Online Multicasting for Network Capacity Maximization in Energy-Constrained Ad Hoc Networks

Weifa Liang, *Senior Member, IEEE*, and Xiaoxing Guo, *Student Member, IEEE*

Abstract—In this paper, we present new algorithms for online multicast routing in ad hoc networks where nodes are energy-constrained. The objective is to maximize the total amount of multicast message data routed successfully over the network without any knowledge of future multicast request arrivals and generation rates. Specifically, we first propose an online algorithm for the problem based on an exponential function of energy utilization at each node. The competitive ratio of the proposed algorithm is analyzed if admission control of multicast requests is permitted. We then provide another online algorithm for the problem, which is based on minimizing transmission energy consumption for each multicast request and guaranteeing that the local network lifetime is no less than γ times of the optimum, where γ is constant with $0 < \gamma \leq 1$. We finally conduct extensive experiments by simulations to analyze the performance of the proposed algorithms, in terms of network capacity, network lifetime, and transmission energy consumption for each multicast request. The experimental results clearly indicate that, for online multicast routing in ad hoc wireless networks, the network capacity is proportional to the network lifetime if the transmission energy consumption for each multicast request is at the same time minimized. This is in contrast to the implication by Kar et al. that the network lifetime is proportional to the network capacity when they considered the online unicast routing by devising an algorithm based on the exponential function of energy utilization at each node.

Index Terms—Wireless communication network, approximation algorithm, power awareness, ad hoc networks, energy consumption optimization, multicasting, broadcasting, network lifetime.

1 Introduction

In recent years, multihop wireless ad hoc networks have been receiving significant attention due to their potential applications in civil and military domains. A multihop wireless ad hoc network is dynamically formed by a collection of mobile nodes, equipped with limited-energy batteries. In such a network, the communication between two mobile nodes can be either in a single hop transmission, in which case each of the nodes is within the transmission range of the other or in a multihop transmission, where the message is relayed by intermediate mobile nodes. It is well known that wireless communications consume significant amounts of battery energy [15] and the limited battery lifetime imposes a constraint on network performance; therefore, energy efficient operations are critical to prolong the network lifetime. Extensive studies on the energy conservation for such a network have been conducted. For example, energy efficient routing has been addressed in [2], [11], [28], [8], [30], [31] and maintaining network connectivity with minimum-energy consumption has also been dealt with in [27], [34], [20], [30], [25], [18].

Multicast is a fundamental problem in any telecommunication network including the wireless ad hoc network. Multicasting is an efficient means of one to many communication and is typically implemented by creating a multicast tree. Because of the severe battery power and transmission bandwidth limitations in wireless ad hoc networks, energy-efficient multicast routing can significantly improve the network performance. In this paper, we will focus on the design of energy-efficient routing algorithms for online multicasting in energy-constrained wireless ad hoc networks. Our objective is to maximize the total amount of multicast message data successfully routed by the network without any knowledge on the future message arrivals and generation rates. We refer to this problem as the *network capacity maximization problem* for online multicasting.

1.1 Related Work

In recent years, there has been tremendous research interest in the design of energy-aware broadcast/multicast routing protocols for ad hoc networks. Most of this work has focused on minimizing the total energy consumption [35], [4], [23], [22], [3], [5], [10], [33]. Independent studies conducted by Liang [23] and Cagalj et al. [4] have provided proof that the problem of finding a minimum-energy broadcast tree is NP-hard. In particular, the authors in [4] have shown that it is unlikely to have an approximation algorithm with an approximation ratio better than $\Omega(\log n)$ unless P = NP. This implies that the minimum-energy multicast tree problem (MEMT for short) is NP-hard too. Wieselthier et al. [35] first considered MEMT and proposed greedy heuristics based on Prim and Dijkstra's algorithms. Among their heuristic algorithms, the most efficient one is called the Broadcast Incremental Power (BIP). Liang [23] considered the problem in both symmetric and asymmetric wireless networks and proposed approximation algorithms with

- *The authors are with the Department of Computer Science, The Australian National University, Canberra, ACT 0200, Australia. E-mail: {wliang, Xiaoxing.guo}@cs.anu.edu.au.*

Manuscript received 29 Mar. 2004; revised 30 Nov. 2004; accepted 24 June 2005; published online 17 July 2006.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-0126-0304.

approximation ratios of $O(\log^3 n)$ and $O(n^\epsilon)$, respectively, where ϵ is constant with $0 < \epsilon \leq 1$. Bian et al. [3] later developed an approximation algorithm for the problem in symmetric networks which is $2 \log n$ times of the optimum. Das et al. [10] provided exact solutions to the problem by modeling it into linear integer programming (LP). Cartigny et al. [5] presented a distributed algorithm for the problem, based on the local information of each participating node only. For very special networks like Euclidean networks, Wan et al. [33] proved that the approximation ratio of the BIP is between 13/3 and 12, by exploring geometric structures of Euclidean minimum spanning trees. However, for general networks, Liang [23] showed that the approximation ratio of the BIP is as bad as $\Omega(n)$. For the minimum-energy multicast tree problem Wieselthier et al. [35] provided a heuristic called the Multicast Incremental Power (MIP), and Liang [23] gave an approximation algorithm for both symmetric and asymmetric networks. In particular, for asymmetric networks, the approximation ratio of Liang's algorithm is $O(|D|^\epsilon)$, where $|D|$ is the number of nodes in the destination set and ϵ is constant with $0 < \epsilon \leq 1$.

However, in many practical applications in wireless ad hoc networks, the performance measure of actual interest is not only to optimize the overall energy consumption but also to maximize the lifetime of each node in the network. To avoid the failure of the nodes due to exhaustion of their battery power, energy-efficient multicast routing algorithms should evenly distribute transmission load among the nodes in the network, thereby prolonging the network lifetime. Thus, it is critical to define what we describe as the "lifetime of a network." The definition of the *network lifetime* as the time of the first node failure [8] is a meaningful measure in the sense that a single node failure can cause the network partitioned, and any further service will be interrupted. There have been several studies on prolonging the network lifetime [8], [7], [16]. Chang and Tassiulas [8], [7] proposed maximizing network lifetime by avoiding the use of low power nodes and choosing the shortest path in terms of the energy consumption if the packet rate is given in advance. Kang and Poovendran [16] considered maximizing the network lifetime broadcast tree problem in both symmetric and asymmetric wireless networks. Note that in their algorithm, the authors aim to keep the broadcast tree alive as long as possible until the power in a node expires. Gupta and Kumar [13] discussed the critical power at which a node needs to transmit in order to ensure the network is connected. Li et al. [19] considered a residual power routing that aims to maximize the remaining energy capacity at each node, while the total energy consumption for the routing is bounded. Liang and Yang [24] recently provided an improved algorithm for this problem, while Sankar and Liu [29] provided an analytical solution for it, using the multicommunity flow approach. Toh [32] proposed a *conditional max-min battery capacity routing schema*, which adopts different policies at different stages of battery energy consumption. Note that the above mentioned works [19], [24], [29], [32] are for unicast routing.

In fact, either minimizing the transmission energy consumption or maximizing the network lifetime is insufficient from the point of view of energy conservation and prolonging the network lifetime. A better approach is to take both of these into consideration when building a multicast tree for each multicast request. There have been a few studies that have dealt with a special case, which is the maximizing network lifetime and minimizing energy broadcast tree problem. For example, Wieselthier et al. [36] introduced a heuristic algorithm called BIP(β), which discourages the participation of low residual energy nodes in the broadcast session by defining a cost $w(u,v)$ for each link (u,v) as $w(u,v) = e_{u,v}(E(v)/RE(v))^\beta$, where $E(v)$ is the initial battery capacity at v, $RE(v)$ is the residual battery energy before the current broadcast session, β is a parameter that reflects the importance assigned to the impact of the residual energy, and $e_{u,v} = d^\alpha_{u,v}$ is the energy needed to send a unit length message from node u to node v along the link $\langle u,v \rangle$. Clearly, the link cost is simply the power required to maintain the link when $\beta = 0$. A smaller β places more emphasis on the construction of energy-efficient trees, resulting in quick depletion of batteries at some of the nodes. By contrast, when β is too large, it will put too much emphasis on balancing energy usage throughout the network, while underemphasizing the need for energy efficiency. Kang and Poovendran [16] dealt with this problem using the similar heuristics.

Recently, Kar et al. [17] studied the online unicast routing problem and defined the *network capacity* concept, which aims to maximize the total amount of message data successfully routed by the network without any knowledge on the future message arrivals and generation rates. They devised an online algorithm using the ideas developed for wired networks [26]. However, as they mentioned, energy efficient routing in wireless ad hoc networks is different from routing in wired networks. In wired networks, the constraint is on the consumption of link-bandwidth, while the routing problem in wireless ad hoc networks is constrained by the battery capacities of participating nodes and wireless multicast advantage property [36]. This latter problem cannot be transformed to a link capacity constraint problem by node splitting since the transmission power required for different links is different.

Inspired by their work, in this paper, we consider the network capacity maximization problem for online multicasting. For this problem, we should point out that the approach used by Kar et al. is not applicable as, in their case, it is not difficult to transform a node weighted path to an edge-weighted path equivalently in terms of the weighted sum of the nodes/edges in the path, while, in our case, we aim to build a node-weighted multicast tree with the minimization of the weighted sum of its nodes. Therefore, we cannot simply transform this directly into an edge-weighted multicast tree problem. Fortunately, through the construction of an auxiliary graph, we can transform a node-weighted multicast tree problem in the original network into an edge-weighted multicast tree problem equivalently in an auxiliary graph. Although a similar online multicast problem like the network capacity maximization problem in wired networks has been studied [1], which is to build an undirected, edge-weighted multicast tree with minimizing the sum of weighted-edges in the tree, the approach as well as the analysis technique of competitive ratio in that paper cannot be adopted for our purpose because we are dealing with a directed node-weighted multicast tree.

1.2 Contributions

In this paper, we study the network capacity maximization problem for online multicasting by proposing two online algorithms for it. One is based on an exponential function of energy utilization at each node, which delivers an approximate solution with a guaranteed competitive ratio; another is based on minimizing the transmission energy consumption of the multicast tree under the constraint that the "local network lifetime" is no less than γ times of the optimum after the realization of the request for each multicast request, where the "local network lifetime" means that the failure time of the network after realizing a multicast request and γ is a given constant with $0 < \gamma \leq 1$. We also conduct extensive experiments by simulations to evaluate the performance of the two proposed algorithms in terms of network capacity, network lifetime, and transmission energy consumption for each multicast request. The experimental results show that the proposed algorithms have better performance in comparison with the other two existing algorithms. In particular, the online algorithm based on minimizing transmission energy consumption with $\gamma \approx 0.9$ outperforms all the other algorithms that we studied. This implies, for online multicast routing in ad hoc wireless networks, that the network capacity is proportional to the network lifetime if the local network lifetime is guaranteed while the total transmission energy consumption of each multicast request is minimized. This is in contrast to the implication by Kar et al. [17] that the network lifetime is proportional to the network capacity when they considered online unicast routing.

The rest of the paper is organized as follows: In Section 2, we introduce the wireless communication model and the problem definitions. The algorithm for finding the maximum local network lifetime for a multicast request is also given. In Section 3, an online algorithm for the concerned problem based on the exponential function of energy utilization at each node will be proposed and another online algorithm for the problem based on minimizing the transmission energy consumption of the multicast tree with the guarantee that the local network lifetime is no less than γ times of the optimum is also presented, where γ is constant with $0 < \gamma \leq 1$. In Section 4, we discuss the distributed implementation issues of the proposed centralized algorithms. In Section 5, we conduct extensive experiments by simulation to compare the performance of the proposed algorithms and the other existing algorithms. In Section 6, we conclude the paper.

2 Preliminaries

In this section, we first introduce a communication model for an energy-constrained ad hoc network. We then define several multicast problems on this model with different optimization objectives.

2.1 Wireless Communication Model

We consider source-initiated multicast requests. The wireless ad hoc network is modeled by a directed graph $M = (N, A)$, where N is the set of nodes with $|N| = n$ and there is a directed edge $\langle u, v \rangle$ in A if node v is within the transmission range of u. For the sake of simplicity, we assume the nodes in M are stationary. Nevertheless, the impact of node mobility can be incorporated into this static model since the transmission power at a node can be adjusted to accommodate the new location of the node as necessary. We also assume that all the nodes in the network are equipped with omnidirectional antennas. Thus, we focus only on energy-efficient multicast communications and do not consider other issues like contention for the channel, lack of bandwidth resources, etc. In this model, each node has a number of power levels, it can choose one of its power levels to transmit a message. Specifically, assume that there are l_i adjustable power levels at node $v_i \in N$. Let $w_{i,l}$ be the transmission power at level l, $1 \leq l \leq l_i$. Assume that $w_{i,j_1} \leq w_{i,j_2}$ if $j_1 < j_2$, $1 \leq j_1, j_2 \leq l_i$. Among the l_i power levels, there are a minimum operational power level with power $p_{\min}(v_i)$ and a maximum operational power level with power $p_{\max}(v_i)$, $1 \leq i \leq n$.

For a transmission from node u to node v separated by a distance $d_{u,v}$, to guarantee that v can receive the message from u directly, the transmission power at node u is modeled to be proportional to $c' d^{\alpha}_{u,v}$, assuming that the proportionality constant c' is 1 for notational simplicity, where α is a parameter that typically takes on a value between 2 and 4, depending on the characteristics of the communication medium. In other words, to make v within the transmission range of u, the transmission power $e_{u,v}$ at node u is $e_{u,v} = d^{\alpha}_{u,v}$ at least. Thus, the transmission power at level l of u derived from v must meet $w_{u,l} \geq d^{\alpha}_{u,v}$.

The reachability of a node in wireless ad hoc networks is fully determined by the transmission power at that node. Therefore, there is a trade-off between reaching more nodes in a single hop by using higher power versus reaching fewer nodes in a single hop by using lower power. Note that the nodes in any particular multicast tree do not necessarily have to use the same power level, and a node may use different power levels for various multicast trees in which it participates.

2.2 Online Algorithms versus Offline Algorithms

Suppose that our optimization objective is a metric (here, it is the network capacity). In this paper, we assume the multicast requests arrive one by one and a new arrival cannot occur while the current session is still active. We further assume that there is no knowledge on future multicast request arrivals and generation rates. If the sequence of the multicast requests is known ahead of time, it is possible to develop an offline algorithm to maximize the metric. We thus use the offline algorithm as a benchmark to measure the performance of the online algorithm, for which only one multicast request arrives at a time. We would like the online algorithm to perform well compared with the offline algorithm. The *competitive ratio* of an online algorithm is defined to be the worst-case ratio of the metric of the online algorithm to the metric of the offline algorithm over all instances. Ideally, such a competitive ratio is expected to be a small constant. However, as shown in [1], [26], this is not possible in general. In fact, if there is no control on the lengths of multicast messages, then the competitive ratio can be as bad as $O(n)$. Intuitively, the online algorithm can be made to perform arbitrarily bad by an *adversary* who injects multicast requests into the system if 1) the length of the multicast message can be arbitrary and 2) the algorithm is not allowed to take admission control. Thus, to obtain the competitive ratio, we assume that the algorithm can perform admission control, i.e., it is allowed to reject some multicast requests even if there are enough resources in the network to realize them.

2.3 Problem Definitions

In this section, we define several energy-aware multicast routing problems, which will be used in this paper. Unless otherwise specified, for all scenarios, we only take into account transmission energy consumption and assume that other energy consumption, like that by reception, is negligible because the RF transmission energy consumption is dominant in wireless communications. Also, for each multicast request, we aim to construct a multicast tree to realize it. One fundamental issue related to the tree construction is how to collect the residual energy information at each node if it will be used as an input parameter. We address this issue as follows.

If it is a centralized algorithm, we assume that there is a central point (a node) to hold the network topology information and the up-to-date residual energy information of each node, and the information can be collected by either flooding or a spanning tree rooted at the central point, while such a tree with minimizing the total energy consumption can be constructed. We further assume that the total energy consumption for information collection is negligible, compared with that of a multicast tree for multicasting purpose. Otherwise, constructing an energy efficient broadcast/multicast tree would not make any sense. In the distributed implementation of the centralized algorithms, the residual energy collection is no longer a problem. Instead, each node knows its residual energy and its neighboring nodes through sending and receiving the simple Hello/Ack messages.

2.3.1 The Network Capacity Maximization Problem for Online Multicasting

Given a wireless ad hoc network $M = (N, A)$, let $e_{u,v}$ be the amount of energy consumed by transmitting a unit message from u to v along link $\langle u, v\rangle \in A$ directly. Notice that v is the farthest reachable node from u by the transmission energy $e_{u,v}$ and that all other closer nodes to u can also be reached at no additional energy consumption. Let (s_i, D_i, τ_i) be a multicast request i, where s_i and D_i represent the source node and the destination set, respectively, and τ_i represents the message length of multicast request i, $1 \leq i \leq k$. We assume that the message in each multicast request is not splittable and, therefore, must be routed over a single multicast tree. Thus, if the message of multicast request i is transmitted through node u along link $\langle u, v\rangle$, then the energy consumption at node u for this session is $\tau_i e_{u,v}$. We further assume that multicast requests arrive one by one and there is no knowledge of future multicast request arrivals and generation rates. When a multicast request arrives, the system has to make a decision whether or not the request should be realized immediately. Notice that we assume that all messages are treated the same regardless of their duration and the number of destinations in a multicast request. The *network capacity maximization problem for online multicasting* is to maximize the total amount of message data that is successfully carried by the network without making any assumption on future message arrivals and generation rates. Specifically, let $\tau_1, \tau_2, \ldots, \tau_q$ be the message lengths of the multicast requests in a sequence and let $\tau_{i_1}, \tau_{i_2}, \ldots, \tau_{i_p}$ be the message lengths of those admitted multicast requests with $1 \leq i_j \leq q$ and $i_{j_1} < i_{j_2}$ if $i_{j_1} < i_{j_2}$, $1 \leq j \leq p$, and $1 \leq i_{j_1}, i_{j_2} \leq q$. Then, the amount of data transferred by the network is $\sum_{j=1}^{p} \tau_{i_j}$. Thus, the objective is to maximize $\sum_{j=1}^{p} \tau_{i_j}$ under the assumption that the sequence of multicast requests is not given in advance and the requests arrive one by one randomly. In other words, the network capacity is the number of requests realized by the network before it fails. The network capacity maximization aims to maximize the number of multicast requests that will be realized by the network, where a multicast request is *realized* if there is a multicast tree in the network rooted at the source node and spanning all destination nodes. Otherwise, the request will be rejected, no matter whether there is a multicast tree rooted at the source node and spanning *some* but not all of the destination nodes.

2.3.2 The Minimum-Energy Multicast Tree Problem

Given a wireless ad hoc network $M = (N, A)$, a source node s, and a destination set D $(\subset N)$, the *minimum-energy multicast tree problem* is to construct a multicast tree rooted at the source and spanning the nodes in D such that the sum of transmission energy at nonleaf nodes is minimized. The problem involves the choice of transmission nodes as well as the transmitter-power level at every chosen transmission node. Note that leaf nodes, which do not transmit any messages, do not contribute any energy consumption in the multicast request.

2.3.3 The Maximizing Local Network Lifetime Multicast Tree Problem

Given a wireless ad hoc network $M = (N, A)$, a multicast request with a source node s and a destination set D $(\subset N)$, the *maximizing local network lifetime multicast tree problem* is to construct a multicast tree rooted at the source and spanning the nodes in D such that the minimum residual battery energy at a node is maximized after the realization of the multicast request.

2.4 Algorithm for the Maximizing Local Network Lifetime Multicast Tree Problem

In [21], Liang provided an exact solution for the maximizing local network lifetime multicast tree problem through the reduction of the problem to the maximizing local network lifetime broadcast tree problem, while this latter problem is polynomially solvable.

Assume that the message length of a broadcast request is τ. The basic idea for finding the maximum local network lifetime for the broadcast request follows.

Let $OPT(BT)$ be the optimal solution of the problem and $RE(u)$ be the residual energy at node u when the current broadcast request arrives. Initially, $RE(u) = E(u)$ and $E(u)$ is the battery capacity at node u. We construct a directed weighted graph $G' = (N, A', w)$ for a wireless ad hoc network M. There is a directed edge from u to v in A' if v is within the transmission range of u and $RE(u) - \tau d_{u,v}^{\alpha} > 0$. The weight assigned to the edge is $w(u,v) = RE(u) - \tau d_{u,v}^{\alpha}$, which is the residual battery energy at u if u uses this link to transmit the message of length τ to v. Let T be a directed spanning tree in G' rooted at the source and $C(u)$ be the set of children of node u in T. Then, the residual battery energy at u after realizing the broadcast request is $RE'(u) = \min_{v \in C(u)}\{w(u,v)\}$. Define $RE(T) = \min_{u \in N}\{RE'(u)\}$. Let $\mathcal{T}$ be the collection of all directed spanning trees in G' rooted at the source. Then, the problem is to find a spanning tree T from $\mathcal{T}$ with maximizing $RE(T)$. In other words,

$$OPT(BT) = \max_{T \in \mathcal{T}}\{RE(T)\}.$$

Thus, the directed spanning tree T in G' rooted at the source can be constructed using Prim's algorithm as follows: Let V_s be the set of vertices in the tree including the source s and $V' = N - V_s$. Initially, $V_s = \{s\}$. The algorithm proceeds as follows: It picks up a directed edge $e = \langle u, v\rangle \in V_s \times V'$ from u to v and includes it to the tree if the weight $w(e)$ of e is the maximum one among all the directed edges from $u \in V_s$ to $v \in V'$, and set $V_s = V_s \cup \{v\}$ and $V' = V' - \{v\}$. This procedure continues until $V' = \emptyset$, and T is then obtained. It is easy to show that $OPT(BT) = RE(T)$.

Now, we solve the maximizing local network lifetime multicast tree problem. Given a multicast request (s, D, τ), let $OPT(MT)$ be the maximum local network lifetime after realizing the multicast request. A multicast tree for the request is then obtained by pruning those leaf-vertices in the above spanning tree T that are not in S (corresponding to D) until no such vertices in the resulting tree exist. Let T_D be the resulting multicast tree. Liang [21] has shown $OPT(MT) = \min_{v \in T_D}\{RE'(v)\}$ by the following lemma:

Lemma 1. *Let T_D^* be the optimal multicast tree in G' and $\langle u, v\rangle$ be the minimum weighted edge in it. Let $\langle a, b\rangle$ be the minimum weighted edge in T_D, then $w(a,b) = w(u,v)$, i.e., $RE(T_D^*) = w(u,v) = RE(T_D)$.*

Proof. What we need is to show that $w(a,b) \geq w(u,v)$. Assume that $w(a,b) < w(u,v)$. Let $V(T_a)$ and $V(T_b)$ be the sets of nodes in the two subtrees including a and b after the removal of the directed edge $\langle a, b\rangle$ from T, respectively. We further assume that the source is in $V(T_a)$.

If $u \in V(T_a)$ and $v \in V(T_b)$, then $w(a,b) \geq w(u,v)$; otherwise, the directed edge $\langle u, v\rangle$ instead of the directed edge $\langle a, b\rangle$ will be included in T, following the construction of T. Therefore, the claim holds. Otherwise, both u and v are in either $V(T_a)$ or $V(T_b)$. We assume that they both are in $V(T_a)$. Because $\langle a, b\rangle$ is a directed edge in T_D, there must be at least one node d in D such that $d \in V(T_b)$, and there is a unique directed path P in T_D^* from s to d. Thus, there must exist at least a directed edge $\langle u', v'\rangle$ in P such that $u' \in V(T_a)$ and $v' \in V(T_b)$. Obviously, $w(u', v') < w(a,b)$; otherwise, a new spanning tree T' rooted at the source is obtained by adding $\langle u', v'\rangle$ to and removing $\langle a, b\rangle$ from T, and $rc(T') > rc(T)$. This contradicts the fact that T has the maximum $rc(T)$ in all directed spanning trees in G' rooted at the source. Therefore, $w(u', v') < w(a,b)$. However, $\langle u', v'\rangle$ is an edge in T_D^*, and $w(u', v') < w(a,b) < w(u,v)$, which contradicts that $\langle u, v\rangle$ is the minimum weighted edge in T_D^*. Therefore, the lemma follows. □

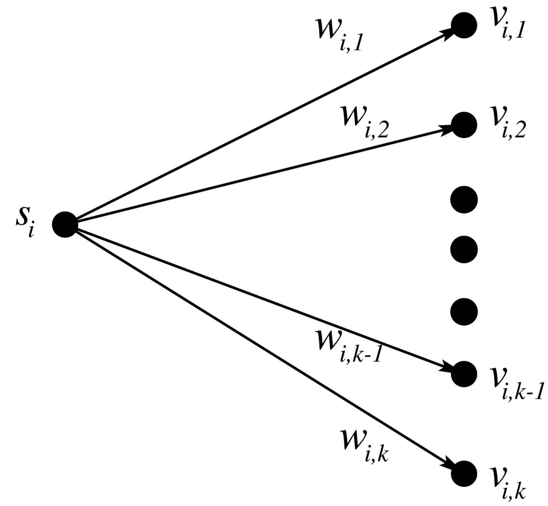


Fig. 1. The widget $G_i = (V_i, E_i)$ for node v_i.

We will later use this algorithm as a subroutine in the design of a proposed algorithm for the network capacity maximization problem for online multicasting. If we define *the cost* of a directed spanning tree as the minimum weight of all its weighted edges, then the maximum local network lifetime broadcast tree can be treated as a sort of a minimum spanning tree. Now, given a multicast request, if there is not such a minimum spanning tree in the current network, a minimum spanning forest then can be found. If there is a tree in the forest including the source node s and the nodes in D, then a multicast tree can be derived easily from the tree by pruning those branches that do not contain any node in D. We shall emphasize that the maximizing local network lifetime multicast tree problem is different from the minimum-energy multicast tree problem in [36], [37], the former is polynomially solvable, while the latter is NP-hard.

3 Multicast Routing Algorithms for Network Capacity Maximization Problem

In this section, we study the network capacity maximization problem for online multicasting by proposing two algorithms.

3.1 Construction of the Auxiliary Directed Graph [23]

For the completeness, we reproduce the construction of an auxiliary graph in [23]. Given an ad hoc network $M(N, A)$, assume that there are l_i adjustable power levels for each node $v_i \in N$, and let $w_{i,1}, w_{i,2} \ldots, w_{i,l_i}$ be the power of l_i adjustable power levels at v_i with $w_{i,l_1} < w_{i,l_2}$, $1 \leq l_1 < l_2 \leq l_i$, and $1 \leq i \leq n$. For node $v_i \in N$, to determine which power level will be used for transmitting messages, a widget $G_i = (V_i, E_i)$ for v_i is built and shown in Fig. 1, $V_i = \{s_i, v_{i,1}, v_{i,2}, \ldots, v_{i,l_i}\}$ and $E_i = \{\langle s_i, v_{i,l}\rangle : 1 \leq l \leq l_i\}$, where s_i represents node v_i, $v_{i,l}$ represents node v_i working at its transmission power level l, the directed edge $\langle s_i, v_{i,l}\rangle$

represents that v_i is working at its the lth power level, and the weight assigned to the edge is the power $w_{i,l}$, $1 \leq l \leq l_i$, and $1 \leq i \leq n$.

Having built the widgets, an auxiliary, edge-weighted, directed graph $G = (V, E, \omega_1)$ is constructed as follows: $V = \cup_{i=1}^{n} V_i$, $E = \cup_{i=1}^{n} E_i \cup E_{dist}$, and $\omega_1 : E \mapsto \mathrm{R}$, where E_{dist} is the set of directed edges and each of the edges is between the two nodes of different widgets, which is defined as follows: For two nodes v_i and v_j with $i \neq j$, there is a directed edge from $v_{i,l}$ to s_j if v_j is within the transmission range of v_i when v_i uses power $w_{i,l}$ to transmit messages. In other words, if $d^{\alpha}_{v_i,v_j} \leq w(v_{i,l})$, then v_j is within the transmission range of v_i, $1 \leq i \leq n$, and $1 \leq l \leq l_i$. Now, consider the weight assignment of edges in G. For each $\langle s_i, v_{i,l} \rangle \in E_i \subset E$, $\omega_1(s_i, v_{i,l}) = w_{i,l}$; for each $\langle v_{i,l}, s_j \rangle \in E_{dist}$, $\omega_1(v_{i,l}, s_j) = 0$.

There is an approximation algorithm for the minimum-energy multicast tree problem in the ad hoc network M, using the auxiliary graph $G(V, E, \omega_1)$, which is described as follows [23].

Find an approximate, directed minimum Steiner tree T in G rooted at s_1 (assume that node $v_1 \in N$ is the source) and spanning the nodes in $D' = \{s_j \mid v_j \in D\}$, using the algorithm in [9]. A multicast tree T_{app} in the ad hoc network M is then derived through the modification to T by merging those edges and nodes in T that are derived from the same mobile node into a single edge and node. The tree T_{app} is an approximate solution to the minimum-energy multicast tree and the total transmission energy consumption in T_{app} is $O(|D|^{\epsilon})$ times of the optimum, where ϵ is constant with $0 < \epsilon \leq 1$. The detailed algorithm for finding T_{app} as well as the construction of the auxiliary graph can be found in [23].

For the sake of convenience, we denote the above algorithm for finding an approximate minimum-energy multicast tree by MEMT, which will be used later as a benchmark for the transmission energy consumption of a multicast tree. Also, in the above discussion, we assumed that a unit length of a multicast message is routed. If the length of the routing message of a multicast request is $\tau\ (> 1)$, then the transmission energy consumption (or the cost) of the multicast tree is $\tau W(T_{app})$, where $W(T_{app})$ is the total transmission energy consumption in T_{app}.

3.2 Algorithm Based on Maximizing the Local Network Lifetime

Here, we propose an online algorithm for the problem of concern based on minimizing the transmission energy consumption of the multicast tree under a constraint that the local network lifetime is no less than γ times of the optimum for each individual multicast request after its realization, where γ is constant with $0 < \gamma \leq 1$. The motivation behind it is that we aim to minimize the total energy consumption for each multicast session, but at the same time, we try to prolong the lifetime of each node (through reducing the energy consumption at the node) by maximizing the minimum residual energy among the nodes.

When a multicast request i arrives, the algorithm finds the maximum local network lifetime first. It then uses the values of the maximum local network lifetime and γ to prune some links in the ad hoc network. It finally constructs a minimum-energy multicast tree in the resulting network so that the local network lifetime of the original network is no less than γ times of the optimum after realizing the multicast request. Assume that a multicast request $(s_i; D_i, \tau_i)$ arrives, the detailed algorithm for the concerned problem is detailed below:

Algorithm BMT(γ): Bound_Lifetime_Network_Capacity_Multicast_Tree $(s_i, D_i, \tau_i, \gamma)$
begin

1. Construct an auxiliary directed graph $G' = (N, A', w)$ for the current ad hoc network using the approach in Section 2.4, where the weight $w(u, v)$ assigned to the edge $\langle u, v \rangle$ is $RE(u) - \tau_i d^{\alpha}_{u,v}$. Find the maximum local network lifetime $OPT(MT)$ for multicast request i using G'.
2. Construct an auxiliary directed graph $G = (V, E, \omega_1)$ using the approach in Section 3.1, assign each directed edge $\langle s_j, v_{j,l} \rangle$ with weight $\omega_1(s_j, v_{j,l}) = RE(v_j) - \tau_i w(v_{j,l})$ and every other directed edge with weight zero.
3. An auxiliary graph $G_1 = (V, E_1, \omega_1)$ is induced from G by removing those edges $\langle s_j, v_{j,l} \rangle$ and the vertex $v_{j,l}$ adjacent to s_j if $\omega_1(s_j, v_{j,l}) < \gamma OPT(MT)$.
4. Another auxiliary graph $G_2 = (V, E_2, \omega_2)$ that has the same topology as G_1 is constructed. The only difference is that each directed edge $\langle s_j, v_{j,l} \rangle$ in G_2 is now assigned a new weight $\omega_2(s_j, v_{j,l}) = w(v_{j,l})$.
5. Find an approximate multicast tree T_{app} using algorithm MEMT, where G_2 is the auxiliary graph.

end

For the sake of convenience, we refer to the above algorithm as BMT(γ). Compared with algorithm MEMT, algorithm BMT(γ) aims to minimize the total transmission energy consumption under the constraint that the local network lifetime is no less than γ times of the optimum after the realization of the request. The later experimental results demonstrate that BMT(γ) with $\gamma \approx 0.9$ outperforms all the other algorithms in terms of the performance measure.

3.3 Algorithm Based on an Exponential Function of Node Energy Utilization

We assume that multicast requests are indexed in the order they arrive. Let $(s_i; D_i, \tau_i)$ be the ith multicast request and $RE_i(v)$ be the residual battery energy at node v when multicast request i arrives, $1 \leq i \leq k$. Assume that $E(v)$ is the initial battery capacity at node v. Obviously, $RE_1(v) = E(v)$. Let $\alpha_i(v) = 1 - RE_i(v)/E(v)$ be the fraction of energy at node v that has been used when multicast request i arrives, which will be referred to as *energy utilization* at node v, $1 \leq i \leq k$. Let $e_{max} = \max_{\langle u,v \rangle \in A}\{e_{u,v}\}$ and $e_{min} = \min_{\langle u,v \rangle \in A}\{e_{u,v}\}$, which are, respectively, the maximum and minimum energy expended by a message of unit length when traversing some link in the network. Let $\rho = e_{max}/e_{min}$. Define $\lambda = 2n\rho$ and $\sigma = (n-1)e_{max}|D_i|^{\epsilon}$, where ϵ is constant with $0 < \epsilon \leq 1$ and $1 \leq i \leq k$.

In the following, we propose an algorithm for the network capacity maximization problem for online multicasting. The basic idea of the proposed algorithm is to use an auxiliary graph $G(V, E, \omega_2)$ to generate an approximate multicast tree with minimizing the transmission energy consumption of it, where $G(V, E, \omega_2)$ has the same topology as the auxiliary graph $G(V, E, \omega_1)$ in Section 3.1 but with a different edge-weighted function ω_2, i.e., for each directed edge $\langle s_j, v_{j,l}\rangle \in E_j \subset E$, $\omega_2(s_j, v_{j,l}) = w(v_{j,l})(\lambda^{\alpha_i(v_j)} - 1)$, where $w(v_{j,l})$ is the power when node v_j sets its power level at l, $1 \le l \le l_j$; for each $\langle v_{j,l}, s_x\rangle \in E_{dist}$, $\omega_2(v_{j,l}, s_x) = 0$. The proposed algorithm for processing multicast request i is as follows:

Algorithm MCM: Max_Capacity_Multicast_Tree $(s_i, D_i, \tau_i, \sigma)$
begin

1. An auxiliary directed graph $G = (V, E, \omega_1)$ is constructed when a multicast request $(s_i; D_i, \tau_i)$ arrives, where $\omega_1(s_j, v_{j,l}) = w(v_{j,l})$ and $\omega_1(v_{j,l}, s_x) = 0$. A subnetwork $G_1(V, E_1, \omega_2)$ of G is obtained after the removal of those edges $\langle s_j, v_{j,l}\rangle$ from G with $\omega_1(s_j, v_{j,l}) > \frac{RE_i(v_j)}{\tau_i}$.
2. Assign each link $\langle s_j, v_{j,l}\rangle$ in G_1 a new weight $\omega_2(s_j, v_{j,l}) = w(v_{j,l})(\lambda^{\alpha_i(v_j)} - 1)$ and, for each link $\langle v_{j,l}, s_x\rangle$ in G_1, $\omega_2(v_{j,l}, s_x) = 0$.
3. Find an approximate, minimum Steiner tree T_i in G_1 rooted at s_i and spanning the vertices that correspond to the nodes in D_i, using algorithm MEMT in [23]. If no such tree exists, this request is ignored and will not be realized; otherwise, an approximate multicast tree T^i_{app} is obtained.
4. Let $W(T^i_{app})$ be the sum of transmission power of vertices in T^i_{app}. If $W(T^i_{app}) \le \sigma$, route the multicast message along the tree T^i_{app}; otherwise, reject the request.

end

For the sake of simplicity, we refer to the above algorithm as MCM. In this algorithm, Step 1 is to construct an auxiliary graph and assign a weight to each directed edge. If there is a directed edge $\langle s_j, v_{j,l}\rangle$ with $\omega_1(s_j, v_{j,l}) > \frac{RE_i(v_j)}{\tau_i}$, it will be removed from the graph because if node v_j as a relay node forwards this multicast message, it will run out of energy before the message is delivered to any of its neighbors. Step 2 is to assign a new weight for each link in the resulting graph obtained at Step 1. The weight on edge $\langle s_j, v_{j,l}\rangle$ is determined by two factors, one is the power level of v_j that will be used for transmission (in this case it is the power level l), and another is the energy utilization $\alpha_i(v_j)$ at v_j. The weight will increase when $\alpha_i(v_j)$ increases or v_j is set at a higher power level. This means that the algorithm tries to avoid links that require high energy for transmission and nodes where the residual energy fractions are low. Step 3 is to find an approximate, minimum Steiner tree in terms of the weighted sum of the edges in the tree, which aims to minimize the total transmission energy consumption for the multicast request. Step 4 employs an admission control mechanism, which will reject the request if the cost for routing the message of a multicast request is too high (above a given threshold σ). Without this option to reject, an adversary can inject messages that consume too much resource destroying the competitive ratio of the algorithm. However, as stated in [17], this is not of practical consequence when messages are generated at random or by an adversary who do not know the routing policy. In practice, rejecting a message when sufficient energy is available is usually unacceptable. The later experimental results indicate that although setting σ to its theoretically determined value might improve the capacity performance, setting σ to infinity results in promising performance with respect to both the capacity and the network lifetime.

In what follows, we analyze the competitive ratio of algorithm MCM. Let $L(k)$ be the total length of messages successfully routed by algorithm MCM till the arrival of multicast request k and $L_{opt}(k)$ the total length of messages successfully routed by an optimal offline algorithm till the arrival of multicast request k. Following the analysis similar to the one given by Kar et al. [17], we have the following theorem.

Theorem 1. *In a wireless ad hoc network $M(N, A)$, for all multicast request (s_i, D_i, τ_i), let $\tau_i \le \frac{\min_{v\in N}\{E(v)\}}{e_{max}\log\lambda}$, $1 \le i \le k$. Then, $\frac{L(k)}{L_{opt}(k)} \ge \frac{1}{1+2K^{\epsilon}\log\lambda}$. In other words, the online algorithm delivers a solution with competitive ratio of $O(K^{\epsilon}\log\frac{ne_{\max}}{e_{\min}})$, where $K = \max_{1\le i\le k}\{|D_i|\}$ and ϵ is constant with $0 < \epsilon \le 1$.*

Proof. See the Appendix. □

From the theoretical point of view, despite the fact that the competitive ratio of algorithm MCM is pessimistic, through the experimental simulations its actual competitive ratio is much better than that.

Remarks. Algorithm MCM is to find an approximate, minimum directed Steiner tree in a directed graph because the weight assigned to each directed edge $\langle u, v\rangle$ is determined by the power level of u and the energy utilization ratio at u at that moment. Thus, the weights of the edges between u and v are not the same. In addition, although the approximation algorithm for finding an approximate undirected Steiner tree provides a better approximation ratio, clearly it cannot be employed in this case.

The proposed algorithm here is similar to the one given by Kar et al. [17] for online unicast routing, the difference is that we employ algorithm MEMT to find an approximate, minimum-energy multicast tree for online multicasting, while they use the shortest path algorithm to find an optimal unicast routing path. It is not difficult to see that the proposed algorithm is more generic, and their algorithm can only deal with a special case of our concerned problem —online unicast routing.

3.4 Extension to the Model where the Power Is Infinitely Adjustable

So far, we have assumed that the transmission power at each node is finitely adjustable. In the following, we show how the proposed algorithms can be extended to a scenario where the transmission power at each node is infinitely adjustable, through a reduction in the following.

Associated with every node v, let $N(v)$ be the neighboring set of v in the current network topology, i.e., the set of the mobile nodes that are within the transmission range of v when v uses its currently maximum power to broadcast a message. Then, there are at most $|N(v)|$ power levels at v, and for each $u \in N(v)$, the power at the level derived from u

is $d_{v,u}^{\alpha}$, where $d_{v,u}$ is the distance between u and v. Clearly, $|N(v)| \leq n-1$.

4 Distributed Implementations

In this section, we discuss the distributed implementation issues of the centralized algorithms presented in this paper. Such a discussion is important for most practical situations where the global topology knowledge is not immediately available to all nodes in the network. Furthermore, the distributed implementation is important in instances where the network topology may change frequently. For the purposes of this discussion, we assume that each node has only local topology knowledge, i.e., the neighboring nodes that are within its transmission range and the distance between it and each of its neighbors. This can be easily found by each node employing a physical-layer probing mechanism using incremental power level increases [34].

4.1 Distributed Implementation of Algorithm MEMT

Algorithm MEMT in [23] serves as a basic building block for the two proposed algorithms. Here, we discuss its distributed implementation. The basic strategy is to embed the auxiliary graph $G(V, E, \omega_1)$ into the physical topology of the network $M(N, A)$, which can be done easily by mapping the widget $G_i(V_i, E_i)$ into node $v_i \in N$, $1 \leq i \leq n$. Each physical node $v_i \in N$ will simulate all the nodes in the widget, and the physical link from $v_i \in N$ to $v_j \in N$ will be used to simulate the links between the nodes in the two widgets, $1 \leq i$ and $j \leq n$. Therefore, any distributed algorithm based on the auxiliary graph can be simulated by the physical network. Following the construction of the auxiliary graph, it can be seen that its diameter is no more than twice that of the physical network.

We now briefly describe the distributed implementation of algorithm MEMT. The key is to provide an efficient distributed implementation of the algorithm for finding an approximate, directed minimum Steiner tree in [9]. The solution delivered by this algorithm is $O(|D|^{\epsilon})$ times of the optimum, where D is the destination set and ϵ is a given constant with $0 < \epsilon \leq 1$. However, to the best of our knowledge, so far, an efficient distributed implementation of the algorithm has not yet been seen. Instead, a simpler algorithm based on single-source shortest paths is employed, which delivers an approximation solution within $|D|$ times of the optimum. Compared with the algorithm in [9], the solution delivered by this latter algorithm may not be as accurate as the former one, but it runs much faster and has an efficient distributed implementation [6]. Hence, the distributed implementation of algorithm MEMT is described as follows.

We first use $M(N, A)$ to simulate the auxiliary graph $G(V, E, \omega_1)$. We then run the algorithm in [6] to find an approximate, directed Steiner tree in G. We finally modify the Steiner tree into an approximate multicast tree in M by a series of local merging operations, which merge those nodes and links derived from the same mobile node into a single node and link.

4.2 Distributed Implementations of Algorithms BMT(γ) and MCM

The distributed implementation of algorithm BMT(γ) can proceed within two stages.

The first stage is to find the optimal local network lifetime $OPT(MT)$ for a given multicast request, and the second stage is to find an approximate, multicast tree in the resulting network, which is induced from M by pruning some of its links using the values of $OPT(MT)$ and γ. It can be seen that an algorithm for finding a directed minimum spanning tree (forest) in $M(N, A)$ is employed in order to find the $OPT(MT)$. Thus, $OPT(MT)$ can be found using the distributed algorithm [14] in the physical network M. Once $OPT(MT)$ is available, the source node can broadcast it to all the other nodes, using the directed minimum spanning tree built.

Now, we proceed to the second stage. For each node $v_i \in N$, the algorithm first prunes some of its neighboring nodes using $OPT(MT)$ and γ, and then builds a widget $G(V_i, E_i)$ using the existing neighboring nodes, $1 \leq i \leq n$. Thus, an auxiliary graph $G_2(V, E_2, \omega_2)$ is constructed. According to the discussion in Section 4.1, an approximate, minimum-energy multicast tree in M can be found by using the distributed algorithm for MEMT.

The distributed implementation of algorithm MCM is identical to that of algorithm MEMT and, therefore, we omit it here.

5 Performance Evaluation

In this section, we evaluate the performance of the proposed algorithms against the existing algorithms through the simulations. Specifically, we first evaluate our algorithms BMT(γ) and MCM against the existing algorithms MEMT in [23] and MIP(β) in [36], where β is an appropriately chosen constant. We then study the impact of γ on the performance of BMT(γ).

5.1 Comparison with Existing Algorithms

We consider networks of 20, 40, 60, 80, and 100 nodes randomly distributed in a $100 \times 100 m^2$ region. For each different network size, 50 network instances are generated by the NS-2 simulator. The energy required to transmit a message of unit length from node u to node v directly is $d_{u,v}^2$, where $d_{u,v}$ is the distance between u and v. The initial battery energy $E(v)$ at each node $v \in N$ is drawn from a uniform distribution between 2.8 and 5.6 *mWh*. We assume that the transmitter at each node has six power levels: 1, 5, 20, 30, 50, and 100 *mW* (as offered by a Cisco Aironet card [12]), and, for this configuration, the corresponding transmission ranges of each node are 5.0, 11.18, 22.36, 27.38, 35.35, and 50 meters. We also assume each node knows the residual energy at all other nodes.

In order to speed up the running time of the proposed algorithms, we employ a simple approach based on the single-source shortest path algorithm to find an approximate, minimum Steiner tree in the auxiliary graph, instead of a more accurate approximation algorithm in [9] that takes a longer running time. Even so, the experimental results indicate BMT(γ) outperforms all the other algorithms in

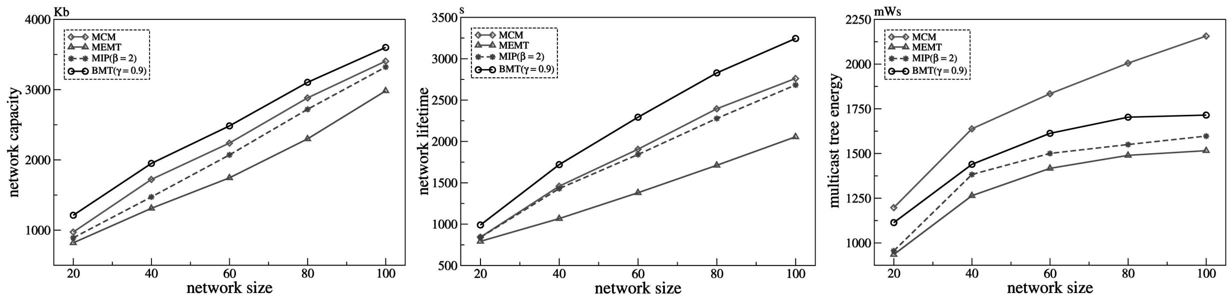


Fig. 2. Mean of the network capacity, network lifetime, and the multicast tree energy consumption out of 50 network instances for multicast group size = 25 percent of the network size; $\alpha = 2$.

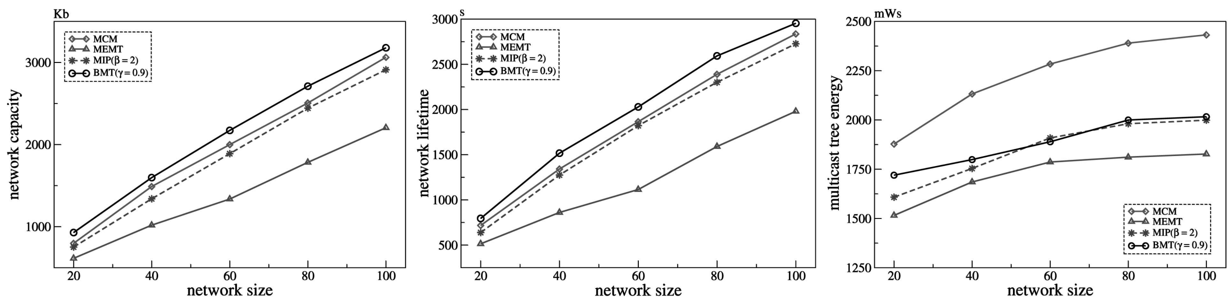


Fig. 3. Mean of the network capacity, network lifetime, and the multicast tree energy consumption out of 50 network instances for multicast group size = 50 percent of the network size; $\alpha = 2$.

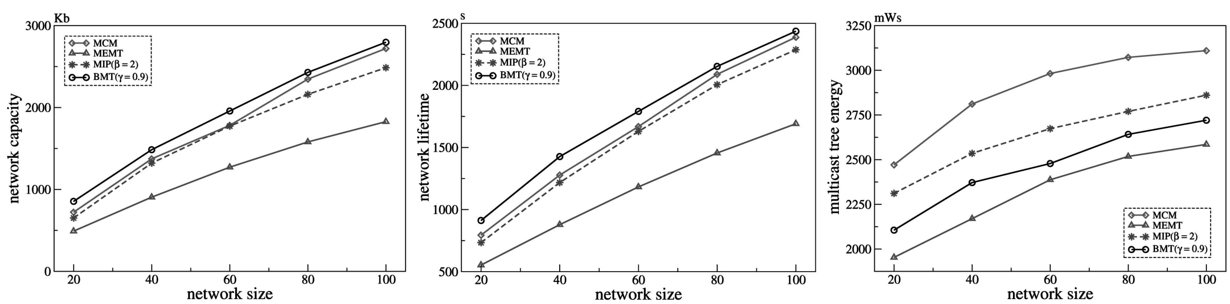


Fig. 4. Mean of the network capacity, network lifetime, and the multicast tree energy consumption out of 50 network instances for multicast group size = 75 percent of the network size; $\alpha = 2$.

terms of the network capacity and the network lifetime when $\gamma \approx 0.9$.

We compare our algorithms with two known algorithms MEMT [23] and MIP(β), where β is an appropriately chosen constant and a larger β reflects more emphasis on balancing the energy use among the nodes throughout the network [36]. To implement algorithm MIP(β), we applied the sweep operation to the tree produced by MIP to eliminate unnecessary transmissions [37], where β is set to be 2, which is claimed by the authors of [36] to be the best in order to achieve a high degree of load balancing that keeps almost all of the nodes alive for a relatively long period when $0.5 \leq \beta \leq 2$.

In our experiment, following the similar setting as given in [17], we injected 1,000 multicast requests of various message lengths into the network one by one. We assumed that each multicast request carries a constant bit rate (1 *Kbps*) traffic and the message length of the request is randomly assigned between 1 and 10 *Kbits*. We also assumed that the source node and the destination set of each multicast request are chosen randomly. To measure the network capacity, we did not terminate the simulation until all the 1,000 requests had been processed, i.e., each request is either realized or rejected. As for the implementation of algorithm MCM, the appropriate value of λ is chosen for different network sizes, and we set $\sigma = \infty$, which represents an absence of admission control. Therefore, a request is accepted only when all destinations in the request can be reached, and a request is rejected due to insufficient resources (energy) for realizing it.

Fig. 2, Fig. 3, Fig. 4, and Fig. 5 illustrate the network capacity, network lifetime, and the transmission energy consumption of the multicast tree when the ratios between the number of destination nodes in a multicast request and the network size are 25, 50, 75, and 100 percent (broadcast), respectively. We note that, among the algorithms that we studied, BMT(γ) with $\gamma \approx 0.9$ has the best performance in maximizing network capacity and the network lifetime in all testing cases, while its overall energy consumption for

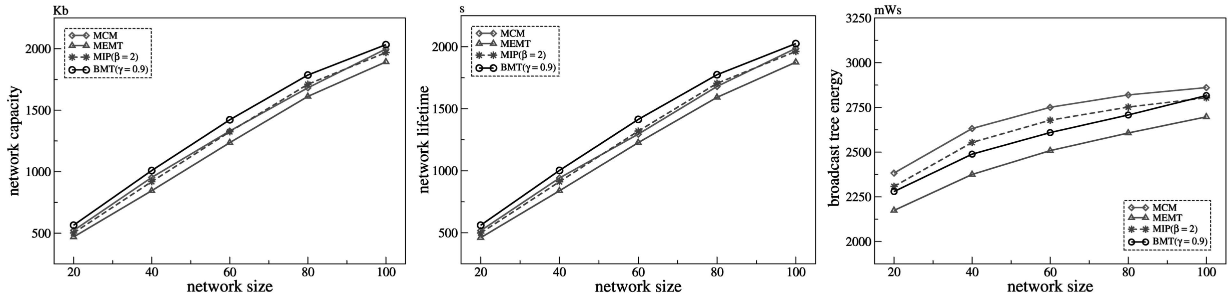


Fig. 5. Mean of the network capacity, network lifetime, and the broadcast tree energy consumption out of 50 network instances for broadcast; $\alpha = 2$.

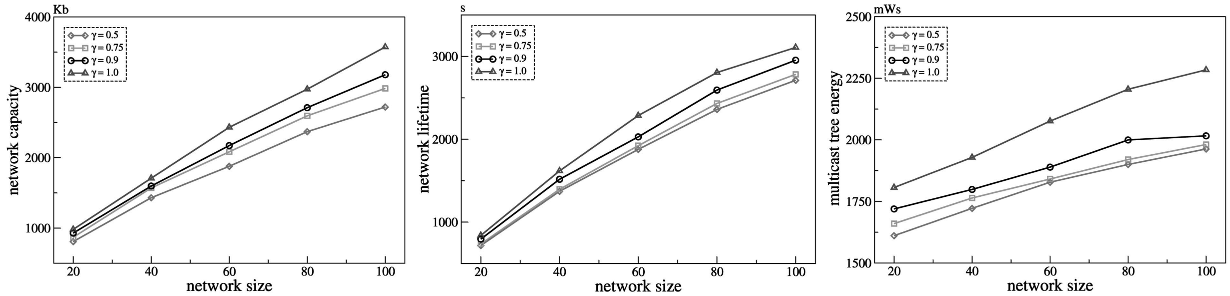


Fig. 6. Mean of the network capacity, network lifetime, and the multicast tree energy consumption out of 50 network instances for multicast group size = 50 percent of the network size under various values of γ; $\alpha = 2$.

each multicast request remains relatively low, compared to that of MCM or MIP(β) with $\beta = 2$. The average fractions of multicast requests successfully realized out of the total 1,000 multicast requests with arbitrary multicast group sizes of MEMT, MIP(β) with $\beta = 2$, MCM, and BMT(γ) with $\gamma \approx 0.9$ are 25.79, 31.10, 40.16, and 46.98 percent, respectively.

5.2 Impact of γ on the Performance of BMT(γ)

We now analyze the impact of different values of γ on the performance of BMT(γ). The constant γ with $0 < \gamma \leq 1$ provides a performance guarantee of BMT(γ) in terms of the local network lifetime. Fig. 6 illustrates the impact on the performance of BMT(γ) under various γs. The network capacity increases when the longer network lifetime is given (a larger γ), and this also leads to a higher total energy consumption throughout the multicast trees. Although BMT(γ) with $\gamma = 1$ provides the maximum local network lifetime and larger network capacity, the transmission energy consumption of the multicast tree grows significantly compared with the case when $\gamma \approx 0.9$.

Through the experimental simulation, it is not difficult to find that the transmission energy consumption of the multicast tree increases with the growth of γ and the growth rate of the transmission energy consumption of the multicast tree is relatively slow until γ is over 0.9. After that, the transmission energy consumption of the multicast tree grows significantly. The experimental results imply that, for online multicasting in ad hoc networks, the network capacity is proportional to the network lifetime.

6 Conclusions

In this paper, we have considered the network capacity maximization problem for online multicasting by proposing energy-efficient online algorithms. We have also conducted extensive experiments using various simulations. The experiments demonstrate that the proposed algorithms outperform the existing algorithms in terms of both network capacity and network lifetime. The experimental results imply that, for online multicast problems in energy-constrained ad hoc networks, the network capacity is proportional to the network lifetime if the transmission energy consumption for each multicast request is at the same time minimized.

Appendix

Proof of Theorem 1

The competitive analysis is along the same line as the analysis presented in [17]. Associate a cost $c(v_j)$ with each node $v_j \in N$. The cost $c(v_j)$ of v_j before the arrival of multicast request i is defined as

$$c_i(v_j) = E(v_j)(\lambda^{\alpha_i(v_j)} - 1). \quad (1)$$

We start with the following lemma:

Lemma 1. *When the kth multicast request arrives, the total cost of the nodes in the network is bounded by $2(n-1)e_{\max}L(k)\log\lambda$, i.e., $\sum_{v_j \in N} c_k(v_j) \leq 2(n-1)e_{\max}L(k)\log\lambda$.*

Proof. Let $S(k)$ be the set of multicast requests realized successfully by algorithm MCM till the arrival of multicast request k. Consider the multicast request k' in $S(k)$, for any $v_j \in N$, we have

$$\begin{aligned} c_{k'+1}(v_j) - c_{k'}(v_j) &\leq E(v_j)(\lambda^{\alpha_{k'+1}(v_j)} - \lambda^{\alpha_{k'}(v_j)}) \\ &= E(v_j)\lambda^{\alpha_{k'}(v_j)}(\lambda^{(RE_{k'}(v_j) - RE_{k'+1}(v_j))/E(v_j)} - 1) \\ &= E(v_j)\lambda^{\alpha_{k'}(v_j)}(\lambda^{\tau_{k'} w(v_{j,l})/E(v_j)} - 1) \quad (2) \\ &= E(v_j)\lambda^{\alpha_{k'}(v_j)}(2^{\tau_{k'} w(v_{j,l}) \log\lambda/E(v_j)} - 1). \quad (3) \end{aligned}$$

In (2), we assume that v_j uses its power level l for transmitting the multicast request k'. Following the assumption of $\tau_{k'} \le \frac{\min_{v\in N}\{E(v)\}}{e_{\max}\log\lambda}$ for any multicast request k', we have $(\tau_{k'} w(v_{j,l})\log\lambda)/E(v_j) \le 1$ and $2^x - 1 \le x$ for all x with $0 \le x \le 1$. We then have

$$c_{k'+1}(v_j) - c_{k'}(v_j) \le \lambda^{\alpha_{k'}(v_j)} \tau_{k'} w(v_{j,l}) \log\lambda. \quad (4)$$

Let $M(k')$ be the multicast tree on which the multicast request k' is realized. Note that, since multicast request k' is accepted by algorithm MCM, therefore, following Step 4 of MCM, the weighted sum of the edges in $M(k')$ is

$$\sum_{\langle u,v\rangle\in M(k')} \omega_2(u,v) \le \sigma = (n-1)e_{\max}. \quad (5)$$

Thus, we have,

$$\begin{aligned}\sum_{v_j\in N}(c_{k'+1}(v_j) - c_{k'}(v_j)) &= \sum_{\langle u,v\rangle\in M(k')}\omega_2(u,v)\\ &\le \sum_{\langle v_j,v_x\rangle\in M(k')}\tau_{k'} w(v_{j,l})\log\lambda\\ &= \tau_{k'}\log\lambda \sum_{\langle v_j,v_x\rangle\in M(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)}-1)\\ &\quad + \tau_{k'}\log\lambda\sum_{\langle v_j,v_x\rangle\in M(k')} w(v_{j,l})\\ &\le \tau_{k'}\log\lambda\sigma + \tau_{k'}\log\lambda(n-1)e_{\max}\\ &\le 2\tau_{k'}(n-1)e_{\max}\log\lambda.\end{aligned} \quad (6)$$

Note that if $k' \notin S(k)$, we have $c_{k'+1}(v_j) - c_{k'}(v_j) = 0$. Also, note that $c_1(v_j) = 0$ for all $v_j \in N$. So,

$$\begin{aligned}\sum_{v_j\in N} c_k(v_j) &= \sum_{k'=1}^{k-1}\sum_{v_j\in N}(c_{k'+1}(v_j) - c_{k'}(v_j))\\ &= \sum_{k'\in S(k)}(c_{k'+1}(v_j) - c_{k'}(v_j))\\ &\le \sum_{k'\in S(k)} 2\tau_{k'}(n-1)e_{\max}\log\lambda\\ &= 2(n-1)e_{\max}L(k)\log\lambda.\end{aligned} \quad (7)$$

□

Let $T(k)$ be the set of multicast requests realized by the optimal offline algorithm but rejected by algorithm MCM until the arrival of multicast request k. In the following, we show that the total cost of a multicast request rejected by algorithm MCM is no more than $\frac{(n-1)e_{\max}}{K^\epsilon}$ by the following lemma:

Lemma 3. *For all multicast requests $k' \in T(k)$, let $M(k')$ be the minimum multicast tree found by an optimal offline algorithm. Then,*

$$\frac{(n-1)e_{\max}}{K^\epsilon} \le \sum_{\langle v_j,v_x\rangle\in M(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)} - 1), \quad (8)$$

where $K = \max_{1\le i\le k}\{|D_i|\}$ and ϵ is constant with $0 < \epsilon \le 1$.

Proof. A multicast request is rejected by algorithm MCM, because either 1) there is no such multicast tree with sufficient energy to route the message (Step 1 of the algorithm) or 2) the cost for routing the multicast message is too high ($> \sigma$ at Step 4).

We first show that the lemma holds if multicast request k' is rejected by algorithm MCM due to the choice of σ. Let $W_{\rm MCM}(k')$ be the total cost of the multicast tree delivered by algorithm MCM for multicast request k', and let $W_{\rm OPT}(k')$ be the total cost of the multicast tree delivered by the optimal offline algorithm. Then, following the result in [23], we have $W_{\rm MCM}(k') \le |D_{k'}| W_{\rm OPT}(k')$. Meanwhile, we know $W_{\rm MCM}(k') > \sigma$. Thus,

$$W_{\rm OPT}(k') \ge \frac{W_{\rm MCM}(k')}{|D_{k'}|} > \frac{\sigma}{|D_{k'}|},$$

i.e.,

$$\sum_{\langle v_j,v_x\rangle\in M(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)} - 1) > \frac{(n-1)e_{\max}}{|D_{k'}|^\epsilon} \ge \frac{(n-1)e_{\max}}{K^\epsilon}.$$

It is now obvious that

$$\frac{(n-1)e_{\max}}{K^\epsilon} \le \sum_{\langle v_j,v_x\rangle\in M(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)} - 1). \quad (9)$$

We then consider the case where a multicast request k' is rejected at Step 1 of MCM. Let $OPT(k')$ be the minimum multicast tree over which the multicast request is realized. Since the message is rejected by our algorithm, there is a directed edge $\langle v_{j'}, v_x\rangle \in OPT(k')$ at least such that $RE_{k'}(v_{j'}) < \tau_{k'} w(v_{j',l'})$. Thus, $\alpha_{k'}(v_{j'}) > 1 - (\tau_{k'} w(v_{j',l'})/E(v_{j'})) \ge 1 - 1/\log\lambda$, following the assumption that $\tau_{k'} \le \frac{\min_{v\in N}\{E(v)\}}{e_{\max}\log\lambda}$ and $\lambda = 2n\rho$, therefore,

$$\begin{aligned}\sum_{\langle v_j,v_x\rangle\in OPT(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)} - 1) &\ge w(v_{j',l'})(\lambda^{\alpha_{k'}(v_{j'})} - 1)\\ &> w(v_{j',l'})(\lambda^{1-1/\log\lambda} - 1)\\ &= w(v_{j',l'})(\lambda/2 - 1)\\ &\ge e_{\min}(\lambda/2 - 1)\\ &= (n-1)e_{\max},\end{aligned} \quad (10)$$

i.e.,

$$(n-1)e_{\max} \le \sum_{\langle v_j,v_x\rangle\in OPT(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)} - 1). \quad (11)$$

Combining (9) and (11), we have

$$\frac{(n-1)e_{\max}}{K^\epsilon} \le \sum_{\langle v_j,v_x\rangle\in M(k')} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)} - 1).$$

□

We now bound the difference between the sum of the costs of the offline and online algorithms by the following important lemma:

Lemma 4. $(n-1)e_{\max}(L_{opt}(k) - L(k)) \le K^\epsilon \sum_{v\in N} c_k(v)$.

Proof. Following Lemma 3, we have

$$\begin{aligned}
&\frac{(n-1)e_{\max}(L_{opt}(k)-L(k))}{K^{\epsilon}} \\
&= \sum_{k'\in T(k)} \frac{(n-1)e_{\max}\tau_{k'}}{K^{\epsilon}} \\
&\le \sum_{k'\in T(k)} \sum_{\langle v_j,v_x\rangle\in OPT(k')} \tau_{k'} w(v_{j,l})(\lambda^{\alpha_{k'}(v_j)}-1) \\
&\le \sum_{k'\in T(k)} \sum_{\langle v_j,v_x\rangle\in OPT(k')} \tau_{k'} w(v_{j,l}) c_{k'}(v_j)/E(v_j) \qquad (12) \\
&\le \sum_{k'\in T(k)} \sum_{\langle v_j,v_x\rangle\in OPT(k')} \tau_{k'} w(v_{j,l}) c_{k}(v_j)/E(v_j) \qquad (13) \\
&\le \sum_{v\in N} c_k(v) \sum_{k'\in T(k),\langle v_j,v_x\rangle\in OPT(k')} (\tau_{k'} w(v_{j,l})/E(v_j)) \qquad (14) \\
&\le \sum_{v\in N} c_k(v). \qquad (15)
\end{aligned}$$

Relation (13) holds because the node costs are nondecreasing, and relation (15) follows the fact that the total energy expended at a node cannot exceed the initial energy at the node. From Lemmas 2 and 4, it follows that $\frac{L(k)}{L_{opt}(k)} \ge \frac{1}{1+2K^{\epsilon}\log\lambda}$, where $K=\max_{1\le i\le k}\{|D_i|\}$ and ϵ is constant with $0<\epsilon\le 1$. Theorem 1 then follows. □

ACKNOWLEDGMENTS

The authors would like to thank the anonymous referees for their constructive comments and suggestions which have helped improve the quality and presentation of the paper.

REFERENCES

[1] J. Aspnes, Y. Azar, A. Fait, S. Plotkin, and O. Warrts, "Online Routing of Virtual Circuits with Applications to Load Balancing and Machine Scheduling," *J. ACM*, vol. 44, pp. 486-504, 1997.

[2] D.J. Baker and A. Ephremides, "The Architectural Organization of a Mobile Radio Network via Distributed Algorithm," *IEEE Trans. Comm.*, vol. 29, pp. 56-73, 1981.

[3] F. Bian, A. Goel, C.S. Raghavendra, and X. Li, "Energy-Efficient Broadcasting in Wireless Ad Hoc Networks: Lower Bounds and Algorithms," *J. Interconnection Networks*, vol. 3, pp. 149-166, 2002.

[4] M. Cagalj, J.-P. Hubaux, and C. Enz, "Minimum-Energy Broadcast in All-Wireless Networks: NP-Completeness and Distribution Issues," *Proc. MobiCom '02 Conf.*, 2002.

[5] J. Cartigny, D. Simplot, and I. Stojmenovic, "Localized Minimum-Energy Broadcasting in Ad Hoc Networks," *Proc. INFOCOM '03 Conf.*, 2003.

[6] K.M. Chandy and J. Misra, "Distributed Computation on Graphs: Shortest Path Algorithms," *Comm. ACM*, vol. 25, pp. 833-837, 1982.

[7] J.-H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad Hoc Networks," *Proc. INFOCOM '00 Conf.*, 2000.

[8] J.-H. Chang and L. Tassiulas, "Fast Approximate Algorithms for Maximum Lifetime Routing in Wireless Ad-Hoc Networks," *Proc. IFIP-TC6/European Commission Int'l Conf.*, 2000.

[9] M. Charikar, C. Chekuri, T.-Y. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li, "Approximation Algorithms for Directed Steiner Problems," *J. Algorithms*, vol. 33, no. 1, pp. 73-91, 1999.

[10] A.K. Das, R.J. Marks, M. El-Sharkawi, P. Arabshahi, and A. Gray, "Minimum Power Broadcast Trees for Wireless Networks: Integer Programming Formulations," *Proc. INFOCOM '03 Conf.*, pp. 1001-1010, 2003.

[11] A. Ephremides, J.E. Wieselthier, and D.J. Baker, "A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling," *Proc. IEEE*, vol. 75, pp. 56-58, 1987.

[12] M. Gerharz, C. de Waal, P. Martini, and P. James, "A Cooperative Nearest Neighbours Topology Control Algorithm for Wireless Ad Hoc Networks," *Proc. 12th Int'l Conf. Computer Comm. and Networks*, pp. 412-417, 2003.

[13] P. Gupta and P.R. Kumar, "Critical Power for Asymptotic Connectivity in Wireless Networks," *Analysis, Control, Optimization and Applications: A Volume in Honor of W.H. Fleming*, pp. 547-566, 1998.

[14] P.A. Humblet, "A Distributed Algorithm for Minimum Weight Directed Spanning Trees," *IEEE Trans. Comm.*, vol. 31, pp. 756-762, 1983.

[15] C.E. Jones, K.M. Sivalingam, P. Agrawal, and J.C. Chen, "A Survey of Energy Efficient Network Protocols for Wireless Networks," *Wireless Networks*, vol. 7, pp. 343-358, 2001.

[16] I. Kang and R. Poovendran, "Maximizing Static Network Lifetime of Wireless Broadcast Ad Hoc Networks," *Proc. Int'l Conf. Comm. (ICC '03)*, 2003.

[17] K. Kar, M. Kodialam, T.V. Lakshman, and L. Tassiulas, "Routing for Network Capacity Maximization in Energy-Constrained Ad-Hoc Networks," *Proc. INFOCOM '03 Conf.*, 2003.

[18] V. Kawadia and P.R. Kumar, "Power Control and Clustering in Ad Hoc Networks," *Proc. INFOCOM '03 Conf.*, 2003.

[19] Q. Li, J. Aslam, and D. Rus, "Online Power-Aware Routing in Wireless Ad Hoc Networks," *Proc. ACM SIGMobile '01 Conf.*, pp. 97-107, July 2001.

[20] N. Li, J.C. Hou, and L. Sha, "Design and Analysis of an MST-Based Topology Control Algorithm," *Proc. INFOCOM '03 Conf.*, 2003.

[21] W. Liang, "Maximizing Network Lifetime and Minimizing Energy Multicasting in Wireless Ad Hoc Networks," *Proc. Int'l Conf. Comm. (ICC '05)*, May 2005.

[22] W. Liang, "Approximate Minimum-Energy Multicasting in Wireless Ad Hoc Networks," *IEEE Trans. Mobile Computing*, 2005.

[23] W. Liang, "Constructing Minimum-Energy Broadcast Trees in Wireless Ad Hoc Networks," *Proc. MOBIHOC '02 Conf.*, pp. 112-122, 2002.

[24] W. Liang and Y. Yang, "Maximizing Battery Life Routing in Wireless Ad Hoc Networks," *Proc. 37th Hawaii Int'l Conf. System Sciences*, 2004.

[25] E.L. Lloyd, R. Liu, M.V. Marathe, R. Ramanathan, and S.S. Ravi, "Algorithmic Aspects of Topology Control Problems for Ad Hoc Networks," *Proc. MOBIHOC '02 Conf.*, pp. 123-134, 2002.

[26] S. Plotkin, "Competitive Routing of Virtual Circuits in ATM Networks," *IEEE J. Selected Areas in Comm.*, vol. 13, pp. 1128-1136, 1995.

[27] R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment," *Proc. INFOCOM '00 Conf.*, 2000.

[28] V. Rodoplu and T.H. Meng, "Minimum Energy Mobile Wireless Networks," *Proc. Int'l Conf. Comm. (ICC '98)*, vol. 3, pp. 1633-1639, 1998.

[29] A. Sankar and Z. Liu, "Maximum Lifetime Routing in Wireless Ad-Hoc Networks," *Proc. INFOCOM '04 Conf.*, 2004.

[30] S. Singh, M. Woo, and C.S. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks," *Proc. MOBICOM '98 Conf.*, pp. 181-190, 1998.

[31] A. Srinivas and E. Modiano, "Minimum Energy Disjoint Path Routing in Wireless Ad-Hoc Networks," *Proc. MOBICOM '03 Conf.*, 2003.

[32] C.-K. Toh, "Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless As Hoc Networks," *IEEE Comm.*, vol. 39, pp. 138-147, 2001.

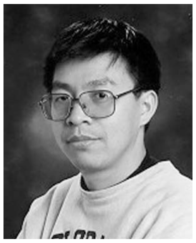
[33] P.-J. Wan, G. Calinescu, and X.-Y. Li, "Minimum-Energy Broadcast Routing in Static Ad Hoc Wireless Networks," *Proc. INFOCOM '01 Conf.*, 2001.

[34] R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks," *Proc. INFOCOM '01 Conf.*, 2001.

[35] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides, "On Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks," *Proc. INFOCOM '00 Conf.*, 2000.

[36] 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*, vol. 39, pp. 113-131, 2002.

[37] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides, "Energy-Efficient Broadcast and Multicast Trees in Wireless Networks," *Mobile Networks and Applications*, vol. 7, pp. 481-492, 2002.



Weifa Liang received the BSc degree from Wuhan University, China, in 1984, the ME degree from the University of Science and Technology of China in 1989, and the PhD degree from the Australian National University in 1998, all in computer science. He is currently a senior lecturer in the Department of Computer Science at the Australian National University. His research interests include the design of energy-efficient routing protocols for wireless ad hoc networks, routing protocol design for WDM optical networks, design and analysis of parallel and distributed algorithms, data warehousing and OLAP, query optimization, and graph theory. He is a senior member of the IEEE and the IEEE Computer Society.

Xiaoxing Guo received the BS degree in computer science and engineering from the National University of Defense Technology, China, in 2002. He has been an MPhil candidate in computer science at the Australian National University since 2003. His research interests include wireless ad hoc and sensor networks, mobile computing, and digital image processing. He is a student member of the IEEE.