

Energy-Aware Self-Stabilization in Mobile Ad Hoc Networks: A Multicasting Case Study¹

Tridib Mukherjee, Ganesh Sridharan, and Sandeep K. S. Gupta
The IMPACT Laboratory
School of Computing and Informatics
Arizona State University, Tempe, AZ, USA
<http://impact.asu.edu>

Abstract

Dynamic networks, e.g. Mobile Ad Hoc Networks (MANETs), call for adaptive protocols that can tolerate topological changes due to nodes' mobility and depletion of battery power. Also proactivity in these protocols is essential to ensure low latency. Self-stabilization techniques for distributed systems provide both adaptivity and proactivity to make it suitable for the MANETs. However, energy-efficiency - a prime concern in MANETs with battery-powered nodes - is not guaranteed by self-stabilization. In this paper, we propose a node-based energy metric that minimizes the energy consumption of the multicast tree by taking into account the overhearing cost. We apply the metric to Self-Stabilizing Shortest Path Spanning Tree (SS-SPST) protocol to obtain energy-aware SS-SPST (SS-SPST-E). Using simulations, we study the energy-latency tradeoff by comparing SS-SPST-E with SS-SPST and other MANET multicast protocols, such as ODMRP and MAODV.

1 Introduction

Mobile ad hoc networks (MANETs) consisting of mobile hosts which communicate using only wireless links are being increasingly used for local area networks, law enforcement, and military operations. One of the main challenges in developing any protocol for MANETs or any Wireless Ad Hoc Networks (WANETs) is to counter the dynamic nature of these types of networks [30] [22]. In MANETs the network topology changes due to node movements, variations in the radio propagation conditions, and depletion of nodes' battery power. Therefore, adaptation to the topological changes is necessary to design any protocol for MANETs. Self-stabilization scheme can serve this pur-

pose as it provides adaptivity in a proactive and distributed manner. Self-stabilization, proposed by Dijkstra [8] [9], stabilizes a distributed system automatically from any arbitrary initial state based on local actions in the distributed nodes. Thus, there is no concept of initial states in a self-stabilizing system. Moreover, self-stabilization is used as a fault tolerance scheme [5] as it converges to a valid system state from any faulty state. Topological changes in MANETs can be thought of as faults and thus self stabilization has the ability of adaptation in a proactive manner to topological changes.

Multicasting is an important communication primitive for many applications in MANETs, such as distributed games and teleconferencing [6]. One of the efficient distribution structure to achieve multicasting is multicast tree [15] [19]. A node in MANET has serious resource constraints, such as battery and bandwidth, so multicast tree should be as energy-efficient as possible [10]. Due to the fact that the problem of constructing the optimal energy-efficient broadcast/multicast tree is NP-complete in WANET (WANET can be viewed as a special case of MANET without node's mobility), several heuristic protocols for building energy-efficient multicast tree have been developed [4][17], such as BIP/MIP [28], EWMA [2], S-REMiT [24] and G-REMiT [25]. All of these broadcast/multicast tree construction protocols need initialization phase, which could reduce the availability of multicasting service. However, initialization phase can be avoided by using self-stabilization technique, such as self-stabilizing Shortest Path Spanning Tree (SS-SPST) [13] and self-stabilizing Minimum Spanning tree (SS-MST) [14].

In this paper, we study the problem of energy-aware self-stabilization multicasting in MANETs. To the best of our knowledge, this is the first work on this topic. We propose a node-based energy metric, which takes into account not only the energy expended on the multicast tree but also the energy wasted in the network due to overhearing [1, 7]. Further, we apply the node-based energy metric to get SS-

¹This research was funded by NSF, grant number ANI-0196156.
1-4244-0910-1/07/\$20.00 ©2007 IEEE.

SPST-E (Energy aware SS-SPST) protocol, so that the multicast tree is energy-efficient and adapts to the topological changes with higher availability in MANETs.

We compare the performance of SS-SPST-E with other MANET multicast protocols like Multicast Operation of the Ad Hoc On-Demand Distance Vector Routing Protocol (MAODV) [20] and On-Demand Multicast Routing Protocol (ODMRP) [12]. Our simulation results show that the energy consumption is less in SS-SPST-E compared to other protocols. Energy savings has a “cost” in terms of increased average packet delivery delay for SS-SPST-E compared to SS-SPST; however, this delay is less than other traditional protocols like MAODV and ODMRP.

2 Motivation

The existing self-stabilizing multicast protocols for the MANETs use the hop metric to build multicast trees. However, the hop metric is not the right metric for the MANETs as it does not capture the actual energy consumption of the mobile nodes. In this paper, we propose an energy metric for energy-efficient maintenance of a multicast tree. We also want to keep the advantages of the self-stabilization paradigm which allows adaptation under changing topology. Thus, we can have an energy-aware adaptivity in the maintenance of a multicast tree.

SS-SPST algorithm constructs a multicast tree by minimizing the number of hops from the root node to the leaf nodes. The tree is maintained in a proactive manner and is re-organized upon occurrence of any topological changes. Proactivity is achieved by periodically broadcasting beacon messages with the local information about the tree. All the nodes check the tree status based on the information available in the beacons received from the neighbors. Upon occurrence of a topological change, a node takes necessary actions locally to re-organize the SPST for the changed topology. The tree is built top-down: the root node stabilizes first followed by the nodes at the next higher level and so on until the leaf nodes are stabilized to form the entire tree.

The next logical step is to prune the tree so that messages are not forwarded to non-group members. SS-SPST achieves this by maintaining a flag for each node. The flag becomes true if there is a member in the downstream of the node. This information is gathered in a bottom-up manner from the leaf node to the root node. Flag becomes false if there is no member in the downstream and that portion of the tree is then logically pruned.

Example 1: Figure 1 shows the initial connections of a mobile network. Figure 2 shows the stabilized multicast tree using the hop count metric to construct a SPST. SS-SPST protocol takes 3 rounds (also known as the beacon intervals) to stabilize in this case. First round stabilizes the root followed by consecutive levels in the next rounds. □

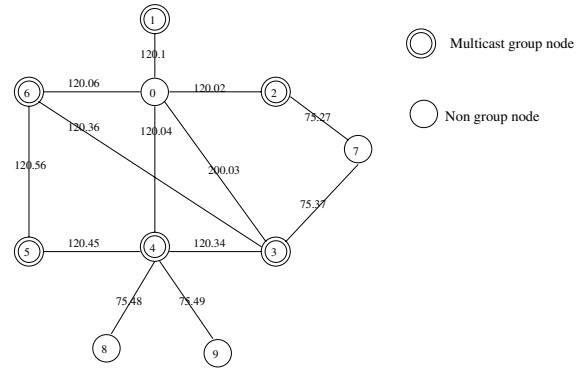


Figure 1. Original Network : Neighboring nodes are connected using straight lines. Node 0 is the multicast source node. The edge weights denote the distance between the two nodes connecting it.

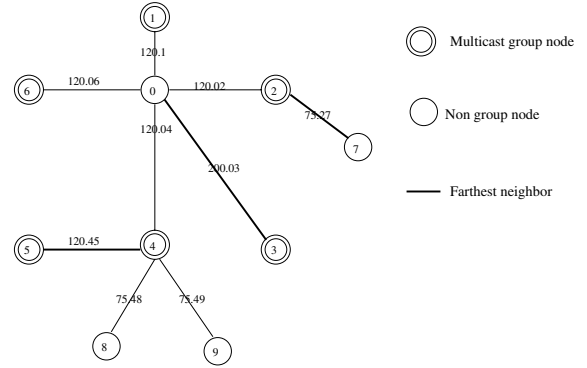


Figure 2. SS-SPST : The tree is stabilized by minimizing the hop count from the source.

The usual time-complexity measure for self-stabilizing algorithms is in terms of *rounds* [11]. In case of self-stabilizing multicast protocols, a round is defined to be the time period in which each node in the system receives at least one beacon message from each of its neighbors and performs computation based on its received information.

But, in spite of having the advantages of adaptivity and low latency due to pro-activity, there are certain drawbacks in the SS-SPST algorithm that make it unfeasible for the MANETs. First and foremost, the hop count, as mentioned earlier, is not a good metric for the MANETs since it does not properly capture the energy cost. As energy should be minimized for the MANETs, it may be necessary to increase the hop count to minimize the energy consumption. Additionally, increase in the hop count can decrease the performance of the protocol since as the number of transmissions and receptions increases the end-to-end delay along with the probability of message being lost also increases. Thus, we need a cost metric which is both energy-efficient

and has a minimal impact on a protocol's performance.

3 System Model

We assume that each node in the MANET is identified by a unique identifier. When a node transmits (*broadcasts*) a message, the nodes in its *coverage area* can (almost) simultaneously hear the message. This assumption is valid since the broadcast nature of the wireless medium provides the *wireless multicast advantage* that we want to cash on. Moreover, all the nodes have power control capabilities to vary their transmission range [16]. T_{ij} denotes the minimum transmission energy required for a packet over the link between nodes i and j .

We also assume that the reception energy is constant for all the nodes. Inclusion of transmission-power dependent reception energy consumption [23] is left for future work. Each node sends its link and node characteristics in the beacon message and all the neighbors calculate the energy cost of the link to the node based on the information available in the beacon. The tree is constructed based on the values calculated in all the nodes. When beacon is not received from a node, all the neighboring nodes sense a disconnection of the node and re-organizes the tree according to a self-stabilization protocol. In essence, any disconnection in the tree is treated as a fault in the system.

All the nodes in the system actively participate in the beaconing. Due to the broadcast nature of the wireless medium, even the non-group nodes expend energy in listening the multicast messages (called *overhearing* [1, 7]) and discard it. We define this wasted energy as the *discard energy* and intend to minimize it in our energy metric.

4 Cost Metrics

According to the nature of the energy cost factor we can broadly classify the cost either as a *link energy cost* or as a *node energy cost*. Link energy costs capture the link variants such as error rate, channel bandwidth and transmission power while node energy costs capture the node variants such as energy level and node load. Before coming up with an energy cost metric for SS-SPST, let us first discuss the issues in employing the energy metrics in the literature.

In [21], [26], and [18] authors have proposed link-based metric which captures the transmission energy spent by the node to maintain a link. By assigning transmission energy expended to the links, SS-SPST can build a tree which minimizes the total transmission energy. Let C_{ij} be the total energy cost of link between i and j . Then we have,

$$C_{ij} = T_{ij}. \quad (1)$$

We name the SS-SPST with the above metric to be SS-SPST-T, where 'T' denotes the transmission cost.

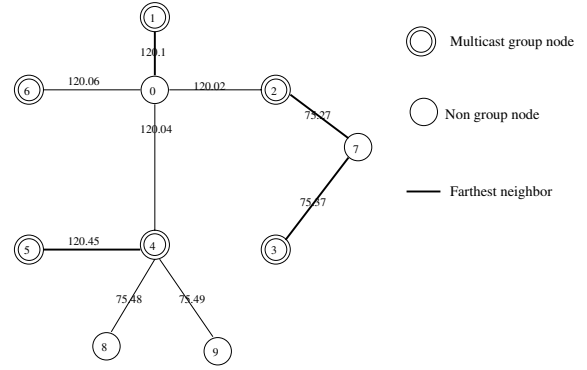


Figure 3. SS-SPST-T - Transmission energy metric.

Example 2: In our topology of Figure 1, when we apply SS-SPST-T algorithm, the resulting tree is constructed as in Figure 3. It is more energy efficient if node 3 makes node 7 as its parent instead of node 0. The protocol takes one more round to reconstruct the tree in this way. So a total of 4 rounds are needed for stabilization. \square

However, this metric does not capture the energy expended in real scenarios as each link is not required to spend separate transmission energy because of the multicast advantage of the wireless medium.

Node based cost metric are considered in [28], [27] and [3] where the authors propose schemes (such as BIP, and MIP) to build energy-efficient broadcast and multicast trees in wireless networks. These schemes take the maximum transmission energy of the transmitting node as the node metric. There are various reasons the transmission energy can vary. For simplicity in our examples, we use the distance as the factor of varying transmission energy. In general, we consider the transmission energy as the cost of a node to transmit to a neighbor. Applying this method to SS-SPST, energy to reach the costliest neighbor is counted as the energy cost of transmission.

Let j be the node which is a neighbor of node i and is connected to node i by a link which requires node i to expend maximum transmission energy. We call node j to be the costliest neighbor of node i and the transmission energy for node i to reach node j is T_{ij} . Note that this does not mean that i is the costliest neighbor of node j . The concept of costliest neighbor is not symmetric. This is another reasoning advocating the use of node based cost metric instead of the link based ones. Let t_i be the total number of tree neighbors of node i , R be the reception energy cost, and C_i be the total energy cost of node i . Then we have,

$$C_i = T_{ij} + t_i \times R, \quad \text{where } j \text{ is costliest neighbor} \quad (2)$$

We name the SS-SPST implementation with the above cost metric as SS-SPST-F, where 'F' signifies the costliest (far-

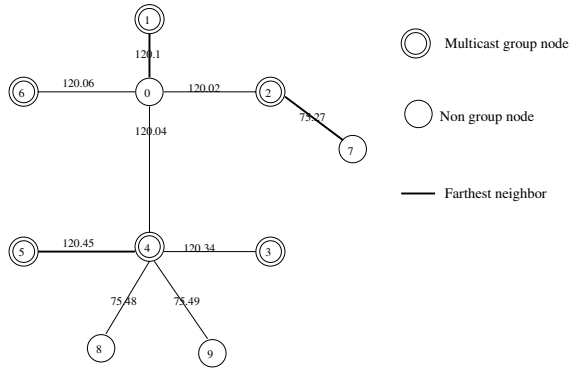


Figure 4. SS-SPST-F - Costliest group node transmission energy.

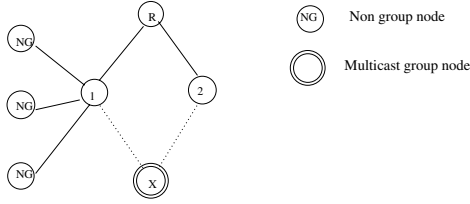


Figure 5. Discard energy wastage.

thest) node metric.

Example 3: Figure 4 shows the tree constructed by applying SS-SPST-F algorithm to topology in Figure 1. After the first three rounds, node 3 selects node 4 as its parent instead of node 0; since the incremental cost of joining node 4 is very low as node 5 is the costliest neighbor of node 4. After node 3, which is a former costliest child of node 0, changes its parent in round 4, node 1 in the transmission range of node 0 becomes the costliest child. The costs are adjusted based on the new costliest neighbor in round 5. Thus the protocol takes five rounds to stabilize. \square

Node based metrics characterize the energy expended in a better way than a link based metric, however, none of the energy metrics in the literature consider the discard energy discussed earlier.

Example 4: Consider the example in Figure 5. Assume node X wants to join the multicast group and it has to make decision between node 1 and node 2 as its parent node in the multicast tree. Further, assume that the costs to reach the root node R through both the nodes 1 and node 2 are the same. In this example, total energy consumed will be reduced if node X chooses node 2 as its parent since node 1 has three non-group nodes as its neighbor and every time node 1 transmits, those three nodes will have to spend energy to receive the packet and discard it. Our energy cost metric tries to minimize this discard energy. \square

To take into account discard energy, we modify the ‘‘F’’ metric as follows. Let, L_i be the discard energy spent by nodes to discard a packet when node i transmits to its costli-

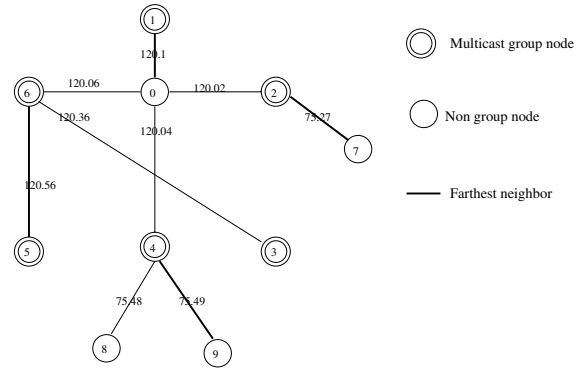


Figure 6. SS-SPST-E - Farthest group node transmission energy with discarding energy.

est neighbor node j . Let N_i be the total number of neighbors of node i . Now the total discard energy for node i , L_i can be presented as follows

$$L_i = R \times (N_i - t_i) \quad (3)$$

We can now represent our energy metric C_i as follows

$$C_i = T_{ij} + L_i + t_i \times R, \text{ where } j \text{ is } i\text{'s costliest child} \quad (4)$$

We name the protocol with the proposed energy cost metric, given above, as SS-SPST-E. From the above equation we can see that the source node is assigned an energy cost based on its current transmission power and the wastage energy of the nodes in this transmission range.

Example 5: Finally, Figure 6 shows the multicast tree with SS-SPST-E, our proposed protocol, for the topology of Figure 1. It builds a tree similar to SS-SPST-F based on costliest neighbor while trying to minimize the discard energy expended by non-group neighbors. In Figure 4, whenever node 4 transmits a data packet, nodes 8 and 9 listen to it and discard it. Energy wise, it will be better for nodes 5 and 3 to join node 6 instead of node 4. SS-SPST-E takes the same stabilization time as SS-SPST-F. \square

5 SS-SPST-E Algorithm

This section formalizes the SS-SPST-E algorithms. The topology of the ad hoc network is modeled by an undirected graph $G = (V, E)$, where V is the set of nodes and E is the set of links between neighboring nodes. Each node $i \in V$ maintains a local variable C_i to denote the current energy cost which is calculated as explained above. In addition, each node i maintains a hop count variable $H_i(r)$ to denote the hop count to the root node and a parent variable P_i to denote the current parent to the root. Let us denote the set of all nodes adjacent to node i as $Adj(i)$. A valid value to the variable P_i points to a node which is in the set $Adj(i)$. Each node calculates the overhead energy cost involved to join a particular node in the tree and tries to join the node with

the least overhead energy cost. Let O_i denote the current overhead energy cost estimated at node i and $O_i(k), k \in Adj(i)$ is the overhead energy cost of node i joining node k .

Overhead energy cost of node i to join node k is calculated as $O_i(k) = \Delta C_k^i$, where ΔC_k^i represents the energy cost difference experienced by node k with and without node i as its child. The total energy cost of a tree is given by the sum of energy cost of all nodes in the tree. We introduce a constant $FLOOR$, which gives the minimum possible value for the total energy cost of the tree. When all the nodes are in the most energy-efficient path the total energy cost of tree should be equal to the value of $FLOOR$. Each node in the network, when it is not connected to the tree has a energy cost as $EMax$, which is greater than the maximum possible energy cost of the tree which is nothing but the energy required for the root node to reach all the nodes if we assume only 1 hop communication. To control the maximum number of hops a node can be from the root we fix the maximum number of hops to be the total number of nodes $NCOUNT$. Let O_i^{min} represent the minimum possible value of O_i .

Remark 1 For all i , O_i^{min} is determined by the topology of the (network) graph and group information.

For any node i in the graph we define the following sets:

- $\mathcal{VP}(i) = \{j \mid j \in Adj(i) \wedge H_j(r) < NCOUNT\}$.
- $\mathcal{N}(i) = \{j \mid j \in \mathcal{VP}(i) \wedge O_i(j) = \min_{k \in \mathcal{VP}(i)} (O_i(k))\}$.

The set $\mathcal{VP}(i)$ contains neighboring nodes of i that have a hop count less than the maximum value, where as the set $\mathcal{N}(i)$ contains neighboring nodes of i that are on currently estimated energy-efficient paths from node i to r . Formally we can describe SS-SPST-E as follows:

$$\begin{aligned} & \text{if } (i = \text{root} \wedge (H_i \neq 0 \vee P_i \neq \text{NULL})) \\ & \text{then } H_i = 0 \& P_i = \text{NULL} \& O_i = 0 \\ & \text{else if } \left(i \neq r \wedge (O_i \neq \right. \\ & \quad \left. \min_{j \in \mathcal{VP}(i)} (O_i(j)) \vee P_i \notin \mathcal{N}(i) \vee H_i \neq H_{P_i} + 1) \right) \\ & \text{then } O_i = \min_{j \in \mathcal{VP}(i)} (O_i(j)) \& P_i = j, j \in \mathcal{N}(i) \& \\ & \quad H_i = H_j + 1 \end{aligned}$$

The pruning of the multicast tree built by SS-SPST-E is done in the bottom-up manner similar to SS-SPST.

5.1 Correctness

A system S is said to be self-stabilizing with respect to predicate P (defined on global system states) if it satisfies the following two properties:

1. Convergence: Starting from any arbitrary global state, S is guaranteed to reach a global state, satisfying P in finite number of state transitions.

2. Closure: Once S reaches a global state satisfying P , it cannot be falsified.

All global states in which the predicate P holds are called *legitimate states*. For SS-SPST-E the legitimate states are defined as follows:

Definition 1 The system is in the legitimate state iff $\forall i \in V : O_i = O_i^{min} \wedge H_i \leq NCOUNT \wedge ((i = r \wedge P_i = \text{NULL} \wedge H_i = 0) \vee (i \neq r \wedge P_i \in \mathcal{N}(i) \wedge H_i = H_{P_i} + 1))$

Any state which is not a legitimate state is an illegitimate state. SS-SPST-E algorithm tries to bring an illegitimate state to a legitimate state and thereby reducing the total energy consumption. Each node looks at the states of its neighbors and checks the local satisfiability and if the node is not locally legitimate, it triggers self-stabilization routine and tries to bring the whole system to a legitimate state. This is very similar to SS-SPST.

To prove the correctness of the algorithm we have to prove the convergence property, closure property and the correctness of the algorithm to maintain a single connected tree at any time. We also prove that our protocol does not have the count-to-infinity problem. This can happen as we do not concentrate on minimizing the hop count only.

Lemma 1 Convergence: Starting from a given illegitimate state the total energy cost of the graph reduces after every round till all the nodes in the system are stabilized.

Proof: We prove the above lemma using induction. When the node is not connected to the tree it has a energy cost of $EMax$. After the first round, root node stabilizes and sets its cost variable. Since $EMax > C_r$, the total graph energy cost reduces after first round. Now let us assume the energy cost of the graph after k rounds is TC_k . We now have to prove that if TC_{k+1} is the energy cost of the graph after $k+1$ rounds then $TC_{k+1} < TC_k$. There are two possibilities,

Case 1: There exists at least one node that is not connected to the tree. By our assumption, the graph is connected. So there will be at least one tree neighbor of the not connected nodes and in the $k+1^{st}$ round, it will be connected to the tree and its energy cost will reduce from $EMax$ to a lesser value. Hence, $TC_{k+1} < TC_k$.

Case 2: All nodes are connected to the tree. After k rounds, each node will have a parent in the tree and will have a energy cost associated with itself. In the $k+1^{st}$ round, when a node changes its parent then the overhead associated with that node reduces. The reduction of energy cost of the old parent will be greater than the increase of energy cost of the new parent. Hence, $TC_{k+1} < TC_k$.

Thus the energy cost of the graph will keep reducing till it reaches a minimum value, which in our case is $FLOOR$.

Lemma 2 Closure: Once the energy cost of the graph equals $FLOOR$, it stays there until further faults occur.

Proof: If the total energy cost of the graph decreases after it attains the *FLOOR* value, then our assumption that *FLOOR* is the minimum possible energy tree value does not hold true. Hence the total energy cost of the graph cannot reduce further than *FLOOR* value. The total energy cost of the graph will never increase after a round as shown by Lemma 1. Hence, the total energy cost of the graph will neither decrease nor increase after it attains *FLOOR*.

Lemma 3 *Count-to-infinity:* Tree will always remain connected and there will not be any loops.

Proof: There is a possibility that a loop may be created during the tree formation. As soon as a loop forms the hop count value for each node to the root will increase with every round and soon it will exceed the *NCOUNT* value and hence the nodes will be removed from the sets $\mathcal{VP}(i)$ and $\mathcal{N}(i)$. Thus, they will not be considered as the parent node by any node. In this way, the loop will be detected and necessary stabilizing actions will repair it. Since by our assumption all the nodes are connected in the graph, the resultant tree will also remain connected as per Lemma 1.

6 Simulation Model

To analyze the performance of SS-SPST protocol and analyze the impact of proposed cost metric on it, we simulate SS-SPST, SS-SPST-E, MAODV and ODMRP protocols in diverse scenarios. This section explains the model we used to simulate these protocols and the performance metrics we measured. We used the network simulator (ns-2.1b9a) to compare the performance of the protocols. The main aim of this simulation study was to characterize the scenarios for which performance of self-stabilizing protocols can be better (worse) than other multicast routing protocols. Further, we wanted to analyze various trade-offs involved in saving energy. We modeled a 750m x 750m simulation area with 50 nodes placed at random positions. Random way-point mobility model was used in our simulations. As pointed by Noble et al. [29] one has to guard against the velocity-decay problem of this model. The use of the random-way point model in our simulation conforms to the fix suggested by Noble et al. Specifically, the settings of simulation parameters ensures that the nodes use non-zero minimum velocity. We used various scenario files with different initial node locations and different mobility rates and took an average value to plot the graphs. Each simulation ran for 1800 seconds of simulated time. We used the same scenarios to evaluate all the protocols.

We assumed each node to be equipped with an omnidirectional antenna which can dynamically vary the transmission power. From the given nodes, we chose one node to be the source of the multicast session sending CBR data

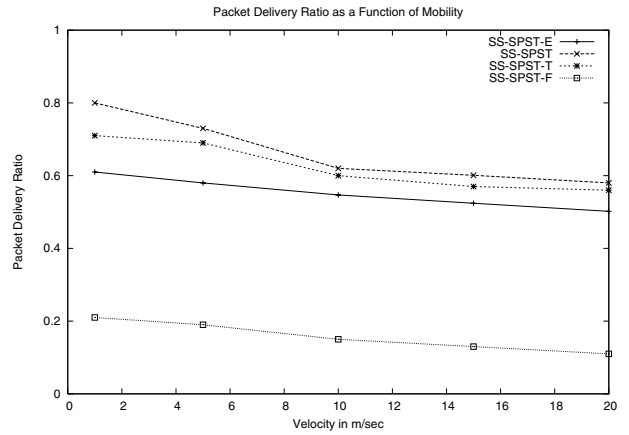


Figure 7. Packet Delivery Ratio vs. Velocity.

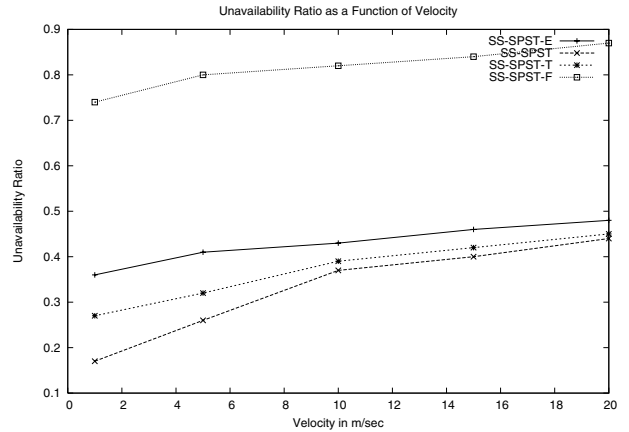


Figure 8. Unavailability Ratio vs. Velocity.

packets at the rate of 64Kbps. We measure the performance of the protocols based on the following metrics.

1. *Packet delivery ratio (PDR):* The ratio of the number of data packets actually delivered to the receivers and the number of data packets supposed to be received by the receivers. This metric shows the effectiveness of a protocol to deliver data packets to multiple recipients.
2. *Energy consumed per packet delivered:* We measure the energy consumed per packet delivered by dividing the total energy consumed by all the nodes in the network by the total number of data packets delivered.

The performance of the protocols is studied by varying the mobility speed and the multicast group size. Moreover, to analyze the impact of beacon interval on the protocols, we also varied the beacon interval.

7 Simulation Results

We analyze the simulation results based on the performance metrics given in the section 6. For simulations in sections 7.1 and 7.3 we fixed the beacon interval to be 2 sec.

and varied the maximum mobility speed from 1-20 m/s. For results in section 7.2, we fixed the maximum mobility speed to be 5 m/s and varied the beacon interval from 1 to 4 sec.

7.1 Comparison of different cost metrics

In order to evaluate the performance of SS-SPST with different cost metrics, we simulated the protocols SS-SPST, SS-SPST-T, SS-SPST-F and SS-SPST-E and analyzed the results under diverse scenarios. Our main aim is to achieve optimal energy efficiency which we achieve by our proposed SS-SPST-E algorithm. We first discuss the results and conclude this section listing out the cost involved in saving energy.

1. Packet Delivery Ratio: From the Figure 7 we see that as the mobility speed of individual nodes increases the packet delivery ratio drops for all the protocols. In the context of dynamic systems, self-stabilization refers to the stabilization time after the "final" failure. Stabilization is only guaranteed *eventually*, there is an inherent assumption in self-stabilizing system that faults eventually stop to occur for the system to stabilize. As the mobility speed increases, the faults occur at a faster rate than the stabilization time and hence the packet delivery ratio for all the protocols decreases as the mobility speed increases.

Our energy metric forces the SS-SPST-E protocol to choose an energy-efficient path available. Often, transmitting a packet in a single hop might consume more energy than relaying it along a tandem of nodes in a straight line. Thus, the SS-SPST-E tree is deeper than the SS-SPST tree. This reason contributes to reduction in PDR value for SS-SPST-E protocol when compared to that of SS-SPST. The reason SS-SPST-F has a very low PDR value is because of its dynamic nature which causes instability. From the graph we see that PDR value of SS-SPST-T is slightly better than SS-SPST-E because changes to the tree structure cause changes to the existing cost and hence the protocol needs more time to stabilize.

The unavailability ratio which gives the ratio between the time for which the multicast service is unavailable and the total multicast duration is related to PDR and we show the Unavailability Ratio in our simulation results in Figure 8. We observe that for high PDR, unavailability ratio is low and for low PDR the unavailability ratio is high.

2. Energy Consumption: Energy consumption per packet of the nodes is mainly due to transmission and reception of control and data packets. The energy spent to transmit control packets is the same at different mobility speeds. As we see from Figure 9, the energy consumed per node in the network increases as the velocity increases. As the velocity increases, the number of faults that occur in the system increases. This causes more beacon packets to stabilize the system after which a data packet can be delivered to some nodes. So more control packet overhead causes in-

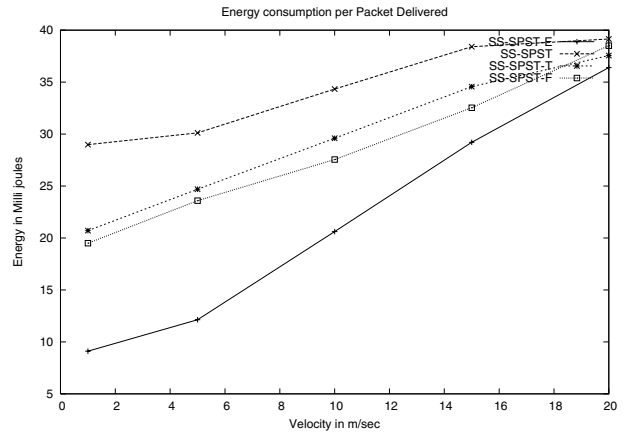


Figure 9. Energy Consumption vs. Velocity.

crease in energy consumption per packet.

Further from the graph, we see that SS-SPST has the highest energy consumption and SS-SPST-E has the lowest. The other two protocols SS-SPST-F and SS-SPST-T have lesser energy consumption than SS-SPST with the hop count metric but they do not give the most energy-efficient solution to build the multicast tree. Thus our simulation results coincide with our initial analysis. When the mobility speed is relatively slow, the rate at which the fault occurs because of mobility is also slow and hence the SS-SPST-E protocol transmits most of its packets through energy-efficient paths. As the mobility speed increases the fault rate increases and hence energy saving of SS-SPST-E compared to that of SS-SPST decreases. At high mobility speeds both the protocols consume almost the similar amount of energy per data packet delivered.

From the above discussion we see that SS-SPST-E is the most energy-efficient among the other protocols. Though incorporation of the new energy metric causes a slight drop in the packet delivery ratio and slight increase in unavailability time, the amount of energy saved per packet is still higher than SS-SPST and SS-SPST with other metrics. Our conclusion is that SS-SPST-E is the best choice in a energy constrained networks like Wireless Ad Hoc Networks.

7.2 Beacon Interval

One important parameter in evaluating SS-SPST and SS-SPST-E is the beacon interval. In this section we analyze the impact beacon interval has performance of a protocol.

1. Packet Delivery Ratio: Figure 10 shows the variation of packet delivery ratio (PDR) with the beacon interval. As the beacon interval increases the PDR value drops for both the protocols, SS-SPST and SS-SPST-E. This drop is due to the fact that time to realize that a fault has occurred in the system increases with the beacon interval. With low beacon interval values the faults are identified and corrected quicker and hence we see a higher packet delivery ratio. From the graph, we see that the decrease in PDR value is



Figure 10. PDR vs. Beacon Interval.

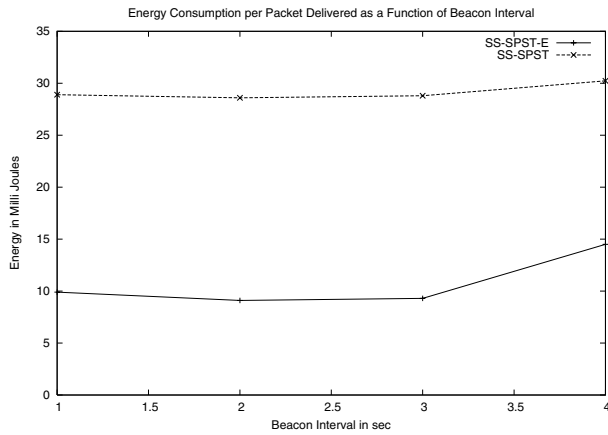


Figure 11. Energy Consumption vs. Beacon Interval.

linear till a point and then drops at a faster rate. This fact is because the increase in the beacon interval makes some nodes to move out of range completely. In other words, number of faults that occurs when beacon interval is more than 3 seconds increases. Hence, we see a sharp drop after beacon interval of 3 seconds. From this we conclude, for better packet delivery ratio results we can fix a low value for the beacon interval and a value above 3 seconds will not depict the real performance of SS-SPST.

2. Energy Consumption: Figure 11 shows the variation of energy consumption with the beacon interval. Energy consumption per packet delivered value is altered by total energy consumption and the number of packets delivered. As the beacon interval increases, one expects the energy consumed to decrease as the number of control packets sent will decrease. From the graph we see that the energy consumed per packet delivered value decreases for a while and then increases. This increase is attributed to the reduction in the number of packets delivered with increase in beacon

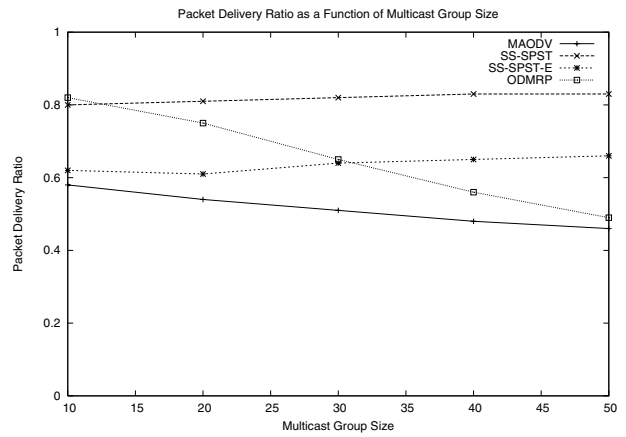


Figure 12. PDR vs. Multicast Group Size

interval as explained by Figure 10. At higher velocities the rate of decrease in packet delivery ratio is more than the rate of increase in energy saving thus energy expended per packet increases beyond certain velocity.

Use of short beacon intervals increases number of control messages and use of long beacon intervals decreases the packet delivery ratio. Our conclusion is that for beacon interval value around 2 sec., we have an optimum energy consumed per packet delivered. Hence, for the rest of the paper, we use a beacon interval of 2 sec.

7.3 Comparison with other multicast routing protocols

In this section we compare SS-SPST and SS-SPST-E protocols with other multicast routing protocols like MAODV and ODMRP. We analyze the comparison results and bring out the scenarios when self-stabilizing algorithms will be useful over on-demand multicast protocols.

1. Packet Delivery Ratio: We varied the multicast group size to calculate the packet delivery ratio. Each node in the network had a maximum speed of 1m/s. The multicast group size was varied from 10 to 50 members in steps of 10.

From the graph (Figure 12) we see that self-stabilizing protocols are group-scalable i.e., when the multicast group nodes are more, the self-stabilizing protocols perform as well as they do when the multicast group nodes are less. In other words self-stabilizing protocols scale well with increase in the number of multicast group nodes. We see that packet delivery ratio is almost the same for different set of multicast nodes. ODMRP outperform self-stabilizing protocols when the number of multicast nodes is less. It does not scale well with the increase in the multicast group nodes. As we see from the graph the packet delivery ratio drops drastically as the number of multicast nodes increases. This decrease in packet delivery ratio is attributed to the increase in overhead caused by redundant paths.

Of all the four protocols, packet delivery ratio of

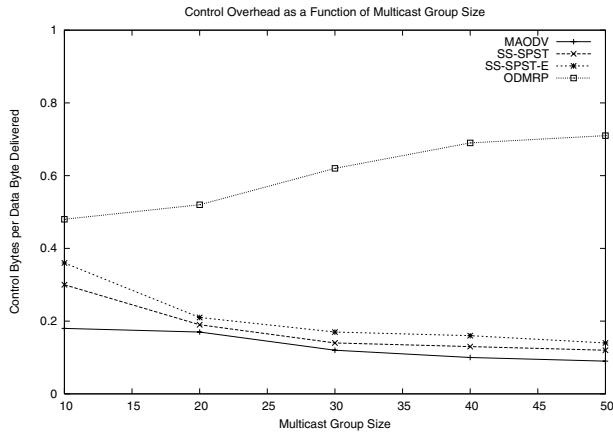


Figure 13. Control Byte Overhead vs. Multicast Group Size.

MAODV is the least. Though packet delivery ratio is not affected too much by the increase in number of multicast nodes, it is less when compared to other protocols. The graph (Figure 13) depicts the cost involved in multicasting using different protocols. This figure represents the control byte overhead when the data is sent continuously. Since ODMRP is a mesh-based protocol, it uses more control bytes for building the multicast tree. ODMRP constructs more than one route from source to destination and hence uses many control messages. As the number of group members increase, ODMRP behaves similar to *flooding* algorithm and hence the control byte overhead increases with increase in the number of group members.

Of all the four protocols, MAODV uses the least control byte overhead. It generates control messages only when there is a need for multicasting. SPST and SS-SPST-E are proactive multicast protocols. The number of control packets sent by these two protocols is almost the same. When multicasting of data packet takes place continuously, these protocols have a very less control overhead but when there is no multicasting, these protocols still send out control packets to maintain the tree structure. Thus these self-stabilizing protocols will have higher multicasting overhead when there is no multicasting operation. Further, SS-SPST-E sends additional information in its beacon packet to conserve energy and hence has more control byte overhead.

We further varied the mobility speed of the nodes in the network from 1m/s to 20 m/s while fixing the multicast group size at 20. By varying the mobility we measured the packet delivery ratio. Mobility against packet delivery ratio graph (Figure 14) shows the performance of the multicasting protocols in presence of mobility. We see that packet delivery ratio in all the protocols drops as the mobility of the nodes increases. Self-stabilizing protocols take finite time to build a multicast tree. As the underlying nodes move, the

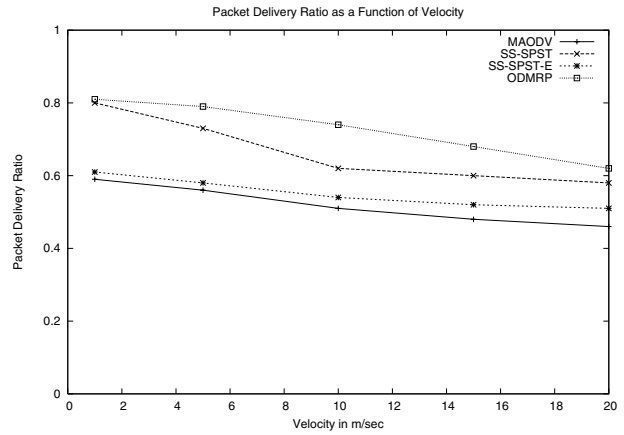


Figure 14. PDR vs. Velocity.

multicasting tree should be reconstructed. From our simulations, we found that SS-SPST-E takes more time to build a tree than SS-SPST. This is due to the fact explained previously in section 7.1.

ODMRP's packet delivery ratio is the highest among the protocols compared even at high mobility speeds. By providing alternate paths, ODMRP performs better than other protocols even under high mobility. MAODV, which is a tree-based protocol performs in a similar fashion to SS-SPST. As the mobility speed increases the packet delivery ratio decreases. In Figure 15, we show the average delay for all the four protocols and find out that both the self-stabilizing protocols have less delay compared to the others. This supports our claim that proactivity in self-stabilization induces less delay. However, SS-SPST has less delay than SS-SPST-E as it has less hop-counts in the multicast tree.

2. Energy Consumption per packet delivered: By varying the mobility speed we captured the energy spent to successfully deliver a packet to the receiver in all the protocols. Our aim was to capture the fact how energy-efficient is SS-SPST-E when compared to other protocols. From Figure 16, we see that ODMRP has the highest energy consumption per packet and SS-SPST-E has the least. Energy is expended on transmission of control messages to build multicast structure and data messages along the route. ODMRP constructs more than one path to deliver the data message to the receiver and hence consumes maximum energy among all the other protocols. MAODV and SS-SPST construct tree structure to deliver multicast messages. Although SS-SPST has more control overhead than MAODV, the packet delivery ratio of SS-SPST is also more. Hence, SS-SPST spends less energy per packet delivered than MAODV.

8 Conclusions

In this paper, we focused on energy aware adaptivity for MANETs. We used the self-stabilization paradigm for the

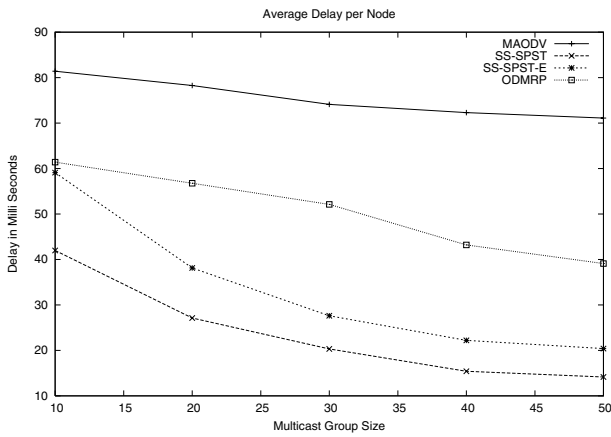


Figure 15. Delay vs. Multicast Group Size.

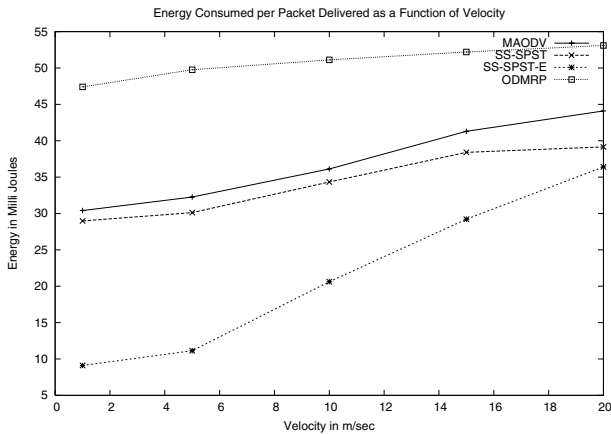


Figure 16. Energy Consumption per Packet Delivered vs. Velocity.

distributed systems for the MANETs due to its adaptive and proactive nature. We selected multicasting in MANETs as our example to find out the trade offs needed in energy efficiency in a self stabilizing system. We picked SS-SPST as the self-stabilizing multicast protocol for MANETs and incorporated in it an energy-aware node-based metric which takes into account overhearing cost. Our simulation results show that to save energy we need to compromise on the latency. We also analyzed SS-SPST and SS-SPST-E with other traditional non-self-stabilizing protocols. A general conclusion is that, self-stabilizing protocols are group-scalable and have a better packet delivery ratio even when the number of multicast group nodes is large.

Acknowledgments

We are thankful to the IPDPS reviewers and Georgios Varsamopoulos for helpful suggestions for improving the quality of this paper.

References

- [1] P. Basu and J. Redi, "Effect of Overhearing Transmissions on Energy-Efficiency in Dense Sensor Networks," In Proc. ISPN'04, April, 2004
- [2] M. Cagalj, J. P. Hubaux, and C. Enz, "Minimum-energy broadcast in all-wireless networks: NP-Completeness and distribution issues," In Proc. of ACM MobiCom 2002, Atlanta, Georgia, Sept. 2002, pp. 172-182.
- [3] J.-H. Chang, and L. Tassiulas, "Energy Conserving Routing in Wireless Ad Hoc Networks,"; In Proc. of IEEE INFOCOM 2000, 2000, pp. 22-31.
- [4] A. E. F. Clementi, P. Crescenzi, P. Penna, G. Rossi, and P. Vocco, "On the complexity of computing minimum energy consumption broadcast sub-graphs," In Proc. of 18th Annual Theoretical Aspects of Comp. Sc. (STACS), vol. 2010, Springer-Verlag, 2001, pp. 121-131.
- [5] A. Cournier, A. K. Datta, F. Petit, and V. Villain, "Self-Stabilizing PIF Algorithm in Arbitrary Rooted Networks", ICDCS 2001.
- [6] S. Deering, "Host Extensions for IP Multicasting", RFC 1112, August 1989, Available from <http://www.ietf.org/rfc/rfc1112.txt>
- [7] G. Deng and S. K. S. Gupta, "On Maximizing Network Lifetime of Broadcast in WANETs under an Overhearing Cost Model". In Proc. ICDCN'06, Dec. 2006. LNCS 4308, pp 215-226.
- [8] E. W. Dijkstra, "Self Stabilizing systems in spite of distributed control", In Proc. Communications of the ACM, November 1974.
- [9] E. W. Dijkstra, "A Belated Proof of Self-Stabilization", Distributed Computing, 1986.
- [10] A. Ephremides. "Energy Concerns in Wireless Networks". In Proc. of IEEE Wireless Communications, vol. 9, no. 4, pp. 46-59, Aug. 2002.
- [11] C. Genolini and S. Tixeuil. "A lower bound on dynamic k-stabilization in asynchronous systems". SRDS 2002 21st Symposium on Reliable Distributed Systems, IEEE Computer Society Press, pp. 211-221, 2002.
- [12] M. Gerla, S.-J. Lee and C.-C. Chang, "On-Demand multicast routing protocol (ODMRP) for ad hoc networks". In Proc. Of IEEE Wireless Communications and Networking Conference 1999, LA, September 1999.
- [13] S. K. S. Gupta, A. Bouabdallah and P. K. Srimani. "Self-Stabilizing Protocol for Shortest Path Tree for Multi-cast Routing in Mobile Networks (Research Note)". In Proc. of Euro-Par 2000, May 2000.
- [14] S. K. S. Gupta and P. K. Srimani. "Self-Stabilizing Multicast Protocols for Ad Hoc Networks". Journal of Parallel and Distributed Computing, 2003.
- [15] S. K. S. Gupta and P. K. Srimani. "Cored-Based Tree with Forwarding Regions (CBT-FR): A Protocol for Reliable Multicasting in Mobile Ad Hoc Networks". J. Parallel Distrib. Comput. 61(9): 1249-1277 (2001).
- [16] P. Karn, "Maca - a new channel access method for packet radio," In Proc. of the 9th ARRL Computer Networking Conference, Canada, 1990.
- [17] F. Li and I. Nikolaidis, "On minimum-energy broadcasting in all-wireless networks," In Proc. of the 26th Annual IEEE Conference on Local Computer Networks (LCN 2001), Tampa, Florida, Nov. 2001, pp. 193-202.
- [18] A. Misra and S. Banerjee. "Minimum Energy Paths for Reliable Communication in Multi-Hop Wireless Networks". Technical Report, Department of Computer Science, University of Maryland, College Park, December 2001.
- [19] S. Ramanathan. "Multicast Tree Generation in Networks with Asymmetric Links". IEEE/ACM Transaction on Networking, November 1996.
- [20] E. Royer and C. E. Perkins. "Multicast operation of the ad-hoc on-demand distance vector routing protocol". In Proc. Of the 5th ACM/IEEE Annual Conf. On Mobile Computing and Networking, August 1999.
- [21] S. Singh and C. S. Raghavendra. "PAMAS-Power Aware Multi-Access Protocol with Signaling for Ad Hoc Networks". ACM Communications Review, July 1998.
- [22] A. S. Tanenbaum. "Computer Networks". Prentice Hall, 4th edition, 2002.
- [23] S. Vaudevan, C. Zhang, D. Goeckel, D. Towsley. "Optimal Power Allocation in Wireless Networks with Transmitter-Receiver Power Tradeoff Proc. INFOCOM'06, 2006.
- [24] B. Wang and S. K. S. Gupta, "S-REMIT: An algorithm for enhancing energy-efficiency of multicast trees in wireless ad hoc networks," In Proc. of IEEE 2003 Global Communication Conference (GLOBECOM 2003), vol. 6, San Francisco, CA, Dec. 2003, pp. 3519-3524.
- [25] B. Wang and S. K. S. Gupta, "G-REMIT: An algorithm for building energy efficient multicast trees in wireless ad hoc networks," In Proc. of 2nd IEEE International Symposium on Network Computing and Applications (NCA-03), Cambridge, MA, Apr. 2003, pp. 265-272.
- [26] J. E. Wieselthier, G. D. Nguyen and A. Ephremides. "Algorithms for Energy-Efficient Multicasting in Static Ad-Hoc Wire ss Networks". In Proc. Journal on Mobile Networks (MONET), vol. 6, 2001, pp. 2 51-63.
- [27] J. E. Wieselthier, G. D. Nguyen and A. Ephremides. "Resource Management in Energy-Limited, Bandwidth-Limited, Transceiver-Limited Wireless Networks for Session-Based Multicasting". Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol. 39, Issue 5, June 2002.
- [28] J. E. Wieselthier, G. D. Nguyen and A. Ephremides, "On the Construction of Energy-Efficient Broadcast and Multicast trees in Wireless Networks". In Proc. IEEE INFOCOM 2000, 2000, pp. 585-594.
- [29] J. Yoon, M. Liu, B. Noble "Random Waypoint Considered Harmful" Proc. INFOCOM'03, pp. 1312-1321, 2003.
- [30] Internet Engineering Task Force (IETF) Mobile Ad Hoc Networks (MANET) Working Group Charter. <http://www.ietf.org/html.charters/manet-charter.html>.