Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2007, Article ID 98938, 10 pages
doi:10.1155/2007/98938

Research Article

Tree-Based Distributed Multicast Algorithms for Directional Communications and Lifetime Optimization in Wireless Ad Hoc Networks

Song Guo,[1] Oliver W. W. Yang,[2] and Victor C. M. Leung[1]

[1] *Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada V6T 1Z4*
[2] *School of Information Technology and Engineering, University of Ottawa, Ottawa, ON, Canada K1N 6N5*

Received 1 June 2006; Revised 29 October 2006; Accepted 30 October 2006

Recommended by Xiuzhen Cheng

We consider the problem of maximizing the network lifetime in WANETs (wireless ad hoc networks) with limited energy resources using omnidirectional or directional antennas. Unlike most solutions that use a centralized multicast algorithm, we use graph-theoretic approach to solve the problem in a distributed manner. After providing a globally optimal solution for the special case of single multicast session using omnidirectional antenna, this approach leads us to a group of distributed algorithms for multiple multicast in WANETs using directional antennas. Experimental results show that our distributed multicast algorithms for directional communications outperform other centralized multicast algorithms significantly in terms of network lifetime for both single-session and multiple-session scenarios.

1. INTRODUCTION

There is an increasing interest in wireless ad hoc networks in many application domains where instant infrastructure is needed and no central backbone system and administration (like base stations and wired backbone in a cellular system) exist. Each communicating node in these networks acts as a router in addition to a host in order to communicate with each other over a limited number of shared radio channels. A communication session can be achieved either through a single-hop transmission if the communicating nodes are close enough to each other, or through multiple hops by relaying through intermediate nodes. Since each node in such a network is usually powered by a battery with limited amount of energy, the wireless ad hoc network will become unusable after the batteries are drained. Consequently, energy efficiency is an important design consideration for wireless ad hoc networks.

Over the last few years, energy efficient communication in wireless ad hoc networks with directional antennas has received more and more attention. This is because directional communications can save transmission power by concentrating RF energy where it is needed [1, 2]. On the other hand, the broadcast/multicast communication is also an important issue as many routing protocols for wireless ad hoc networks need this mechanism to maintain the routes between nodes. Therefore, one would be interested in finding an algorithm that would provide the maximum lifetime to the multicast session. The optimization metric is typically defined as the duration of the network operation time until the battery depletion of the first node in the network.

Some work has considered maximizing the network lifetime in a WANET with omnidirectional antennas for a single broadcast session, for example, [3–6], or a single multicast session, for example, [6–10]. The same problem with directional antennas has been studied in [1, 2, 11–14]. It has been proven to be an NP-hard problem [13]. The only exact solution for such difficult problem is the MILP formulation presented in [12]. In [1, 2], the authors extend the minimum energy metric by incorporating residual battery energy based on the observation that long-lived multicast/broadcast trees should consume less energy and should avoid nodes with small residual energy as well. The MLR-MD (for maximum lifetime routing for multicast with directional antenna) algorithm has been proposed recently in [13]. The basic idea of the MLR-MD algorithm is to start with a multicast routing solution first (e.g., a single beam from the source covering all multicast destination nodes) and then iteratively improve lifetime performance of the current solution by identifying the node with the smallest lifetime and revising routing topology as well as corresponding beamforming behavior for an increased network lifetime. All existing solutions are centralized, meaning that at least one node needs global network information in order to construct an energy efficient multicast tree.

In this paper, we explore the energy conservation offered by directional communications for providing long-lived broadcasting/multicasting in wireless ad hoc networks. Our focus is on establishing source-initiated multicast trees to maximize network operating time in energy-limited wireless ad hoc networks with single or multiple multicast sessions. Similar to previous research on the same problems [1–14], we only consider static networks because mobility adds a whole new dimension to the problem and it is out of the scope of this paper.

Unlike the previous work, we would like to design the distributed algorithms that can run on the wireless nodes with limited recourses (i.e., bandwidth, memory, computational capacity, and power). We first use graph-theoretic approach to solve the special case of single multicast session using omnidirectional antenna. This graph-theoretic approach provides us insights into more general case of using directional antennas, and inspires us to produce a group of distributed algorithms. We will extend these solutions to maximize the network lifetime over multiple sessions as well in more realistic scenarios for a wide range of potential civil and military applications. A straightforward approach is that the same trees that were optimized for single session operation are used for the multiple session operations.

The main contribution of this paper is that we present a group of distributed multicast algorithms for the network lifetime maximization problem in WANETs with omnidirectional antennas or directional antennas. In particular, we prove that our distributed algorithm for a single multicast session using omnidirectional antennas is globally optimal. Experimental results also show that our distributed multicast algorithms for directional communications outperform other centralized multicast algorithms significantly in terms of network lifetime for both single-session and multiple-session scenarios.

The rest of this paper is organized as follows. Section 2 develops the system model. Section 3 exploits some important properties of a min-max tree and proposes a group of distributed algorithms for both omnidirectional and directional antenna scenarios. Section 4 demonstrates the performance of our algorithms through a simulation study. Section 5 gives the conclusion on the results.

The following symbols and notations listed in Table 1 will pertain to the remainder of this paper.

Table 1: Symbols and notations.

$G(A,N)$	A directed graph modeling the wireless ad hoc network with a node set N and an arc set A corresponding to the unidirectional wireless communication link
$A(T_s)$	The arc set of a multicast tree T_s
C_v	The child node set of node v
D	The set of destination nodes of a multicast session
M	The set of multicast members including source node and all destination nodes
$N(T_s)$	The node set of a multicast tree T_s
N_v	A set of neighboring nodes of node v located within its maximum transmission range
TN_v	A tree node set in which each node belongs to the multicast tree T_s and lies in the maximum transmission range of node v
T_s	A multicast tree rooted at a source node s
p_{vu}	The RF transmission power needed for the link from node v to node u
$p_{\max}$	The maximum RF transmission power level that a node can choose
p_{recv}	The minimum power needed for reception processing
p_{tran}	The minimum power needed for transmission processing
r_{vu}	The distance between node v and node u
w_{vu}	The weight for an arc(v,u) in graph G
α	The propagation loss exponent
$\delta(T_s)$	The maximum weight of the arc in T_s
$\delta_{\min}$	The minimum $\delta(T_s)$ for all T_s over Ω_M
δ_{LB}^v	The lower bound of $\delta_{\min}$ estimated at node v
δ_{LB}	A lower bound of $\delta_{\min}$
ε_v	The residual battery energy of node v
θ_v	The antenna beamwidth of node v ($\theta_{\min} \le \theta_v \le \theta_{\max}$)
$\theta_v(C_v)$	The minimum possible antenna beamwidth for node v to cover a node set C_v
τ_{vu}	The maximal lifetime of a tree arc
Ω_M	The family of trees T_s of G spanning all the nodes in M

2. SYSTEM MODEL

We model our wireless ad hoc network as a simple directed graph G with a finite node set N and an arc set A corresponding to the unidirectional wireless communication links. Each node is equipped with a directional antenna, which concentrates RF transmission energy to where it is needed. We assume an ideal MAC layer that provides bandwidth availability, that is, frequency channels, time slots, or CDMA orthogonal codes, depending on the access schemes.

Assuming the transmitted energy at node v to be uniformly distributed across the beamwidth θ_v ($\theta_{\min} \le \theta_v \le \theta_{\max}$), the minimal transmitted power required by node v to support a link between two nodes v and u separated by a distance r_{vu} ($r_{vu} > 1$) is proportional to r_{vu}^{α} and beamwidth θ_v, where the propagation loss exponent α typically takes on a value between 2 and 4. Without loss of generality, all receivers are assumed to have the same signal detection threshold, which is typically normalized to one. Then the transmission power p_{vu} needed by node v to reach node u can be expressed as

$$p_{vu} = \frac{r_{vu}^{\alpha} \cdot \theta_v}{360}. \quad (1)$$

Any node $v \in N$ can choose its power level, not to exceed some maximum value $p_{\max}$. In addition to RF propagation, energy may be also expended for transmission processing (on modulation, encoding, etc.) and reception processing (on demodulation, decoding, etc.). For simplicity, these quantities are the same for any node, denoted as p_{tran} and p_{recv}, respectively.

We consider a source-initiated multicast with a multicast set $M = \{s\} \cup D$, where s is the source node and D is the set of destination nodes. All the nodes involved in the multicast form a multicast tree rooted at the node s, that is, a rooted tree T_s, with a tree node set $N(T_s)$, and a tree arc set $A(T_s)$. We define a rooted tree as a directed acyclic graph with a source node with no incoming arcs, and each other node v has exactly one incoming arc. A node with no out-going arcs is called a leaf node, and all other nodes are internal nodes (also called relay nodes). An important property of a rooted tree is that for any node v in the rooted tree T_s, there must exist a single directed acyclic path in the tree.

Let the energy supply $\varepsilon = \{\varepsilon_u \mid u \in N\}$ be the initial energy level associated with each node in G. The residual lifetime τ_{vu} of a tree arc(v, u) is therefore

$$\tau_{vu} = \begin{cases} \dfrac{\varepsilon_v}{p_{vu} + p_{\text{tran}} + p_{\text{recv}}}, & v \neq s, \\ \dfrac{\varepsilon_v}{p_{vu} + p_{\text{tran}}}, & v = s. \end{cases} \quad (2)$$

3. DISTRIBUTED MIN-MAX TREE ALGORITHMS

We first consider the graph representation of the WANET with omnidirectional antennas ($\theta_v = 360$), and assign

$$w_{vu} = \frac{1}{\tau_{vu}} = \begin{cases} \dfrac{r_{vu}^{\alpha} + p_{\text{tran}} + p_{\text{recv}}}{\varepsilon_v}, & v \neq s, \\ \dfrac{r_{vu}^{\alpha} + p_{\text{tran}}}{\varepsilon_v}, & v = s, \end{cases} \quad (3)$$

as the arc weight in the graph. It has been shown in [11] that the single session-based maximum lifetime multicast problem is equivalent to the *min-max tree problem*, which is to determine a directed tree T_s spanning all the multicast members (i.e., $M \subseteq A(T_s)$) such that the maximum of the tree arc weight $\delta(T_s)$ is minimized, where

$$\delta(T_s) \equiv \max\{w_{vu} \mid (v, u) \in A(T_s)\}. \quad (4)$$

Due to their equivalence, we will only investigate the properties of the min-max tree in this section. In the following, we will provide a related theorem that is used to derive our efficient algorithms.

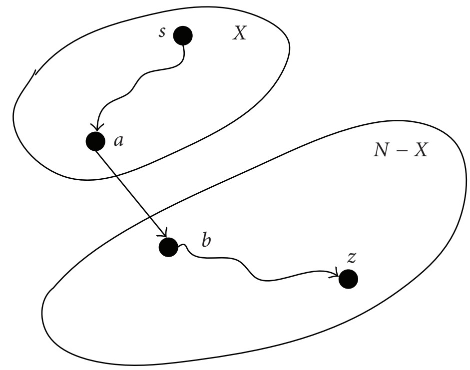

Figure 1: Illustration of the proof for Theorem 1. (The arrow line denotes the directed tree link and arrow curve denotes the directed tree path.)

3.1. A min-max tree theorem

Let T_s^* be the min-max tree and Ω_M is the family of the trees spanning all the nodes in M, we therefore have

$$\delta_{\min} \equiv \delta(T_s^*) \leq \delta(T_s), \quad \forall T_s \in \Omega_M. \quad (5)$$

A tree link (v, u) is called the bottleneck link of the tree T_s if $w_{vu} = \delta(T_s)$.

Theorem 1. *Let (v, u) be the bottleneck link of the multicast tree $T_s \in \Omega_M$. If there exists a node set X, $s \in X$ and $D \cap (N - X) \neq \phi$, such that $w_{vu} \leq w_{xy}$ for any $x \in X$ and $y \in N - X$, then T_s is a* min-max *tree.*

Proof. For any multicast tree $T_s' \in \Omega_M$, let (v', u') be its bottleneck link. Note that there is at least one multicast member z $(z \neq s)$ belonging to $N - X$, that is, $z \in D \cap (N - X)$, since otherwise it contradicts the fact $D \cap (N - X) \neq \phi$. Therefore, there must exist an arc$(a, b) \in A(T_s')$, as shown in Figure 1, connecting X and $N - X$ (i.e., $a \in X$ and $b \in N - X$) in order to satisfy the requirement that there exists a directed path from s to the multicast member z.

From the given condition in Theorem 1, we have $w_{vu} \leq w_{ab}$. Furthermore, since $(a, b) \in A(T_s')$, the bottleneck link weight $\delta(T_s')$ of tree T_s' must be equal to or greater than the weight of any other tree link, for example, link (a, b). That is, $w_{ab} \leq \delta(T_s')$. We thus can derive that $\delta(T_s) = w_{vu} \leq w_{ab} \leq \delta(T_s')$ for any $T_s' \in \Omega_M$, that is, T_s is a min-max tree. □

3.2. Min-max tree algorithm

Theorem 1 immediately suggests an MMT (min-max tree) algorithm for the maximum lifetime multicast problem as follows.

Initially, the multicast tree T_s only contains the source node. It then iteratively performs the following *search-and-grow* procedure until the tree contains all the nodes in M.

The MMT(G, s) algorithm

(1) Initialize T_s by setting $N(T_s) = \{s\}$ and $A(T_s) = \phi$.

(2) Repeat

 (i) *Search phase*:

 Find the arc(v, u) connecting $N(T_s)$ and $N - N(T_s)$ with minimum value w_{vu}, and then add (v, u) into the tree by setting $N(T_s) = N(T_s) \cup \{u\}$ and $A(T_s) = A(T_s) \cup \{(v, u)\}$.

 (ii) *Grow phase*:

 while (exist link (x, y) connecting $N(T_s)$ and $N - N(T_s)$ such that $w_{xy} \le w_{vu}$)

 Add (x, y) into the tree by setting $N(T_s) = N(T_s) \cup \{x\}$ and $A(T_s) = A(T_s) \cup \{(x, y)\}$.

 until $(M \subseteq N(T_s))$.

(3) Obtain the final multicast tree by pruning the broadcast tree T_s.

Algorithm 1: The MMT algorithm.

Search-and-grow procedure

Find the link (v, u) connecting tree node set and nontree node set with minimum weight w_{vu}, and then include it into the multicast tree. Consequently, the tree T_s would grow by including as many nontree links (x, y) as possible into the multicast tree if $w_{xy} \le w_{vu}$ until no more such links can be found.

A pseudocode of the MMT algorithm is given in Algorithm 1.

We will use a ten-node network as a simple example to illustrate the basic tree construction steps in MMT. All nodes are multicast members and node 0 is the source. Each node has the same initial energy supply in a 10×10 square as shown in Figure 2. The maximum transmission range is set to 5 and a propagation loss exponent is $\alpha = 2$.

Step 1. Initially, the tree consists of only the source node 0.

Step 2. In the first iteration, the link $(0, 4)$ connecting node sets $\{0\}$ and $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is found with minimum weight, and then added into the tree as shown by the dark arc in Figure 2(a). There is no any other link included in the tree in the following grow operation.

Step 3. In the second iteration, the link $(0, 7)$ connecting node sets $\{0, 4\}$ and $\{1, 2, 3, 5, 6, 7, 8, 9\}$ is found with minimum weight and added into the tree. The tree then grows by including link $(7, 9)$ as shown by the light arcs in Figure 2(b) since $w_{79} < w_{07}$.

Step 4. In the third iteration, the link $(9, 1)$ connecting node sets $\{0, 4, 7, 9\}$ and $\{1, 2, 3, 5, 6, 8\}$ is found with minimum weight and added into the tree. The tree then grows by including links $(1, 3)$, $(1, 5)$, $(1, 6)$, $(3, 8)$, and $(6, 2)$ since their weights are all less than w_{91}. The min-max tree is eventually obtained as shown in Figure 2(c) with the bottleneck link $(9, 1)$ that is found in the last iteration.

We have the following observations for the *search-and-grow* process.

(1) Only one link is chosen in search phase, for example, link (v, u) as shown in Figure 3, where T_s is a partially constructed multicast tree at the beginning of this search phase.

(2) The weight w_{vu}, denoted as δ_{LB}, must be a lower bound of $\delta_{\min}$ and it is given by

$$\delta_{LB} = \min \{ w_{xy} \mid (x, y) \in A,\ x \in N(T_s),\ y \in N - N(T_s) \}. \tag{6}$$

(3) There would be multiple links to be included into the multicast tree in a subsequent grow phase. A larger constructed multicast tree T_s' is then obtained by the end of the *search-and-grow* process.

(4) The new added links grow from certain nodes (e.g., node v), called grow points, by absorbing as many new links as possible denoted as the tree branches in the darker shaded area in Figure 3. It is interesting to note that there would be multiple such grow points in T_s, for example, node v', if $w_{vu} = w_{v'u'}$.

(5) The sequence of the weight w_{vu} in the min-max tree formation is in an increasing order and the final one in the sequence is equal to $\delta_{\min}$.

(6) After the multicast members are all in the tree, all redundant links, indicated by the dotted arrows in Figure 3, should be pruned from the tree.

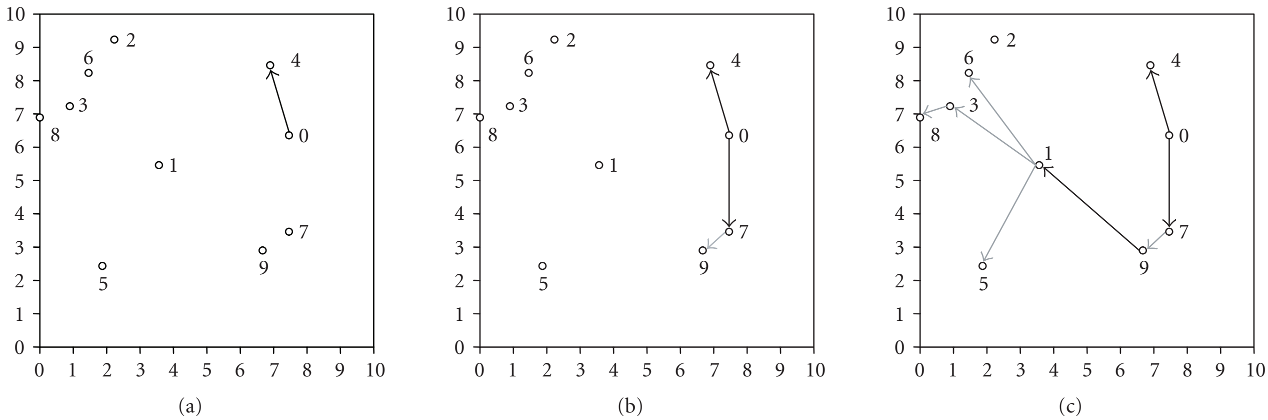

FIGURE 2: Examples of min-max tree construction using the MMT algorithm.

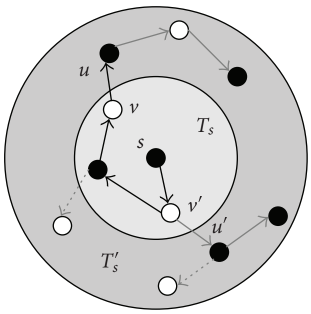

FIGURE 3: Illustration of the *search-and-grow* process. (The dark nodes indicate the multicast members, and light nodes indicate the nonmembers. The dark arrows indicate links that are included into the tree in *search* phases and the light arrows indicate the links that are included into the tree in *grow* phases.)

Finally, it remains to show that the multicast tree discovered by the MMT algorithm is a min-max tree. This is stipulated as follows.

Lemma 1. *At least one bottleneck link of the tree constructed by MMT is included in the tree in a search operation.*

Proof. We prove it by contradiction. Suppose that each bottleneck link, for example, (x, y), of the tree constructed by MMT is added in the tree in a grow operation, and the link (v, u) is included into the tree just in the preceding search operation. From the *search-and-grow* procedure, we have $w_{xy} \leq w_{vu}$. On the other hand, $w_{vu} \leq w_{xy}$ because (x, y) is a bottleneck link of the tree. Therefore, we derive $w_{xy} = w_{vu}$, that is, (v, u) is also a bottleneck link, which contradicts the above assumption that all bottleneck links are included in grow operations. □

Theorem 2. *MMT constructs a* min-max *tree.*

Proof. From the conclusion of Lemma 1, there exists a bottleneck link that is added into the tree in a search operation. Let T_s be the partially constructed multicast tree before entering such search operation. At this situation, the node set $X = N(T_s)$ satisfies the conditions in Theorem 1 and therefore we conclude that the final tree obtained from the MMT algorithm is a min-max tree. □

3.3. The DMMT-OA algorithm

The above analysis would allow us to design distributed algorithm. Our DMMT-OA (distributed MMT algorithm for omnidirectional antenna) uses *search-and-grow* cycles to discover a min-max tree. Such feature is beneficial to implement it in a distributed fashion. We have formulated a data structure to maintain locally the multicast forwarding state at each tree node v: a membership status and the neighborhood table N_v. The membership status indicates if this node is a *source*, *receiver*, or *forwarder*. A node can be both a receiver and forwarder. The neighborhood table N_v contains one entry for each neighbor u within its maximum transmission range. Each entry in the table includes a flag to indicate if the node u is a *tree* node or a *nontree* node. More specifically, if u is a tree node, the relationship to node v is further indicated as *parent*, *child*, or *other* (neither *parent* nor *child*). All tree nodes within N_v are denoted as TN_v.

The distributed algorithm assumes an underlying beaconing protocol which allows each node to be aware of the existence of all its neighbors and the information w_{xy} between any two neighbor nodes x and y. After the neighbor discovery, any node v will create an entry for each neighbor u and set node u as *nontree*. When there is a multicast request, the source will begin to construct a min-max tree as follows.

In a search operation, each tree node v (initially only source node s) first locally calculates an estimation of the lower bound of $\delta_{\min}$ as follows:

$$\delta_{\mathrm{LB}}^{v} = \min\{w_{vu} \mid u \in N_v - TN_v\}. \tag{7}$$

It would unicast a multicast-join-reply (MJREP) message back to its parent with the parameter δ_{LB}^{v} if v is a leaf node, or with the parameter $\min\{\delta_{\mathrm{LB}}^{v},\ \delta_{\mathrm{LB}}^{x} \mid x \text{ is a child node of } v\}$

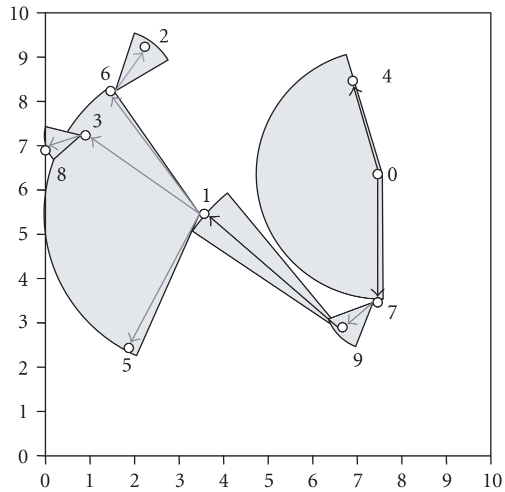

FIGURE 4: DMMT-OA for directional antenna networks.

after collecting all MJREPs from its children if v is a relay node. Note that node v does not send this message if the *parent* flag is not set yet. Furthermore, if v is a multicast member, it also attaches its own address in the MJREP message, which will be propagated to the source to notify its attendance to the multicast.

In this manner, the source will eventually obtain the lower bound δ_{LB} just as given in (6) once all MJREPs are received from its children. If not all multicast members are included in the tree, the source will initiate the grow operation by propagating the multicast-join-request (MJREQ) messages with the parameter δ_{LB} all over the network.

When receiving the first MJREQ message, each intermediate node v will first set the transmitting node (from which MJREQ is received) as *parent* in its neighborhood table, then send back an acknowledgment message which allows its parent node to set itself as a *child*. Node v would also forward MJREQ to any node u only if $w_{vu} \le \delta_{\text{LB}}$. All subsequent duplicate MJREQs (with the same request ID) from other nodes are simply dropped, while the corresponding relationship flag is set as *other* for each of these nodes in node v's neighborhood table. The multicast forwarding state at each tree node v is set as follows. If node v is a destination, it will set it as *receiver*. In addition, if node v is a relay node (i.e., there is at least one entry with a *child* flag in its neighborhood table), it will set its membership status as *forwarder*.

After a short period of time, no more MJREQs would be received at node v. This means that the grow operation completes around node v, and it then goes to the search operation again as described earlier. Finally, a forwarding tree is created in these *search-and-grow* cycles until all members join the tree. After that, a min-max multicast tree is obtained by pruning all the unnecessary links in a distributed fashion from the nonmember leaf nodes.

The above DMMT-OA algorithm for the omnidirectional antenna networks can be straightforward applied for directional communications. Figure 4 shows the result by running the DMMT-OA algorithm for the scenario with $\theta_{\min} = 30$ and $\theta_{\max} = 360$, in which the shaded sectors indicate the areas covered by the directional antennas. This simple process is to reduce the antennas beamwidth of each internal node v to the smallest possible value that provides beam coverage of all its downstream neighbors in the tree, subject to the constraint $\theta_{\min} \le \theta_v \le \theta_{\max}$.

3.4. The DMMT-DA algorithm

The DMMT-DA (distributed MMT algorithm for directional antennas) algorithm is similar in principle to DMMT-OA for the formation of min-max tree, in the sense that new nodes are added into the tree in *search-and-grow* cycles. We must first incorporate the antenna beamwidth into the arc weight as follows:

$$w_{vu} = \begin{cases} \dfrac{r_{vu}^{\alpha} \cdot \theta_v(C_v)}{360 \cdot \varepsilon_v} + \dfrac{p_{\text{tran}} + p_{\text{recv}}}{\varepsilon_v}, & v \neq s, \\ \dfrac{r_{vu}^{\alpha} \cdot \theta_v(C_v)}{360 \cdot \varepsilon_v} + \dfrac{p_{\text{tran}}}{\varepsilon_v}, & v = s, \end{cases} \tag{8}$$

where $\theta_v(C_v) \in [\theta_{\min}, \theta_{\max}]$ is the minimum possible antenna beamwidth for node v to cover all its children C_v in the tree.

Let T_s be the partially constructed tree obtained at the beginning of a search phase. In order to obtain the lower bound provided by (7) in this search phase, each tree node v needs to recalculate the weight w_{vu} using (8), in which the node set C_v is given as follows:

$$C_v = \{x \mid (v, x) \in A(T_s)\} \cup \{u\}. \tag{9}$$

In a grow operation, the new children, for example, node x, of each tree node v, should be included into the tree as many as possible if a tree structure is still maintained and w_{vx} is not greater than the lower bound δ_{LB} that is obtained from the previous search operation, that is,

$$C_v = \arg\max_{C_v} \left| \{x \mid x \in N_v - TN_v \wedge w_{vx} \le \delta_{\text{LB}}\} \right|. \tag{10}$$

Finally, we use the same network configuration in Figure 2 to illustrate the tree construction steps in DMMT-DA.

Step 1. Initially, the tree consists of only the source node 0.

Step 2. In the first iteration, the link $(0, 4)$ is found and added into the tree with minimum beamwidth $\theta_0(\{4\}) = 30$ as shown by the shaded sector in Figure 5(a). There is no any other link included in the tree in the following grow operation.

Step 3. In the second iteration, the link $(4, 1)$ is found and added into the tree with minimum beamwidth in the search operation. The tree then grows by including links $(1, 3)$, $(1, 6)$, $(3, 8)$, and $(6, 2)$ as shown in Figure 5(b), where

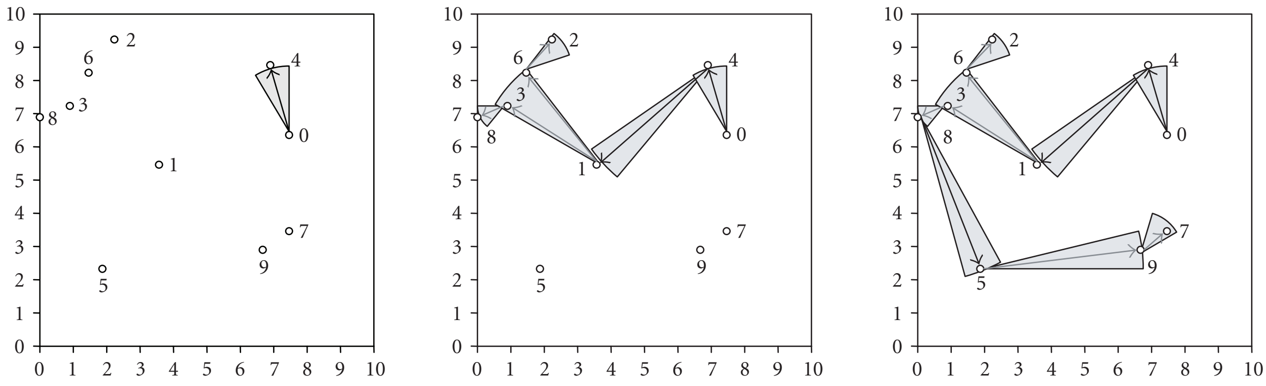

FIGURE 5: Examples of min-max tree construction using DMMT-DA algorithm.

TABLE 2: Parameter values for simulation.

Parameters	Description	Values
n	Network size	100
$\theta_{\min}$	Minimum antenna beamwidth	10°, 30°, 60°, 90°, 180°, and 360°
$\theta_{\max}$	Maximum antenna beamwidth	360°
$p_{\max}$	Maximum RF power level	100
p_{tran}	Minimum power needed for transmission processing	0.1*
p_{recv}	Minimum power needed for reception processing	1*
$E(\varepsilon)$	Mean of the initial energy	500**
$D(\varepsilon)$	Variance of the initial energy	200**
α	Propagation loss exponent	2

*We have also used other values of $(p_{\text{tran}}, p_{\text{recv}}) = (0,0)$ and $(0.01, 0.1)$, and have observed similar simulation results.
**Can be arbitrary units that are consistent with the units of distance.

$\theta_1(\{3,6\}) = \angle 316$[1], $\theta_3(\{8\}) = 30$, and $\theta_6(\{2\}) = 30$, since the weights w_{13}, w_{16}, w_{38}, and w_{62} are all less than w_{41}.

Step 4. In the third iteration, the link $(8,5)$ is found and added into the tree with minimum beamwidth. The tree then grows by including links $(5,9)$, and $(9,7)$. The min-max tree is eventually obtained as shown in Figure 5(c) with the bottleneck link $(8,5)$ that is found in the last iteration.

4. PERFORMANCE EVALUATION

We have evaluated the performance of our distributed algorithms in many network examples. The evaluation is done via simulation written in C++ for the set of heuristic algorithms $I = \{\text{DMMT-OA, DMMT-DA, RB-MIP-}\beta, \text{D-MIP-}\beta\}$, where β is a parameter that reflects the importance assigned to the impact of residual energy[2] [2]. We use RB-MIP-β and D-MIP-β to denote algorithms RB-MIP and D-MIP with different values of β, respectively. We have only considered $\beta = 0, 1$, and 2. In each network example, a number of nodes are randomly generated within a square region 10×10. The values of parameters used in simulation are given in Table 2.

We use the metric *normalized network lifetime* to evaluate and compare algorithm performance. It is defined as the ratio of actual network lifetime obtained using heuristic algorithm to the best solution obtained by choosing the maximum lifetime from all heuristic algorithms. Such metric provides a measure of how close each algorithm comes to provide the longest lifetime tree. Thus allows us to facilitate the comparison of different algorithms over a wide range of network examples.

4.1. Performance in single session scenarios

In experiments based on single sessions, multicast groups of a specified size m ($m = 5, 25, 50, 100$) are chosen randomly from the overall set of nodes. One of the nodes is randomly chosen to be the source. We randomly generated 100 different network examples, and we present here the average over those examples for all cases.

[1] The symbol $\angle abc$ indicates the degree of angle between arc(b,a) and arc(b,c).
[2] The cost of a link (v,u) is defined as $c_{vu} = p_{vu} \cdot (E_v(0)/E_v(t))^{\beta}$, where $E_v(t)$ is the residual energy at node v at time t.

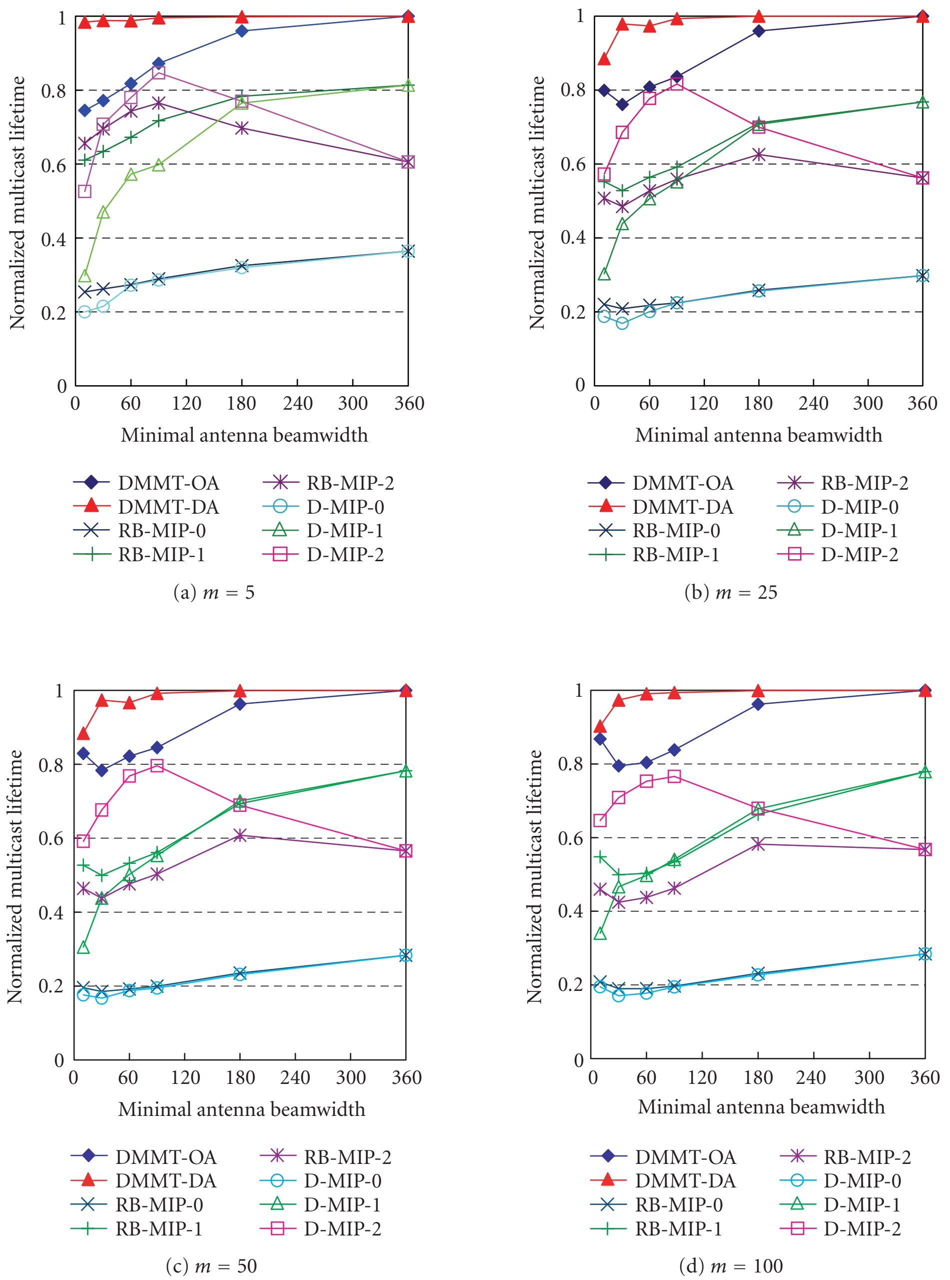

FIGURE 6: Performance comparison based on normalized network lifetime for 100-node networks with single multicast session.

Figure 6 illustrates the mean *normalized network lifetime* as a function of multicast group size and minimal antenna beamwidth for all algorithms. In all cases, DMMT-DA provides much better performance than other algorithms, and its superiority is even greater in network examples with larger $\theta_{\min}$, for example, always within 5% close to the best solution when $\theta_{\min} \geq 30^\circ$. In fact, as guaranteed by Theorem 2, DMMT-DA degenerates into DMMT-OA and therefore both achieve the globally optimal solutions for the case of using omnidirectional antennas.

4.2. Performance in multiple session scenarios

In multiple session-based experiments, multicast requests arrive with interarrival times that are exponentially distributed with rate $1/n$ at each node. Session durations are exponentially distributed with mean 1. Multicast groups are chosen randomly for each session request; the number of destinations is uniformly distributed between 1 and $n-1$. Similarly, we randomly generated a sequence of multicast requests in each scenario and the experimental results are obtained from

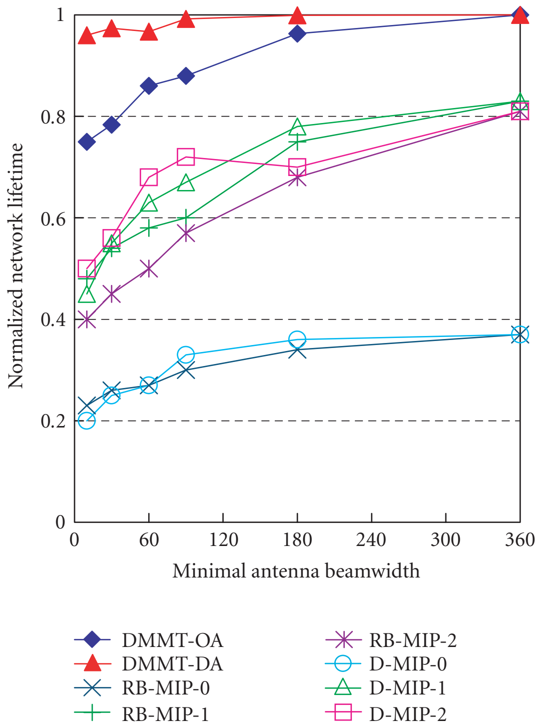

Figure 7: Performance comparison based on normalized network lifetime for 100-node networks with multiple multicast sessions.

100 different scenarios. Note that the same random multicast request sequence is used for each algorithm, thereby facilitating a meaningful comparison of results.

Figure 7 shows how the *normalized network lifetime* changes as the minimal antenna beamwidth varies under multiple multicast sessions for all algorithms. In all cases, both DMMT-OA and DMMT-DA have better performance than other algorithms, and DMMT-DA is even better and always performs very close (within 5%) to the best solutions.

Our key observations from all these experiments are the following.

(1) In single session scenarios, both DMMT-OA and DMMT-DA provide global optimal solutions for WANETs with omnidirectional antennas, and DMMT-DA outperforms all other algorithms for WANETs with directional antennas.

(2) In multiple session scenarios, DMMT-DA shows superior performance than other heuristic algorithms for both directional and omnidirectional antenna networks.

(3) The minimal total energy consumption does not guarantee maximum lifetime either for a network with single multicast session or for a network with multiple multicast sessions, as shown in Figures 6 and 7, respectively.

(4) The revised minimum energy multicast algorithms, like RB-MIP-β/D-MIP-β ($\beta = 1$ and 2), by incorporating residual energy into the cost metric, could provide longer lifetime for both single and multiple session scenarios as shown in Figures 6-7.

5. CONCLUSION

We have presented a group of distributed multicast algorithms for static WANETs with omnidirectional/directional antennas. The correctness of our algorithm in providing a maximum lifetime multicast tree has been proved as well for WANETs with omnidirectional antennas and single session. The performance of our algorithms in terms of network lifetime has been also validated using the simulations over a large number of network examples.

ACKNOWLEDGMENTS

This research was supported in part by the NSERC (Canada) Discovery Grant no. OGP0044286, NSERC Research Grant no. OGP0042878 and an NSERC Postdoctoral Fellowship Award.

REFERENCES

[1] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-limited wireless networking with directional antennas: the case of session-based multicasting," in *Proceedings of IEEE 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, vol. 1, pp. 190–199, New York, NY, USA, June 2002.

[2] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-aware wireless networking with directional antennas: the case of session-based broadcasting and multicasting," *IEEE Transactions on Mobile Computing*, vol. 1, no. 3, pp. 176–191, 2002.

[3] I. Kang and R. Poovendran, "On the lifetime extension of energy-efficient multihop broadcast networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '02)*, vol. 1, pp. 365–370, Honolulu, Hawaii, USA, May 2002.

[4] I. Kang and R. Poovendran, "Maximizing static network lifetime of wireless broadcast ad hoc networks," in *Proceedings of IEEE International Conference on Communications (ICC '03)*, vol. 3, pp. 2256–2261, Anchorage, Alaska, USA, May 2003.

[5] A. K. Das, R. J. Marks, M. El-Sharkawi, P. Arabshahi, and A. Gray, "MDLT: a polynomial time optimal algorithm for maximization of time-to-first-failure in energy constrained wireless broadcast networks," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '03)*, vol. 1, pp. 362–366, San Francisco, Calif, USA, December 2003.

[6] M. X. Cheng, J. Sun, M. Min, and D.-Z. Du, "Energy-efficient broadcast and multicast routing in ad hoc wireless networks," in *Proceedings of the 22nd IEEE International Performance, Computing and Communications Conference (IPCCC '03)*, pp. 87–94, Phoenix, Ariz, USA, April 2003.

[7] B. Wang and S. K. S. Gupta, "On maximizing lifetime of multicast trees in wireless ad hoc networks," in *Proceedings of the International Conference on Parallel Processing (ICPP '03)*, pp. 333–340, Taiwan, China, October 2003.

[8] P. Floréen, P. Kaski, J. Kohonen, and P. Orponen, "Multicast time maximization in energy-constrained wireless networks," in *Proceedings of the Joint Workshop on Foundations of Mobile Computing*, pp. 50–58, San Diego, Calif, USA, September 2003.

[9] L. Georgiadis, "Bottleneck multicast trees in linear time," *IEEE Communications Letters*, vol. 7, no. 11, pp. 564–566, 2003.

[10] S. Guo, V. C. M. Leung, and O. W. W. Yang, “A scalable distributed multicast algorithm for lifetime maximization in large-scale resource-limited multihop wireless networks,” in *Proceedings of the International Wireless Communications and Mobile Computing Conference (IWCMC ’06)*, pp. 419–424, Vancouver, BC, Canada, July 2006.

[11] S. Guo and O. W. W. Yang, “Multicast lifetime maximization for energy-constrained wireless ad hoc networks with directional antennas,” in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM ’04)*, vol. 6, pp. 4120–4124, Dallas, Tex, USA, November-December 2004.

[12] S. Guo and O. W. W. Yang, “Formulation of optimal tree construction for maximum lifetime multicasting in wireless ad hoc networks with adaptive antennas,” in *Proceedings of IEEE International Conference on Communications (ICC ’05)*, vol. 5, pp. 3370–3374, Seoul, Korea, May 2005.

[13] Y. T. Hou, Y. Shi, H. D. Sherali, and J. E. Wieselthier, “Online lifetime-centric multicast routing for ad hoc networks with directional antennas,” in *Proceedings of IEEE the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’05)*, vol. 1, pp. 761–772, Miami, Fla, USA, March 2005.

[14] S. Guo, V. C. M. Leung, and O. W. W. Yang, “Distributed multicast algorithms for lifetime maximization in wireless ad hoc networks with omni-directional and directional antennas,” in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM ’06)*, San Francisco, Calif, USA, November-December 2006.