauniv.edu

**Abstract.** Mobile Ad hoc Networks are characterized by multi-hop wireless links, without any infrastructure, and frequent host mobility. A plethora of routing protocols has been proposed. A class of routing protocols called *on-demand* protocols has recently gained attention because of their efficiency and low routing overhead. As the mobile nodes in the network work on low power batteries, the need to take into account their *power consumption* arises. This paper focuses on a particular on-demand routing protocol, called *Dynamic Source Routing*, and shows how an efficient heuristic based Multipath technique can improve the mean time to node failure and maintain the variance in the power of all the nodes as low as possible. In the *Multipath Power Sensitive Routing Protocol* (MPSR) every node in the network is treated equally and the overall network is stable for a long time. An interesting feature of using this protocol is that the end-to-end packet delay does not increase significantly. The results of extensive simulation show that the performance of *MPSR* protocol is on an increasing trend as mobility increases when compared to the *Dynamic Source Routing*.

## 1 Introduction

Mobile hosts and wireless networking hardware are becoming widely available, and extensive work has been done recently in integrating these elements into traditional networks. However, mobile users may want to communicate in situations in which no fixed wired infrastructure is available, because either it may not be economically practical or physically possible to provide the necessary infrastructure or because the expediency of the situation does not permit its installation. A mobile ad hoc network is an autonomous system of mobile hosts connected by wireless links. There is no static infrastructure such as base stations. If two hosts are not within radio range, all message communication between them must pass through one or more intermediate hosts that act as routers. The hosts are free to move around randomly, thus changing the network topology dynamically.

The design of an efficient routing protocol is a major challenge in such dynamic wireless networks. A lot of work has been done in this area right from the seventies, when the U.S. Defense Research Agency, DARPA supported the PRNET (Packet Radio Network) [7]. Routing protocols must deal with the typical limitations of these

networks that include high power consumption, low bandwidth and high error rates. These routing protocols may generally be categorized as *table driven* and *source initiated on demand driven*.

On-demand routing is the most popular approach in ad hoc networks. Instead of periodically exchanging route messages to maintain a permanent route table of the full topology, on-demand routing protocols build routes only when a node needs to send data packets to a destination. Most proposed protocols of this type (for example, Dynamic Source Routing (DSR) [3] and Ad hoc On-demand Distance Vector (AODV) [2]) however, use a single route for each session.

Multiple paths can be useful in improving the effective bandwidth of communication, responding to congestion and heavy traffic, and increasing delivery reliability. Multipath routing protocols in wired networks has been widely developed in [9], [6], [4], and [19]. These protocols use table-driven algorithms (link state or distance vector) to compute multiple routes. Studies show however, that proactive protocols perform poorly in mobile networks because of excessive routing overhead [8], [13]. Multipath routing in ad hoc networks has been proposed in [16], [1], [21], and [10]. Although these protocols build multiple routes on demand they are not much concerned with the power of each node in the route. Providing multiple routes is beneficial in network communications, particularly in mobile wireless networks where routes become obsolete frequently because of mobility and poor wireless link quality.

We approach to find more than one efficient path between a source and destination to mask link failures in the network. This requires three components: A route discovery mechanism, a mechanism for sending packets along the selected route and a high level protocol for selecting the most reliable set of routes from the many paths that may exist in the route cache of the source node. The first two of these, discovery and forwarding mechanisms, are relatively well-understood. In order to accomplish the third issue, we use a heuristic to find which of the potentially efficient routes in the cache the routing layer should use to achieve high network stability and maintain the variance in power of the nodes to a minimum.

This paper presents Multipath Power Sensitive Routing (MPSR) Protocol that builds multiple routes between the source and the destination nodes and uses a heuristic to switch between routes such that the burden of routing is distributed evenly to all the nodes in the network. The authors believe that every node in the network must be treated equally for the stability of the network. This paper aims to achieve this stability by maintaining the variance of the remaining power of each node as low as possible so that the mean time to failure of the nodes increases.

The remainder of this paper is organized as follows. In Section 2 we survey the applied work on Multipath routing. Section 3 briefly describes the DSR protocol and the circumstances in which its performance is not satisfactory. Section 4 describes the proposed Multipath Power Sensitive Routing (MPSR) Protocol in detail. Performance evaluation by extensive simulation is described in Section 5. Section 6 describes the future work intended to be performed. Section 7 concludes the paper.

## 2 Related Work

Past work on multipath routing protocols has mainly focused on quick failure recovery, finding disjoint paths between source and destination. Here we approach to use a heuristic (a combined metric using the shortest path and the average remaining power of the nodes in the route) to effectively select multiple paths.

In On-demand multipath routing for mobile adhoc networks [1] the route requests that are replied to are those that carry a source route that is link wise disjoint from the primary source route. When the primary route breaks, the shortest remaining alternate route is used. It shows that providing all intermediate nodes in the primary (shortest) route with alternative paths has a significantly better performance than providing only the source with alternate paths.

The AODV-BR: Backup routing [16] algorithm discovers more than one route in order to replace a broken one with one of the backup routes. It relies on variants of the on-demand routing protocol, AODV, to discover multiple routes. The goal is to improve the packet delivery ratio and the average delay per packet by falling back to an operational backup route when the primary route breaks.

Split Multipath Routing with Maximally Disjoint Paths [17] approaches to use both primary and backup paths simultaneously to route data. Such a multipath routing approach can better distribute load, resulting in significant decreases in packet loss and, in the case that packets are dispersed across the path set with increased fault tolerance. [17] has examined how to establish two maximally disjoint paths and select routes on a per-packet basis. These protocols do not address the issue of path selection and are limited to route replies provided by the routing protocol, and [1] does not provide a specific method for selecting the maximally disjoint path.

The Path Set Selection [14] deals with finding redundant paths that are disjoint in nature. This is to ensure that the correlation between the failures of the paths is as small as possible. The Alternate Path Routing [12] selects the two routes with the least number of hops after decomposing the route replies into constituent links. Further more this protocol does not provide a metric to justify route selection scheme.

Here we take care that the route replies are sent to the source node in such a way that the efficient routes reach the source in order of their efficiency. We define the efficiency of the routes by using the average remaining power of the nodes in the route and hop count so that the end-to-end delay does not increase significantly. The forwarding function chooses the currently available efficient route and sends packets through this route.

## 3 Dynamic Source Routing Protocol

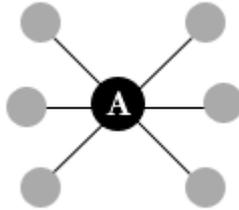
In the Dynamic Source Routing (DSR) Protocol, a process known as *Route Discovery* establishes route between a source and a destination node. Here the source floods a route request packet and as the route request packet moves towards the destination node, the route gets built. When the route request packet reaches the destination it sends the route reply packet to the source. Any intermediate node knowing the route to the destination can also send a route reply. After knowing the route the source adds the route to the route cache it maintains and also to each data packet that it wants to

send to that particular destination. The route discovery process is done on-demand. Even though many route requests reach the destination it sends the reply packet corresponding to the shortest route.

The link break is taken care by the *Route Maintenance* process, which sends a route error packet to the source saying that a particular link is not valid. The source then initiates a route request if it has more data to send to that destination.

### 3.1 Circumstances in Which Performance of DSR Is Not Satisfactory

As DSR uses the shortest path as the routing criteria, it may overload some of the nodes that occur in more number of shortest paths. Consider the situation as in Fig 1.



**Fig. 1.** A network illustrating the situation in which shortest path routing performs poorly

Consider node A that is present in more than one shortest route of different source destination pairs. As it is overloaded to carry many of the data packets, the power of this node decreases more rapidly than other nodes. If node A fails due to power loss, then all the routes that pass through node A become invalid which results in a series of route discovery process. This results in a significant routing overhead in the network and increases the traffic. The aim of this paper is to reduce this type of overhead by associating a weight to each node and making the routing decision based on the weight as well as the shortest path so that it does not significantly affect the end-to-end delay.

Link breaks can be due to mobility or node failure resulting from power loss. As DSR caches only the shortest path, it needs to initiate a route request once a cached route becomes invalid. As mobility increases or when the power of the nodes in the route decreases node failures occur more frequently and the route request packets adds significantly to the routing overhead. When the source nodes cache multiple routes, it can switch over to another cached route so that the time for route discovery is eliminated and network overhead due to route request is reduced. It also ensures that the multiple routes cached in the route cache are not stale by maintaining a time out value for each entry.

These facts motivated us towards a heuristic based multipath power sensitive routing protocol that performs well in the above-mentioned volatile circumstances.

## 4 Multipath Power Sensitive Routing Protocol

Multipath Power Sensitive Routing Protocol (MPSR) is an on-demand source routing protocol that facilitates the source node to accept multiple route replies. The source can cache these multiple routes and switch between them based on the network conditions. The routing function constructs the routing table based on the weight (remaining power) of the nodes in the network and the forwarding function selects the routes based on a heuristic that aims at reducing the power consumption of the nodes in the route. Here we assume a non-hostile environment in which the nodes relay their original weights. A counter called *route count* (*rCount*) is maintained. This specifies the maximum number of routes that can be cached for a particular destination. *rCount* can preferably have values such as 3, 4, 5 etc based on our simulations. These values were chosen considering the number of nodes in the network and the diameter of the network.

The value of *rCount* is known to all the nodes and is used in the route discovery process. Any node can have at most *rCount* routes for a particular destination. Whenever a source node needs to send a data packet to a destination node it searches its route cache for a route to that particular destination. If the route does not exist, it initiates a route discovery process.

In the route discovery process the source node floods the *route request* packet. The weight of the nodes in the route is added along with their address. The weight of the nodes is its current *remaining power*. As each node relays the *route request* it appends its address and weight. When the *route request* reaches the destination node it adds its address and weight and sends the *route reply* packet. Any intermediate node that has a route to the destination can also send the route reply to the source. Figure 2 shows the route with address and remaining power.

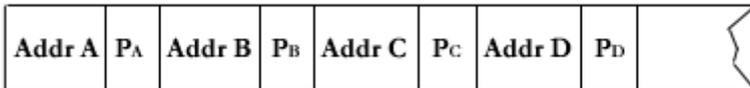


Fig. 2. Address and Power of the nodes in the route request/reply packets

In the destination node the average power of each path is calculated. Let *N* be the hop count of a path and *n* be the average power of the path. According to the MPSR routing criteria, the source chooses the path with *high* average power and *low* hop count. So when the destination sends the route reply packet it waits for a time that is equivalent to,

$$(N / n) * t \tag{1}$$

where *t* is a random amount of time. Using this we can find a route with low hop count and high average power though not with both minimum hop count and maximum average power. This metric is used to get a compromise between the hop count and the average remaining power. This ensures that the more efficient paths reach the source before the source gets *rCount* routes so that the source caches only to

a maximum of  $rCount$  efficient routes. The intermediate node that sends the route reply also follows this strategy. This strategy serves for two purposes:

- The destination sends the route with low hop and high average power before any other routes to the source.
- To prevent the traffic around the source node due to excessive route replies. As all the nodes other than the source are potential candidates to send the route reply, there may be lots of route reply reaching the source node.

The route cache in each node is modified such that it can cache more than one route for a destination. The cache is maintained as a priority queue with the most efficient path in the front of the queue. When a new route reaches the source, it checks the number of routes with its  $rCount$ . If it can cache more routes it adds this route according to the priority queue in the correct position. Thus the insertion of route takes about  $O(rCount)$  time. As the value of  $rCount$  is of the order of 3, 4, 5 etc this does not affect the computational time for route insertion. As the most efficient route is available in the front of the priority queue the route selection can be done in  $\Theta(1)$ . We maintain a time out interval for each cache entry so as to eliminate stale routes. In our simulations we used the Timer- Based Route Expiry as described in [11], which is a dynamic mechanism that allows each node to choose timeout values independently based on its observed route stability. Moving on to forwarding, the route selection is based on two strategies.

#### 4.1 Min Power Strategy

For each valid route in the cache for a particular destination, the remaining power of each node in the route is known. Let  $minPower$  be the power of the node with minimum remaining power in the route. A parameter called the threshold ( $t_h$ ) is defined as the value of the safe lower bound of the remaining power. Each node must have the remaining power greater than  $t_h$  to function properly. Another parameter  $diff$  is defined as follows:

$$diff = minpower - t_h \quad (2)$$

The power discharge pattern of alkaline batteries is linear [18] while the lithium ion batteries have a precipitous discharge in battery life which is quadratic in nature [15]. Considering this power discharge pattern of batteries in the nodes and the value of  $diff$ , the numbers of packets that can be safely transmitted through this route are calculated based on this criterion. Consider this value to be  $N_p$ . Here  $N_p = diff / power$  for transmitting each packet based on the discharge function of the battery in the node.

#### 4.2 Round Robin Strategy

The source node has a count of the total number of data packets to be sent to a particular destination. Using *Round Robin* scheme, let the number of packets that can be sent through each of the routes be  $N_r$ . Let  $N_t$  be the total number of data packets to

a destination and *numRoutes* be the current number of routes to that destination present in the route cache. Then,

$$N_r = N_t / \text{numRoutes} \quad (3)$$

Using the Round Robin strategy, we equally distribute the packets among the routes in the cache.

If the number of packets determined by the round robin strategy ( $N_r$ ) is greater than the number of packets that can be transmitted based on the Min Power Strategy ( $N_r > N_p$ ), then it is not possible to adopt round robin strategy. So the number of packets that can be transmitted through a route is  $\min(N_p, N_r)$ .

Consider the situation in which  $N_p > N_r$ : In this case if the min power strategy alone is followed, then it might result in draining the power of the node to the minimum threshold so that the mean time to failure of the node decreases and the probability of link failure increases. So the combined strategy works well in the above-mentioned situations.

In the route maintenance process, a link break is detected and a route error packet is transmitted to the source analogous to the DSR protocol. The routes with the advertised link are deleted from the route cache.

Thus the Multipath Power Sensitive Routing (MPSR) Protocol maintains the variance of the power among the nodes in the network to be as low as possible thereby contributing to the longer lifetime of the network.

## 5 Performance Evaluation

We used a detailed simulation study to evaluate the effectiveness of the Multipath Power Sensitive Routing Protocol described in the last section. The performance of the MPSR was compared to the base DSR protocol. In the following sub-section, we first describe the simulation environment and the performance metrics used, and then present and analyze the simulation results.

### 5.1 Simulation Environment

The detailed simulation was done using the Global Mobile Simulator (*GloMoSim*) Library [20]. The network was modeled with 50 mobile hosts placed *randomly* within a terrain area of dimension 2000m x 2000m area. The radio model to transmit and receive packets is RADIO-ACCNOISE which is the standard radio model used. The packet reception model is SNR-BOUNDED where a node receives the signal without error if the *Signal to Noise Ratio* (SNR) is more than a specified threshold value. The radio transmission power is set to 15.0 dBm and the sensitivity of the radio is set to -91.0 dBm. Each node has a channel capacity was 2 Mb/s.

The IEEE 802.11 Distributed Coordination Function (DCF) [5] was used as the medium access control protocol. The *random waypoint* model was adopted as the mobility model. In the random waypoint model, a node randomly selects a destination from the physical terrain. It moves in the direction of the destination in a speed uniformly chosen between a minimum and maximum speed specified. After it reaches

its destination, the node stays there for a time period specified as the pause time. The minimum and the maximum speed were set constant to zero and 10 m/s, respectively. The various mobility scenarios were modeled by using different pause times. If the pause time increases mobility decreases and vice versa.

## 5.2 Performance Metrics

The metrics used for performance evaluation were: (i) *Standard deviation of the Routing load* which is the deviation in the number of control packets (*route request, route reply, and route error*). (ii) *Packet delivery ratio* — the ratio obtained by dividing the number of data packets correctly received by the destination by the number of data packets originated by the source. (iii) *Standard Deviation of remaining power* of the nodes in the network. (iv) *Average end-to-end delay* of data packets - this includes all possible delays caused by buffering during route discovery, queuing delay at the interface, retransmission delays at the MAC, propagation and transfer times; (v) *Number of packet drops*— the total number packets dropped that includes data as well as control packets. (vi) *Average hop count*-the arithmetic mean of the hop counts of all the routes present in the route cache of all nodes.

## 5.3 Simulation Results

The measurement of all the above-mentioned parameters was performed once for each of the different mobility scenarios.

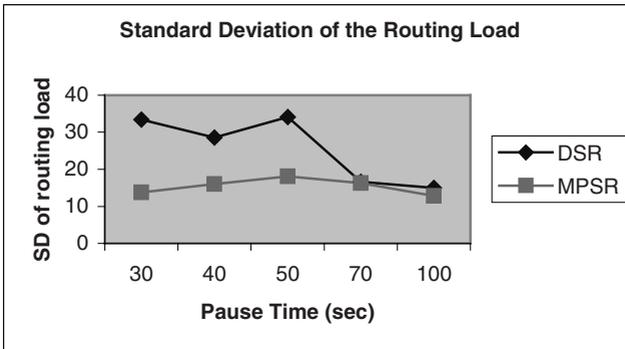


Fig. 3. Plot of the Standard Deviation of the Routing Load with respect to mobility.

Fig. 3 shows the standard deviation (SD) of the routing load which included the control packets viz., route request, route reply and route error. The SD of the routing load in MPSR is observed to be low compared to DSR especially when the mobility is more (when the pause time is less). This is because with high mobility, link breaks are common and DSR needs to initiate new route request and this adds significantly to the routing overhead. Also when the power of the node decreases and when links break

due to link failure, the route error packets are more in DSR. If a link break occurs due to increased mobility, DSR has to initiate a route discovery process. But in MPSR, the probability of all the routes becoming invalid is less and hence the source node can choose another possible route in its route cache to the intended destination. This justifies that there will not be any additional overhead in MPSR when the mobility increases. The low SD of MPSR shows that every node in the network has almost equal routing overhead thus contributing to maintain the SD in the power consumed by the nodes. This increases the mean time to failure of the nodes and the stability of the network.

Fig. 4 shows the packet delivery ratio of each protocol with respect to mobility. The fact that MPSR outperforms DSR is visually obvious, especially when the mobility is higher (i.e., the pause time decreases).

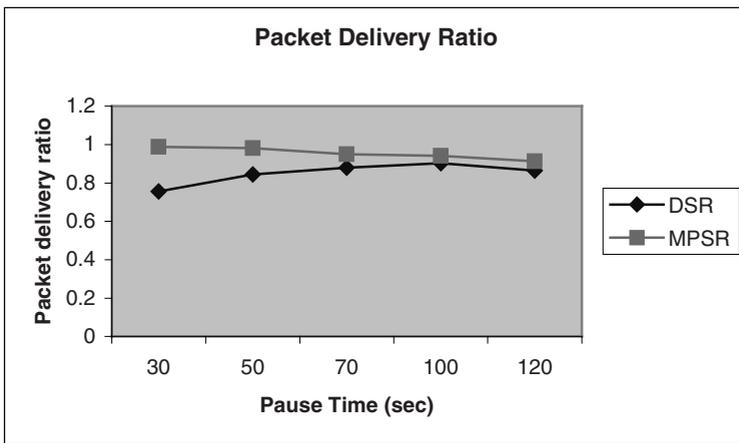


Fig. 4. Plot of the Average Packet Delivery Ratio of nodes with respect to mobility.

In DSR, only one route is used for each session and when that route is invalidated it has no route to that destination. In that case, it sends a RREQ to discover a new route. DSR however, does not apply any aging mechanism for cached route entries, and hence routes stored in the cache (either by the source or the intermediate nodes) may be stale. After a route break, source nodes will use these newly acquired but obsolete routes only to learn that they are also invalid, and will attempt another route discovery. Many data packets are dropped during this process and more delay is needed to discover correct routes.

In MPSR though we maintain multiple routes, a time out interval takes care that the routes are not stale. The performance of MPSR is significant at high mobility.

Fig. 5 shows how the standard deviation in power is lower in MPSR as compared to DSR. The low variance in the control packets is one cause for the low variance in power consumed. Also if the power of a node reaches the lower bound, MPSR routes the packet through other possible safe routes. This ensures that the power consumed by each node is equal. This increases the stability of the network.

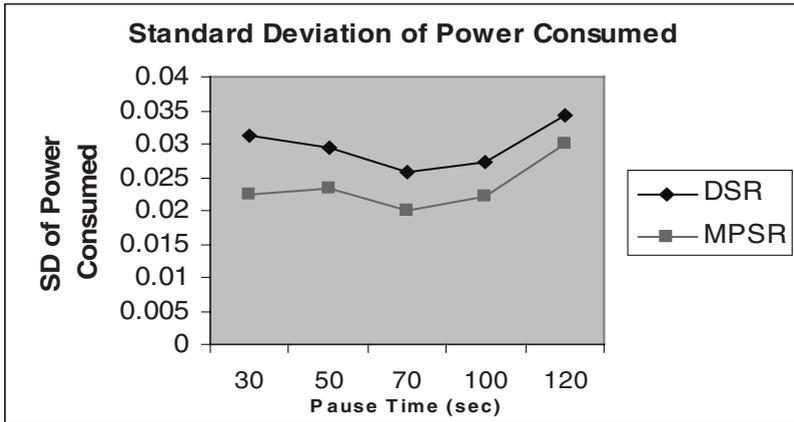


Fig. 5. Plot of the Standard Deviation of the Power Consumed with respect to mobility

The curves for end-to-end delay were not plotted because there was no significant difference in average packet delay between DSR and MPSR. This result was surprising because we had expected a slight worsening in delay for packets (in the MPSR case) as they get routed around nodes with high cost (or high remaining power). This was the result of consideration of the minimum hop ( $N$ ) and the maximum average power ( $n$ ) as described in section 3. On closer examination of the simulation trace it was found that some packets did indeed take longer routes and of these some did have higher delay (measured in time steps). However the number of these packets was not large and as a result did not contribute to a statistically significant result. In more congested situations, MPSR performed better than DSR. In addition, DSR yields longer delays reconstructing routes and the period of time the data packets are buffered at the source node during route discovery results in larger end-to-end delays. MPSR on the other hand, uses the remaining valid routes when one of the multiple routes is disconnected, and hence no route acquisition latency is required. So, overall, we conclude that packet delay is unaffected when using MPSR.

Fig. 6 illustrates the number of packets dropped by each protocol. Both data and control packets are measured. The reasons for packet drops can be incorrect route information, mobility, and node failure due to power loss, collisions, and congestion. DSR cannot maintain precise routes and drops more packets as nodes move more often (i.e., less pause time). The usage of stale routes from caches is the major reason of DSR packet drops. MPSR has considerably fewer packet drops compared to DSR. This is because MPSR invokes fewer route discovery processes and consequently, transmits less control.

Fig. 7 reports the average hop count of each protocol. DSR has the shortest hop distance when there is no mobility because MPSR may sometime choose route with longer distance than the shortest route as it considers the remaining power of the

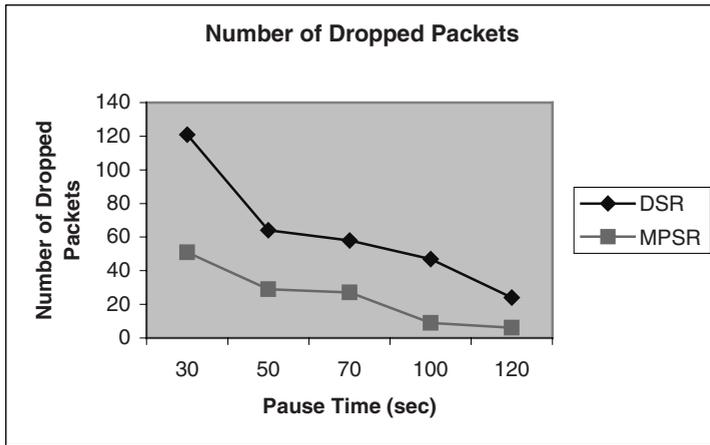


Fig. 6. Plot of the Average number of Dropped Packets with respect to mobility

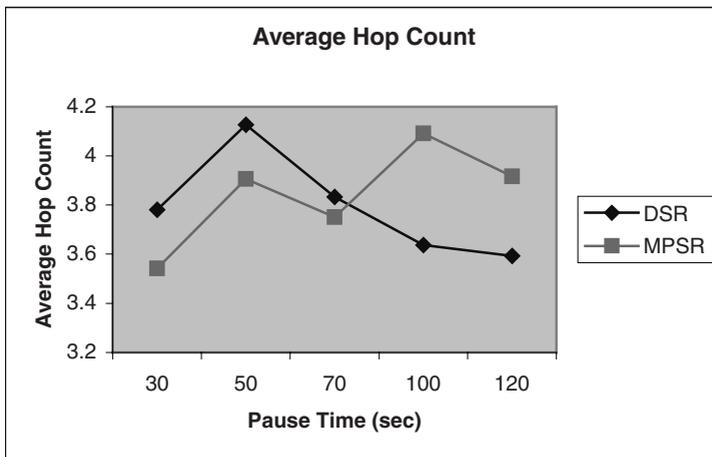


Fig. 7. Plot of the Average Hop Count with respect to mobility

nodes in the path into account. With mobility however, the hop count of DSR grows and becomes larger than those of MPSR protocol. If the route is established directly from the destination, it can be the shortest route since it is built based on the most recent information and accounts for node locations after movements. DSR, however, uses cached routes from intermediate nodes. As there is no timeout interval for the cached routes they may not be fresh and do not exploit the current network topology. DSR therefore builds longer routes than the MPSR protocol. Longer paths have more chance of having route breaks since one-link disconnection results in route invalidation.

## 6 Future Work

The protocol presented assumes a non-hostile working environment in which the mobile nodes advertise their true weights. We are working on the ways of adapting this protocol in the presence of malicious nodes that advertise their weights lower than what they are, to avoid traffic to pass through them.

## 7 Conclusion

In this paper, the Multipath Power Sensitive Routing (MPSR) Protocol for Mobile Ad hoc Networks has been presented. MPSR is an on-demand source routing protocol that establishes multiple routes between a source destination pair and switches between the routes based on a heuristic. This heuristic takes the current network conditions into consideration. The remaining power of each node and the hop count are taken as the routing criteria and care is taken such that routing burden is equally distributed over all nodes in the network. This increases the mean time to failure of the nodes and eventually results in the stability of the network. Providing multiple paths is useful in ad hoc networks because when one of the routes is disconnected, the source can simply use other available routes without performing the route discovery process again.

Extensive simulations, which were performed to evaluate the performance of the MPSR and DSR protocols, indicate that MPSR outperforms DSR because multiple routes provide robustness to mobility. The performance difference becomes evident as the degree of mobility increases. MPSR had considerably fewer packet drops compared to DSR. As the packets are transmitted through the route selected based on a combined strategy (*Min Power* and *Round Robin*) we distribute the load evenly to all the nodes and the network is stable for a longer time.

## References

1. Nasipuri and S.R. Das, "On-Demand Multipath Routing for Mobile Ad Hoc Networks," Proceedings of IEEE ICCCN'99, Boston, MA, Oct. 1999, pp. 64–70.
2. C.E. Perkins and E.M. Royer, "Ad-Hoc On Demand Distance Vector Routing," Proceedings of IEEE WMCSA'99, New Orleans, LA, Feb. 1999, pp. 90–100.
3. D.B. Johnson and D.A. Maltz, "Dynamic Source Routing in Ad Hoc Wire-less Networks," In Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Chapter 5, Kluwer Academic Publishers, 1996, pp. 153–181.
4. D. Sidhu, R. Nair, and S. Abdallah, "Finding Disjoint Paths in Networks," Proceedings of ACM SIGCOMM'91, Zurich, Switzerland, Sep. 1991, pp.43–51.
5. IEEE Computer Society LAN MAN Standards Committee, Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY) Specification, IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.
6. Cidon, R. Rom, and Y. Shavitt, "Analysis of Multi-Path Routing," IEEE/ACM Transactions on Networking, vol. 7, no. 6, Dec. 1999, pp. 885–896.
7. John Jubin and Janet D. Tornow, "The DARPA packet radio network protocols," Proceedings of the IEEE, 75(1):21–32, January 1987.

8. J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," Proceedings of ACM/IEEE MOBICOM'98, Dallas, TX, Oct. 1998, pp. 85–97.
9. J. Chen, P. Druschel, and D. Subramanian, "An Efficient Multipath Forwarding Method," Proceedings of IEEE INFOCOM'98, San Francisco, CA, Mar. 1998, pp. 1418–1425.
10. J. Raju and J.J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-Free Multipath Routing," Proceedings of IEEE ICCCN'99, Boston, MA, Oct. 1999, pp. 522–527.
11. M.K Marina, S. R. Das, "Performance of Route Caching Strategies in Dynamic Source Routing," In the Proceedings of 2nd Wireless Networking and Mobile Computing (WNMC), Phoenix, April 2001, In conjunction with the Int'l Conference on Distributed Computing Systems (ICDS) 2001
12. M.R. Pearlman and Z.J. Haas, P. Sholander, S.S. Tabrizi. "On the impact of alternate path routing for load balancing in mobile ad hoc networks." Proceedings of the first workshop on mobile and ad hoc networking and computing (MobiHoc 2000), Boston, MA, Aug. 2000.
13. P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks," Proceedings of ACM/IEEE MOBICOM'99, Seattle, WA, Aug. 1999, pp. 195–206.
14. P. Papadimitratos, Z. J. Haas, E.G. Sirer, " Path Set Selection in Mobile Ad Hoc Networks," In the Proceedings of the Third ACM Symposium on Mobile AdHoc Networking & Computing (MobiHoc 2002) Lausanne, Switzerland, June 9–11 2002
15. S. Gold, "A PSPICE Macromodel for Lithium-Ion Batteries", The 12th Annual Battery Conference on Applications and Advances, California State Univ., Long Beach, CA, Jan 14–17, 1997
16. S.-J. Lee and M. Gerla, "AODV-BR: Backup Routing in Ad hoc Networks," Proceedings of IEEE WCNC2000, Chicago, IL, Sep. 2000.
17. S.-J. Lee and M. Gerla, "Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks," Proceedings of ICC 2001, Helsinki, Finland, June 2001.
18. S. Singh, M. Woo and C.S. Raghavendra, " Power Aware Routing in Mobile Adhoc Networks," In the Proceedings of the Fourth Annual International Conference on Mobile Computing and Networking (MOBICOM'98) 1998.
19. S. Vutukury and J.J. Garcia-Luna-Aceves, "An Algorithm for Multipath Computation Using Distance Vectors with Predecessor Information," Proceedings of IEEE ICCCN'99, Boston, MA, Oct. 1999, pp. 534–539.
20. UCLA Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory, GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network Systems, <http://pcl.cs.ucla.edu/projects/domains/gloimosim.html>.
21. V.D. Park and M.S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," Proceedings of IEEE INFO-COM'97, Kobe, Japan, Apr. 1997, pp. 1405–1413.
22. W.T. Zaumen and J.J. Garcia-Luna-Aceves, "Loop-Free Multipath Routing Using Generalized Diffusing Computations," Proceedings of IEEE INFOCOM'98, San Francisco, CA, Mar. 1998, pp. 1408–1417.