International Journal of Computer Applications (0975 – 8887)
Volume 28– No.8, August 2011

Performance Evaluation of Energy Efficient Routing Protocols for MANET

P.Sivasankar
Assistant Professor
NITTTR, Chennai

Dr.C.Chellappan
Professor, DCSE
Annauniversity, Chennai

S.Balaji
Software Engineeer
IBM, Bangalore

ABSTRACT
MANET are a dynamic and infra structureless networks. The major constraint of this type ao networks are Energy optimization because the nodes involved in these type of networks are battery operated. To improve the lifetime of these network can be improving the energy levels of the individual nodes of the network. This paper presents an analysis of the effects of different design choices for this on-demand routing protocols in wireless ad hoc networks. The analysis of this paper is based on the Dynamic Source Routing protocol (DSR), which operates entirely on-demand. This paper compares an existing DSR protocol with the proposed two modified DSR protocols named Maximised Energy Efficient Routing protocol (MEER) and Cluster Based Energy Efficient Routing protocol(CBEER). Based on the simulation, it is found that the energy efficiency and percentage of reliable delivery of packets in the MEER and CBEER protocol are considerably higher than the existing DSR protocol. Also these proposed protocols considerably increases the life time of the network.

General Terms
Energy, Cache, delay, throughput, speed.

Keywords
CBEER, Cluster, DSR, GloMoSim, MANET and Route cache.

1. INTRODUCTION
Mobile ad hoc networks[1] (MANETs) are instantly deployable, dynamic and without any wired base station or fixed infrastructure. A node communicates directly with the nodes within radio range and indirectly with all others using a dynamically determined multi-hop route.

A critical issue for MANETs is that the activity of nodes is energy-constrained. In the past few years, extensive research has been carried out in developing routing protocols for MANETs. Past research for reducing energy consumption has focused on the hardware and the operating system level. However, significant energy savings can be obtained at the routing level by designing minimum energy routing protocols that take into consideration the energy costs of a route when choosing the appropriate route.

This paper is worked on the network layer/routing layer & Radio layer and focuses on design and implementation of Maximised Energy Efficient[12,14,16] Routing(MEER), and Cluster Based Energy Efficient Routing Algorithm(CBEER) in the existing DSR[2] protocol. These algorithms are designed and implemented using Global Mobile Simulator (GloMoSim). Also the performance of the protocol is evaluated and compared with the existing DSR protocol.

The rest of the paper is organized as follows. Section 2 of this paper gives an overview of the basic operation of the DSR protocol. The Maximised Energy Efficient Routing Algorithm (MEER) and Cluster Based Energy Efficient Routing Algorithm(CBEER) are discussed in Section 3 and 4 respectively. In Section 5, we describe the methodology of the simulation study, including simulator features, performance metrics evaluated , and results and analysis are described. The conclusion of the paper and the simulation study of the proposed protocols are presented in section 6.

2. DYNAMIC SOURCE ROUTING PROTOCOL
DSR[1] is an on demand, source routing protocol, with each packet carrying in its header the complete, ordered list of nodes through which the packet will be routed. DSR consists of two mechanisms: *route discovery* and *route maintenance[3,8]***.** When a node **s** has a packet to send for which it does not have a route, it initiates route discovery by broadcasting a route request (RREQ). The request is propagated in a controlled manner through the network until it reaches either the destination node d or some intermediate node, ***n,*** that knows of a route to node ***d.*** Node ***n*** (or node **d**) then sends a route *reply (RREP)* to node s with the new route. In the case that multiple routes are located (i.e., multiple route replies are received), nodes **s** selects the one with the best metric (e.g., hop count).To reduce the cost of route discovery, each node maintains a cache of source routes it has learned through route discovery or other means such as snooping route in route replies and data packets. Each node also operates its network interfaces in promiscuous mode, disabling the address filtering function on each interface and causing the interface to receive all packets that it overhears. This section describes the basic operation of Route Discovery and Route Maintenance, although a number of optimizations to this basic operation exist [1, 9] that are not discussed here due to space limitations.

Route Maintenance is the mechanism by which a node detects whether or not a route kept in its cahe has become stale as result of host mobility and topology change. When an (intermediate) node n detects that the next link in a packets route is broken, it first sends a route error (RERR) message to the source node s that generated the packets route.

2.1 Route Caching in DSR

All routing information needed by a node participating in ad hoc network using DSR is stored in that node's route cache. Each node in the network maintains its own route cache. A node adds information to its route cache as it learns of new links between nodes in the ad hoc network; for example, a node may learn of new links when it receives a packet carrying either a Route Reply or a DSR Routing header. Likewise, a node removes information from its Route Cache as it learns that existing links in the Ad hoc network have broken; for example, a node may learn of a broken link when it receives a packet carrying a Route Error or through the link-layer retransmission mechanism reporting a failure in forwarding a packet to its next-hop destination.

The route cache should support storing more than one route to each destination. In searching the route cache for a route to some destination node, the route cache is indexed by destination node address. We describe the properties of this searching function on a route cache as follows: Each implementation of DSR at any node may choose any appropriate strategy and algorithm for searching its route cache and selecting a "best" route to the destination from among those found. For example, a node may choose to select the shortest route to the destination (the shortest sequence of hops), or it may use an alternate metric to select the route from the cache.

An implementation of a route cache may provide a fixed capacity for the cache, or the cache size may be variable. We describe the properties of the available space within a node's Route Cache as follows: Each implementation of DSR at each node may choose any appropriate policy for managing the entries in its route cache, such as when limited cache capacity requires a choice of which entries to retain in the cache. For example, a node may chose a "least recently used" (LRU) cache replacement policy, in which the entry last used longest ago is discarded from the cache if a decision needs to be made to allow space in the cache for some new entry being added. However, the Route Cache replacement policy should allow routes to be categorized based upon "preference", where routes with a higher preference are less likely to be removed from the cache. For example, a node could prefer routes for which it initiated a Route Discovery to routes that it learned as the result of promiscuous snooping on other packets. In particular, a node should prefer routes that it is presently using to those that it is not..

3. MAXIMISED ENERGY EFFICIENT ROUTING ALGORITHM (MEER)

The basic principles of this algorithm is in the selection of routes on the remaining energy levels of the nodes that constitute the route. This is obtained by modifying the DSR in such a way that the source node 'knows' about the energy levels[4,5,7] of the intermediate nodes and hence can choose the most energy efficient route. MEER differs from the conventional DSR only in the Route Discovery. The other aspects of DSR remain essentially the same.

3.1 Route Discovery

This selection of the best route is based on the following algorithm: the destination node first determines the least power level in each route that is reported to it by the RREQ packets. Then it compares these least power levels and chooses the highest among them. It then selects the corresponding route. The destination node then transmits the RREP packet through this route. It should be noted that the destination copies the energy information from the RREQ packet to the RREP packet. Thus, by our algorithm, the destination node selects the route with the highest life time from a set of available routes (since the least energy level is maximum, the selected route has the highest life time among the available routes).

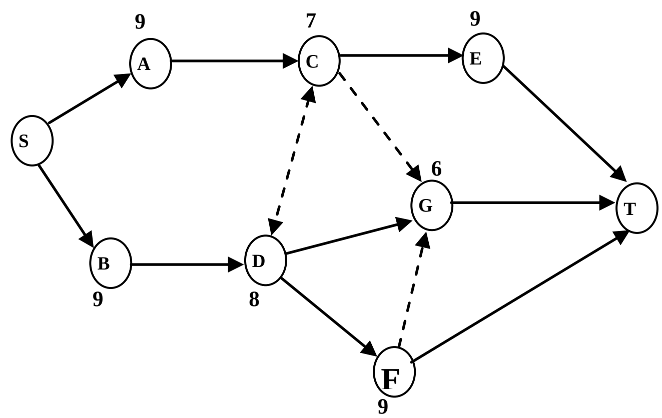

Fig 1: MEER algorithm

This algorithm can be explained using the network given in above Figure. The nodes and their energy levels at any arbitrary time are shown. Let us assume that the node S wants a route to node T. The possible routes from S to T are: (S-A-C-E-T), (S-B-D-F-T), and (S-B-D-G-T). The RREQ packets that originate from S travel along these paths and reach the destination T. Assuming that the path (S-A-C-E-T) is the shortest path, the conventional DSR protocol would have chosen it and rejected the RREQ packets from the other routes. But the MEER algorithm caches all the three routes and compares them. By observation, the least power levels in the above routes are 7, 6 and 8 respectively. Since the energy level 8 is the highest among these least energy levels, the corresponding route (S-B-D-F-T) is chosen and the others discarded. Thus the MEER algorithm increases the lifetime of mobile ad hoc networks at the expense of end delay, routing overhead and system complexity (*route-request cache* is to be implemented).

3.2 Route Cache

All routing information needed by a node participating in ad hoc network using DSR is stored in that node's route cache[6]. Each node in the network maintains its own route cache. A node adds information to its route cache as it learns of new links between nodes in the ad hoc network; for example, a node may learn of new links when it receives a packet carrying either a Route Reply or a DSR Routing header. Likewise, a node removes information from its Route Cache as it learns that existing links in the Ad hoc network have broken; for example, a node may learn of a broken link when it receives a packet carrying a Route Error or through the link-layer retransmission mechanism reporting a failure in forwarding a packet to its next-hop destination.

The route cache should support storing more than one route to each destination. In searching the route cache for a route to some destination node, the route cache is indexed by destination node address. We describe the properties of this searching function on a route cache as follows: Each implementation of DSR at any node may choose any appropriate strategy and algorithm for searching its route cache and selecting a "best" route to the destination from among those found. For example, a node may choose to select the shortest route to the destination (the shortest sequence of hops), or it may use an alternate metric to select the route from the cache.

An implementation of a route cache may provide a fixed capacity for the cache, or the cache size may be variable. We describe the properties of the available space within a node's Route Cache as follows: Each implementation of DSR at each node may choose any appropriate policy for managing the entries in its route cache, such as when limited cache capacity requires a choice of which entries to retain in the cache. For example, a node may chose a "least recently used" (LRU) cache replacement policy, in which the entry last used longest ago is discarded from the cache if a decision needs to be made to allow space in the cache for some new entry being added. However, the Route Cache replacement policy should allow routes to be categorized based upon "preference", where routes with a higher preference are less likely to be removed from the cache. For example, a node could prefer routes for which it initiated a Route Discovery to routes that it learned as the result of promiscuous snooping on other packets. In particular, a node should prefer routes that it is presently using to those that it is not.

4. CLUSTER BASED ENERGY EFFICIENT ROUTING ALGORITHM (CBEER)

In this section, we describe a new cache route selection mechanism for MANET routing protocols, what is we call Cluster Based Energy Efficient Routing Algorithm(CBEER). In this algorithm, selection of routes should be based on the remaining battery energy level of the node. The CBEER protocol was designed to bring about energy aware route establishment and route maintenance in the DSR protocol. CBEER differs from DSR in the methods of route establishment and route maintenance as well as in the structures of the route request and route reply packets.

4.1 Route Establishment

The CBEER protocol is composed of two main mechanisms that work together to allow the discovery and maintenance of source routes in the ad hoc network. In this section, we discuss the Route discovery and Route maintenance algorithms in detail.

4.1.1 Route Discovery Algorithm

In this section we present an algorithm, which selects the route based on the Energy metrics of a node participating in the mobile network to route and deliver the packet to the destination. In this algorithm, selection of routes should be based on the remaining battery energy level of the node. The CBEER protocol was designed to bring about energy aware route establishment in order to avoid the full drain of the energy from a node, which in the network forms a gateway to the other zones. The proposed algorithm differs from the existing DSR protocol in the route discovery and energy aware route maintenance with higher cache hit ratio and higher percentage of reliable delivery of packets.

The modification of the protocol results in reduction of unnecessary packets in the network due to the flooding of route error and route discovery packets. Though the route established in the protocol is not shortest path, comprise is done on the energy level remaining in the node for transmission and reception of packets from the nodes in the network and the path needed for the packets to trace to the destination.

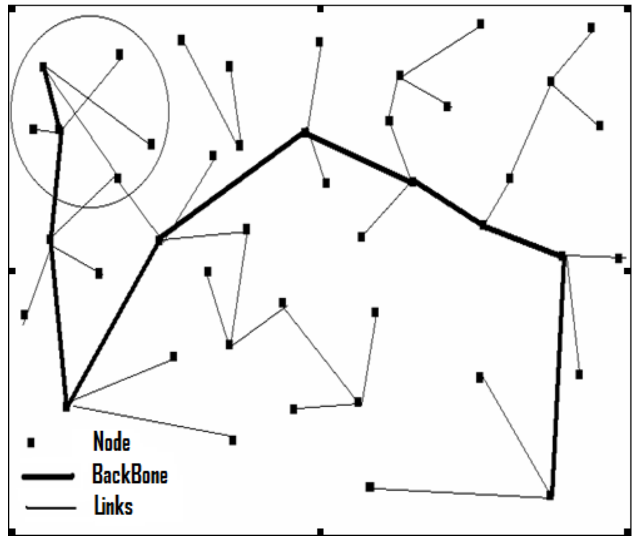

Fig 2: Route Establishment

The network is formed by the “divide and rule” policy for the nodes to deliver the packets to the destination. The tree structure with virtual backbone is used to deliver the packets reliably to the destination with higher cache hit ratio and optimized use of energy in the network. A node on entry to the network gets itself associated to one of the root if its energy level is lesser than the root node else it will act as a root and the node which was a root becomes a leaf node. A virtual backbone is formed with the nodes having the highest energy in the domain to establish routes from one node at one end to other end.

The CBEER protocol is composed of two main mechanisms that work together to allow the discovery and maintenance of source routes in the ad hoc network. In this section, we discuss the Route discovery and Route maintenance algorithms in detail.

4.1.2 Route Discovery

The DSR protocol broadcasts a route discovery packet and the reply is formulated by the node, which has entry of the destination node in its cache or the destination replies with a route reply packet. The proposed algorithm makes sure that the route discovery packet is forwarded to its root and if the root has the cache entry for the packet it will reply back with the route reply or else it will in turn forward the packet to its root. Finally if the route is non-existent till the root node, which participates in the backbone formation, then the route discovery packet is sent through the backbone to the other domain nodes, which may reply back with a route reply. Thus on establishing a route, the route reply containing the route back to the source is routed back. Thus if the destination node is not existent in the domain then after a time out and resend attempts, the source node could find that the destination node is unreachable.

4.2 Route Maintenance Algorithm

The *CBEER algorithm* differs considerably from DSR in Route Maintenance. The route maintenance is easier as the hello packet which contains the cache contains of the node can be interchanged between the leaf node and root nodes. This will ensure that the route is existent and the route reply can be generated from the reference of the cache. The energy based tree formation ensures the participation of nodes in the network though their power remaining is less, by reception of packets intended to them and transmission of packets (acting as a source) but not as a router. A field namely the ROOT_NODE or LEAF_NODE need to be interchanged between the root and the leaf nodes (within the range of the root node) with the sequence number for tracking of the route and identify to which root the node belongs to.

5. SIMULATION TOOL

To evaluate the above said protocols, The GloMoSim (Global Mobile Simulator) is used . About this simulation tool is discussed under the following subsections

5.1 GLOMOSIM

GloMoSim [17] (Global Mobile Information System Simulator) is a scalable simulation environment that effectively utilizes parallel execution to reduce the simulation time of detailed high-fidelity models of large communication networks. GloMoSim is a scalable simulation library for wireless network systems built using the PARSEC simulation environment. Glomosim can be modified to add new protocols and applications to the library. Therefore Glomosim is a good choice for implementing the different traffic sources. GloMoSim is aimed at simulating models that may contain as many as 100,000 mobile nodes with a reasonable execution time; this is done by using node aggregation. As each entity needs to examine packets received only from the nodes located in the region it is simulating, many partitions are used to reduce the total search space for packet delivery. If a packet sent by a node located in Partition (0, 0) cannot reach the border of the partition, no message needs to be sent to the other partitions. Therefore, the other partitions do not have to examine the reception of the packet.

5.2 PARSEC

PARSEC (for Parallel Simulation Environment for Complex systems) is a C-based simulation language developed by the Parallel Computing Laboratory at UCLA, for sequential and parallel execution of discrete-event simulation models. One of the important distinguishing features of PARSEC is its ability to execute a discrete-event simulation model using several different asynchronous parallel simulation protocols on a variety of parallel architectures. PARSEC is designed to cleanly separate the description of a simulation model from the underlying simulation protocol, sequential or parallel, used to execute it.

GloMoSim is being designed using a layered approach similar to the OSI seven-layer network architecture. Simple APIs are defined between different simulation layers. This allows the rapid integration of models developed at different layers by different people. Actual operational code can also be easily integrated into GloMoSim with this layered design, which is ideal for a simulation model as it has already been validated in real life and no abstraction is introduced.

6. RESULT AND ANALYSIS

In this section the result analysis are carried out for various performance metrics. Also the performances of the proposed algorithms are assessed and compared with the exisiting DSR protocol

6.1 Performance Metrics

The various parameters that were measured during the simulation are as follows:

Throughput: It is defined as the ratio of number of packets received to that of the number of packets sent.

Control overhead: It is defined as the sum of number of route requests, route replies & route errors.

End to End Delay: It is the overall average delay experienced by a packet from the source to that of the destination.

Average Energy left: It is taken as the average of the remaining energy levels of all the nodes in the network.

These metrics were measured by varying the following four parameters

1. Speed (m/s) [11,15]
2. Number of Source Destination Pairs
3. Number of nodes

Note: The performance analyses that are carried out in the subsequent sections are based on the extensive simulations and statistics collected amount to 500 Mbytes.

The simulation input parameters apart from varying values used for study and analysis are in the following given Table 1:

Table 1. Simulation parameters

SIMULATION-TIME	15M
SEED	1
NODE-PLACEMENT	UNIFORM
MOBILITY	RANDOM-WAYPOINT
MOBILITY-WP-MIN-SPEED	0
MOBILITY-POSITION-GRANULARITY	0.5
PROPAGATION-LIMIT	-111.0
PROPAGATION-PATHLOSS	FREE-SPACE

NOISE-FIGURE	10.0
TEMPERATURE	290.0
RADIO-TYPE	RADIO-ACCNOISE
RADIO-INITIAL-POWER-LEVEL	4000
RADIO-FREQUENCY	2.4e9
RADIO-BANDWIDTH	2000000
RADIO-RX-TYPE	SNR-BOUNDED
RADIO-RX-SNR-THRESHOLD	10
RADIO-TX-POWER	15
RADIO-ANTENNA-GAIN	0.0
RADIO-RX-SENSITIVITY	-81.0
RADIO-RX-THRESHOLD	-81.0
MAC-PROTOCOL	802.11
PROMISCUOUS-MODE	NO
NETWORK-PROTOCOL	IP
NETWORK-OUTPUT-QUEUE-SIZE-PER-PRIORITY	100

6.2 Simulation results and performance comparison

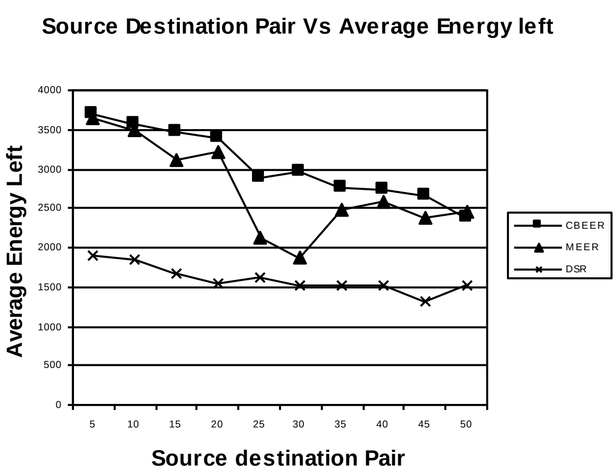

Fig 3: No. of Source Destination pairs Vs Avg. Energy left

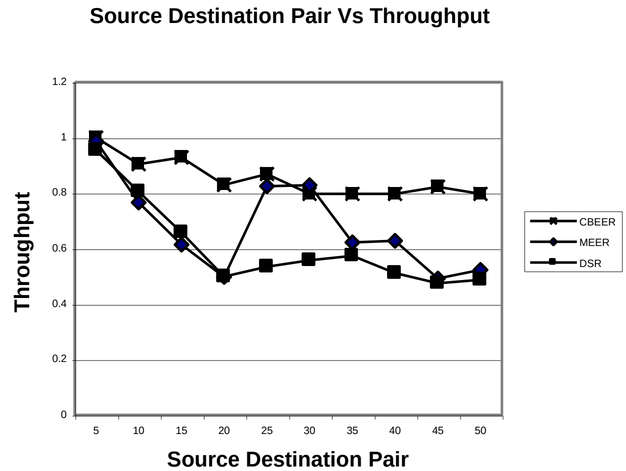

Fig 4: No. of Source Destination pairs Vs Throughput

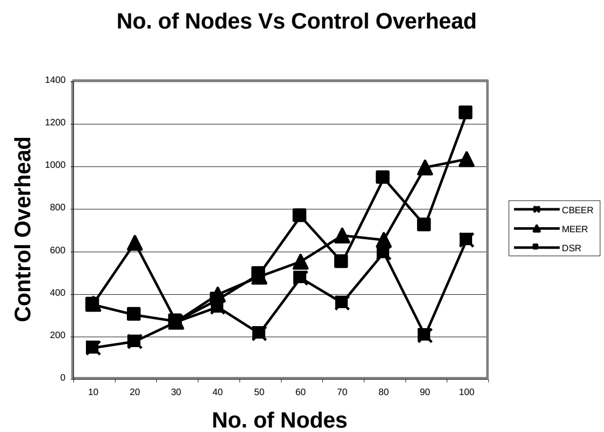

Fig 5: No. of Nodes Vs Control Overhead

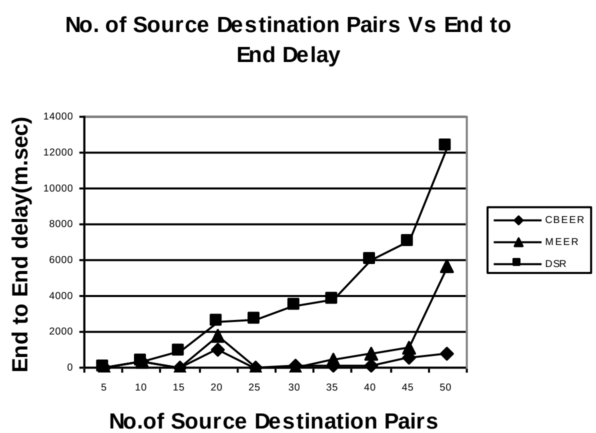

Fig 6: No. of Source Destination pairs Vs End to End delay

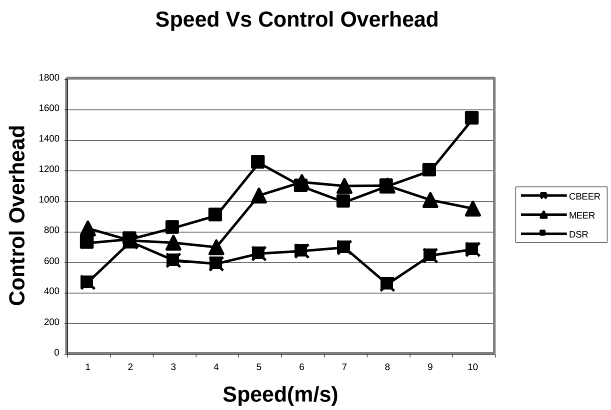

Fig 7: Speed Vs Control Overhead

Figure 3 shows Average remaining energy of the nodes for various numbers of source destination pairs. In this performance comparisons CBEER and MEER algorithms show better performance over DSR. In figure 4, CBEER shows better throughput over MEER and DSR. Figure 5 and 7, shows better performances like reduced control overhead for CBEER and MEER protocols over DSR by varying the parameters like number of nodes and speed of the nodes mobility. Similarly DSR has high end to end delay in delivering the packets to the destination nodes over the CBEER and MEER as in Figure 6.

7. CONCLUSION

The DSR protocol has been implemented and compared with the modified energy aware MEER and CBEER protocols and it is observed that the improved performances of the ad-hoc network. It is found that the modified algorithm has a comparable performance with respect to Average Energy left, End to End Delay, Throughput and Control overhead with the existing DSR protocol. From the simulation results it is concluded that the lifetime and packet delivery ratios of the network is improved with the reduction in control overhead and End to end delay for the proposed protocols over the existing DSR protocol. It has also been observed that by implementing a proper standardized energy model in the existing DSR protocol, our MEER and CBEER protocols are feasible and capable of better energy performance than the preset DSR protocol.

8. REFERENCES

[1] Chih-fan Hisn, Mingyan Liu,"An analytical study of the effect of lifetime of stored routes on routing control overheads".

[2] X.Hong,K.Xu and Gerla , "Scalabe Routing Protocols for MANET", IEEE network vol 16, pp 11-21, july/aug 2002.

[3] Bor rong chen and C.Hwa Chang , "Mobility impact on adhoc networks", IEEE proceedings, may2003.

[4] Dongkyun Kim, J.J Garcia-Luna-Aceves, Katia Obraczka, Juan-Carlos Cano and Pietro Manzoni, "*Power-Aware Routing Based on The Energy Drain Rate for Mobile Ad hoc Networks*"

[5] Huang Jingbo, Hong Peilin and Li Jinshen "Power-aware routing based on DSR for Mobile Ad hoc NETworks", -Journal of Electronics (China), Volume 24, Pages 167-173, Number 2 / March, 2007.

[6] 4. Ben Liang, Zygmunt J. Haas, "*Optimizing Route-Cache Lifetime in Ad hoc Networks*"

[7] Morteza Maleki, Karthik Dantu, Massoud Pedram, "*Power-aware Source Routing Protocol for Mobile Ad hoc Networks*".

[8] Elizabeth M.Royer, C.K TOH,"*A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks*".

[9] Azzedine Bourkerche and Harold Owens II, "Energy aware routing protocol for mobile and wireless adhoc networks", proceedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN'03)2003.

[10] Joo-Han Song, Vincent W.S Wong and Victor C.M Leung , "Efficient on-demand routing for mobile adhoc wireless networks", IEEE Proceedings.

[11] Bor rong chen and C.Hwa Chang , "Mobility impact on adhoc networks", IEEE proceedings, may2003.

[12] Huang Jingbo, Hong Peilin and Li Jinshen "Power-aware routing based on DSR for Mobile Ad hoc NETworks", -Journal of Electronics (China), Volume 24, Pages 167-173, Number 2 / March, 2007.

[13] Rajiv Bagrodia, Xiang zeng and Mario Gerla, "Glomosim: A library for simulation of wireless networks", University of califrnia ,Los Angels.

[14] Eei Yu and jangwon Lee , "DSR based Energy aware Routing Protocols in Ad Hoc networks" Members, IEEE, July 2003.

[15] Bor rong chen and C.Hwa Chang , "Mobility impact on adhoc networks", IEEE proceedings, may2003.

[16] Huang Jingbo, Hong Peilin and Li Jinshen "Power-aware routing based on DSR for Mobile Ad hoc NETworks", -Journal of Electronics (China), Volume 24, Pages 167-173, Number 2 / March, 2007.

[17] Rajiv Bagrodia, Xiang zeng and Mario Gerla, "Glomosim: A library for simulation of wireless networks", University of califrnia ,Los Angels.