Energy Efficient Survivable Broadcasting and Multicasting in wireless Ad Hoc Networks

Jian Tang, Guoliang Xue and Weiyi Zhang
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85287-8809
Email:{jian.tang, xue, weiyi.zhang}@asu.edu

***Abstract*— Survivability and energy efficiency are two critical issues for broadcast and multicast routing in wireless ad hoc networks. Energy efficient routing has been extensively studied. However, survivability issues have not been well addressed in this field. In this paper, we jointly consider both issues and study energy efficient algorithms for survivable broadcast/multicast routing, which is resilient to single node failure. In particular, we propose the minimax survivable broadcasting/multicasting problems, which seek survivable broadcast/multicast trees in which the maximum node transmit power is minimized; and the minimum survivable broadcasting/multicasting problems, which seek survivable broadcast/multicast trees in which the total node transmit power is minimized. For the minimax problems, we present efficient optimal algorithms. For the minimum problems, we present effective heuristics. Preliminary simulation results are also presented.**

I. Introduction

Wireless ad hoc networks are usually used for some critical missions such as rescue, surveillance and so on. Nodes in such networks are prone to fail due to various factors such as enemy attack in the battlefield, harsh environment, and energy depletion. Any communication outage, even just lasting for several seconds, may cause a big loss. Since nodes in wireless ad hoc networks are generally powered by batteries which can not last too long, the sparse energy resource should be used efficiently to prolong network lifetime. We are interested in an energy efficient survivable network system which is able to deliver packets from a source node to working destination nodes after failures of one or multiple nodes.

In this paper, we consider a wireless ad hoc network in which every node has an omni-directional antenna and is able to change its communication range by adjusting its transmit power. Multicasting and broadcasting in wireless networks are very different from those in wired networks because the radio is inherently a broadcast medium, i.e., all nodes within the transmitter's communication range will receive the transmitted data, which is known as Wireless Multicast Advantage (WMA) in [13]. The well known Broadcast Incremental Power (BIP) algorithm is proposed in [13] to achieve energy efficient broadcasting in wireless ad hoc networks by taking advantage of WMA. Recently, Srinivas and Modiano propose to provide survivable unicast routing in wireless ad hoc networks by using disjoint paths in [10]. They present two elegant algorithms to compute node-disjoint paths and link-disjoint paths with minimum total energy for a given source and destination pair in a wireless ad hoc network. To the best of our knowledge, this is the first paper studying energy efficient survivable broadcast and multicast routing in wireless ad hoc networks.

We apply the idea of redundant trees [9] to support survivable broadcast/multicast routing. Basically, a blue tree and a red tree will be constructed when a multicast or broadcast request arrives. The packets will be sent out along both trees. In this way, a packet is guaranteed to reach all working destination nodes via either the blue tree or the red tree even if a node (other than the source node) fails.

We define the **minimax survivable broadcasting/multicasting** problems and the **minimum survivable broadcasting/multicasting** problems. We present efficient optimal algorithms for the minimax survivable broadcasting/multicasting problems which find a pair of survivable broadcast/multicast trees that minimize the maximum transmit power among the nodes. We present effective heuristics for the minimum survivable broadcasting/multicasting problems which find a pair of survivable broadcast/multicast trees that minimize the total transmit power of all nodes.

The rest of this paper is organized as follows. We discuss related work in Section II. We formally define survivable broadcasting and multicasting problems in Section III and present our algorithms in Section IV. We present our simulation results in Section V and conclude the paper in Section VI.

Research supported in part by ARO grant W911NF-04-1-0385, NSF grant CCF-0431167 and a seed grant from CEINT. The information reported here does not reflect the position or the policy of the federal government.

II. Related Work

Energy efficient routing in wireless ad hoc networks has been studied extensively in the literature. In [3], [5], [7], [8], several algorithms have been proposed for wireless ad hoc unicast routing with the goal of either minimizing the total transmit power or maximizing the network lifetime.

The Broadcast Incremental Power (BIP) algorithm is proposed in [13] to construct an energy efficient broadcast tree by considering WMA. A multicast routing algorithm is also presented by pruning the broadcast tree. A Minimum Spanning Tree (MST) based algorithm is presented in [4] for wireless broadcasting, which is able to achieve relatively balanced energy consumption. A formal NP-hardness proof for power optimal broadcast problem in wireless networks is given in [1]. Li *et al.* propose a node weighted Steiner tree based algorithm and prove that its performance ratio is bounded by $(1 + 2\log(n-1))$ in [6]. In [2], the authors propose a localized protocol where each node requires only the knowledge of its distance to all neighboring nodes and distances between its neighboring nodes. They use simulation results to show that it is comparable to the globalized BIP algorithm.

Recently, people began to study survivability issues in wireless ad hoc networks. As we mentioned in the introduction, two elegant algorithms are presented to support both survivability and energy efficiency by computing disjoint paths with minimum total energy in [10]. In [11], Tang and Xue present efficient algorithms for computing a pair of node-disjoint paths which either minimize energy under lifetime constraint or maximizes lifetime under energy consumption constraint. They also study the tradeoffs between path lifetime and total energy consumption in node-disjoint path routing and their effects on network throughput. However, all previous efforts focus on unicast routing only. In this paper, we present energy efficient survivable routing algorithms for both broadcasting and multicasting.

The redundant tree scheme has been proposed by Médard, Finn, Barry and Gallager to support survivable communications in wired networks. In [9], Médard *et al.* propose a redundant tree based preplanned recovery scheme and prove that the trees constructed by their algorithms can survive any single node/link failure. Xue, Chen and Thulasiraman [14] generalize this scheme and study Quality of Protection (QoP) issues and various Quality of Service (QoS) issues such as bandwidth, delay, and cost. Our approach for energy efficient survivable broadcasting and multicasting is along the lines of redundant trees.

III. Problem Definitions

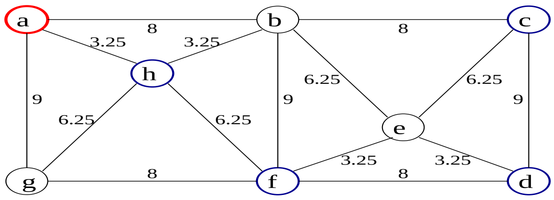
We consider a wireless ad hoc network consisting of n nodes $v_1, v_2, \ldots, v_n$ that have omni-directional antennas and can dynamically adjust their transmit power. All nodes have the same maximum transmit power $\mathcal{T}_{max}$, while each node can transmit at any power level in the range $[0, \mathcal{T}_{max}]$. We follow a commonly used wireless signal propagation model from [13] where the received signal power attenuates proportional to $r^{-\alpha}$, where r is the Euclidean distance between the receiver and the transmitter and α is a constant, typically between 2 and 4, depending on the wireless medium [13].

Assuming that each node in the ad hoc network sets the transmit power to $\mathcal{T}_{max}$, we have an underlying undirected graph G where the vertices of G corresponds to the n nodes in the ad hoc network and there is an undirected edge connecting vertices u and v in G if and only if the nodes corresponding to u and v can communicate with each other. We call this graph the communication graph. For our purpose, the communication graph is also edge weighted, where the cost of an edge connecting vertices u and v is proportional to $d(u, v)^\alpha$, where $d(u, v)$ is the Euclidean distance between the two nodes that correspond to u and v respectively. Therefore the cost of the edge connecting u and v is the transmit power needed at the node corresponding to u so that the node corresponding to v can correctly receive signals from the node corresponding to u. To abuse the notation a little bit, we will use u to mean *the node corresponding to* u, as long as there is no confusion.

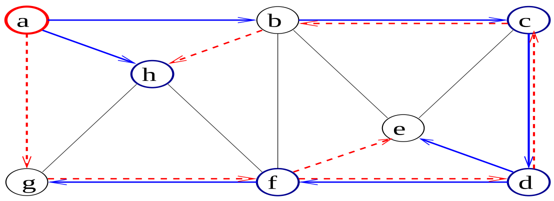
A multicast request is given by a source node s and a set of destination nodes $\{d_1, d_2, \ldots, d_M\}$. In this paper, we are interested in multicast schemes that can survive any single node failure other than the source node. For survivable multicasting, we construct two multicast trees T^B and T^R, both spanning the source node and the destination nodes in such a way that when a single node (other than the source node) fails, every other node can still receive data from the source node via either the multicast tree T^B or the multicast tree T^R. We call T^B and T^R a pair of survivable multicast trees, which is formally defined in Definition 1.

Definition 1: Let $G(V, E)$ be an undirected graph with edge set E and vertex set V. Let a given multicast request consists of a source node $s \in V$ and a set of destination nodes $\{d_1, d_2, \ldots, d_M\} \subseteq V$. Let T^B and T^R be a pair of directed trees such that there is a directed path p_i^B in T^B from s to every destination d_i and a directed path p_i^R in T^R from s to every vertex d_i. T^B and T^R form a pair of **survivable multicast trees** if for any chosen vertex $u \neq s \in V$ and any $d_i \neq u \in V$, p_i^B

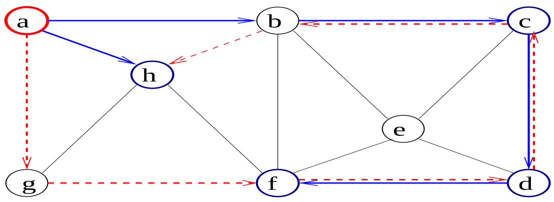
and p_i^R do not both contain node u. We will use p_i^B and p_i^R to denote the two paths discussed above throughout this paper.



(a) A communication graph



(b) A pair of survivable broadcast trees



(c) A pair of survivable multicast trees

Fig. 1. Survivable broadcasting and multicasting using red/blue trees. Solid arcs are for blue trees and dashed arcs are for red trees.

Fig. 1(a) illustrates a sample communication graph with 8 nodes and 15 links. Also illustrated is a multicast request with source node a and destination nodes c, d, f, h. Fig. 1(b) illustrates a pair of survivable broadcast trees rooted at node a, each spanning all other nodes in the network. The tree with solid edges is the *blue tree* and the tree with dashed edges is the *red tree*. When node a wants to send information to all other nodes, it sends the information along both the blue tree and the red tree. Fig. 1(c) illustrates a pair of survivable multicast trees for the sample multicast request. It is obtained by repeatedly pruning non-destination leaf nodes in the pair of survivable broadcast trees.

We illustrate why the pair of multicast trees in Fig. 1(c) can survive a single node failure. Suppose a single node failure occurs at node d. Then the nodes h, c can receive the message from source node a along the blue tree. Node f can not receive the message from node a along the blue tree. However, node f can receive the message from node a along the red tree. Similarly, if a single node failure occurs at node b, then node h can receive the message from node a along the blue tree and nodes c, d, f can receive the message from node a along the red tree. In other words, multicasting along both trees is resilient to single node failure.

For a given pair of survivable multicast trees to work, we also need to assign the transmit power at each node in the network. For example, for the pair of survivable multicast trees in Fig. 1(c), the transmit power at node a must be high enough for the signals from node a to reach nodes b, h and g, where b and h are the downstream neighbors of a in the blue tree and g is the downstream neighbor of a in the red tree. Therefore the transmit power at node a is 9. Similarly, the transmit power at nodes b, c, d, e, f, g, h are $8, 9, 9, 0, 8, 8, 0$, respectively. It is clear that the transmit power at each node can be derived from the pair of survivable multicast trees. We formally define this in the following.

Definition 2: Let T^B and T^R be a pair of survivable multicast trees for a given multicast request. For each node v in the network, the transmit power of v derived from T^B and T^R, denoted by $t(v, T^B, T^R)$, is defined in the following.

- If v is not on either multicast tree or does not have a downstream neighbor on either multicast tree, $t(v, T^B, T^R) = 0$.
- Let $u_1, u_2, \ldots, u_p$ be all the downstream neighbors of v in either T^B or T^R. Then $T(v, T^B, T^R) = \max_{1 \le j \le p} c(v, u_j)$, where $c(v, u_j)$ is the cost of the edge connecting v and u_j in the communication graph.

Definition 3: Let T^B and T^R be a pair of survivable multicast trees for a given multicast request. The maximum transmit power required by T^B and T^R is defined as $t_{max}(T^B, T^R) = \max_{v \in V} t(v, T^B, T^R)$. The total transmit power required by T^B and T^R is defined as $t_{total}(T^B, T^R) = \sum_{v \in V} t(v, T^B, T^R)$.

For the pair of survivable multicast trees in Fig. 1(c), we have $t_{max}(T^B, T^R) = 9$ and $t_{total}(T^B, T^R) = 51$.

Definition 4: Let a multicast request with source node s and destination nodes $d_1, d_2, \ldots, d_M$ be given. A pair of survivable multicast trees T^B and T^R is said to be a pair of **minimax survivable multicast trees** if $t_{max}(T^B, T^R)$ is minimum among all pairs of survivable multicast trees for the given multicast request. The **minimax survivable multicasting** problem seeks for a pair of minimax survivable multicast trees for the corresponding multicast request.

Definition 5: Let a multicast request with source node s and destination nodes $d_1, d_2, \ldots, d_M$ be given. A pair of survivable multicast trees T^B and T^R is said to be a pair of **minimum survivable multicast trees** if $t_{total}(T^B, T^R)$ is minimum among all pairs of survivable multicast trees for the given multicast request. The **minimum survivable multicasting** problem seeks for a pair of minimum survivable multicast trees for the corresponding multicast request.

Note that broadcast is a special case of multicast where the set of destinations consists of $V \setminus \{source\}$. Therefore the terms **survivable broadcast trees**, **minimax survivable broadcast trees**, **minimum survivable broadcast trees** are defined similarly.

In the following sections, we will present efficient algorithms for finding optimal solutions for the minimax survivable broadcasting and multicasting problems, as well as heuristic algorithms for finding good solutions for the minimum survivable broadcasting and multicasting problems. Although broadcasting is a special case of multicasting, it is an important problem by itself. Also, it is easier to present our algorithm for the minimum survivable multicasting problem after the presentation of our algorithm for the minimum survivable broadcasting problem. Therefore we will deal with broadcasting separately when necessary.

IV. SURVIVABLE BROADCASTING AND MULTICASTING

Firstly, we present algorithms for constructing a pair of survivable broadcast trees or multicast trees satisfying certain properties. At the heart of these algorithms is the algorithm for constructing a pair of survivable broadcast trees in an undirected graph. The scheme for finding a pair of survivable broadcast trees (known as the MFBG scheme) has been studied by Médard, Finn, Barry and Gallager [9] and later generalized by Xue, Chen and Thulasiraman [14]. Due to its importance, we present the MFBG scheme [9] for computing a pair of survivable broadcast trees here for future reference.

The MFBG scheme for computing a pair of survivable broadcast trees is listed as Algorithm 1, where it is assumed that s is the source node of the broadcast request. In this scheme, each tree node u is assigned a *blue voltage* $v^B(u)$, which helps ensure that the two trees constructed form a pair of survivable multicast trees. The elegant voltage technique is introduced by [9]. *We note that the voltage is only a tool for easy construction of the red/blue trees. It has nothing to do with the transmit power at the network nodes.*

The MFBG scheme grows the red tree and the blue tree by first finding a cycle containing the root node and then iteratively finding a path connecting a node already on the tree to another node on the tree via nodes not yet on the tree. Once a path is found, we grow the red tree and the blue tree with the help of the voltages. As proved in [9], the voltage rule guarantees the correctness of the algorithms.

Algorithm 1 MFBG

step_1 Initialize T^B and T^R to contain the root node s only. Assign node s both a blue voltage $v^B(s) > 0$ and a red voltage $v^R(s) = 0$.

step_2 Find a cycle $[s, v_1, \ldots, v_k, s]$ with $k \geq 2$. Let $s \rightarrow v_1 \rightarrow \ldots \rightarrow v_k$ be the blue chain and $s \rightarrow v_k \rightarrow \ldots \rightarrow v_1$ be the red chain. Add the blue chain to T^B and the red chain to T^R.
Assign blue voltages at the new nodes in this cycle such that $v^B(s) > v^B(v_1) > \cdots > v^B(v_k) > v^R(s)$.

step_3 if T^B spans all the nodes in G, stop.

step_4 Find a path $[x, v_1, \ldots, v_k, y]$ connecting two distinct nodes x and y on T^B and $k \geq 1$ nodes not on T^B such that $v^B(x) > v(y)$ where $v(y) = v^B(y)$ unless $y = s$, in the latter case $v(y)$ is set to $v^R(s)$.
Let $x \rightarrow v_1 \rightarrow \ldots \rightarrow v_k$ be the blue chain and $y \rightarrow v_k \rightarrow \ldots \rightarrow v_1$ be the red chain. Add the red chain to T^R and the blue chain to T^B.
Let $v^M(x)$ be the maximum of all voltages that are lower than $v^B(x)$. Assign blue voltages at these new nodes on this path such that $v^B(x) > v^B(v_1) > \cdots > v^B(v_k) > v^M(x)$. goto step_3.

Lemma 1: Algorithm 1 correctly finds a pair of survivable broadcast trees T^B and T^R, provided that the graph is biconnected. □

The running time of Algorithm 1 is $O(n^3)$ [9] and Xue *et al.* [15] have shown that there is an $O(n + m)$ time implementation of Algorithm 1, where n and m are the number of vertices and number of edges in the graph.

As an example, we explain how Algorithm 1 constructs the pair of red/blue trees in Fig. 1(b). Node a is chosen as the root node. In step_2, the algorithm finds the cycle $[a, b, c, d, f, g, a]$. It assigns the voltages such that $v^B(a) > v^B(b) > v^B(c) > v^B(d) > v^B(f) > v^B(g) > v^R(a)$ and constructs the blue tree edges $(a, b), (b, c), (c, d), (d, f), (f, g)$ and the red tree edges $(a, g), (g, f), (f, d), (d, c), (c, b)$ in the figure. In step_4, the algorithm finds the path $[d, e, f]$ (note that $v^B(d) > v^B(f)$). Note that $v^M(d)$ equals $v^B(f)$ at this time. It assigns the voltage for e such that $v^B(d) > v^B(e) > v^M(d)$, and constructs the blue tree edge (d, e) and the red tree edge (f, e) in the figure. In step_4, the algorithm finds the path $[a, h, b]$ (note that $v^B(a) > v^B(b)$). Note that $v^M(a)$ equals $v^B(b)$ at this time. It assigns the voltage for h such that $v^B(a) > v^B(h) > v^M(a)$, and constructs the blue tree edge (a, h) and the red tree edge (b, h) in the figure.

A. Minimax Survivable Broadcasting and Multicasting

In this section, we present efficient algorithms for computing a pair of minimax survivable broadcast/multicast trees when the broadcast/multicast request is given.

We note that in an ad hoc network with n nodes, the number of neighbors of each node is bounded by $O(n)$. Therefore it is sufficient to consider only $O(n^2)$ (the number of edges in the communication graph) different transmit power values. Our algorithms for minimax survivable broadcasting and minimax survivable multicasting use bisection on the $O(n^2)$ different transmit power values to find the minimum value $\mathcal{C}$ such that there exists a pair of survivable broadcast/multicast trees T^B and T^R with $t_{max}(T^B, T^R)$ no more than $\mathcal{C}$.

For any positive number $\mathbf{C} \leq \mathcal{T}_{max}$, define $G(\mathbf{C})$ to be the undirected graph G whose vertices are the n nodes in the ad hoc network and there is an edge between nodes u and v if and only if they can correctly receive signals from each other when transmitted at power level $\mathbf{C}$. Our algorithm first uses bisection on the $O(n^2)$ different values of $\mathbf{C}$ to find the minimum value $\mathcal{C}$ such that the source node and all destination nodes of the multicast/broadcast request are in the same biconnected component of $G(\mathcal{C})$. It then applies the corresponding MFBG scheme to $G(\mathcal{C})$ to construct a pair of survivable broadcast trees. Finally, the algorithm prunes the non-destination leaf nodes in the resulting broadcast trees to obtain the corresponding multicast trees. The algorithm is stated in Algorithm 2.

Algorithm 2 Minimax Survivable Broadcasting and Multicasting

- step_1 Use bisection on the $O(n^2)$ power values to find the minimum $\mathcal{C}$ such that the source node and all destinations are in the same biconnected component in $G(\mathcal{C})$.
- step_2 Apply MFBG on the biconnected components of $G(\mathcal{C})$ to construct a pair of survivable broadcast trees T^B and T^R spanning the source node and all other nodes in the biconnected component.
- step_3 Repeatedly delete non-destination leaf nodes in T^B and T^R until every leaf node in T^B and T^R is a destination node. Assign the transmit power for each of the network nodes according to the rules discussed in Definition 2.

Theorem 1: The time complexity of Algorithm 2 is $O(n^2 \log n)$. It correctly computes a pair of survivable multicast trees T^B and T^R such that $t_{max}(T^B, T^R)$ is minimum, provided that the source node and all destination nodes are in the same biconnected component of the communication graph.

PROOF. We can sort the $O(n^2)$ possible power values in $O(n^2 \log n)$ time. For each value of $\mathbf{C}$, we can find out whether the source node and all destination nodes are in the same biconnected components in $O(n^2)$ time [12]. The number of bisections is bounded by $O(\log(n^2)) = O(\log n)$. Therefore the algorithm has a time complexity of $O(n^2 \log n)$.

If there is a pair of survivable multicast trees with the maximum transmit power no more than $\mathbf{C}$, then the source node and all destination nodes are in the same biconnected component of $G(\mathbf{C})$. Similarly, if the source node and all destination nodes are in the same biconnected component of $G(\mathbf{C})$, we can construct a pair of survivable multicast trees with maximum transmit power equal to $\mathbf{C}$. This proves the correctness of the algorithm. □

B. Minimum Survivable Broadcasting and Multicasting

Finding a pair of survivable broadcast trees with minimum total transmit power is difficult since even finding a single broadcast tree with minimum total transmit power is shown to be NP-Complete in [1]. In this section, we will present effective heuristics for constructing a pair of survivable broadcast/multicast trees with small total transmit power.

Again, we will use the frame of red/blue trees. Recall that initially only the root node s is on the trees. We first find a cycle containing s to grow the trees. In each additional step, we add k nodes to the trees by selecting a path $[x, v_1, \ldots, v_k, y]$ connecting two nodes x and y already on the trees via k nodes not yet on the trees. Before each adding procedure, the partially constructed tree pair has a value recording its total energy. Since our goal is to minimize the total energy, we wish the incremental energy caused by adding new nodes in each step is as small as possible. Note that finding a path or cycle with minimum incremental energy is not easy. So we design a heuristics. Firstly, we start with node s and use Depth First Search (DFS) to find a cycle with low energy in terms of the link cost. In the following steps, we use the same method to find a set of feasible low energy paths and then choose the one with the lowest incremental energy. Algorithm 3 is presented as follows to compute a good solution to the minimum survivable broadcasting problem.

Algorithm 3 Minimum Survivable Broadcasting

step_1 Initialize T^B and T^R to contain the root node s only. Assign node s both a blue voltage $v^B(s) > 0$ and a red voltage $v^R(s) = 0$. Also assign the transmit power at node s to zero.

step_2 Find a cycle $[s, v_1, \ldots, v_k, s]$ in the graph with low energy. Let $s \rightarrow v_1 \rightarrow \ldots \rightarrow v_k$ be the blue chain and $s \rightarrow v_k \rightarrow \ldots \rightarrow v_1$ be the red chain. Add the red chain to T^R and the blue chain to T^B. Assign blue voltages at the new nodes in this cycle such that $v^B(s) > v^B(v_1) > \cdots > v^B(v_k) > v^R(s)$. Update the transmit power for every node on the found cycle using the rules described in Definition 2.

step_3 if T^B spans all the nodes in G, stop.

step_4 Find a path $[x, v_1, \ldots, v_k, y]$ with low incremental energy connecting two distinct nodes x and y on T^B and $k \geq 1$ nodes not on T^B such that $v^B(x) > v(y)$ where $v(y) = v^B(y)$ unless $y = s$, in that case $v(y) = v^R(s)$. Let $x \rightarrow v_1 \rightarrow \ldots \rightarrow v_k$ be the blue chain and $y \rightarrow v_k \rightarrow \ldots \rightarrow v_1$ be the red chain. Add the red chain to T^R and the blue chain to T^B. Let $v^M(x)$ be the maximum of all voltages that are lower than $v^B(x)$. Assign blue voltages at these new nodes on this path such that $v^B(x) > v^B(v_1) > \cdots > v^B(v_k) > v^M(x)$. Update the transmit power for every node on the found path using the rules described in Definition 2. goto step_3.

Using analysis similar to that in [14], we can show that Algorithm 3 can compute a pair of survivable broadcast trees as long as the communication graph is biconnected. Its time complexity is $O(n^2(m+n))$ because DFS takes $O(m+n)$ time and we run DFS at most $O(n)$ times in each step.

Then we are ready to describe our heuristic for minimum survivable multicasting. Here we require that the source node and all destination nodes are in the same biconnected component for the given communication graph. Firstly, we invoke Algorithm 3 to construct a pair of survivable broadcast trees for the biconnected component of the given communication graph that contains the source node. Then we repeatedly prune the non-destination leaf nodes in the resulting trees to obtain a pair of survivable multicast trees. We update the transmit power for all affected nodes while pruning. The time complexity is also $O(n^2(m + n))$ since the pruning can be done in linear time.

V. Performance Evaluation

In this section, we evaluate the performance of our algorithms through simulations. We consider wireless ad hoc networks with nodes randomly located in a $750m \times 750m$ region. Every node has the maximum communication range of $250m$. The energy required for transmitting a message with unit size from node u to node v is $0.0001 * d(u, v)^2$. Similar energy consumption models are used in [13]. Each entry in the tables reported here is the average over 100 runs.

In the simulations, we compare our broadcasting algorithms with a *Simple Survivable Broadcast* (SSB) algorithm and a *Link Cost based Survivable Broadcast* (LCSB) algorithm. Basically, the SSB algorithm uses MFBG to construct a pair of survivable trees and then assign the transmit power for every node according to the rules described in Definition 2. The LCSB algorithm employs the similar procedure as SSB to construct the trees, but in each step, uses DFS to find a set of feasible cycles or paths and chooses the one with the lowest total energy in terms of link energy cost. After the survivable tree pair is created, we do the transmit power assignment. The corresponding *Simple Survivable Multicast* (SSM) and *Link Cost based Survivable Multicast* (LCSM) algorithms are using the SSB or LCSB to compute the broadcast trees first and then prune them to obtain multicast trees by applying the pruning method introduced in the previous sections.

TABLE I
Maximum and Variances of Transmit Power of Survivable Broadcast Trees

Network Size	SSB		Ours	
	max	var	max	var
50	5.6263	5.4042	3.6384	1.8772
75	6.2054	5.8452	2.9136	1.1902
100	6.2134	5.2566	2.1678	0.6535
125	6.2206	4.8648	1.7122	0.3795
150	6.2242	4.3571	1.4536	0.2714

First, we evaluate the performance of our broadcasting algorithms. We generate networks with the size of $50, 70, 100, 125$ and 150 nodes respectively. For each network instance, a source node is randomly chosen. In the first simulation, the maximum and variances of transmit power of all wireless nodes are used as the performance metrics to evaluate our algorithm for minimax survivable broadcasting. Table I shows the results.

From Table I, we can see that our algorithm decreases the maximum transmit power dramatically compared with the SSB algorithm. For example, the maximum transmit power assigned by our algorithm is 34.9% of that given by the SSB algorithm. We achieve this improvement because our algorithm is an optimal algorithm in the sense of minimizing maximum transmit power and it is able to avoid using high cost wireless links when constructing the broadcast trees.

In addition, the variances of transmit power assigned by our algorithm are much smaller. This means that the energy consumption by using our broadcast trees is more balanced, a feature which may prolong the lifetime of the routing trees. With the increase of the network size, both the maximum and variance values obtained by our algorithm are decreased since we have more chances to choose low cost links when constructing tree pair in a denser network. However, the increase of network size dose not change results given by the SSB algorithm too much since it may still add high cost links into the tree pair even there exit more low cost links.

In the second simulation, we use almost the same settings. But we measure the total transmit power (rather than the maximum transmit power) to evaluate our algorithm for minimum survivable broadcasting. The results are presented in the Table II. Our algorithm outperforms other algorithms in all cases. In moderately dense networks, such as networks with 100 nodes, the improvement is over 30% compared with the SSB algorithm and about 25% compared with the link cost based algorithm. Our algorithm can perform better over the SSB algorithm because costs of wireless links are added into the trees by the SSB algorithm in each step could be arbitrarily high. Our algorithm can perform better than the link cost based algorithm because the latter ignores WMA.

TABLE II

The Total Transmit Power of Survivable Broadcast Trees

Network Size	SSB	LCSB	Ours
50	114.3106	90.5744	78.4337
75	134.7824	113.4545	89.8671
100	144.8365	131.7537	99.4533
125	154.5760	152.3571	106.5085
150	155.5008	168.7740	111.9993

TABLE III

The Total Transmit Power of Survivable Multicast Trees

Group Size	SSM	LCSM	Ours
20	112.3468	87.6871	71.8006
40	128.1268	107.6246	84.6775
60	135.3598	117.8865	90.9871
80	140.6790	125.3073	95.9970
100	144.8365	131.7537	99.4533

Finally, we run the simulation to evaluate our heuristic for minimum survivable multicasting. We use randomly generated networks with 100 nodes. The multicast sessions are randomly generated. The sizes of multicast groups are set to $20, 40, 60$ and 100 respectively. When the size is set to 100, it actually becomes broadcasting. Same as the second simulation, the total transmit power is counted. From Table III, we see that the solutions given by our algorithm have the least total power in all cases.

VI. CONCLUSIONS

In this paper, we have formulated the minimax survivable multicast/broadcast problems and the minimum survivable multicast/broadcast problems in wireless ad hoc networks. For the minimax problems, we have presented efficient optimal algorithms. For the minimum problems, we have presented simple heuristic algorithms. Our simulation results show that the proposed algorithms perform quite well in practice.

REFERENCES

[1] M. Cagalj, J.P. Hubaux, C. Enz, Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues, *ACM Mobicom'2002*, pp. 172–182.

[2] J. Cartigny, D. Simplot, I. Stojmenovic, Localized minimum-energy broadcasting in ad-hoc networks, *IEEE Infocom'2003*, pp. 2210–2217.

[3] J.H. Chang, L. Tassiulas, Energy conserving routing in wireless ad-hoc networks, *IEEE Infocom'2000*, pp. 22–31.

[4] M. X. Cheng, J. Sun, M. Min, D.-Z. Du, Energy efficient broadcast and multicast routing in ad hoc wireless networks, *IEEE IPCCC'2003*, pp. 87–94.

[5] K. Kar, M. Kodialam, T.V. Lakshman, L. Tassiulas, Routing for network capacity maximization in energy-constrained ad-hoc networks, *IEEE Infocom'2003*, pp. 673–681.

[6] D. Li, X. Jia, H. Liu, Energy efficient broadcast routing in statuc ad hoc wireless networks, *IEEE Transactions on Mobile Computing*, Vol. 3(2004), pp. 144-151.

[7] Q. Li, J. Aslam, D. Rus, Online power-aware routing in wireless ad-hoc networks, *ACM Mobicom'2001*, pp. 97–107.

[8] X.Y. Li, P.J. Wan, Y. Wang, O. Frieder, Constrained shortest paths in wireless networks, *IEEE Milcom'2001*, pp. 884–893.

[9] M. Médard, S.G. Finn, R.A. Barry and R.G. Gallager, Redundant trees for preplanned recovery in arbitrary vertex-redundant or edge-redundant graphs, *IEEE Transactions on Networking*, Vol. 7(1999), pp. 641–652.

[10] A. Srinivas and E. Modiano, Minimum energy disjoint path routing in wireless ad-hoc network, *ACM Mobicom'2003*, pp. 122–133.

[11] J. Tang and G. Xue, Node-disjoint path routing in wireless networks: tradeoff between path lifetime and total energy, *IEEE ICC'2004*, pp. 3812–3816.

[12] R.E. Tarjan, Depth first search and linear graph algorithms, *SIAM Journal on Computing*, Vol. 1(1972), pp. 146–160.

[13] J. E. Wieselthier, G. D. Nguyen and A. Ephremides, On construction of energy-efficient broadcast and multicast trees in wireless networks, *IEEE Infocom'2000*, pp. 585–594.

[14] G. Xue, L. Chen and K. Thulasiraman, Quality of service and quality of protection issues in preplanned recovery schemes using redundant trees, *IEEE JSAC*, Vol. 21(2003), pp. 1332–1345.

[15] G. Xue, R. Gottapu and W. Zhang, Survivable network design using path-based spanners, *submitted for publication*.