

# LIFETIME-AWARE MULTICAST ROUTING IN WIRELESS AD HOC NETWORKS

Morteza Maleki and Massoud Pedram

Dept. of EE-Systems, University of Southern California, Los Angeles, CA 90089

{morteza, pedram}@usc.edu

**Abstract.** *One of the main design constraints in mobile ad hoc networks (MANETs) is that they are energy constrained. Hence, routing algorithms must be developed to consider energy consumption of the nodes in the network as a primary goal. In MANETS, every node has to perform the functions of a router. So if some nodes die early due to lack of energy and/or the network becomes fragmented, then it will not be possible for other nodes in the network to communicate with each other. This paper presents a lifetime-aware multicast routing algorithm that maximizes the ad hoc network lifetime by finding routing solutions that minimize the variance of the remaining energies of the nodes in the network. Extensive simulation results are provided to evaluate the performance of the new routing algorithm compared to a number of different metrics and in comparison to a variety of existing multicast routing algorithms.*

## 1 INTRODUCTION

An ad hoc network is one where in all nodes work independently of any common centralized administrator. Each one of them performs the tasks of a router. They should be self-adapting in that if their connection topology changes, their routing tables should reflect the change. Also, since they are mobile, they largely run on finite energy sources (i.e., batteries.) This means that these nodes are power-constrained. Hence, it is an important design constraint for them to be power-aware or power-efficient. Furthermore, a node should not be greedy about its reserving its own power source since failure of some nodes in the network may result in lack of connectivity among nodes that are alive.

The primary goal of the conventional multicast routing protocols and algorithms has been to reduce the delay since most multicast applications tend to be delay sensitive audio/video broadcasting. Hence, most of the multicast routing protocols are designed to construct a multicast tree that minimizes the communication latency. Since the number of hops is a good heuristic metric for capturing this latency, a multicast tree with the minimum number of hops has been favored by most routing protocols ([1],[2],[3]). We call this tree the Minimum Hop-count Tree (MHT). In wireless ad hoc networks, there are two other criteria that make routing design an even more complicated task, i.e., mobility and power efficiency. The issue of mobility has extensively been addressed in the literature. In fact, the performance of multicast routing protocols has been evaluated in regard to their robustness to link failure due to the mobility ([1],[3],[4],[5]). However, there has been little work on developing a wireless multicast routing protocol in which power is a key objective or constraint. More precisely, although there have been some studies on the construction of energy-efficient broadcast and multicast tree in ad hoc networks ([6],[7]), most of these works require a global view of the network and cannot be applied in a distributed way whereby the nodes have only local knowledge.

This paper<sup>1</sup> addresses the problem of designing a lifetime-aware multicast routing protocol and algorithm that can be applied to an ad hoc network where nodes only have limited knowledge of the network topology and power states of other nodes in the network. The paper is structured as follows. Section 2 contains review of prior work in energy-aware anycast (unicast, multicast, and broadcast) routing in ad hoc networks. Section 3 describes the rationale and details of the proposed lifetime-aware multicast routing algorithm and protocol. Section 4 describes the simulation environment, the implementation, and the experimental results.

## 2 RELATED WORK

### 2.1 Energy-Aware Unicast Routing

Reference [8] proposes a routing algorithm based on minimizing the amount of power (or energy per bit) required to get a packet from a source to some destination. More precisely, the problem is stated as:

$$M_{\pi}^{in} \left\{ \sum_{(i,j) \in \pi} P_{ij} \right\} \quad (1)$$

where  $P_{ij}$  denotes the power expended for transmitting (and receiving) between two consecutive nodes,  $i$  and  $j$  (a.k.a. link cost), in route  $\pi$ . This link cost can be defined for two cases: a) when the transmit power is fixed; b) when the transmit power is varied dynamically as a function of the distance between the transmitter and intended receiver. For the first case, energy for each operation (receive, transmit, broadcast, discard, etc.) on a packet is given by [9]:

$$E(\text{packet}) = b \times \text{packet\_size} + c \quad (2)$$

where  $b$  and  $c$  are operation-specific coefficients. Coefficient  $b$  denotes the packet size-dependent energy consumption whereas  $c$  is a fixed cost that accounts for acquiring the channel and for MAC layer control negotiation.

The main disadvantage of the problem formulation of reference [8] is that it always selects the least-power cost routes. As a result, nodes along these least-power cost routes tend to “die” soon because of the battery energy exhaustion. This is doubly harmful since the nodes that die early are precisely the ones that are needed most to maintain the network connectivity (and hence useful service life). Therefore, it would be better to use a higher power cost route if this routing solution avoids using nodes that have a small amount of remaining battery energy. This observation has given rise to a number of “battery-cost lifetime-aware routing” algorithms ([10],[11]).

<sup>1</sup> This research was sponsored in part by DARPA PAC/C program under contract no. DAAB07-00-C-L516.

## 2.2 Minimum Energy Broadcasting

The main goal of minimum energy broadcasting is to reach from a specific source to all the other nodes in the network in a multihop transmission way with minimum total transmission energy assuming that nodes have variable transmission power. In an ad hoc network this may happen when we flood the network from a specific source. Since the main use of flooding is in route discovery it is important that flooding is done with minimum total energy. Minimum energy broadcasting has been addressed as NP-hard problem and there have been several works for finding heuristics for this problem ([6]).

## 2.3 Energy-Aware Multicast Routing

The goal of energy-efficient multicast routing is to reach a subset of nodes (one-to-many cast) that we will refer to as multicast receivers from a multicast source, such that we have maximum longevity of the paths between the source and the receivers. In general, finding a minimum energy multicast tree is equal to finding a minimum Steiner tree that is known to be an NP-hard problem ([12]). Two related works on the energy-aware multicast tree are as follows:

- (1) *Least-Cost shortest Path Tree (LPT)*: This is a tree obtained by superimposing all least cost paths (or shortest paths) between the source and each multicast receiver. The cost of each path can be calculated by equation 1 or equation 3.
- (2) *Multicast Incremental Power Tree (MIPT)* ([7],[13]): This tree is obtained from the Broadcast Incremental Power (BIP) tree proposed in [7]. The BIP algorithm consists of the following steps. For all nodes  $i$  in the tree and all nodes  $j$  which are not in the tree, we evaluate the incremental power cost of node  $i$  if node  $j$  is connected to the tree through  $i$ . This cost is defined as  $\rho'_{ij} = \rho_{ij} - \rho_i$ , where  $\rho_{ij}$  denotes the power level of node  $i$  if we add node  $j$  to the tree through  $i$  while  $\rho_i$  denotes the current power level of node  $i$ . Initially, the tree includes only the source node (i.e., the broadcast initiator node.) A pair  $(i,j)$  that results in the minimum value of  $\rho'_{ij}$  is chosen and node  $j$  is added to the tree through  $i$ . This procedure is repeated until all nodes are included in the tree. The MIPT is generated by pruning the broadcast (spanning) tree i.e. by eliminating all sub-paths that are not required to reach the multicast receivers. Reference [13] provides an improvement over [7] by dividing  $\rho'_{ij}$  by the remaining battery capacity of node  $i$ , thereby, calculating as the cost function for link  $ij$  in the tree a quantity that represents the *normalized lifetime loss* of node  $i$  if node  $j$  is included in the tree through  $i$ . In [13], the final cost function is obtained as the network lifetime loss to the power  $\alpha$ , where  $\alpha$  is a fixed parameter (See Section 3.3 for a detailed discussion of our proposed approach compared to [13].)

## 3 LIFETIME-AWARE MULTICAST ROUTING

### 3.1 Cost Function

The cost function that we adopt is the same as that is used in Power-aware Source Routing (PSR) [14], which is an on-demand source-initiated unicast routing algorithm that uses information about the state of the charge in battery sources of nodes in the network so as to maximize the network lifetime. More precisely, PSR solves the following problem to find a unicast route  $\pi_{s \rightarrow d}$  from source  $s$  to destination  $d$  at route discovery time  $t$ :

$$\text{Min}_{\pi} C(\pi_{s \rightarrow d}, t) = \sum_{i \in \pi_{s \rightarrow d}, i \neq s, d} C_i(t) \quad (3)$$

$$\text{where } C_i(t) = (1 + \rho_i) \cdot \left( \frac{F_i}{R_i(t)} \right)^{\alpha_i(t)} \quad (4)$$

$\rho_i$ : normalized transmit power of node  $i$

$F_i$ : full-charge battery capacity of node  $i$

$R_i(t)$ : remaining battery capacity of node  $i$  at time  $t$

$\alpha_i(t)$ : a positive weighting factor that increases with  $\frac{F_i}{R_i(t)}$

This cost function has two parts: one part captures the transmit power level, whereas the other captures the remaining battery capacity. The first term helps select a path with the minimum total energy consumed, whereas the second term helps balance power consumption over all nodes in the network as described next.  $\alpha_i(t)$  is inversely proportional to the ratio of the remaining capacity over the full-charge capacity of the battery source. As this ratio decreases and becomes less than a specified set of threshold values one at a time,  $\alpha_i(t)$  increases super linearly according to a fixed schedule. In this way, nodes with very low remaining battery capacity contribute a much higher value to the total path cost. In other words, when a path from source to destination has some nodes with very low residual battery, the cost of the path will be very high, and therefore, PSR will behave similar to the Max-Min battery cost routing. Note that  $\rho_i$  is normalized to the unicast (and/or broadcast) reception cost of a node.

### 3.2 Neighbor Cost Effect in Multicast Routing

Assume that a multicast tree from the source to several receivers has been constructed. The packet flow is coming out from source and is terminated at leaves of the tree where the receivers are located. We will refer to those intermediate nodes in the multicast tree that have more than one child in the tree as multi-fanout nodes (e.g., node A in Figure 1.) In ad hoc networks since the MAC layer does not have the ability of multicasting ([9]), there are two distinct methods to send out the packets from a multi-fanout node:

- (1) *Multiple unicast*: The parent node separately sends unicast packets to every child node in the multicast tree,
- (2) *Single broadcast*: The parent broadcasts the packets to all nodes in its immediate neighborhood (which may include nodes that are not in the multicast tree).

Reference [9] experimentally studied the power-optimal choice between these two methods. According to its results, the multiple unicast method results in much higher power consumption for the sender (i.e., the parent node in the multicast tree.) Based on these results, in our work, we adopt the single broadcast method at multi-fanout nodes.

It should be noted that in sending broadcast packets as opposed to sending unicast packets, there is no handshaking and acknowledgement in the form of *Request-To-Send/Clear-To-Send (RTS/CTS)* and *ACK* packets. Therefore, when using the single broadcast method, all the nodes that are in the radio range of the sender listen to the channel and receive the packet. As a result, non-destined nodes will unnecessarily consume power to receive the broadcast packet. In fact, these non-destined receiver nodes of broadcast packets realize at the routing (or IP) layer that they are not in the multicast tree that is identified in the IP header of received packets. Therefore, it is difficult for the physical layer to

filter out non-destined packets at IP layer. This kind of filtering has indeed not been implemented in current commercial WLAN adaptors. As a result, multi-fanout nodes will find the single broadcast method to be more beneficial to them from a power dissipation viewpoint. One must, therefore, consider the power consumption cost of all neighbors of nodes that broadcast packets when calculating the cost of a multicast tree in which multi-fanout nodes use a single broadcast method. This phenomenon, which we will refer to as the *neighbor cost effect*, makes the problem of finding a multicast tree with optimal cost quite complex.

Assuming that each node uses the cost function of equation (4), the complete cost of a multicast tree,  $C(T,t)$ , employing the single broadcast method at time  $t$  can be written as follows:

$$C(T,t) = \sum_{i \in T; i \neq s, i \notin Y} \left\{ (1 + \rho_i) \cdot \left( \frac{F_i}{R_i(t)} \right)^{\alpha_i(t)} + \right. \\ \left. \left( \text{if } \theta(i) \geq 2 \text{ then } \sum_{j \in \text{neigh}(i) \wedge j \notin T} \left( \frac{F_j}{R_j(t)} \right)^{\alpha_j(t)} \text{ else } 0 \right) \right\} \quad (5)$$

$\theta(i)$  : the degree of node  $i$  in  $T$  (including incoming and outgoing edges)

$\text{neigh}(i)$  : the set of nodes that are in the radio range of node  $i$

$s, Y$  : multicast source and set of multicast receivers

As described earlier, the cost function of equation (4) includes a power balancing part, which applies to all nodes that may consume power. Because of the neighbor cost effect, some nodes around the multicast tree consume power in their receiver parts. Equation (5) adds to total cost of the tree cost of the nodes that are not in the tree, but are in fact affected by the packet broadcasts at the multi-fanout nodes.

Another issue concerning the single broadcast method at multi-fanout nodes is that the farthest child from the parent determines the broadcast transmission power of that transmitting node. For example in Figure 1, the transmission power at node A is  $\text{Max}(\rho_1, \rho_2)$ . With considering the neighbor cost effect in multi-fanout nodes, the multicast routing problem would even be more challenging. Recall that finding a minimum energy-cost multicast tree without considering the neighbor cost effect is equivalent to that of finding a minimum Steiner tree which is NP-hard. As a result the problem of finding a lifetime-aware multicast tree with consideration of the neighbor cost effect is also an NP-hard problem.

There are many algorithms for finding a tree with near optimal cost ([15],[16]). Although it is possible to modify some of these algorithms to account for the neighbor cost effect at multi-fanout nodes, this approach is ill-advised in our context because these algorithms are too complex and require global information about the network connectivity graph in order to be applied. However, we are interested in finding solutions that can be deployed in an ad hoc network where nodes only have local knowledge about themselves and perhaps their neighboring nodes and must do the route discovery in a distributed, ad hoc manner (no global depository of information exists.) Furthermore, in ad hoc networks, the underlying network topology (connectivity graph) changes dynamically due to the mobility and link failure. Hence, ad hoc routing algorithms should be able to periodically update their routes. The routing update cost should be rather low.

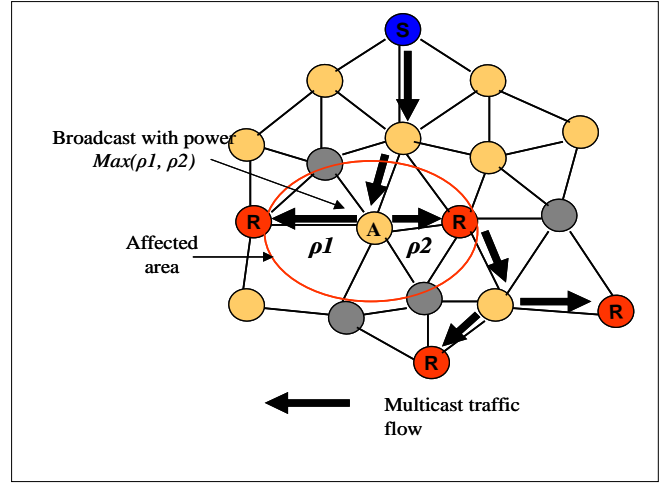


Figure 1: Neighbor cost effect in wireless networks.

### 3.3 Algorithm Design for Constructing a Lifetime-aware Multicast Tree

In this section, we present an algorithm for constructing a Lifetime-aware Multicast Tree (LMT) for multicast routing in ad hoc networks (with single broadcast at multi-fanout nodes of the tree.) We will show that this algorithm can be deployed in ad hoc environments with some flooding overhead. Flow of the LMT algorithm is as follows:

- (1) Using the cost function given by equations (3) and (4), find least-cost path between the source and each receiver in the multicast tree
- (2) Sort receivers in order of increasing path cost to the source
- (3) Include the least-cost path from the source to the first receiver in the sorted list as part of the LMT
- (4) Extend the LMT by finding the least-cost path from the existing LMT to the next receiver in the sorted list
- (5) Repeat step 4 until all receivers are connected to LMT

The rationale for connecting receivers with lower path costs to the source earlier than those with higher path costs (step 2) is that the paths included earlier in the LMT tend to carry (relay) more data traffic (because of step 4 in which subsequent receivers may connect to some intermediate node in the partially constructed LMT in order to establish their connection to the source). Step 4 is the key step in the LMT algorithm and is explained below.

The cost of a path,  $C(\pi_r, t)$ , from a multicast receiver,  $r$ , to a partially constructed multicast tree,  $T$ , at contact point  $j$  and at time instance  $t$  can be calculated as follows:

$$C(\pi_{r \rightarrow j}, t) = \sum_{i \in \pi_{r \rightarrow j}; i \neq r} (1 + \rho_i) \cdot \left( \frac{F_i}{R_i(t)} \right)^{\alpha_i(t)} + \sum_{k \in \text{neigh}(j) \wedge k \notin T} \left( \frac{F_k}{R_k(t)} \right)^{\alpha_k(t)} \quad (6)$$

$$C(\pi_r, t) = \text{Min}_{j \in T} C(\pi_{r \rightarrow j}, t) \quad (7)$$

In equation (6),  $\pi_{r \rightarrow j}$  denotes the set of nodes from receiver  $r$  (exclusive) to contact point  $j$  (inclusive).  $neigh(j)$  consists of all nodes that are in the transmit range of node  $j$  which is determined by the farthest child of node  $j$  (which is on  $\pi_{r \rightarrow j}$ ). The normalized transmit power for node  $j$  (the contact point) is recalculated to be included in equation (6) if node  $j$  has to increase its power level to connect receiver  $r$  through path  $\pi_{r \rightarrow j}$  to the tree.

Notice that compared to [13], our proposed multicast tree construction algorithm is more efficient and can be easily deployed in a lifetime-aware multicast routing protocol (see below.) More precisely, at each step in the LMT algorithm, we only calculate the least cost path from the set of unconnected multicast receivers to the partially constructed tree  $T$ . In contrast, at each step of the MIPT algorithm, we ought to calculate the cost of all links between nodes already included in  $T$  and those that are not included in order to identify the lowest cost link. These modifications in tree construction results in a significant performance improvement for the LPT over MIPT when the number of multicast receivers is small compared to the total number of nodes in the ad hoc network. In addition, [13] uses a fixed value for  $\alpha$  and does not consider the neighbor cost effect.

### 3.4 Protocol Design for Constructing a Lifetime-aware Multicast Tree

In this part we propose a high-level routing protocol for LMT. We have selected an on-demand approach that has already been deployed in ODMRP ([1]). In ODMRP group membership and multicast routes are established and updated by the source. While a multicast source has data to send, it floods the networks with Join-Request (JREQ) packets. Flooding is like asking every node in the network to find a group of multicast receivers. When a node receives a non-duplicate JREQ, it rebroadcasts it in the network. When a multicast receiver receives the JREQ packet it replies back to the source. The result of flooding and reply-back procedure is that a multicast tree rooted at the source is constructed. However, the metric for constructing this tree in [1] is the number of hops. In other words, this is a least-cost path tree with respect to the number of hops between the source and each of the multicast receivers. The reason is that during the flooding process, every node simply rebroadcasts only the first arriving JREQ packet and drops any other copies of the packet that arrive later and from other paths. In addition, receivers immediately reply back to the first arriving JREQ packet. The first arriving JREQ has generally traversed the path that has least number of hops because that path usually has the lowest delay. In ODMRP the source periodically floods the network to refresh the membership information and update the routes and thereby the multicast tree.

The process of finding the least-cost path tree is similar to that used in ODMRP except that we use cost function given in equations (2) and (3). Furthermore, during the flooding process, every node may pass on several copies of the JREQ packet from a multicast source. In particular, every node passes the first arriving JREQ and then turns on a timer whose count is proportional to the number of hops in the first arriving JREQ. All JREQ's will be passed through this node until the timer expires after which the node blocks subsequent JREQ's from the multicast source. As long as the node timer is not expired, if the node receives a new copy of a JREQ packet, it examines the cost in the header of JREQ packet. If that cost is less than the cost of a previous copy

of that JREQ packet that has already passed through the node, then it will pass on the new copy as well; otherwise, it will drop it. Furthermore, when a node passes on a JREQ, it adds its own cost to the header of the packet so that the cost of a packet is updated as it traverses along a path. Multicast receivers also have their local timers and reply back on (i.e., select) the path that has the least-cost when their timers expire. To implement the LMT algorithm in an ad hoc network, a number of steps must be followed as described below:

- (1) The source of multicast tree floods the network to find a least-cost path tree to all multicast receivers. This process has already been explained. One more note is that the multicast receivers add the cost of the path that they select to the header of the reply-back packet before sending this packet to the source.
- (2) The source sorts the receivers in increasing order of their respective path costs and sends a path confirmation packet to the receiver whose path has the least cost. Next, it removes that receiver from the sorted list and initializes the multicast tree,  $T$ , to consist of the least-cost path from the source to that receiver.
- (3) The source sends a FLOOD command to the next receiver in the sorted list and asks this receiver to flood the network and search for any of the nodes in  $T$ .
- (4) The receiver that receives a FLOOD command from the source, starts flooding the network searching for the nodes in  $T$ . This is equivalent to starting another source initiated flooding in multicast routing where this flooding source is the receiver under considerations and the receivers are all the nodes in  $T$ .
- (5) Every node that is in  $T$  replies back to the flooding receiver along the least-cost path from that node to the flooding receiver while accounting for the neighbor cost. Note that nodes can easily obtain neighbor information, including number of their neighbors, from the MAC layer.
- (6) The flooding receiver chooses the path with least cost to some node in  $T$ . It then sends information about the subpath that must be added to  $T$ , to the multicast source.
- (7) The multicast source updates  $T$  to include the new subpath to the flooding receiver, removes that receiver from the sorted list and repeats steps 3 through 6 until all receivers in the multicast group are connected to  $T$  (the list becomes empty.)
- (8) The proposed LMT protocol optimizes the transmit power level of each node of the multicast tree after  $T$  has been constructed in the way that each node in the LMT will determine its own minimum transmit power level based on the distance of its farthest child node in the tree.

Clearly the LMT protocol causes an increase in the number of flooding procedures, which is proportional to the number of receivers in the multicast group (cf. step 4 above.) However, the protocol works particularly well for a scenario with the "hard state" method as follows. Initially a small set of multicast receivers exists and the source forms an LMT tree according to the abovementioned protocol. When a new multicast receiver arrives, it should join the tree by following steps (4) through (6) except that the source does not need to send any FLOOD command to the new receiver as described in step (4) and the new



receiver starts flooding the network to find a node as the contact point to the tree. Recall that in the “hard state” join/leave method, there is no periodic refresh to update the multicast tree. Instead nodes send explicit commands to join or leave the tree whenever they want to do so. Furthermore, with in the “soft state” join/leave method, if the number of initial receivers is large, then as the size of the multicast group increases, the network lifetime gain of the LMT increases (cf. Section 4). Finally, flooding is done by sending small control packets. There are ways to reduce the overhead of flooding process ([17],[18]).Therefore, the cost of the flooding processes can be made small.

## 4 EXPERIMENTAL RESULTS

### 4.1 Simulation Setup

To simulate our routing algorithms, we developed an event-driven simulator (called LRSim for Lifetime-aware Routing Simulator) in C++. The block diagram of the simulator is shown in Figure 2. LRSim is a high-level simulation environment that can implement any kind of routing algorithm without implementing the MAC layer. Nodes are randomly distributed in an area and their locations are given as input to LRSim. For our simulations, we used 50 nodes, which are uniformly located in a 1000 by 1000 m<sup>2</sup> area. The nodes were randomly chosen as a source or a receiver member of a multicast connection. 100 multicast connections with random duration and random starting times were established during the simulation time, which is 20,000 secs. Connections had CBR (Constant Bit Rate) traffic with the rate of 3 packets per sec. Each node was randomly assigned an initial energy, which varied between 1500, and 3000 units of energy. The nodes have four different transmission power levels for four radio ranges: 150, 125, 75, and 50 meters. The energy cost for transmitting and receiving a packet are as equation (2) where the coefficients (*a* and *b*) are taken from [9] which were experimentally measured. The coefficients for different states (and for transmit range of 150) are as follows:

Unicast send	1.9	x	packet_size+454	( $\mu$ W.sec)
Broadcast send	1.9	x	packet_size+266	( $\mu$ W.sec)
Unicast receive	0.5	x	packet_size+356	( $\mu$ W.sec)
Broadcast receive	0.5	x	packet_size+56	( $\mu$ W.sec)

In transmit mode, value of *a* varies as a function of transmission power level or range ([20]) and accordingly the above coefficients are calculated for other transmission ranges (125, 75 and 50 meters.) In each simulation run, some routing algorithms are simulated and finally the results of different simulation runs are analyzed and compared. After each packet generation event, the remaining energies of the nodes participating in data transmission are decremented to reflect the cost of transmitting, relaying or receiving the packet. When the remaining energy of a node reaches zero that node is considered dead for the rest of the simulation and does not participate in any subsequent transmission or reception.

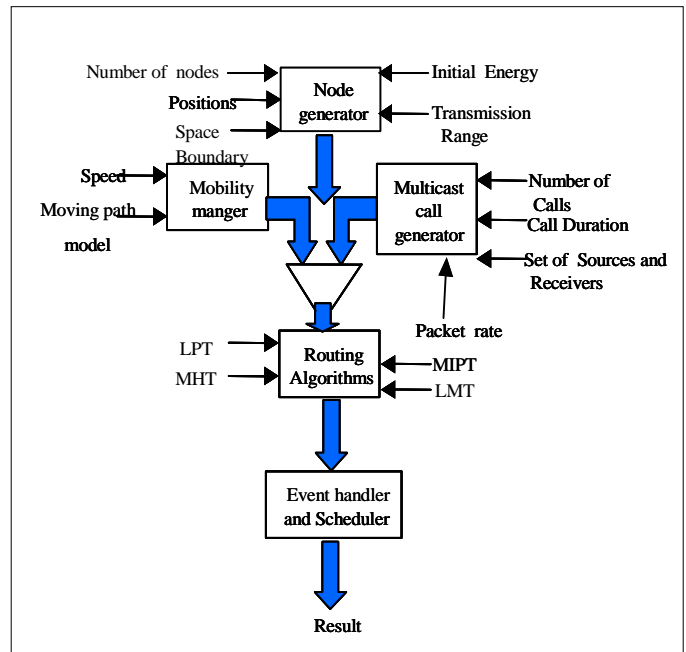


Figure 2: LRSim flow diagram.

### 4.2 Simulation Results

We have done several simulations and have measured the effect of a number of different multicast routing algorithms on the *network lifetime*, the *RMS value of remaining energy*, the *packet delivery ratio*, and *energy consumption per transmitted packet*. The performance of the minimum hop-count tree (MHT), the multicast incremental power tree (MIPT), the least-cost path tree (LPT), and the lifetime-aware multicast tree (LMT) have been evaluated and compared with respect to the abovementioned metrics. We have selected for our experimental results a variant of the MIPT algorithm as proposed in reference [13], where the MIPT cost function is proportional to the normalized lifetime loss. For a fair comparison of MIPT and LMT, the value of  $\alpha$  that has been chosen for MIPT is equal to the average value of  $\alpha(t)$  during the lifetime of each node.

We have applied an on-demand approach, which periodically refreshes the multicast tree. The timing period of refresh in our simulation was set to 50 sec. In terms of the cost function, MHT uses the hop-count metric whereas MIPT, LPT and LMT rely on the cost function given in equation (4). We can state that the performance of MHT is the same as that of the ODMRP.

We may define the network lifetime as the total elapsed time from the state of full battery charge for all nodes in the network to a state in which a fixed number (or percentage) of the nodes in the network die due to energy source exhaustion. Figure 3 shows the ratio of a number of nodes that are alive to the total number of nodes during part of simulation time at different time instances (when multicast group size is 15.) As can be seen, in MHT nodes start dying out sooner but with a rather uniform rate. In LMT, LPT and MIPT, the nodes start dying later but die more rapidly. In terms of the network lifetime as will be shown, LPT performs better than MHT but worse than MIPT and LMT for large group size. Although the initial multicast group size was set to 15 in this

simulation, when some nodes die out, multicast connection start or continue with less group size. As will be shown subsequently, the multicast group size has a significant impact on the performance of different routing algorithms.

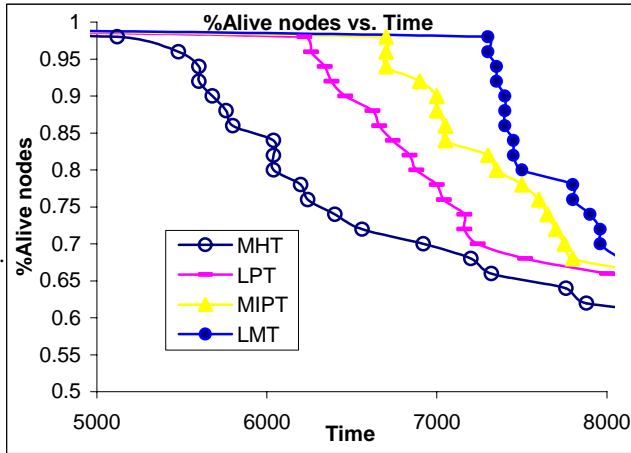


Figure 3: Ratio of alive nodes to total number of nodes vs. time.

Maximizing the network lifetime as defined above is roughly equivalent to minimizing *the variance of the residual energy of the network nodes*. Figure 3 is not very expressive in this regard. A histogram of the snapshots of the energy consumption in each of the nodes at different time instances is more informative in this regard. One way to present such histograms is to report the RMS value of the consumed energy ( $E_{RMS}$ ) at different time instances as in [19]. Figure 4 shows the evaluation of  $E_{RMS}$  as a function of time for different multicast routing algorithms. In this figure, the  $E_{RMS}$  values are reported at time instances before any node dies out in any of the routing algorithms.

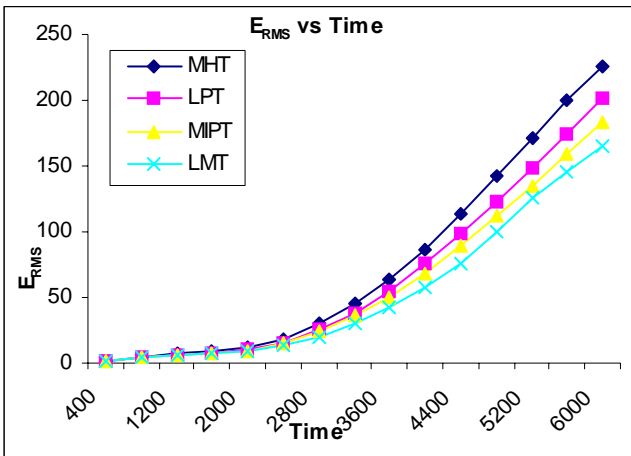


Figure 4: ERMS value as a function of time.

Let's define the *connectivity ratio* as the ratio of the cardinality of the largest connected component in the network graph to that of the initial (connected) network graph. We may then define the network lifetime as the total elapsed time from the state of network connectedness to a state in which the network

connectivity ratio drops to 40%. We chose 40% because we observed that near this percentage, the network graph becomes so disconnected that it cannot be considered as functional anymore. Of course, any other ratio may be used, depending on user preferences. It should be noted that because multicast sources and receivers have been distributed uniformly over all nodes of the network graph, disconnection in the network graph results in disconnectedness of most of the remaining multicast connections. We run the simulations for a large number of cases in which the node locations were uniformly distributed in the area. Figure 5 shows the network lifetime based on this definition as a function of the number of multicast receivers.

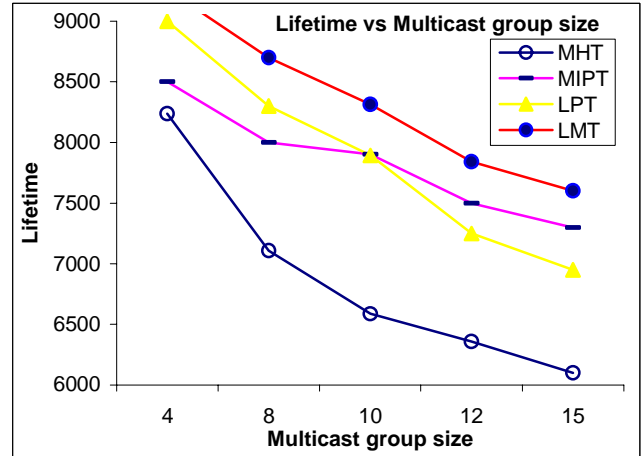
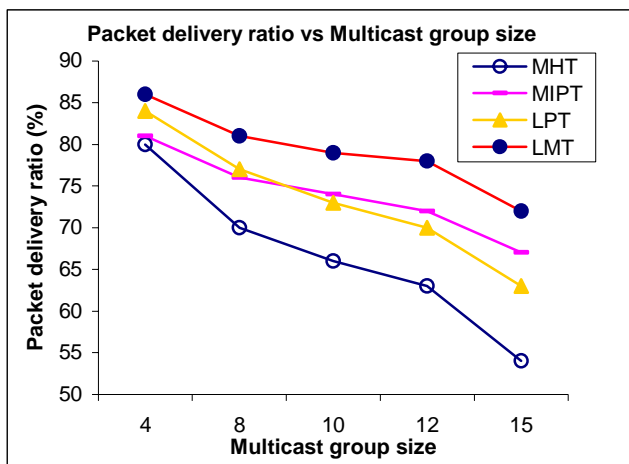


Figure 5: Lifetime for various multicast group sizes where lifetime is defined as from time 0 until network connectivity ratio drops to 40%.

From these results we can conclude that when the multicast group size is small, algorithms that are based on broadcast and prune (such as MIPT) exhibit poor performance. Similarly, MHT has poor performance. However, LPT does better in this case. We believe this is the case because LPT produces a good approximation to optimal cost tree, which is the minimum Steiner Tree when the number of multicast receivers is small. On the other hand, when the multicast group size grows, MIPT exhibits a higher performance compared to MHT whereas, in relative terms, the performance of LPT degrades. In both cases of small and large multicast group size, LMT delivers the best results in terms of the network lifetime. Furthermore, its performance gain over other algorithms increases as the group size increases. The reason is that as the group size increases, the multicast tree becomes fatter (i.e., the number of multi-fanout nodes in  $T$  increases), resulting in a larger number of local broadcasts along the tree, which in turn increases, the neighbor cost effect. Notice that only LMT correctly accounts for the neighbor cost effect.

Packet delivery ratio is defined as the number of delivered data packets to the number of generated data packets in all nodes. We generate as many as 300,000 data packets during the simulation. They are generated between random multicast source and receiver groups at random times. Many of these might not have reached their intended destination due to the lack of existence of a route between the source and target destination and this may occur

when some nodes die out. For each transmitted (or generated) packet, by a specific source in a multicast connection, the delivery ratio of that packet is defined as the ratio ( $|Y_s| / |Y_{max}|$ ) where  $|Y_s|$  is the number of reachable (or alive) multicast receivers at the time of transmission of that packet and  $|Y_{max}|$  is the maximum number of multicast receivers for the specific connection. The total packet delivery ratio is calculated by averaging the delivery ratio of all transmitted packets for all multicast connections. Obviously, the network lifetime influences this ratio. If the network was alive for a longer period of time, then more data traffic will go through. Figure 6 shows the packet delivery ratio for different multicast group sizes. The expectation is that all of the multicast receivers in each connection receive all the packets generated by the source of that connection. We assumed that after refreshing a multicast tree connection (which is done periodically), source of that connection checks for the reachability of its multicast receivers group. If some receivers are not reachable, then this means that we are not be able to deliver packets that are generated from that time until end of the connection to those receivers. Since the packet delivery ratio in our setup is only affected by node failure, it is a function of the network lifetime (which defined according to the number of dead or alive nodes), which in turn means longer lifetime results in a higher delivery ratio. This figure confirms this expectation and may be used to validate/justify our definition of the network lifetime. As the group size increases, the delivery ratio decreases since more nodes participate in the multicast tree connection and the probability that some receivers in each connection become unreachable increases.



**Figure 6:** Packet delivery ratio as a function of multicast group size.

## 5 CONCLUSIONS

In this paper, we described LMT, a lifetime-aware multicast routing algorithm for ad hoc networks that maximizes the network lifetime by finding routing solutions that minimize the variance of the remaining energies of the nodes in the network. Extensive simulation results were provided to evaluate the performance of LMT with respect to a number of different metrics (i.e., two definitions of the network lifetime, the RMS value of remaining

energy, the packet delivery ratio, and the energy consumption per transmitted packet) in comparison to a variety of existing multicast routing algorithms (i.e., MHT, MIPT, and LPT). These results clearly demonstrate the effectiveness of LMT over a wide range of simulated scenarios.

## 6 REFERENCES

1. S.-J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," *IETF Internet Draft*, draft-ietf-manet-odmrp-02.txt, Jan. 2000.
2. J. J. Aceves and E. Madruga, "The Core-Assisted Mesh Protocol," *IEEE JSAC*, vol. 17, no. 8, pp. 1380-1394, Aug. 1999.
3. E. Royer and C. Perkins, "Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing," *IETF Internet Draft*, draft-ietf-manet-maodv-00.txt.
4. S.-J. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, "A Performance Comparison Study of Ad Hoc Wireless Multicast Protocols," *Proc. of Infocom*, pp. 565-574, Mar. 2000.
5. M. Gerla, C.-C. Chiang, and L. Zhang, "Tree Multicast Strategies in Mobile, Multihop Wireless Networks," *ACM/Kluwer Mobile Networks and Applications*, vol. 4, no. 3, pp. 193-207, 1999.
6. M. Cagalj, J-Phubaux and C.Enz, "Minimum-Energy Broadcast in All-Wireless Networks: NP-Completeness and Distribution Issues," *Proc. of Mobicom*, pp. 172-182, Sep. 2002.
7. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-Efficient Broadcast and Multicast Trees in Wireless Networks," *Mobile Networks and Applications (MONET)*, vol. 7, no. 6, pp. 481-492, Dec. 2002.
8. S. Singh, M.Woo and C.S. Raghavendra, "Power-Aware Routing in Mobile Ad hoc Networks," *Proc. of Mobicom*, pp. 181-190, 1999.
9. L.M. Freney and M. Nilsson, "Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment," *Proc. of Infocom*, pp.1548-1557, Apr. 2001.
10. C.K. Toh, "Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad hoc Networks," *IEEE Communication Magazine*, pp. 138-147, June 2001.
11. Misra and S.Banerjee, "MRPC: Maximizing Network Lifetime for reliable routing in Wireless Environments," *Proc. of Wireless Communications and Networking Conf. (WCNC)*, pp. 800-806, Mar. 2002.
12. M. Imase and B. Waxman, "Dynamic Steiner Tree Problem," *Journal of Discrete Math*, vol. 4, no. 3, pp. 369-384, Aug. 1991.
13. J.E. Wieselthier, G.D. Nguyen and A. Ephremides, "Resource management in energy-limited, bandwidth-limited, transceiver-limited wireless networks for session-based multicasting," *International Journal of Computer and Telecommunications Networking*, vol. 39, no. 2, pp. 113-131, June 2002.
14. M. Maleki, K. Dantu and M. Pedram, "Power-aware Source Routing in mobile ad hoc networks," *Proc. of ISLPED*, pp. 72-75, Aug. 2002.
15. M. Parsa, Q. Zhu and J.J. Garcia-Luna-Aceves, "An Iterative Algorithm for Delay -Constrained Minimum-Cost Multicasting," *IEEE/ACM Trans. on Networking*, vol. 6, no.4, pp. 461-474, Aug. 1998.
16. J. Cong, A. B. Kahng, G. Robins, M. Sarrafzadeh, and C. K. Wong, "Provably Good Performance-driven Global Routing," *IEEE Trans. Computer-Aided Design*, vol. 11, pp. 739-752, 1992.
17. Y. Ko and N. H. Vaidya, "Location-Aided Routing (LAR) Mobile Ad Hoc Networks," *Proc. of Mobicom*, pp. 66-75, Oct.1998.
18. Y. Ko and N. Vaidya, "Geocasting in Mobile Ad Hoc Networks: Location-Based Multicast Algorithms," *Proc. of Workshop on Mobile Computing Systems & Applications (WMCSA)*, Feb. 1999.
19. M. Maleki, K. Dantu and M. Pedram, "Lifetime Prediction Routing in Mobile Ad Hoc Networks," *Proc. Wireless Communications and Networking Conf.(WCNC)*, Mar. 2003.
20. W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan,, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *Proc. of the 33rd Annual Hawaii International Conference on System Sciences(HICSS)*, pp. 3005-3014, Jan. 2000