Cooperative Multicast for Maximum Network Lifetime

Ivana Maric
WINLAB, Rutgers University
ivanam@winlab.rutgers.edu

Roy Yates
WINLAB, Rutgers University
ryates@winlab.rutgers.edu

Abstract

We consider cooperative data multicast in a wireless network with the objective to maximize the network lifetime. For a *static* power assignment at the nodes, in which a node power is constant throughout the multicast session, we present the *Maximum Lifetime Accumulative Broadcast (MLAB)* algorithm that specifies the nodes' order of transmission and transmit power levels. We prove that the solution found by MLAB is optimal. The power levels found by the algorithm ensure that the lifetimes of the active relays are the same, causing them to fail simultaneously. For the same battery levels at all the nodes, the optimum transmit powers become the same.

The simplicity of the solution is made possible by allowing the nodes that are out of the transmission range of a transmitter to collect the energy of unreliably received overheard signals. As a message is forwarded through the network, nodes will have multiple opportunities to reliably receive the message by collecting energy during each retransmission. We refer to this cooperative strategy as *accumulative multicast.* Cooperative multicast not only increases the multicast energy-efficiency by allowing for more energy radiated in the network to be collected, but also facilitates load balancing by relaxing the constraint that a relay has to transmit with power sufficient to reach its most disadvantaged child. When the message is to be delivered to all network nodes this cooperative strategy becomes *accumulative broadcast* [14].

A more general *dynamic* multicast problem allows variable power allocation at each node during the multicast. We show that this problem is a special case of a static multicommodity multicast problem. We then demonstrate that, unlike single-source static multicast, the dynamic cooperative multicast problem, and thus the multicommodity problem as well, are NP-complete.

Keywords: Cooperative multicast, cooperative broadcast, maximum network lifetime, optimum transmit powers, dynamic strategy, multiple source broadcast problem.

1 Introduction

We consider the problem of energy-efficient multicasting in a wireless network. In the multicast problem, a message from a *source* node is to be delivered efficiently to a set of *destination* nodes. When the set of destination nodes includes all the network nodes (except the source), the multicast problem reduces to the broadcast problem. When there is only one destination node, multicast reduces to unicast and the problem becomes that of routing to one destination node. Prior work on this subject has been focused on the minimum-energy broadcast problem with the objective of minimizing the total transmitted power in the network. This problem was shown in [2, 3, 13] to be NP-complete. Several heuristics for constructing energy-efficient broadcast trees have been proposed; see [2–4, 12, 19, 21] and references therein.

However, broadcasting data through an energy-efficient tree drains the batteries at the nodes unevenly causing higher drain relays to fail first. A performance objective that addresses this issue is *network lifetime* which is defined to be the time duration until the first node battery is fully drained [5]. Finding a broadcast tree that maximizes network lifetime was considered in [9–11, 22]. The problem of maximizing the network lifetime during a multicast was addressed in [7]. In [20], it was shown that the use of directional antennas can improve both the energy-efficiency and the lifetime of the network as compared to the omnidirectional case. Because the energies of the nodes in a tree are drained unevenly, the optimal tree changes in time and therefore the authors [7,9,11] distinguished between the *static* and *dynamic* maximum lifetime problem. In a static problem, a single tree is used throughout the broadcast session whereas the dynamic problem allows a sequence of trees to be used. Since the latter approach balances the traffic more evenly over time, it generally performs better. For the static problem, an algorithm was proposed that finds the optimum tree [11]. For the special case of identical initial battery energy at the nodes, the optimum tree was shown to be the minimum spanning tree. In a dynamic problem, a series of trees were used that were periodically updated [11] or used with assigned duty cycles [9].

Wireless formulations of the above broadcast problems assume that a node can benefit from a transmission only if the received power is above a threshold required for reliable communication. This is a pessimistic assumption. A node for which the received power is below the required threshold, but above the receiver noise floor, can collect energy from the unreliable reception of the transmitted information.

Moreover, it was observed in the relay channel [6] that utilizing unreliable overheard information is essential to achieving capacity. This idea is particularly suited for the multicast problem, where a node has multiple opportunities to receive a message as the message is forwarded through the network. We borrow this idea and re-examine the multicast problem under the assumption that nodes accumulate the energy of unreliable receptions. We refer to this particular cooperative strategy as *accumulative multicast* and in the special case of broadcast, as *accumulative broadcast* [14]. The minimum energy accumulative broadcast problem was formulated and addressed in [14–16]. The problem was shown to be NP-complete. An energy-efficient heuristic was proposed that demonstrated the improvement of accumulative broadcast over the conventional broadcast. Under a different physical model, this problem was independently considered in [8] and again shown to be NP-complete. Furthermore, the same idea, for a packet level system model with the additional constraint of a power threshold for signal acquisition, was recently proposed under the name *Hitch-hiking* [1].

In this paper, we address the problem of maximizing the network lifetime by employing the accumulative multicast. As in the conventional broadcast problem, we impose a *reliable forwarding* constraint that a node can forward a message only after reliably decoding that message.

We first consider the *static* cooperative multicast in which node powers are constant throughout the multicast seesion. We show that the static maximum lifetime multicast problem has a simple optimal solution and propose the *Maximum Lifetime Accumulative Broadcast (MLAB)* algorithm that finds it. The solution specifies the order of transmissions and transmit power levels at the nodes. The power levels given by the solution ensure that the lifetimes of relay nodes are the same and thus, their batteries die simultaneously. Moreover, the simplicity of the solution allows us to formulate a distributed MLAB algorithm for the accumulative broadcast [17] that uses local information at the nodes and is thus better suited for networks with large number of nodes.

We then consider more general multicast problem that relaxes the constant power constraint and allows for variable powers at the nodes. We show that this problem is a special case of a static, multicommodity cooperative multicast problem. We then demonstrate that, unlike single-source static multicast, the dynamic multicast problem, and thus the multicommodity problem as well, are NP-complete.

The paper is organized as follows. In the next section, we give the network model and in Section 3, we formulate the problem. In Section 3.1 we present the MLAB algorithm that finds the optimal solution to the static multicast problem. We then consider more general multicast problems by allowing variable node powers and by considering multiple sources in Section 4,.

2 System model

We consider a wireless network of N nodes such that from each transmitting node k to each receiving node m, there exists an AWGN channel of bandwidth W characterized by a frequency non-selective link gain h_{mk}. We further assume large enough bandwidth resources to enable each transmission to occur in an orthogonal channel, thus causing no interference to other transmissions. Each node has both transmitter and receiver capable of operating over all channels.

A receiver node j is said to be in the transmission range of transmitter i if the received power at j is above a threshold that ensures the capacity of the channel from i to j is above the code rate of node i. We assume that each node can use different power levels, which will determine its transmission range. The nodes beyond the transmission range will receive an unreliable copy of a transmitted signal. Those nodes can exploit the fact that a message is sent through multiple hops on its way to other nodes. Repeated transmissions act as a repetition code for all nodes beyond the transmission range.

During a multicast session, a sequence of messages are transmitted from each source node. After a certain message has been transmitted from a certain source, labeled node 1, sequence of retransmissions at appropriate power levels will ensure that eventually every destination node has reliably decoded the message. When we consider a multicast of a certain message, we say that a node is *reliable* with respect to that message, once it has reliably decoded that message. Under the reliable forwarding constraint, a node is permitted to retransmit (forward) only after reliably decoding the message. During the multicast, the message is repeatedly transmitted until the set of destination nodes $\mathcal{D}$ becomes reliable.

The constraint of reliable forwarding imposes an ordering on the network nodes. In particular, a node m will decode a message from the transmissions of a specific set of transmitting nodes that became reliable prior to node m. Starting with node 1, the source, as the first reliable node, a solution to the cooperative multicast problem will be characterized by a *reliability schedule*, which specifies the order in which the nodes become reliable. Since the multicast stops after the message has been delivered to D destination nodes, a reliability schedule will not necessarily contain all the network nodes. In general, a multicast reliability schedule is an ordered subsequence of the list of nodes of length M, $D < M \leq N$, that starts with node 1, and contains all destination nodes and a subset a network nodes that relay the message. In the broadcast case, a reliability schedule $[n_1, n_2, n_2, \ldots, n_N]$ is simply a permutation of $[1, 2, \ldots, N]$ that always starts with the source node $n_1 = 1$.

For a given reliability schedule, we refer to the ith node in the schedule as simply node i. After each node $k \in \{1, \ldots, m-1\}$ transmits with average power p_k, the rate in bits per second that can be achieved at node m is [18]

$$r_m = W \log_2 \left(1 + \frac{\sum_{k=1}^{m-1} h_{mk} p_k}{N_0 W}\right) \qquad \text{bits/s}, \tag{1}$$

where N_0 is the one-sided power spectral density of the additive white Gaussian receiver noise.

Let the required data rate $\overline{r}$ be given by

$$\overline{r} = W \log_2 \left(1 + \frac{\overline{P}}{N_0 W}\right) \qquad \text{bits/s}. \tag{2}$$

From (1) and (2), achieving $r_m = \overline{r}$ implies that the total received power at node m has to be above the threshold $\overline{P}$, that is,

$$\sum_{k=1}^{m-1} h_{mk} p_k \geq \overline{P}. \tag{3}$$

After the data has been successfully delivered to the destination nodes, all those nodes are reliable and the feasibility constraint (3) is satisfied at every destination node m. When communicating at rate $\overline{r}$, the required signal energy per bit is $E_b = \overline{P}/\overline{r}$ Joules/bit. This energy can be collected at a node m during one transmission interval $[0, T]$ from a transmission of a single node k with power $p_k = \overline{P}/h_{mk}$, as commonly assumed in wireless broadcasting problems [2, 3, 7, 9–11, 19]. However, using the accumulative strategy, the required energy E_b is collected from $m - 1$ prior transmissions.

3 Problem Formulation

The objective of the network lifetime problem is to maximize the amount of data, that is, the number of bits, sent at the source and delivered to a set of destination nodes, for a given energy budget at each node. For a given data rate $\overline{r}$, this objective translates to maximizing the time during which data is received at the destination nodes.

From a node perspective, for a node i with a battery budget e_i and transmitting with an average power p_i, the time until its battery is drained determines the *lifetime* of a node i as $T_i = e_i/p_i$. A network in which a large number of dimensions is available, may faciliate *pipe-lining*, by allowing a node to transmit a new message in every transmit interval, in a new orthogonal channel. If all the messages follow a same sequence of hops, a node i will have a new message to forward in each transmission interval using p_i. Assuming that the data is delivered until a first network node dies, the smallest T_i determines the amount of data delivered to destination nodes.

On the other hand, depending on the underlaying protocol, a node i, after forwarding a message, may be idle for time $\overline{T}$, until the message reaches the destination nodes, a new message is sent at a source and it reaches i reliably. This reduces the average power at node i to $p'_i = p_i T_i / \overline{T}$ and the node lifetime appears to be larger that in the previous case, although the amount of data sent at a node and delivered to the destinations is the same.

To unify the two situation and capture the effect that a node lifetime reflect only the time a node is actually transmiting, we consider the power at the node i to be energy spent per transmition of an actual message. The maximum network lifetime problem is then to maximize the number of messages L that are reliably delivered to a set of destination nodes, $\mathcal{D}$.

In the *dynamic* cooperative multicast that delivers L messages in a multicast session, each node i is assigned L transmit power levels $[p_i(1), \ldots, p_i(L)]$ to forward the messages. For message l, power assignment at all nodes is given by a vector $\mathbf{p}(l) = [p_1(l), \ldots, p_N(l)]^T$. We refer to power allocation $\mathbf{P} = \{\mathbf{p}(1), \ldots \mathbf{p}(L)\}$ as a *dynamic power assignment* [7].

In its special case, when the transmit power at a node i is kept constant, p_i, for all L, a dynamic power assignment reduces to a *static* power assignment and the above problem becomes the static cooperative multicast problem. A number of messages a node i can transmit is simply e_i/p_i and it is maximized by choosing p_i as small as possible. We next present the solution for the static single-source multicast problem. Theorem proofs and a distributed implementation are given in [17].

In the static multicast, it is enough to consider a multicast of a single message. The optimum solution must specify the reliability schedule as well as the transmit power level at each node to deliver that message. Given a schedule, we can formulate a linear program (LP) that will find the optimum solution for that schedule. Such a solution will identify those nodes that should transmit and their transmission power levels. A schedule is an ordered subsequence of M nodes from a network of N nodes,

$$\mathbf{x} = [x_1, \ldots, x_M], \tag{4}$$

with $x_1 = 1$. We say that the *length* of the subsequence $\mathbf{x}$ in (4) is $\|\mathbf{x}\| = M$. Let

$$\{\mathbf{x}\} = \{x_1, \ldots, x_{\|\mathbf{x}\|}\} \tag{5}$$

denote the set of nodes in a schedule $\mathbf{x}$ and let Π_N denote the set of all variable-length ordered subsequences of $\{1, \ldots, N\}$. It follows that the family of all possible schedules is $\mathcal{X}_N(\mathcal{D}, 1)$ where

$$\mathcal{X}_N(\mathcal{D}, s) = \{\mathbf{x} \in \Pi_N | \mathcal{D} \in \{\mathbf{x}\}, x_1 = s\} \tag{6}$$

Given a schedule $\mathbf{x}$, we define a gain matrix $\mathbf{G}(\mathbf{x})$ to have i, jth element

$$[\mathbf{G}(\mathbf{x})]_{ij} = \begin{cases} h_{x_i x_j} & i > j, \\ 0 & \text{otherwise}, \end{cases} \tag{7}$$

for $1 \leq i, j \leq M$. When a node j does not participate in the retransmission of the message, reliable reception by that node is unnecessary and that node can be omitted from the problem formulation. Thus, channel gains corresponding to any node j that is not in schedule $\mathbf{x}$ are not included in $\mathbf{G}(\mathbf{x})$. We can define the problem of maximizing the network lifetime for schedule $\mathbf{x}$ in terms of the vector $\mathbf{p}$ of transmitted powers as

$$\min \max_{\mathbf{p}} \frac{p_i}{e_i} \tag{8}$$

$$\text{subject to } \mathbf{G}(\mathbf{x})\mathbf{p} \geq \mathbf{1}\overline{P}, \tag{8a}$$

$$\mathbf{p} \geq \mathbf{0}. \tag{8b}$$

The inequality (8a) contains $M-1$ constraints as in (3), requiring that the accumulated received power at all nodes in schedule $\mathbf{x}$ (except the source) is above the threshold $\overline{P}$. It should be apparent that power p_i in $\mathbf{p}$ corresponds to the transmit power of node x_i in the schedule $\mathbf{x}$. Alternatively, we can define the problem in terms of *normalized* node powers $\overline{p}_i = p_i e_1/e_i$ that account for different battery capacities at the nodes; the lifetime at every node i in terms of the normalized power is as if all the batteries were the same: $T_i = e_i/p_i = e_1/\overline{p}_i$. In terms of normalized node powers, Problem (8) can be defined as

$$\min \max_{\overline{\mathbf{p}}} \overline{p}_i \tag{9}$$

$$\text{subject to } \overline{\mathbf{G}}(\mathbf{x})\overline{\mathbf{p}} \geq \mathbf{1}\overline{P},$$

$$\overline{\mathbf{p}} \geq \mathbf{0}$$

where each column $\overline{\mathbf{g}}_i$ of the normalized gain matrix $\overline{\mathbf{G}}(\mathbf{x})$ is obtained from the corresponding column $\mathbf{g}_i$ of matrix $\mathbf{G}(\mathbf{x})$ as $\overline{\mathbf{g}}_i = \mathbf{g}_i e_i / e_1$.

For any schedule $\mathbf{x}$, we can formulate Problem (9) as a linear program in terms of transmit power levels $\overline{\mathbf{p}}$,

$$\hat{p}^*(\mathbf{x}) = \min_{\overline{\mathbf{p}}} \ \hat{p} \tag{10}$$

$$\text{subject to } \overline{\mathbf{G}}(\mathbf{x})\overline{\mathbf{p}} \geq \mathbf{1}\overline{P}, \tag{10a}$$

$$\overline{\mathbf{p}} \leq \mathbf{1}\hat{p} \tag{10b}$$

$$\overline{\mathbf{p}} \geq \mathbf{0}. \tag{10c}$$

If $\hat{p} = \hat{p}^*(\mathbf{x})$, then there exists a power vector $\overline{\mathbf{p}}$ such that (10a) and (10b) are satisfied. It follows that for any $p > \hat{p}$, $\overline{\mathbf{p}} \leq \mathbf{1}p$. Thus, for any power $\hat{p} \geq \hat{p}^*(\mathbf{x})$, we say that power $\hat{p}$ is *feasible* for schedule $\mathbf{x}$. Over all possible schedules, the optimum power is

$$p^* = \min_{\mathbf{x} \in \mathcal{X}_N(\mathcal{D})} \hat{p}^*(\mathbf{x}). \tag{11}$$

Equation (11) is a formal statement of the problem from which finding the best schedule corresponding to p^* is not apparent. We will see that the power p^*, may, in fact, be the solution to (10) for a set of schedules, $\mathcal{X}^*$. In the rest of the paper, we will consider only normalized powers and we therefore drop the overline notation; $\mathbf{H}$ will denote the ordinary gain matrix, $\mathbf{G}(\mathbf{x})$ will denote the gain matrix permuted for schedule $\mathbf{x}$, and the power vector will be simply $\mathbf{p}$, with p_i representing either the power of node i or node x_i, as appropriate for the context.

Rather than identifying $\mathcal{X}^*$, we employ a simple procedure that for any power p, determines a collection of schedules for which power p is feasible. In particular, to distribute a message, we let each node retransmit with power p *as soon as possible*, namely as soon as it becomes reliable. We refer to such a distribution as the *ASAP*(p) *distribution*. During the ASAP(p) distribution, the message will be resent in a sequence of retransmission stages from sets of nodes $Z_1(p), Z_2(p), \ldots$ with power p where in each stage i, a set Z_i that became reliable during stage $i-1$, transmits and makes Z_{i+1} reliable.

Let $S_i(p)$ and $U_i(p)$ denote the reliable nodes and unreliable nodes at the start of stage i. $U_{D,i}(p) \subset U_i(p)$ is the set of unreliable destination nodes at the start of stage i. Then, $Z_1(p) = 1$ and $S_i(p) = Z_1(p) \cup \ldots \cup Z_i(p)$. The set $Z_{i+1}(p)$ is given by

$$Z_{i+1}(p) = \{z \in U_i(p) : p \sum_{k \in S_i(p)} h_{zk} \geq \overline{P}\}. \tag{12}$$

Note that if power p is too small, the ASAP(p) distribution can *stall* at stage i with $S_{i+1}(p) = S_i(p)$ and $U_{D,i}(p) \neq \emptyset$, the empty set. In this case, ASAP(p) fails to distribute the message to all destination nodes. When $U_{D,i}(p) = \emptyset$ at a stage i, the ASAP(p) distribution *terminates successfully*. We will say that ASAP(p) distribution is a *feasible multicast* if it terminates successfully.

The partial node ordering, $Z_1(p), Z_2(p), \ldots$, specifies the sequence in which nodes became reliable during the ASAP(p) distribution. In particular, any schedule $\mathbf{x}$ that is consistent with this partial ordering is a feasible schedule for power p. Nodes that become reliable during the same stage of ASAP(p) can be scheduled in an arbitrary order among themselves since these nodes do not contribute to each other's received power. The following theorem verifies that in terms of maximizing the network lifetime it is sufficient to consider only schedules consistent with the ASAP(p) distribution.

Theorem 1 *If $\tilde{p}$ is a feasible power for a schedule $\tilde{\mathbf{x}}$, then the ASAP$(\tilde{p})$ distribution is a feasible multicast.*

In particular, Theorem 1 implies that for optimum power p^*, the ASAP(p^*) distribution is feasible.

We next present the *Maximum Lifetime Accumulative Broadcast (MLAB)* algorithm, that determines the optimum power p^*. Once the power p^* is given, broadcasting with ASAP(p^*) will maximize the network lifetime.

3.1 The MLAB algorithm

We label node 1 as the source and 2 as its closest neighbor (more precisely, the node with the highest link gain to the source). The MLAB algorithm finds the optimum power p^* through a series of ASAP(p) distributions, starting with the smallest possible *candidate broadcast power*, $p = \overline{P}/h_{21}$. Whether ASAP(p) stalls or terminates successfully, we define $\tau(p)$ as the terminating stage. When $p = p^*$, the ASAP(p^*) distribution will terminate in $\tau^* = \tau(p^*)$ stages. When the ASAP(p) distribution stalls at stage $\tau(p)$, we determine the minimum power increase δ for which ASAP$(p + \delta)$ will not stall at stage $\tau(p)$, in the following way. The increase in broadcast power δ_j needed to make a node $j \in U_{\tau(p)}(p)$ reliable must satisfy

$$\overline{P} = (p + \delta_j) \sum_{k \in S_{\tau(p)}(p)} h_{jk}. \tag{13}$$

We choose $\delta = \min_{j \in U_{\tau(p)}(p)} \delta_j$. We then increase p to $p + \delta$ and *restart* the MLAB algorithm. The algorithm stops when an ASAP(p) distribution terminates successfully.

The MLAB algorithm ends after at most $N - 1$ restarts. There exists a set of feasible schedules that are consistent with the partial ordering given by the ASAP(p) distribution. The normalized transmit power at all nodes in $S_{\tau(p)}(p)$ is p. Note that the last transmitting set $Z_{\tau(p)}$ could in fact, transmit with power less than p if it is enough for the last set of unreliable destination nodes, $U_{D,\tau(p)}(p)$, to become reliable. Thus, choosing the power level at all nodes to be p is not necessarily a unique solution. While this won't change the network lifetime, the latter solution will reduce the total transmit power in the network. Next theorem shows that the power found by MLAB is in fact the optimum power, that is, $p = p^*$.

Theorem 2 *The MLAB algorithm finds the optimum power p^* such that the ASAP(p^*) distribution maximizes the network lifetime.*

Finally, we note that the full restarts of the MLAB algorithm are used primarily to simplify the proof of Theorem 2. In fact, when MLAB stalls, it is sufficient for the reliable nodes to offer incremental retransmissions at power Δ^*. This observation will be the basis of distributed algorithm proposed in [17].

4 Dynamic Cooperative Multicast

The ASAP(p^*) solution found by MLAB is *static* since it stays constant throughout the multicast session. In conventional broadcast, the constraint that a node is made reliable by the transmission of a single relay, causes the relay with the most disadvantaged child to drain its battery fastest. Consequentially, the optimality of a spanning tree that maximizes the network lifetime for a given initial battery levels is temporary and dynamic tree updates [9, 10] are

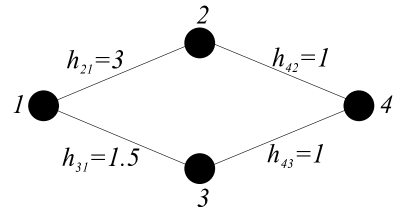

Figure 1: Four node network example.

needed for load balancing. In a cooperative multicast using the ASAP(p^*) distribution, all relays will be draining their batteries evenly; however, a set of *leaf nodes* $U_{\mathcal{T}^*}(p^*)$ will never transmit and will have full batteries even when the relay nodes die. An significant question is whether the undepleted batteries of these leaf nodes can be exploited by a dynamic multicast strategy.

After multiple uses of the ASAP(p^*) distribution, re-examination of the maximum lifetime problem (11), as expressed in terms of normalized powers, will show for each non-relay node $j \in U_{\mathcal{T}^*}(p^*)$ that the outgoing normalized link gains h_{kj} have increased by the ratio of the full battery energy of node j to the depleted battery of node 1. Although one can show that reconfiguring the multicast distribution to maximize the residual network lifetime results in the very same ASAP(p^*) distribution, it would be mistake to conclude that ASAP(p^*) policy is an optimal dynamic policy. In fact, similar to the conventional broadcast, a dynamic strategy with time varying powers can extend the network lifetime. For example, in the four node network shown in Figure 1, the source node 1 wishes to send messages to the destination node 4. With initial battery powers $e_i = 2$ and required received power $\overline{P} = 1$, the ASAP solution uses transmissions by nodes 1, 2 and 3, each with power $p^* = 2/3$. The number of messages delivered to node 4 is $e_i/p^* = 3$. On the other hand, alternating between schedule $\mathbf{x} = [1, 2]$ with power vector $\mathbf{p} = [1/3, 1, 0]$ and schedule $\mathbf{x}' = [1, 3]$ with power vector $\mathbf{p}' = [2/3, 0, 1]$ results in a system which has average transmit power of $1/2$ for each node and delivers $L = 4$ messages to node 4 by using each schedule twice. In this case, dynamic switching between schedules, corresponding to routing packets along multiple routes, delivers a larger number of messages than the ASAP distribution, the optimal static policy. With the solution to the static problem at our hand, a question whether a simple general solution for the dynamic cooperative multicast exists, begs for an answer.

The optimum solution for the dynamic multicast problem is given by the reliability schedule sequence $\mathbf{X} = \{\mathbf{x}(1), \ldots, \mathbf{x}(L)\}$ that specifies a schedule $\mathbf{x}(l)$ for each message $l \in \{1, \ldots L\}$, and similarly, a dynamic power assignment $\mathbf{P} = \{\mathbf{p}(1), \ldots, \mathbf{p}(L)\}$. For the subsequent discussion, it will be convenient to formulate the optimization problem in the form of decision problem in which the question is whether the nodes' battery energies are sufficient for the broadcast of L messages. Employing the constraints given in (10), we have the dynamic maximum lifetime accumulative broadcast (DMLAB) problem.

DMLAB Given a nonnegative matrix specified by $\{h_{j,k} | 1 \leq j \leq n, 0 \leq k \leq n\}$, a vector of battery energies $\mathbf{e} = \begin{bmatrix} e_1 & \cdots & e_n \end{bmatrix}$, and a constant $\overline{P}$, does there exist a sequence of schedules $\mathbf{x}(1), \ldots, \mathbf{x}(L)$ and power vectors $\mathbf{p}(1), \ldots, \mathbf{p}(L)$ such that

$$\mathbf{G}(\mathbf{x}(l))\mathbf{p}(l) \geq \mathbf{1}\overline{P}, \qquad l = 1, \ldots, L, \tag{14}$$

$$\sum_{l=1}^{L} \mathbf{p}(l) \leq \mathbf{e}, \tag{15}$$

$$\mathbf{p}(l) \geq \mathbf{0}, \qquad l = 1, \ldots, L, \tag{16}$$

$$\mathbf{x}(l) \in \mathcal{X}_N(V, 1), \qquad l = 1, \ldots, L. \tag{17}$$

Note that $\mathcal{X}_N(V, 1)$ in the constraint (17) is defined by (6) as the set of all reliability schedules starting with node 1 and including the entire set of nodes V. Thus constraint (17) ensures that each schedule starts with the source node 1 and makes all nodes $v \in V$ reliable.

As we will see, DMLAB is a closely connected to a static, multi-commodity broadcast problem in which more than one source has data to broadcast to a subset of nodes. We refer to this problem as the multiple-source maximum lifetime accumulative broadcast (MS-MLAB) problem. In MS-MLAB, messages, encoded at the same code rate, from each of L sources $\{s_1, \ldots, s_L\}$, are delivered to all nodes $v \in V$ over L static accumulative broadcast schedules $\mathbf{x}(1), \ldots, \mathbf{x}(L)$. Each source s_l employs a single schedule $\mathbf{x}(l)$ and a static power assignment $\mathbf{p}(l)$. The solution specifies L schedules $\{\mathbf{x}(l)\}_{l=1}^{L}$ and transmit power vectors, $\mathbf{P} = \{\mathbf{p}(l) | l = 1, \ldots, L\}$, one for each source. As before, a choice of schedules $\{\mathbf{x}(l)\}_{l=1}^{L}$ admits the following decision formulation of the problem.

MS-MLAB Given a nonnegative matrix specified by $\{h_{j,k} | 1 \leq j \leq n, 0 \leq k \leq n\}$, a vector of battery energies $\mathbf{e} = \begin{bmatrix} e_1 & \cdots & e_n \end{bmatrix}$, and a constant $\overline{P}$, does there exist a sequence of schedules $\mathbf{x}(1), \ldots, \mathbf{x}(L)$ and power vectors $\mathbf{p}(1), \ldots, \mathbf{p}(L)$ such that

$$\mathbf{G}(\mathbf{x}(l))\mathbf{p}(l) \geq \mathbf{1}\overline{P}, \qquad l = 1, \ldots, L, \tag{18}$$

$$\sum_{l=1}^{L} \mathbf{p}(l) \leq \mathbf{e}, \tag{19}$$

$$\mathbf{p}(l) \geq \mathbf{0}, \qquad l = 1, \ldots, L. \tag{20}$$

$$\mathbf{x}(l) \in \mathcal{X}_N(V, s_l), \qquad l = 1, \ldots, L. \tag{21}$$

When each source s_l is node 1, the MS-MLAB becomes an instance D-MLAB. Thus to examine the complexity of these problems it is sufficient to focus on the dynamic accumulative broadcast problem.

Theorem 3 *DMLAB is NP complete.*

Since DMLAB is a special case of maximum lifetime cooperative multicast, results extend readily to the multicast problem.

Proof: Theorem 3

In [**?**, Theorem 4], it is observed that the non-accumulative DYNAMIC MAXIMUM LIFETIME BROADCAST (D-MLB) problem is NP hard. In DMLB, the schedule $\mathbf{x}(l)$ becomes a tree T_l and node i uses power $p_i(T_l)$ sufficient to ensure that its children are made reliable. A formal statement of the decision formulation of this problem is

DMLB Given a directed complete weighted graph $G = (V, c)$, the power of a node u in a directed spanning subgraph T is given by $p_u(T) = \max_{(u,v)\in E(T)} c(u, v)$, where $e(T)$ denote the set of edges in subgraph T. Given that each node v has battery energy e_v, do there exist directed spanning subgraphs $T_1, \ldots, T_L$ such that $\sum_{l=1}^{L} p_u(T) \leq e_u$ for each u.

In the context of wireless networks, the DMLB edge cost $c(u, v)$ corresponds to $\overline{P}/h_{vu}$, the power required for node u to transmit reliably to node v.

We now transform any instance of DMLB into an instance of DMLAB with graph G' and node set V' in which the link gain matrix is designed to preclude any actual accumulation. For each node $v \in V$, we create a cluster of n nodes in V' consisting of an input node set $I_v = \{i_{v,u} | u = 1, \ldots, n, u \neq v\}$ and a cluster head o_v. For the source node 1, the set of input nodes I_1 is empty and cluster 1 consists of just the cluster leader o_1. Node $i_{v,u}$ has link gain $h(i_{v,u}, o_u) = h_{vu}$ from node o_u and *zero* link gain from all other nodes outside cluster v. The cluster head o_v is given battery energy e_v. Each node $i_{v,u}$ has link gain $1/\epsilon$ to every node within cluster v and zero link to any node outside cluster v. This implies that node $i_{v,u}$ can employ transmit power $\overline{P}\epsilon$ to transmit reliably to every other node in cluster v. We choose ϵ so that $L\overline{P}\epsilon < e_{v,u}$, the battery energy of node $i_{v,u}$. Thus any input node $i_{v,u}$ has sufficient energy to forward all L messages reliably to the other nodes in its cluster and the battery constraints of the input nodes are not binding on DMLB.

To prove that DMLAB is NP-complete, we show that DMLB has a solution of duration L if and only DMLAB with $\overline{P} = c$ has solution of duration L.The graph G' for DMLAB has the following key properties:

- Node $i_{v,u}$ can only be made reliable by a transmission from outside of its cluster by the cluster leader o_u and cannot accumulate energy from any other transmissions outside cluster v.
- The nodes in cluster v become reliable if and only if at least one node $i_{v,u}$ is made reliable by a transmission from a cluster head o_u.

Consider a solution $T(1) \ldots, T(L)$ for MLB. In the broadcast tree $T(l)$, node u uses power $p_u(l)$ to broadcast to child nodes $v_1, \ldots, v_m$. In the induced instance of DMLAB, node o_u is assigned transmit power $p_j(l)$ to make reliable the nodes $i_{v_1,u}, \ldots, i_{v_m,u}$. Each node $i_{v_k,u}$ then transmits with power ϵ to make reliable all other nodes in cluster v_k, including the cluster head o_{v_k}. The key observation is that there is a one-to-one correspondence between node v transmitting with power $p_v(l)$ for tree $T(l)$ in DMLB and cluster head o_v transmitting with the same power $p_v(l)$ for schedule $\mathbf{x}(l)$ in DMLAB. However, this power assignment on graph G' admits a set of possible schedules $\mathbf{x}(l)$ such that if node $v \in V$ is a child of node u in the tree $T(l)$, then node $i_{v,u}$ follows the cluster leader o_u and all other nodes in cluster v follow node $i_{v,u}$. The existence of the tree $T(l)$ for DMLB implies there are many such DMLAB schedules $\mathbf{x}(l)$. Thus a feasible solution for DMLB is also a feasible solution for DMLAB.

For the reverse direction, suppose we have a length L feasible sequence of schedules $\mathbf{x}(1), \ldots, \mathbf{x}(L)$ for DMLAB under graph G'. For each schedule $\mathbf{x}(l)$, each node in each cluster v is made reliable. An arbitrary cluster v is made reliable if and only if a node $i_{v,u}$ for some u is made reliable by the transmission of node o_u preceding all nodes in cluster C_v in the schedule $\mathbf{x}(l)$. In the graph G, we define the tree $T(l)$ such that u is a parent of v if in DMLAB, cluster leader o_u is the earliest node in the schedule $\mathbf{x}(l)$ that transmits to make a node $i_{v,u} \in I_v$ reliable. Since $\mathbf{x}(l)$ is a feasible schedule for DMLAB, the tree $T(l)$ and power assignment $\mathbf{p}(l)$ is a feasible broadcast tree for DMLB. □

References

[1] M. Agarwal, J. Cho, L. Gao, and J. Wu. Energy efficient broadcast in wireless ad hoc networks with hitch-hiking. In *Proc. of INFOCOM'04*, June 2004.

[2] A. Ahluwalia, E. Modiano, and L. Shu. On the complexity and distributed construction of energy-efficient broadcast trees in static ad hoc wireless networks. In *Proc. of Conf. on Information Science and Systems*, pages 807–813, March 2002.

[3] M. Cagalj, J. Hubaux, and C. Enz. Energy-efficient broadcast in all-wireless networks. *ACM/Kluwer Mobile Networks and Applications (MONET); to appear*, 2003.

[4] J. Cartigny, D. Simplot, and I. Stojmenovic. Localized minimum-energy broadcasting in ad hoc networks. In *Proc. of INFOCOM'03*, March 2003.

[5] J. H. Chang and L. Tassiulas. Routing for maximum system lifetime in wireless ad-hoc networks. In *Proc. of 37-th Annual Allerton Conference on Communication, Control and Computing*, September 1999.

[6] T. Cover and A. El Gamal. Capacity theorems for the relay channel. *IEEE Trans. on Information Theory*, 25(5):572–584, September 1979.

[7] P. Floreen, P. Kaski, J. Kohonen, and P. Orponen. Multicast time maximization in energy constrained wireless networks. In *Proc. of ACM/IEEE Mobicom 2003 Workshop on Foundations of Mobile Computing*, September 2003.

[8] Y. Hong and A. Scaglione. Energy-efficient broadcasting with cooperative transmission in wireless sensory ad hoc networks. In *Proc. of Allerton Conference*, October 2003.

[9] R. J. Marks II, A. K. Das, and M. El-Sharkawi. Maximizing lifetime in an energy constrained wireless sensor array using team optimization of cooperating systems. In *Proc. of the International Joint Conf. on Neural Networks, IEEE World Congress on Computational Intelligence*, May 2002.

[10] I. Kang and R. Poovendran. Maximizing network lifetime of broadcasting over wireless stationary adhoc networks. *submitted*.

[11] I. Kang and R. Poovendran. Maximizing static network lifetime of wireless broadcast adhoc networks. In *Proc. of ICC'03*, May 2003.

[12] F. Li and I. Nikolaidis. On minimum-energy broadcasting in all-wireless networks. In *Proc. of Local Computer Networks (LCN 2001)*, November 2001.

[13] W. Liang. Constructing minimum-energy broadcast trees in wireless ad hoc networks. In *Proc. of International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02)*, pages 112–122, June 2002.

[14] I. Maric and R. Yates. *Efficient Multihop Broadcast for Wideband Systems*, volume 62. American Mathematical Society, Book chapter: multiantenna channels: capacity, coding and signal processing, DIMACS workshop on signal processing for wireless transmission edition, October 2002.

[15] I. Maric and R. Yates. Efficient multihop broadcast for wideband systems. In *Proc.of the Allerton Conference on Communications, Control and Computing, Monticello, IL*, October 2002.

[16] I. Maric and R. Yates. Efficient multihop broadcast for wireless networks. *IEEE J. Sel. Areas Commun.*, 22(6):1080 – 1088, August 2004. Special issue on fundamental performance limits of wireless sensor networks.

[17] I. Maric and R. Yates. Maximum lifetime of cooperative broadcast in wireless networks. *IEEE JSAC Special Issue on Wireless Ad Hoc Networks, accepted*, June 2004.

[18] E. Telatar. Capacity of multi-antenna gaussian channels. In *Europ. Trans. Telecommunications*, volume 10, pages 585–595, November 1999.

[19] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *Proc. of INFOCOM'00*, pages 585–594, March 2000.

[20] 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*, 1(3):176–191, July 2002.

[21] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides. Energy-efficient broadcast and multicast trees in wireless networks. *Mobile Networks and Applications (MONET)*, 7:481–492, December 2002.

[22] 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*, 39:113–131, June 2002.