

Modified Peer Protocol For Energy Efficient Routing In Mobile Ad Hoc Network

S. Divya

Research Scholar

Dr.SNS Rajalakshmi College of Arts and Science

B. Murugesakumar

Assistant professor

*Dept of Master of Computer Applications
Dr.SNS Rajalakshmi College of Arts and Science*

IJERT

Abstract

Mobile Adhoc Network (MANET) is a self-configuring infrastructureless network of mobile hosts connected by wireless links. The wireless links in the network are highly error prone and can go down frequently due to mobility of nodes. Therefore, energy efficient routing over MANET is still a critical task due to highly dynamic environment. In this paper, a link cost model is derived to accurately track the energy consumption on different factors. Then the route discovery and route maintenance is found with the minimum energy routing protocol and the PEER protocol. IEEE 802.11 DCF uses exponential backoff algorithm. In this, average backoff value is increased after every collision due to the large size of contention window. So, considerable proportion of bandwidth is lost due to collision. Due to that the energy consumption is increased. we propose a Modified Progressive Energy Efficient Routing Protocol (MPEER) protocol using FIB with a quick and low overhead path discovery scheme and an efficient path maintenance scheme for reducing energy consumption especially in mobile environment. "Fibonacci Increment Back off (FIB)" mechanism is used to improve the performance of the bandwidth for energy efficient routing in ad hoc networks. The experimental analysis of proposed protocol has been carried out using network simulator NS-2.

1. Introduction

MANETs consists of mobile hosts equipped with wireless communication devices. The transmission of a mobile host is received by all hosts within its transmission range due to the broadcast nature of wireless communication and omnidirectional antennae. These are networks in which all nodes are mobile and communicate exclusively via wireless connections. Usually, the nodes are equipped with a single, omni-directional wireless antenna. There is no fixed infrastructure in the network, and there is no hierarchy; all nodes are equal, and can function both as end points of data communication and as routers, forwarding data for each other in multi-hop fashion. A group of users carrying Wi-Fi enabled devices such as mobile phones, laptops etc., moving in a particular area and forming a dynamic wireless network among them. A MANET is a type of ad hoc network that can change locations and configure itself on the fly. Protocols are evaluated based on measure such as the packet drop rate, end-to-end packet delays, network throughput

etc. The routers are free to move randomly and organize themselves arbitrarily. Topology may change rapidly and unpredictably.

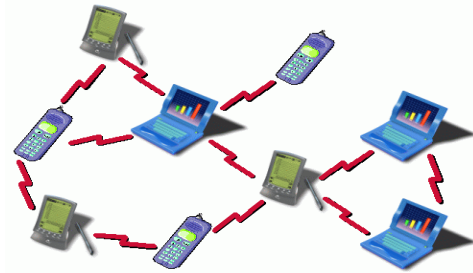


Figure 1: An Example Structure of MANET

An Example Structure of MANET is shown in Figure 1. Generally, MANET is defined as a collection of digital data terminals equipped with wireless transceivers that can communicate with one another without using any fixed networking infrastructure. Wireless ad hoc networks usually consist of mobile battery operated computing devices that communicate over the wireless medium. While the processing capacity and the memory space of computing devices increase at a very fast speed, the battery technique lags far behind. Therefore, it is critical to derive energy conservation schemes to increase the device and network operation time.

In wireless networks, [20] the transmitted signal is attenuated at the rate of $1/d^n$, where d is the distance between a sender and a receiver and n is the path loss exponent with value between 2 and 6 depending on the operational environment. Instead of using the maximum transmission power all the time, with power control, a sender can adjust the transmission power according to d . However, link level power control cannot ensure that the end-to-end energy consumption from a source to a destination is minimum. To conserve energy, many energy efficient routing protocols have been proposed [1]-[9]. These protocols can be generally classified into two categories: Minimum Energy routing protocols[1]-[6] and Maximum Network Lifetime routing protocols[8][9]. Minimum Energy routing protocols search for the most energy efficient path from the source to the destination, while Maximum Network Lifetime routing protocols attempt to balance the remaining battery-power at each node when searching for the energy efficient path.

Since Minimum Energy routing scheme is also an important part in most recent Maximum Network Lifetime routing protocols such as Conditional Max-Min Battery Capacity Routing (CMMBCR) [8] and Conditional Maximum Residual Packet Capacity (CMRPC) routing [9], we will focus on developing more efficient Minimum Energy routing protocols in this paper. Minimum Energy routing protocols can be

further divided into three classes based on the types of link costs: Minimum Total Transmission Power (MTTP), Minimum Total Transceiving Power (MTTCP), and Minimum Total Reliable Transmission Power (MTRTP). MTTP protocols use the transmission power as the link cost metric and search for the path with minimum total transmission power between the source and the destination.

MTTCP protocols use the transmission power plus the receiving power as the link cost. Authors in [3] used distributed Bellman-Ford algorithm to obtain the minimum total transceiving power path. MTTP and MTTCP protocols proposed in the literature, however, did not consider the energy consumption due to data packet retransmissions. Instead, authors in [4] proposed a MTRTP protocol to take into account the energy consumption of packet retransmissions. The total transmission power consumed for reliably transmitting a data packet from one node to its neighboring node is used as the link cost.

In existing minimum energy routing protocols, signaling packets are often transmitted at the maximum power to reduce the hidden terminal problem as a result of using asymmetric transmission powers from different neighbouring nodes. The signalling packet that experiences more collisions, for example the RTS packet in 802.11, would consume significant amount of power. Without taking into account the energy used for signalling, the path discovered could consume much more energy than a path selected based on a more accurate energy consumption model. In addition, most of literature work focused only on the development of new link cost metric. Once a new link cost is derived, the traditional shortest path routing protocols, such as AODV (Ad hoc On Demand Distance Vector) and DSR (Dynamic Source Routing) protocols, are modified to search for the minimum cost path. However, such straightforward modification would lead to several problems. First, the routing overhead in route discovery phase is very high, which not only consumes a significant amount of energy but also leads to a long path setup delay. Second, the route maintenance scheme used in conventional shortest path routing protocol is not suitable for maintaining energy efficient path in a mobile environment.

2. Progressive Energy Efficient Routing Protocol (PEER) Protocol

2.1. Energy Consumption Model for 802.11

PEER is a cost-based energy efficient routing protocol. In a cost-based routing protocol, the total cost of all the links on each available path between the source node and the destination node will be calculated, and a minimum cost path (meeting certain criteria) will be selected. As link cost is very

important in the cost-based energy efficient routing protocols, it is critical to derive an accurate link cost metric to obtain an optimal path.

In this section, [39] we will derive the link cost and show how to estimate the parameters needed for link cost calculation. As Modified PEER will run over 802.11 MAC, in the following, will derive the link cost for 802.11 wireless networks. Two MAC schemes have been specified in IEEE 802.11 DCF[23] (Distributed Coordination Function) and PCF (Point Coordination Function). As PCF is a centralized protocol, we will only consider DCF at MAC layer in this paper as it will be used to work with Modified PEER. For better describing our link cost model, we first give a brief overview of DCF. IEEE 802.11 DCF is based on CSMA/CA (Carrier Sensing Multiple Access with Collision Avoidance) mechanism.

It consists of two carrier sensing schemes, namely physical carrier sensing and virtual carrier sensing. The virtual carrier sensing scheme is implemented with NAV (Network Allocation Vector). If a node receives a packet (such as RTS, CTS and DATA packet), it will update NAV with the duration included in the received packet. The NAV value indicates when the on-going transmission session will end.

If a node has data packets to send to another node, it first checks its NAV. If its NAV is larger than 0, it has to wait until NAV reaches 0.[35] After that, the sender transmits a RTS packet after the channel is available for a period longer than DIFS (DCF Inter Frame Space) or the backoff timer reaches zero.

The receiver responds with a CTS packet after receiving the RTS packet. If the sender does not receive the CTS packet within a predetermined time interval, it will retransmit the RTS packet. After receiving the CTS, the sender will send out the data packet and the receiver will reply with an ACK packet after receiving the data packet successfully. If the sender doesn't receive the ACK packet within a predefined time period, the whole process will be repeated.

Parameter Estimation for Link Cost

To simplify the expressions in the analysis, [39] we denote the data size, the 802.11 header size, the RTS packet size, the CTS packet size and ACK packet size by N , N_{hdr} , N_{rts} , N_{cts} , and N_{ack} , respectively. And we also define the following symbols:

$$N_8 = N + N_{hdr} + N_{phy}; N_r = N_{rts} + N_{phy};$$

$$N_c = N_{cts} + N_{phy}; \text{ and } N_a = N_{ack} + N_{phy};$$

where N_{phy} is the size of physical layer overhead. Then the average total transmission power for successfully transmitting a packet from node i to node j is

$$\overline{P_T}(i,j) = \frac{P_m N_r}{P_{r,i,j} P_{c,i,j} P_{i,j} P_{a,i,j}} + \frac{P_m N_c}{P_{c,i,j} P_{i,j} P_{a,i,j}} + \frac{P_{i,j} N_s}{P_{i,j} P_{a,i,j}} + \frac{P_{i,j} N_d}{P_{a,i,j}} \\ \frac{P_m(N_r + N_c P_{i,j})}{P_{r,i,j} P_{c,i,j} P_{i,j} P_{a,i,j}} + \frac{N_s P_{i,j} + P_{i,j} N_d P_{i,j}}{P_{i,j} P_{a,i,j}} \quad (3.1)$$

In addition, denoting the receiving power as P_r , then the average total receiving power for successfully receiving a packet from node i to node j is

$$\overline{P_R}(i,j) = P_r \frac{N_r + (N_c + P_{i,j} + \frac{N_d}{N_s} P_{i,j} P_{a,i,j}) P_{c,i,j}}{P_{c,i,j} P_{i,j} P_{a,i,j}} \quad (3.2)$$

Assume there are $M-1$ intermediate nodes between a source and a destination. Let the nodes along the path from the source to the destination be numbered from 0 to M in that order. Then the average total power for reliable transmission along the path from the source (node 0) to the destination (node M) is

$$\overline{P_{total}} = \sum_{i=0}^{M-1} (\overline{P_T}(i,i+1) + \overline{P_R}(i,i+1)) \quad (3.3)$$

Based on this formula, it is apparent that

$$\overline{P_{total}} = \sum_{i=0}^{M-1} (\overline{P_T}(i,i+1) + \overline{P_R}(i,i+1))$$

would be the link cost between node i and $i+1$.

Most parameters in the link cost model (Equ. (3.5)) can be easily obtained except the transmission powers ($P_{i,j}$ and $P_{j,i}$) and the packet error rates ($P_{r,i,j}$, $P_{c,j,i}$, $P_{i,j}$ and $P_{a,j,i}$). In this section, we will show how to estimate these parameters. For parameter estimation purpose, we make the following assumptions: (1) the path loss between two nodes is symmetric on both directions; (2) the physical layer can provide the information on the average power level of a packet (such as RTS/CTS) received and the average interference level to the MAC layer. These are common assumptions made in many power control schemes as well as energy efficient routing protocols.

Since the wireless signal is attenuated at the rate of $\frac{1}{d^n}$ (d is the distance and n is the path loss exponent), the received power level (P_r) at the receiver is proportional to $\frac{P_t}{d^n}$, where P_t is transmission power level. That is,

$$P_r = k \frac{P_t}{d^n} \quad (3.4)$$

where K is a factor depending on the environment. With this formula, a node can send a packet at a

known power level, and calculate the desired transmission power for other packets based on the received power level of the known packet and the target receiving power. For example, if node

A receives a packet P_e (e.g., RTS, CTS and broadcast packets) at per bit power level P_r , and it knows that the packet was sent by B using maximum per bit transmission power (P_m), then node A can calculate the necessary per bit transmission power node B needs to use to transmit other packets to A with the following equations:

$$\begin{cases} P_r = K X P_m / d^n \\ P_r^{th} = K X P_t(B, A) / d^n \end{cases}$$

where P_r^{th} is the minimum necessary received power level. It is easy to obtain $P_t(B;A)$ as

$$p_t(B,A) = P_r^{th} \times \frac{P_m}{P_r} \quad (3.5)$$

As we assume that the path loss is the same on both directions, $P_t(A;B)$ for a packet from A to B will be the same as $P_t(B;A)$. That is, node A can estimate its necessary transmission power as well as the necessary transmission power B should use to transmit a packet to itself. Packet error is mainly caused by collision, interference and noise.

Here we distinguish the concept of collision and interference by the carrier sensing zone. If the error is caused by the nodes within the carrier sensing zone, we call it collision, otherwise interference. It is easy to obtain the interference and noise level since each node can monitor it when the channel is free. With interference and noise level measured, we can then calculate the bit error rate (BER) based on the received power level and modulation scheme. Once we get the BER, we can calculate the packet error rate (PER) caused by the interference and noise as

$$PER = 1 - (1 - BER)^L,$$

where L is the number of bits in the packet.

Most collisions happen during the transmission of RTS. Therefore, we only need to consider the packet error rate caused by the collision of RTS packet.

$$p_c(t+1) = \alpha p_c(t) + \frac{1-\alpha}{N} \sum_{i=0}^{N-1} C_{t-i} \quad (3.6)$$

where $p_c(t)$ is the estimated collision probability at time t , α is the remembering rate, and C_{t-i} with $i=0; \dots; N-1$ are the last N slot samples. C_i is equal to 0 if the i -th slot is free or the node transmits successfully in such slot; otherwise C_i is 1.

Therefore, the packet error rates for CTS, DATA and ACK packets are calculated based on the interference and noise power, the receiving power, and the packet size. While for RTS packet, we need to take into account the packet error rate caused by both interference and collision. Denote the packet error rate due to interference and noise by p_{int} , the

packet error rate due to collision by p_c , the packet error rate of RTS packet can be calculated as:

$$P_{r,i,j} = P_{int} + P_c - P_{int} * P_c \quad (3.7)$$

This work proposes and evaluates FIB algorithm to check if it could replace BEB for variable and high traffic load networks. Results show that the proposed FIB outperforms BEB in some cases which depend on the range of the Fibonacci series the algorithm shall follow keeping in mind that FIB range must not exceed 10 numbers in order to conserve energy.

2.2. PEER Protocol

A routing strategy should not get some arbitrary path quickly and rely on a route maintenance scheme to adjust the path later to an energy efficient one, as it may take much more time and create a larger overhead to adapt the route and there is no guarantee that such adaptation could find a path that leads to energy saving comparable to the minimum energy one.

2.2.1. Route Discovery Process

In this section, introduce the route discovery strategy of PEER.[19] The quickest way to find a path between two nodes would be through a shortest path routing scheme. However, there may exist a few shortest (smallest number of hops) paths between the source node and destination node. For example, in Fig. 2, assuming all the intermediate nodes (A, B, E, F, G, H) are the neighboring nodes of both S and D while S and D are beyond transmission range, then there are six shortest (2 hops) paths (SAD, SBD, SED, SFD, SGD, SHD). Among all the shortest paths, it is better to pick the most energy efficient one (we call it minimum energy shortest path).

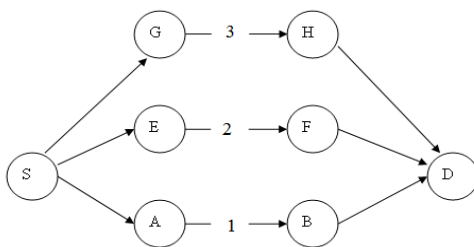


Figure 2: Three routes between node S and D.

Denote the set of paths between the source and the destination by L , the number of hops for path l by N_l , and the energy consumption for link i in path l by $E_{l,i}$, then the set of shortest paths L_s would be

$$L_s = \arg \min(N_l), l \in L \quad (3.8)$$

The set of minimum energy shortest paths L_{ms} would be

$$L_{ms} = \arg \min \left(\sum_{i=1}^{N_l} E_{l,i} \right), l \in L_s \quad (3.9)$$

Even though there may be more than one minimum energy shortest path in L_{ms} , the routing protocol can pick a unique one by some criterion, such as route request packet arriving time. Based on the previous definition, the basic searching algorithm would be: (1) search for all shortest (fewest hops) paths; (2) pick the minimum energy path(s) among the shortest paths in (1).

To implement this algorithm, the route request packet should carry two pieces of information: one is the hop count, the other is the energy consumption. The source node first broadcasts the route request packet with both hop count and energy consumption set to 0.

Once an intermediate node receives a route request packet, it first updates the hop count (increased by 1) and energy consumption (increased by the energy consumption between the sender and itself) information in the route request packet. Then it will rebroadcast such packet only if one of the following conditions holds:

- The node hasn't received such a packet before or the packet comes from a shorter (smaller number of hops) path;
- The packet comes from a path with the same number of hops as the best path so far, but the energy consumption is lower.

The first condition ensures that the shortest path is selected, while the second condition selects the minimum energy path from all the shortest ones. This algorithm, however, has similar path selection issues as other energy efficient routing protocols.

That is, the destination node may receive many route request packets from different possible minimum energy shortest paths, but it could not tell which one is the best until it receives all possible packets. As the destination node has no knowledge on the number of route request packets it will receive, it may not be able to make the decision even if it has already received all the route request packets.

For example, assuming all six shortest paths (SAD, SBD, SED, SFD, SGD, SHD) in Fig. 3.2 have the same energy consumption and the destination D has received all of them, D may still not be able to select the best one as it does not know when is the best time to make the decision.

There are several ways to deal with this issue at the destination node. One option is that the destination sends a route reply for each route request it receives. As the destination may need to send out many route reply messages, this method will waste

energy. Also, the source node might transmit some data packets on less energy efficient path before the best one is found.

Another option is that the destination sets up a timer after receiving a route request packet. If it receives another route request before the timer goes off, it will reset the timer. Otherwise, it will select the best path found before the timer goes off and reply the source with a route reply packet. The third option is to set up a time window, and the destination will select the best path within the time window.

The last two methods help reduce the energy consumption, but it may increase the route setup time. We use the second one as it can adapt to the number of arriving route request packets. If there are only very few route request packets arriving at the destination, the destination can send back route reply packet quickly to reduce the route setup time.

On the other hand, it can wait for a period of time for the route request packet from a more energy efficient path to arrive if there are more route request packets arriving at the destination and there is no significant time difference between two consecutive request packets.

2.2.2 Route Maintenance

The route obtained in path discovery phase is suboptimal and may still lead to a higher end-to-end energy consumption than that of the minimum energy path.[19] In addition, the network environment can change dramatically due to node movements and dynamic channel conditions, and the previous energy efficient route may no longer be efficient as time goes on. Therefore, the route maintenance phase is very critical for energy efficient routing protocols.

TABLE 1

An Example Link Energy Table Recorded by a Node D

(a)	(b)	(c)	(d)	(e)	(f)	(g)
A	B	5	S1	D1	1	0
B	C	4	S1	D1	1	1
D	B	3	S2	D2	3	3
F	G	7	S3	D3	5	4
B	E	2	S2	D2	3	5

As extra signaling messages will consume more energy, the route maintenance scheme of PEER will not use additional periodic messages. Instead, an observing node will passively monitor data packets exchanged in its neighborhood and collaborate with its neighbors to look for a more energy-efficient path. Each node can estimate the necessary transmission

power and the link cost to a neighboring node once it receives RTS, CTS or broadcast packet from this node.

In PEER, each forwarding node will insert the link cost into the IP header of the packet targeted for its next-hop receiver as an IP option, and every node will monitor the data packets exchanged in its neighborhood to intercept the corresponding link costs and use these link costs to estimate the cost of a path segment. For each data packet transmitted, received, or overheard by a node, it will record the following information into a link cost table: a) sender, b) receiver, c) link cost between the sender and the receiver, d) source, e) destination, f) IP header ID, g) The current time. Among these parameters, (a) and (b) can be obtained from the MAC header, while (c) to (f) can be obtained from the IP header. The information for a link will be kept only for a short time for accurate information and reducing storage overhead.

From the link cost table, a node can know how a packet passes through its neighborhood and the total link cost for that. For example, node D's link energy table is shown in Table 1. As the parameters (source, destination, and IP header ID) can identify a packet. We can see in the table that node D records the path info for three packets: P1(S1, D1, 1), P2(S2, D2, 3) and P3(S3, D3, 5). The first packet (P1) goes through a two-hop path segment (A→B →C) in D's neighborhood and the total cost of the path segment is 9 (5 + 4). The second packet (P2) goes through another two-hop path segment (D→B →E) and the total cost of the path segment is 5 (3+2). The third packet (P3) goes through a one-hop path segment (F→ G) and the link cost is 7.

3. Modified Progressive Energy Efficient Routing protocol (MPEER) Protocol

The existing minimum energy based routing schemes often introduce big overhead during path discovery and the path setup time is very long. On the other hand, in this paper a Modified PEER Protocol is used much more time and create a larger overhead to adapt the route and there is no guarantee that such adaptation could find a path that leads to energy saving comparable to the minimum energy one. Therefore, a good strategy is to find a path close to the minimum energy one quickly and then use a maintenance scheme to adjust the path for further energy reduction.

Taking this into consideration, the Modified PEER search for the energy efficient path quickly during route discovery process, and maintains the route actively so that it can respond to topology and channel changes quickly. In the following, we show

how PEER achieves both goals. In that when the network is heavily loaded, the rate of collisions and the number of retransmissions heavily increase. This leads to a considerable decrease in network throughput and hence decrease the QoS as more and more bandwidth is wasted.

MAC layer facilitates sharing the channel among all the nodes in the network effectively. It also enables perfect channel utilization. In order to minimize collision, IEEE 802.11 DCF uses exponential backoff algorithm. In this, average backoff value is increased after every collision due the large size of contention window. The backoff process consumes lot of time. So, considerable proportion of bandwidth is lost due to collision. Moreover, the overhead is also introduced by backoff algorithm.

To overcome these issues, in this proposed approach, a new form of Fibonacci called "Fibonacci Increment Back off (FIB)" mechanism is used to improve the performance of the bandwidth estimation for IEEE 802.11 in ad hoc networks. This algorithm uses Fibonacci increment back off in which the differences between consecutive contention window sizes are reduced.

Experimental results show that the proposed approach which uses the Fibonacci Increment Back off achieves better throughput than the other back off algorithms. Another problem of BEB is stability. BEB has been designed to be stable for large number of nodes. However, a number of studies have shown that BEB could suffer from instability. Due to the disadvantages of BEB we going for FIB Algorithm.

3.1. Fibonacci Increment Backoff Algorithm

In this paper, a new backoff algorithm is referred to be a "Fibonacci Increment Backoff (FIB)" that can overcome the limitation of the existing BEB. In the FIB algorithm the difference between consecutive contention window sizes are reduced according to a Fibonacci sequence. Results from simulation experiments reveal that the proposed algorithm achieves higher throughput than the BEB when used in a mobile ad hoc environment.

This work proposes a new backoff strategy that allows nodes to wait for an incremental period formulated from the mathematical Fibonacci series when applying CSMA/CA algorithm as they need to access the channel. The Fibonacci Increment is a well known mathematical series that is formulated incrementally as each new subsequent number is formulated by adding the direct previous two numbers according to (3.10):

$$F(x) = F(x-1) + F(x-2), \text{ where } x \geq 0, F(0) = 0, F(1) = 1, x \geq 0 \quad (3.10)$$

This series has a number of interesting characteristics. Amongst these characteristics is a special value called the golden section property, the golden section property is obtained by calculating the ratio between every two successive terms in the Fibonacci series. After a certain number of terms, the ratio converges to a limit of

$$\frac{1 + \sqrt{5}}{2} \approx 1.618 \quad (3.11)$$

Following is the pseudo code for the proposed FIB algorithm where n is the maximum Fibonacci sequence number to be chosen, in other words, it is the range size FIB algorithm will go through. BP refers to the backoff period which typically equals to 0.00032 s.

```

For i=1 to n
    wtime= fib(i)* BP
int fib( n){
    if(n=1) return 1
else
    return fib(n-1)+fib(n-2)}

```

Proposed FIB Algorithm

The incremental behavior of Fibonacci mechanism is expected to minimize the possibility that two or more nodes choose the same backoff period in dense networks or networks with intensive traffic loads and hence avoid collision caused by BEB mechanism. As it stated previously; in order to save power which is the primary constrain for which the IEEE 802.11 is proposed, nodes are allowed to choose a random backoff period from a small range of [0-2BE-1].

For this reason, the BE value of BEB algorithm is allowed to reach 5 and not more, otherwise, longer values will lead to longer waiting time and longer delay which will directly drain the power causing dissipation which contradicts the IEEE 802.11e design goals. So, the BEB time each node shall wait before starting transmission can be calculated according to (3.12).

$$wtime = rand * BP \quad (3.12)$$

Where wtime is the waiting time each node shall wait, rand is the random value chosen from the range of [0 - 2 BE-1] and BP refers to the backoff period which equals to 0.32 ms. According to the previous formula, the BEB minimum wtime = 0, while the maximum = 31* 0.32 = 9.92 ms. This work proposes and evaluates FIB algorithm to check if it could replace BEB for variable and high traffic load networks. Results show that the proposed FIB outperforms BEB in some cases which depend on the range of the Fibonacci series the algorithm shall

follow keeping in mind that FIB range must not exceed 10 numbers in order to conserve energy.

4. Experimental Results

We have simulated PEER, Ad hoc On Demand Routing Protocol (AODV) and Modified PEER protocol in Network Simulator2 (NS2). The results have been derived by writing a tcl script and generating corresponding trace and nam files. The packet size is 512 bytes. The original standard MAC protocol has been modified to implement the variations of the backoff algorithms. Modifications have mainly targeted the mathematical formulas used to calculate new CW sizes. Several topologies and mobility scenarios have been created to test the algorithm as intensively as possible.

In order to assess the performance of different backoff mechanisms, values of mobility speed, traffic rate and network size had to be fed into the simulator. Testing for speed values, ranging from 2 m/s to 20 m/s has given useful information concerning the efficiency of the proposed algorithms for both slow and highly mobile MANETs as well. This simulation setup can increase the performances of the FIB on Energy Efficiency in Mobile Ad hoc network.

Energy Consumption

Figure 4.1 shows the Energy Consumption of the protocols as Energy Consumed and Time as its axis. The graph is drawn for the comparison of three protocols an Normal Energy Efficient Protocol – Ad hoc On-Demand Distance Vector Routing (AODV), Progressive Energy Efficient Routing Protocol (PEER), and The Modified Progressive Energy Efficient Routing (MPEER). Here the Energy consumption is minimum for the MPEER as compared with the AODV and PEER. It uses FIB which is more energy efficient than BEB used in the other protocols.

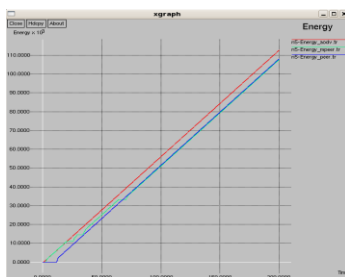


Figure 4.1 Energy Consumption

Throughput and Delay analysis FIB

The new FIB has improved the total throughput of the network simulated in our work. When the number of nodes is increased, the contention is higher to gain access to the channel. Because of the reduced amount of increment on the window size, a larger size of data was successfully received by nodes over the network. The same enhancement is noticed even while increasing mobility speed.

One of the major obstacles in the way of developing a MAC protocol for MANETs is mobility. Having a long backoff value allows the node to move outside the transmission range before being allowed to retry accessing the channel. With FIB, the ceiling of backoff periods is controlled to prevent extremely long backoff periods. One more factor participating in increasing throughput is reducing idle times. With smooth increments of contention window size, idle time is reduced. The FIB makes the protocol to speed up the path setup the path setup and minimize end-to-end delay and the energy consumption.



Figure 4.2 Throughput Analysis

Figure 4.2 shows the Throughput Analysis. The throughput analysis is drawn between simulation Time and the Kb/s. The graph is drawn for three protocols ADOV, PEER Protocol and the Modified PEER Protocol. Here the Throughput Analysis is maximum in mobile and hoc network for Modified PEER Protocol when it is compared with other two protocols like ADOV and PEER. The proposed algorithm increased the total throughput especially when the system size is large. In which FIB algorithm achieves higher throughput than the BEB that is used in a mobile ad hoc network. Hence FIB acts as an efficient one in energy saving mechanism.

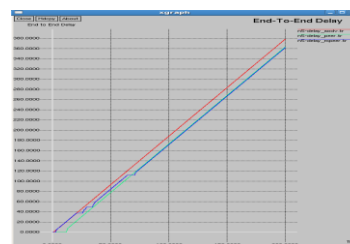


Figure 4.3: Delay Analysis

Figure 4.3 shows the Delay Analysis graph. The experimental graph is drawn for the End-To-End Delay and the simulation time. The graph is drawn for three protocols: ADOV, PEER Protocol and the Modified PEER Protocol. Here the Delay Analysis is minimum when it is used in mobile and hoc network for Modified PEER Protocol when it is compared with other two protocols like ADOV and PEER. The MPEER uses the FIB to minimize the Delay, which is better than BEB used in the other protocols.

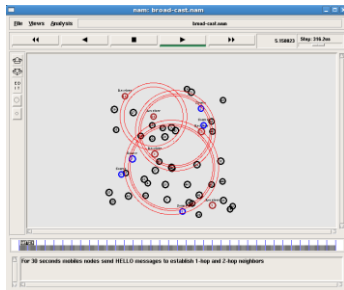


Figure 4.4 Network Animator for MPEER

Figure 4.4 shows the network animator file generated for the Modified PEER Protocol.

5. Conclusion and Future Work

It is important to design energy efficient routing protocols for mobile ad hoc networks. However, without a careful design, an energy efficient routing protocol could have much worse performance than a normal routing protocol. Specially, an energy efficient routing protocol could incur much higher control overhead and path setup delay as demonstrated by our simulations, and consume even more energy than a normal routing protocol in mobile environment.

In this paper, first derived a link cost model to accurately track the energy consumption due to various factors. We then discussed the issues in path discovery and route maintenance associated with the minimum energy routing protocols. Based on these observations and link cost metric, we propose a Modified progressive energy efficient routing (M-PEER) protocol with a quick and low overhead path discovery scheme and an efficient path maintenance scheme for reducing energy consumption especially in mobile environment. However, energy consumption still remains one of the main obstacles to the diffusion of this technology, especially in application scenarios where a long network lifetime and a high quality of service are required. Energy saving schemes aimed at minimizing the radio activity might be insufficient to fully address the energy savings issue and need to be complemented with (or replaced by) techniques for energy-efficient management at the sensor level. Intuitively, these

techniques operate to reduce the number of data acquisitions (i.e., data samples) rather than the number of transmitted messages.

Existing research on backoff algorithms for Energy Saving strategies mainly focuses on using external information, as opposed to information available from within the node, to decide the length of backoff timers. Such information includes network traffic load, transmission failures of other nodes and the total number of nodes in the network. In a mobile network, acquiring such information is not feasible at all times. To address this point, this thesis proposes new backoff algorithms called as Fibonacci Increment Backoff (FIB) in which it has an effective power saving mechanism in the Mobile Ad Hoc Network than the existing BEB. These algorithms use internal information only to make their decisions. The Fibonacci Increment Backoff algorithm is used to reduce the increment factor for large contention window sizes. Results from simulations have demonstrated that the proposed algorithm increased the total throughput of mobile ad hoc networks especially when the system size is large.

The total throughput has been increased highly for wireless network while used in the Energy saving mechanism. The throughput power which can be increased widely by reducing the idle times, with smooth increments of contention window size, idle time is reduced. The End-To-End Delay is minimized and it makes energy efficient and to speedup path setup without any delay and it can be maintained continuously. Hence the FIB is more energy efficient than the existing BEB and gives an energy efficient routing but there is always some scope of enhancement in MPEER is to simulate it with other QoS parameters and balancing the loads so that we could have an elegant approach towards efficient, energy saving and adaptable routing.

REFERENCES

- [1] K. Scott and N. Bambos, "Routing and Channel Assignment for Low Power Transmission in PCS", ICUPC '96, Oct. 1996
- [2] S. Doshi, S. Bhandare, and T. X Brown, "An On-demand Minimum Energy Routing Protocol for a Wireless Ad Hoc Network", ACM Mobile Computing and Communications Review, vol. 6, no. 3, July 2002
- [3] V. Rodoplu and T. Meng, "Minimum Energy Mobile Wireless Networks", IEEE Journal on Selected Areas on Communications, vol. 17, Aug. 1999. S. Banerjee and A. Misra, "Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks", MOBIHOC'02, June. 2002.

- [4] S. Banerjee and A. Misra, "Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks", MOBIHOC'02, June. 2002.
- [5] J. Gomez, A. T. Campbell, M. Naghshineh, and C. Bisdikian, "Conserving Transmission Power in Wireless Ad Hoc Networks", IEEE Conference on Network Protocols, Nov. 2001
- [6] J. Zhu, C. Qiao and X. Wang, "A Comprehensive Minimum Energy Routing Protocol for Wireless Ad Hoc Networks", INFOCOM'04, Mar. 2004
- [7] C. K. Toh, H. Cobb and D. Scott, "Performance Evaluation of Battery-Life-Aware Routing Schemes for Wireless Ad Hoc Networks", ICC'01, June 2001
- [8] A. Misra and S. Banerjee, "MRPC: Maximizing Network Lifetime for Reliable Routing in Wireless Environments", WCNC'02, Mar. 2002
- [9] ANSI/IEEE Std 802.11, 1999 Edition
- [10] E. Jung and N. H. Vaidya, "A Power Control MAC Protocol for Ad Hoc Networks", MOBICOM'02, Sept. 2002.
- [11] G. Bianchi and I. Tinnirello, "Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 network", INFOCOM'03, 2003
- [12] C-K Toh, "Ad Hoc Mobile Wireless Networks Protocols and Systems", Prentice Hall, 2002.
- [13] T. H. Cormen, C. E. Leiserson and R. L. Rivest, "Introduction to Algorithms", MIT Press, 1998
- [14] S. Gabriel, D. Moss and R. Melhem, "Mitigating the Flooding Waves Problem in Energy-Efficient Routing for MANETs", ICDCS'06, 2006
- [15] C. E. Perkins and E. M. Royer, "Ad hoc On-Demand Distance Vector Routing", Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, February 1999
- [16] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks", Mobile Computing, Kluwer Academic Publishers, 1996
- [17] DS. PalChaudhuri and D. B. Johnson, "Power Mode Scheduling for Ad Hoc Networks", ICNP, 2002.
- [18] W. Navidi and T. Camp, "Stationary Distributions for the Random Waypoint Mobility Model", IEEE Trans. On Mobile Computing, vol. 3, no. 1, pp. 99- 108, 2004.
- [19] J. Zhu and X. Wang, "PEER: A Progressive Energy Efficient Routing Protocol for Wireless Ad Hoc Networks", INFOCOM'05, Mar. 2005
- [20] J. Zhu, C. Qiao, and X. Wang, "On Accurate Energy Consumption Model for Wireless Ad-Hoc Networks", IEEE Trans. Wireless Comm., vol.5, no.11, pp.3077-3086, Nov. 2006.
- [21] J. Goodman et al., "Stability of Binary Exponential Backoff", app. In the Proc. of the 17-th Annual ACM Symp. Theory of Comp., Providence, May 1985.
- [22] J. R. Shoch and J. A. Hupp, "Measured performance of an ethernet local network," *Commun. ACM*, vol. 23, no. 12, pp. 711–721, Dec. 1980.
- [23] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *J. Select. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000
- [24] C. Hu, H. Kim, and J. Hou. "An Analysis of the Binary Exponential Backoff Algorithm in Distributed MAC Protocols", Technical Report, University of Illinois – Urbana Champaign, 2005.
- [25] K. Sakakibara, T. Seto, D. Yoshimura, and J. Yamakita, "On the stability of slotted ALOHA systems with exponential backoff and retransmission cutoff in slow-frequency-hopping channels," in *Proc. 4th Int. Symp. Wireless Personal Multimedia Communications*, Aalborg, Denmark, Sep. 2001.
- [26] Z. Fang, et al., "Performance evaluation of a fair backoff algorithm for IEEE 802.11 DFWMAC." International Symposium on Mobile Ad Hoc Networking & Computing
- [27] V. Bharghavan, et al., "MACAW: a media access protocol for wireless LAN's", Proceedings of the conference on Communications architectures, protocols and applications, 1994, pp. 210 – 225.
- [28] K. Jang, "A New Backoff Algorithm to Guarantee Quality of Service over IEEE 802.11 Wireless Local Area Networks," LNCS 298, pp 371-376, 2004.
- [29] J. Håstad, T. Leighton, and B. Rogoff, "Analysis of backoff protocols for multiple access channels," *SIAM J. Comput.*, vol. 25, no. 4, pp. 740–744, 1996.
- [30] S. Manaseer and M. Ould-khoua, "A New Backoff Algorithm for MAC Protocol in MANETs," 21st Annual UK Performance Engineering Workshop, pp 159-164, 2005.
- [31] J. Jang and M. Lee, "A Modified Backoff Algorithm to Improve Multiple Access Fairness for Optical Wireless LAN", ICACT, 2000.
- [32] K. Xu, et al., "How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks", IEEE Global Telecommunications Conference, 2002, Vol. 1, pp. 72 – 76.
- [33] F.Y. Li, M. Haugea, A. Hafslund, O. Kure, and P. Spilling, "Estimating Residual Bandwidth in 802.11-Based Ad Hoc Networks: An Empirical Approach," Proc. Seventh Int'l Symp. Wireless Personal Multimedia Comm. (WPMC '04), Sept. 2004.
- [34] S. Manaseer, M. Ould-Khoua, and L. Mackenzie, "Fibonacci Backoff Algorithm for Mobile Ad Hoc Networks", Liverpool John Moores University, PGNET 06, Liverpool, 2006.
- [35] J. Zhu and X. Wang, "Model and Protocol for Energy Efficient Routing over Mobile Ad-Hoc Networks", IEEE Trans. Mobile Computing., vol.10, no.11, pp.1546-1557, Nov. 2011.

IJERT