



Energy-Efficient Broadcast and Multicast Trees in Wireless Networks*

JEFFREY E. WIESELTHIER** and GAM D. NGUYEN

Code 5521, Information Technology Division, Naval Research Laboratory, Washington, DC 20375, USA

ANTHONY EPHREMIDES

Electrical & Computer Eng. Dept. and Institute for Systems Research, University of Maryland, College Park, MD 20742, USA

Abstract. The wireless networking environment presents formidable challenges to the study of broadcasting and multicasting problems. In this paper we focus on the problem of multicast tree construction, and we introduce and evaluate algorithms for tree construction in infrastructureless, all-wireless applications. The performance metric used to evaluate broadcast and multicast trees is energy-efficiency. We develop the Broadcast Incremental Power (BIP) algorithm, and adapt it to multicast operation by introducing the Multicast Incremental Power (MIP) algorithm. These algorithms exploit the broadcast nature of the wireless communication environment, and address the need for energy-efficient operation. We demonstrate that our algorithms provide better performance than algorithms that have been developed for the link-based, wired environment.

Keywords: wireless broadcast, wireless multicast, energy efficient, ad hoc network, network algorithm

1. Introduction

In [14] we discussed the fundamental issues associated with energy-efficient multicasting in all-wireless (i.e., ad hoc, infrastructureless, or peer-to-peer) networks, and demonstrated the benefits that can be obtained by exploiting the characteristics of the wireless medium. Our approach departed from the traditional layered structure in that we jointly addressed the issues of transmitted power levels (and hence network connectivity, which is a Physical layer function) and multicast tree formation (a routing function, associated with the Network layer). Such joint decisions on connectivity and routing result in improved energy efficiency. Several approaches for multicasting were introduced and evaluated, and performance comparisons were provided.

Our approach in this paper exploits the “node-based” nature of wireless communications; one crucial aspect of this is the property that a single transmission can be heard by several nearby nodes. By contrast, most previously developed models for multicasting (e.g., [2,6,15]) have been “link-based” models, under which a node must transmit separately to each of its neighbors. Although link-based models are appropriate for wired applications (and to some extent can be adapted to wireless networks [14]), they do not reflect properly the properties of the all-wireless network environment.

We study source-initiated broadcasting (one-to-all) and multicasting (one-to-many) of “session” (or connection-oriented) traffic. In either case, our objective is to form a minimum-energy tree, rooted at the source, that reaches all of the desired destinations. Here we consider only the energy used for transmission, neglecting for the present the energy associ-

ated with reception and signal processing; the joint study of all forms of energy expenditure and the associated trade-offs is a subject for future research. After demonstrating the fundamental differences between wired and wireless networks, we introduce the node-based Broadcast Incremental Power (BIP) algorithm. We then show how BIP can be adapted for multicast operation by introducing the Multicast Incremental Power (MIP) algorithm. Our performance results demonstrate that BIP and MIP provide better performance than conventional link-based schemes over a wide range of networks and system parameters.

A crucial issue in wireless networks is the trade-off between the “reach” of wireless transmission (namely the simultaneous reception by many nodes of a transmitted message) and the resulting interference by that transmission. We assume that the power level of a transmission can be chosen within a given range of values. Therefore, there is a trade-off between reaching more nodes in a single hop by using higher power (but at a higher interference cost) versus reaching fewer nodes in that single hop by using lower power (but at a lower interference cost). Another crucial issue is that of energy consumption, because of the nonlinear attenuation properties of radio signals. We focus on energy consumption in this paper.

Few studies have addressed multicasting specifically for wireless networks. For example, the problem of multicast scheduling in cellular mobile networks was studied in [8], and a forwarding multicast protocol for noncellular networks was studied in [5]. Although [7] addressed the multicasting problem with a goal toward reaching efficient and near-minimum-cost algorithms for wireless networks, their approach was link-based, and hence does not take into consideration the node-based nature of wireless communications. Most other

* This work was supported by the Office of Naval Research.

** Corresponding author.

multicasting studies have been limited to the case of stationary networks that are not wireless (e.g., [2,6,15]).

In this paper, we focus on the determination of minimum-cost (in our case, minimum-energy) broadcast and multicast trees. The problem we address involves the designation of which nodes are to transmit, and the power levels at which they are to do so. To assess the complex trade-offs one at a time, we assume in this paper that there is no mobility. Nevertheless, the impact of mobility can be incorporated into our models because transmitter power can be adjusted to accommodate the new locations of the nodes, as necessary. In other words, the capability to adjust transmission power provides considerable “elasticity” to the topological connectivity, and hence may reduce the need for hand-offs and tracking. Also, we do not address medium-access control issues. Therefore, we are able to concentrate on the development of node-based algorithms for wireless networks that are similar in principle to algorithms for the minimum-cost spanning tree (MST) algorithm for wired networks, but which exploit the characteristics of the wireless medium.

2. Architectural issues in all-wireless networks

The all-wireless networks studied here are quite different from the cellular systems and wireless LANs that have been developed in the commercial domain. Cellular systems have fixed base stations, which communicate among themselves using dedicated non-wireless lines; thus, the only multicast problems that are new in those systems involve tracking the mobile users. Otherwise, wireless communication is limited to that between mobile users and base stations. In fully-connected wireless LANs, since there is single-hop connectivity among all the nodes, the multicasting problem is trivial. However, in ad hoc wireless networks it is possible to establish a link between any pair of nodes, provided that each has a transceiver available for this purpose and that the signal-to-noise ratio at the receiving node is sufficiently high. Thus, unlike the case of wired networks, the set of network links and their capacities are not determined a priori, but depend on factors such as distance between nodes, transmitted power, error-control schemes, other-user interference, and background noise. Thus, even when the physical locations of the nodes are fixed, many of the factors that affect network topology (and hence network control schemes) are (at least partially) influenced by the actions of the network nodes. Furthermore, in ad hoc networks no distinction can be made between uplink and downlink traffic, thus greatly complicating the interference environment. Therefore, the wireless networking environment poses many new challenges not encountered in non-wireless or cellular networks, even when mobility is not addressed.

In this paper, we focus on wireless networks in which the node locations are fixed, and the channel conditions unchanging. The wireless channel is distinguished by its *broadcast* nature; when omnidirectional antennas are used, every transmission by a node can be received by all nodes that lie

within its communication range. Consequently, if the multicast group membership includes multiple nodes in the immediate communication vicinity of the transmitting node, a single transmission suffices for reaching all these receivers.

In addition to interference, another undesirable impact of the use of high transmitter power is that it results in increased energy usage. Since the propagation loss varies nonlinearly with distance (at somewhere between the second and fourth power), in unicast applications it is best (from the perspective of transmission energy consumption) to transmit at the lowest possible power level, even though doing so requires multiple hops to reach the destination. However, in multicast applications it is not prudent to draw such conclusions a priori because the use of higher power may permit simultaneous connectivity to a sufficiently large number of nodes, so that the total energy required to reach all members of the multicast group may be actually reduced. Furthermore, even for unicast applications, the use of lower power (and, hence, multiple hops) necessitates the complex coordination of more signals and therefore may actually result in higher total energy expenditure.

Thus, the choice of transmitted power levels depends ultimately on complex trade-offs between energy limitations and the demands of protocol operation. In view of the complex interdependencies among many aspects of network design (e.g., transmitted power levels, signal processing considerations, spectral efficiency, mobility effects, etc.), it may be beneficial to design protocols that span several of the traditional layers to address appropriately the unique characteristics of the all-wireless environment [9,10]. Our studies do, in fact, support this conjecture.

3. Wireless communications model

We consider source-initiated, circuit-switched, multicast sessions. The network consists of N nodes, which are randomly distributed over a specified region. Each node has several transceivers, and can thus support several multicast sessions simultaneously. Any node is permitted to initiate multicast sessions. Multicast requests and session durations are generated randomly at the network nodes. Each multicast group consists of the source node plus at least one destination node. Additional nodes may be needed as relays either to provide connectivity to all members of the multicast group or to reduce overall energy consumption or both. The set of nodes that support a multicast session (the source node, all destination nodes, and all relay nodes) is referred to as a *multicast tree*.

The connectivity of the network depends on the transmission power. We assume that each node can choose its power level, not to exceed some maximum value p_{\max} . The nodes in any particular multicast tree do not necessarily have to use the same power levels; moreover, a node may use different power levels for the various multicast trees in which it participates.

We assume that the received signal power is equal to $pr^{-\alpha}$, where p is the transmission power, r is the distance and α is a

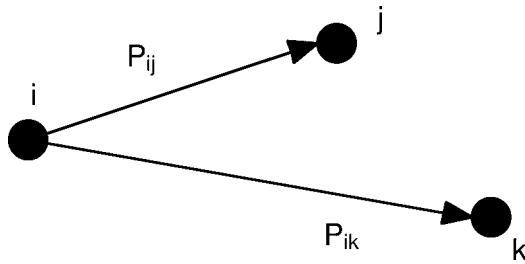


Figure 1. The “wireless multicast advantage”: $P_{i,(j,k)} = \max\{P_{ij}, P_{ik}\}$.

parameter that typically takes on a value between 2 and 4, depending on the characteristics of the communication medium. We use a simplified interference model in which we assume that the interference level is independent of network traffic and the same at all nodes. Based on this model the transmitted power required to support a link between two nodes separated by distance r is proportional to r^α , since the received power must exceed some threshold (which depends on factors such as signal parameters, detector structure, and noise levels). Without loss of generality, we set the threshold constant equal to 1, resulting in:

$$\begin{aligned} p_{ij} &= \text{power needed for link between Node } i \text{ and Node } j \\ &= r_{ij}^\alpha, \end{aligned}$$

where r_{ij} is the distance between Node i and Node j . If the maximum permitted transmitter power p_{\max} is sufficiently large, the nodes will be able to transmit at sufficiently high power so that the network is fully connected.

We assume the use of omnidirectional antennas; thus all nodes within communication range of a transmitting node can receive its transmission. It is important to note how the broadcast property of wireless communication can be exploited in multicast applications. Consider the example shown in figure 1, in which a subset of the multicast tree involves Node i , which is transmitting to its neighbors, Node j and Node k . The power required to reach Node j is P_{ij} and the power required to reach Node k is P_{ik} . A single transmission at power $P_{i,(j,k)} = \max\{P_{ij}, P_{ik}\}$ is sufficient to reach both Node j and Node k , based on our assumption of omnidirectional antennas. The ability to exploit this property of wireless communication, which we refer to as the “wireless multicast advantage,” makes multicasting an excellent setting in which to study the potential benefits of energy-efficient protocols.

As a result of the wireless multicast advantage, the correct view of the omnidirectional wireless communication medium is as a *node-based* environment that is characterized by the following properties:

- A node’s transmission is capable of reaching another node if the latter is within communication range, which in turn means that the received signal-to-interference-plus-noise ratio exceeds a given threshold and that the receiving nodes have allocated (scheduled) receiver resources for this purpose.

- The total power required to reach a set of other nodes is simply the *maximum* required to reach any of them individually.

By contrast, in wired models, so long as there is a wire or cable link connecting two nodes, the reception is ensured over that link, and the cost of Node i ’s transmission to Node j and Node k would be the *sum* of the costs to the individual nodes.¹ Thus, wired networks can be viewed correctly as *link-based*.

4. Minimum-energy broadcast trees

We first address the problem of constructing the minimum-energy, source-based broadcast tree for each newly arriving broadcast session request.² Doing so involves the choice of transmitter-power levels and relay nodes. As noted earlier, we address only the energy spent for RF transmission. Thus, the total energy of the broadcast tree is simply the sum of the energy expended at each of the transmitting nodes in the tree; leaf nodes (which do not transmit) do not contribute to this quantity. Since we are considering session traffic, all transmitting nodes transmit for the entire duration of each session. Therefore, the total transmission energy is proportional to the total power needed to maintain the tree. Hence, we evaluate performance in terms of the total power required to maintain the tree.

Our focus in this paper is on the “logical” problem of establishing minimum-energy trees, rather than on the development of practical protocols for full-fledged multicast implementation. Thus, we do not address medium-access issues or the constraints of insufficient equipment or bandwidth resources. In [12,13] we applied the MIP algorithm introduced in this paper to simulations in which a sequence of many multicast requests compete for a finite quantity of transceivers and/or frequencies. An insufficient quantity of either of these resources can result in the construction of trees that do not reach all destinations, use more than the minimum energy (because only suboptimal trees can be constructed), or both. Also, we leave for future investigation the impact of the protocol overhead associated with the establishment of the trees.

It is important to emphasize a crucial difference between wired and wireless network multicasting. In wired networks, the broadcasting problem can be formulated as the well-known minimum-cost spanning tree (MST) problem. This formulation is based on the existence of a cost associated with each link in the network; the total cost of the broadcast tree is the sum of the link costs. The situation in wireless networks

¹ In wired networks, energy is not a concern; the cost of a link would typically be related to bandwidth and congestion (and hence delay) considerations. The case of wireless applications with highly directive antennas is similar to the case of wired networks in the sense that multiple beams may be needed to reach multiple destinations; thus the total cost of a node’s transmissions to its neighbors would be equal to the sum of the cost of the individual beams needed to reach each individual destination.

² An alternative approach to tree construction is the use of Core-Based Trees (CBT) [3], under which the same tree is used for all communication within a multicast or broadcast group. We study source-based trees exclusively in this paper.

is different, however, because of the “wireless multicast advantage” property, discussed in section 3, which permits all nodes within communication range to receive a transmission without additional expenditure of transmitter power. Therefore, the standard MST problem, which reflects the *link-based* nature of wired networks, does not capture the *node-based* nature of wireless networks. We do not know of any scalable solutions to the node-based version of this problem.³ In this paper we introduce a heuristic that takes into account the wireless multicast advantage in the formation of low-energy broadcast trees. We use such low-energy broadcast trees (including versions based on both link-based and node-based versions) as the basis for some of our heuristics for the construction of suboptimal multicast trees in wireless networks.

5. The Broadcast Incremental Power algorithm

Our objective is the determination of the minimum-power tree, rooted at the source node, that reaches all of the other nodes in the network. For wireless networks, this is a difficult problem for which no scalable solutions appear to be available at this time. Therefore, it is necessary to develop heuristics. The total power associated with the tree is simply the sum of the powers at all transmitting nodes. Clearly, this is a node-based (rather than link-based) metric because it enables us to exploit the wireless multicast advantage. Nevertheless, some of the algorithms we have studied are based on the minimization of link-based costs (as a heuristic for the minimization of the true, node-based cost).

In this section, we introduce and describe the Broadcast Incremental Power (BIP) algorithm, which is the major contribution of this paper. We also discuss two additional approaches, which are based on conventional networking techniques. Unlike the other two approaches, BIP exploits the wireless multicast advantage in the construction of the broadcast tree.

In all of the algorithms studied here, each transmission by a node is characterized by its transmitter power level, as well as a designation of which (possibly several) of the nodes receiving this transmission are to forward it toward which of the ultimate destination nodes. In small examples, we are able to compare our results to those obtained by an exhaustive search algorithm that is based on the recursive procedure discussed in [14].

5.1. Description of BIP

We describe the basic operation of BIP here by presenting a simple example of tree construction. A pseudocode description is provided in appendix A. The objective is to construct a minimum-cost (in our case, minimum-power) tree, rooted at the Source.

³ It was recently shown that the related (although somewhat different) problem of finding a minimal-size multipoint relay set in wireless networks is NP-complete [11].

Step 1. Figure 2(a) shows a ten-node network, in which Node 10 is the Source. A propagation constant of $\alpha = 2$ is assumed. Initially, the tree consists of only the Source. We begin by determining the node that the Source can reach with minimum expenditure of power, i.e., the Source’s nearest neighbor, which is Node 9. This node is added to the tree. Thus, at this point, two nodes are included in the tree, namely Node 10 and Node 9 (figure 2(a)). The notation $10 \rightarrow 9$ means that the addition to the tree at this step is the transmission from Node 10 to Node 9.

Step 2. We then determine which “new” node can be added to the tree at *minimum additional* cost. There are two alternatives. Either Node 10 can increase its power to reach a second node, or Node 9 can transmit to its nearest neighbor that is not already in the tree. In this example, Node 10 increases its power level to reach Node 6 (figure 2(b)). Note that the cost associated with the addition of Node 6 to the tree is the *incremental* cost associated with increasing Node 10’s power from a level sufficient to reach Node 9 to a level sufficient to reach Node 6. The cost of a transmission between Node 10 and Node 9 is $r_{10,9}^\alpha$, and the cost of a transmission between Node 10 and Node 6 is $r_{10,6}^\alpha$. The incremental cost associated with adding Node 6 to the preliminary tree consisting of Node 9 and Node 10 is $r_{10,6}^\alpha - r_{10,9}^\alpha$. We are able to exploit the wireless multicast advantage because both Node 6 and Node 9 can be reached when Node 10 transmits with sufficient power to reach Node 6.

Step 3. There are now three nodes in the tree, namely Node 6, Node 9, and Node 10. For each of these nodes, we determine the incremental cost to reach a new node. (Since Node 6 and Node 9 were not previously transmitting, their respective incremental costs will equal their full transmission powers if they are chosen to transmit; since Node 10 was already transmitting, its incremental cost is only the required increase in its transmission power.) The node that can be added with minimum incremental cost (which turns out to be Node 7) is added to the tree (figure 2(c)).

Continue. This procedure is continued until all nodes are included in the tree, as shown in figure 2(d). The order in which the nodes were added in steps 4 through 9 is: $6 \rightarrow 8$, $6 \rightarrow 5$, $9 \rightarrow 1$, $9 \rightarrow 3$, $9 \rightarrow 4$, $9 \rightarrow 2$.

Evaluation of transmitted power. The total power required to maintain this tree is the sum of the transmitted powers at each of the transmitting nodes. In this example, Nodes 10, 6, and 9 transmit, while the other nodes, which are leaf nodes, do not. The overall transmitter power is therefore

$$\begin{aligned} P &= \max(r_{10,6}^\alpha, r_{10,9}^\alpha) + \max(r_{6,5}^\alpha, r_{6,7}^\alpha, r_{6,8}^\alpha) \\ &\quad + \max(r_{9,1}^\alpha, r_{9,2}^\alpha, r_{9,3}^\alpha, r_{9,4}^\alpha) \\ &= r_{10,6}^\alpha + r_{6,5}^\alpha + r_{9,2}^\alpha. \end{aligned}$$

For the present case of $\alpha = 2$, we have $P = 10.90$.

BIP is similar in principle to Prim’s algorithm for the formation of MSTs, in the sense that new nodes are added to the tree one at a time (on a minimum-cost basis) until all nodes are included in the tree. In fact, the implementation of

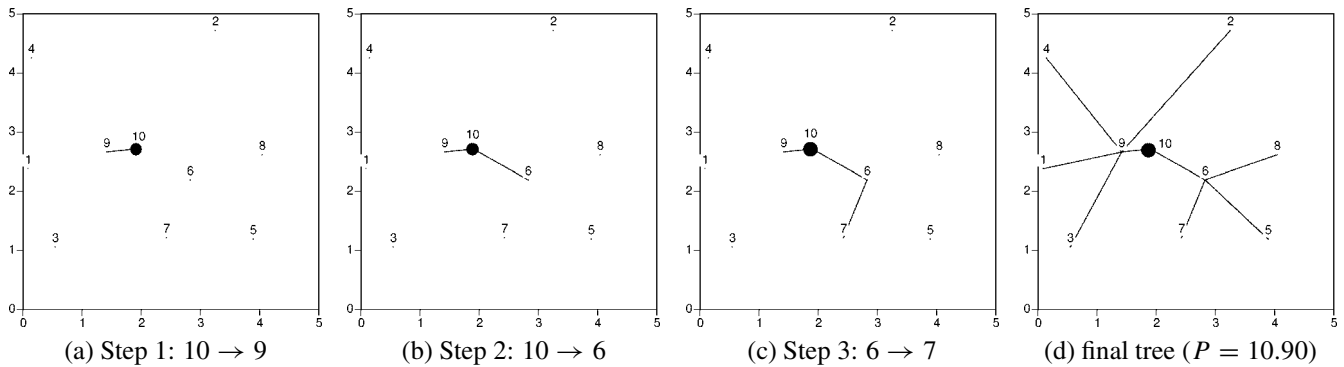


Figure 2. Example of tree construction using BIP (first three steps and final tree); $\alpha = 2$.

this algorithm is based on the standard Prim algorithm, with one fundamental difference. Whereas the inputs to Prim’s algorithm are the link costs P_{ij} (which remain unchanged throughout the execution of the algorithm), BIP must dynamically update the costs at each step (i.e., whenever a new node is added to the tree) to reflect the fact that the cost of adding new nodes to a transmitting node’s list of neighbors is the incremental cost. Consider an example in which Node i is already in the tree (it may be either a transmitting node or a leaf node), and Node j is not yet in the tree. For all such Nodes i (i.e., all nodes already in the tree), and Nodes j (i.e., nodes not yet in the tree), the following is evaluated:

$$P'_{ij} = P_{ij} - P(i), \quad (1)$$

where P_{ij} is the link-based cost of a transmission between Node i and Node j (i.e., it is r_{ij}^α), and $P(i)$ is the power level at which Node i is already transmitting (prior to the addition of Node j ; if Node i is currently a leaf node, $P(i) = 0$). The quantity P'_{ij} represents the incremental cost associated with adding Node j to the set of nodes to which Node i already transmits. The pair $\{i, j\}$ that results in the minimum value of P'_{ij} is selected, i.e., Node i transmits at a power level sufficient to reach Node j . Thus, one new node is added to the tree at every step of the algorithm.

Unlike Prim’s algorithm, which guarantees the formation of minimum-cost spanning trees for link-based costs (as in wired networks), BIP does not necessarily provide minimum-cost trees for wireless networks. However, neither do any other scalable algorithms that we are aware of. The performance results of section 7 demonstrate nonetheless that this algorithm does, in fact, provide improved performance over a wide range of examples.

5.2. Broadcast algorithms based on link-based techniques

Two of the algorithms we have studied are based on well-known techniques, namely the use of shortest unicast paths and the use of spanning trees, both of which use link-based costs.

5.2.1. Broadcast Least-Unicast-cost (BLU) algorithm

A straightforward (but far from optimal) approach is the use of broadcast trees that consist of the superposition of the best

unicast paths to each individual destination (see, e.g., [4]). It is assumed that an underlying unicast algorithm (such as the Bellman–Ford or Dijkstra algorithm) provides “minimum-distance” paths from the source node to every other node. Since BLU is based on the use of a scalable unicast algorithm, it also is scalable.

It is significant to note that, although algorithms based on minimum-distance paths are normally used for packet-switched applications, we are using this approach here for session-oriented traffic. We feel that it is appropriate to do so in wireless applications because a cost (involving power and possibly congestion) can be defined for each link in the network. By contrast, in circuit-switched wired applications it is difficult to define a link cost because energy is not of concern and because delay is not an appropriate metric (as it would be in packet-switched applications) since resources are reserved in circuit-switched applications. Instead, blocking probability is the only overall objective, and there is no known way of mapping that objective to individual link metrics. Summarizing the above, we have:

BLU: A minimum-cost path from the source node to every other node is established. The broadcast tree consists of the superposition of these unicast paths.

Figure 3(a) shows the multicast tree produced by BLU for the same example shown in figure 2. The power required to maintain this tree is $P = 12.17$. Note that whereas both Node 9 and Node 10 transmit with relatively high power under BLU, only Node 9 transmits with relatively high power under BIP. Consequently, the overall power consumption under BIP is somewhat lower for this example. Thus, the failure of BLU to exploit the wireless multicast advantage in the tree construction process results in higher overall power expenditure. However, the evaluation of the cost of the resulting tree (i.e., the total power needed to sustain the broadcast tree) does take into consideration the wireless multicast advantage.

5.2.2. Broadcast Link-based MST (BLiMST) algorithm

This algorithm is based on the use of the standard MST formulation (as in wired networks) in which a link cost is associated with each pair of nodes (i.e., the power to sustain the link). Thus, the “wireless multicast advantage” is ignored in the construction of the MST. Since the MST problem is of

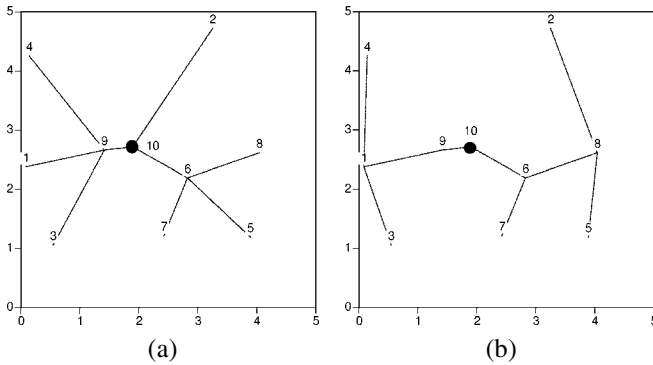


Figure 3. Trees produced by link-based algorithms ($\alpha = 2$). (a) Tree produced by BLU ($P = 12.17$). (b) Tree produced by BLiMST ($P = 13.22$).

polynomial complexity, it is scalable. Once the MST is constructed in this manner, the evaluation of its cost does take into consideration the wireless multicast advantage. Summarizing the above, we have:

BLiMST: A minimum-cost (minimum-power) spanning tree is formed using standard (link-based) MST techniques.

Figure 3(b) shows the multicast tree produced by BLiMST for the same example shown in figure 2. The power required to maintain this tree is $P = 13.22$. Similarly to the case of BLU, the failure of BLiMST to exploit the wireless multicast advantage results in higher overall power expenditure.

5.3. Complexity considerations

The complexity of BLU, when implemented by means of the Dijkstra algorithm, is $O(N^2)$, where N is the number of nodes in the network [1, p. 111].

The complexity of BLiMST, when implemented by means of Prim's algorithm, is $O(N^3)$ when a straightforward implementation is used [1, p. 524]. However, a more-sophisticated implementation using a Fibonacci heap yields complexity $O(M + N \log N) = O(N^2)$, where $M = N(N - 1)/2$ is the number of links (in a fully connected network).

We have noted that BIP is based on Prim's algorithm. Therefore, it also has complexity $O(N^3)$. Because of the need to update the costs P'_{ij} at each step of the algorithm, it is not yet clear whether the Fibonacci heap technique is applicable here.

5.4. The sweep operation: removing unnecessary transmissions

The performance of the algorithms presented here can be improved by eliminating unnecessary transmissions by means of what we call the "sweep" operation. Before describing the procedure, we show two examples. Figure 4(a) shows the tree that results from applying the sweep operation to the tree produced by BIP (figure 2(d)). In this example, Node 9 and Node 6 serve as relays. Nodes 1, 2, 3, 4, 5, 7, and 8 are leaf nodes; thus they do not transmit, and hence do not contribute

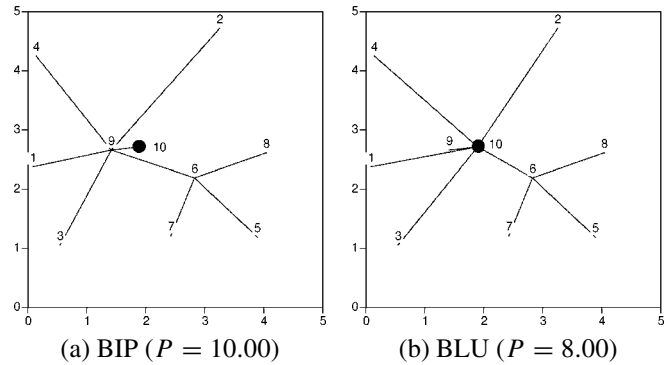


Figure 4. Post-sweep trees produced by BIP and BLU for the network of figure 2 ($\alpha = 2$).

to the power consumption. It is easy to see (by simple geometry, since we assume the use of omnidirectional antennas) that Node 9's transmitted power is sufficient to reach Node 6; thus Node 10 can reduce its power so that it reaches only Node 9. By doing so, the overall power consumption is reduced from $P = 10.90$ to $P = 10.00$.

Figure 4(b) shows a sweep example based on BLU. Again (see figure 3(a)), Node 9 and Node 6 serve as relays and Nodes 1, 2, 3, 4, 5, 7, and 8 are leaf nodes. The total transmitter power required to maintain this tree is $P = 12.17$. It is easy to see that the transmission by Node 10 (which is sufficient to reach Node 2) can also reach Nodes 1, 3, and 4 without further expenditure of power. Therefore, the transmission by Node 9 is unnecessary. The resulting power consumption is reduced from $P = 12.17$ to $P = 8.00$.

The sweep procedure, for which a pseudocode description is provided in Appendix B, is summarized as follows. We examine the nodes in ascending ID order, i.e., from 1 through N .⁴ Leaf nodes are ignored because they do not transmit. In our example for BLU, the non-leaf node with the lowest ID is Node 6, whose downstream neighbors are Nodes 5, 7, and 8; thus Node 6 is the first candidate for restructuring. Since the transmission by Node 6 does not reach any neighbors of Nodes 5, 7, and 8, no changes are made here. The situation at Node 9 is similar. Finally, we reach Node 10. As noted above, Node 10 reaches all of Node 9's downstream neighbors; therefore, the transmission by Node 9 can be eliminated.⁵ The complexity of the sweep, as currently implemented, appears to be $O(N^2)$.

We have applied the same sweep procedure to all of our algorithms. Typically, a single application of the sweep operation provides significant improvement; small further improvement can often be obtained by repeating the sweep once more, but little improvement has been found by additional applications of this procedure. In this particular example, BLU provides a lower power post-sweep tree than BIP; however, in

⁴ Alternative schemes for ordering the sequence in which the nodes are examined are currently under investigation, including schemes that start from the Source and progress outwardly along the tree.

⁵ In some cases, the sweep operation discovers opportunities to reduce transmitted power (although without eliminating a transmission entirely), namely when the responsibility of relaying to a particular node can be transferred to another node.

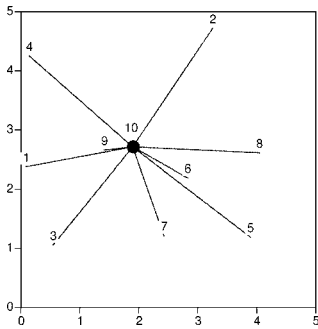


Figure 5. Optimal multicast tree for our example network for $\alpha = 2$ ($P = 6.30$).

most cases (see section 7) the tree produced by BIP has lower power, both before and after the sweep. BLiMST provides a tree with $P = 13.22$ before the sweep, with no improvement obtained by sweeping in this particular network.

5.5. The optimal tree

The optimal (lowest power) tree, obtained by exhaustive search, is a hub network (figure 5), in which the Source Node 10 transmits with sufficient power to reach all of the other nodes; the resulting tree power is $P = 6.30$.⁶ In most of our examples for which the true optimal solution is available (i.e., ten-node networks), however, the performance provided by the three algorithms we have studied is much closer to that of the optimal solution than in this particular example.

6. Algorithms for multicasting

It is well known that the determination of a minimum-cost multicast tree in wired networks is a difficult problem, which can be modeled as the NP-complete Steiner tree problem. The wireless multicast tree problem appears to be at least as hard in wireless networks as it is in wired networks. As we noted earlier, we know of no scalable algorithms for the minimum-energy broadcast problem. Thus, heuristics are definitely needed.

In this section we discuss three of the multicasting algorithms we have studied. They are direct analogs of the broadcasting algorithms discussed in section 5, and “Multicast” replaces “Broadcast” in their names.

Multicast Incremental Power (MIP) algorithm. A broadcast tree is formed using BIP. To obtain the multicast tree, the broadcast tree is *pruned* by eliminating all transmissions that are not needed to reach the members of the multicast group. More specifically, nodes with no downstream destinations will not transmit, and some nodes will be able to reduce their transmitted power (i.e., if their more-distant downstream neighbors have been pruned from the tree).

Multicast Least-Unicast-cost (MLU) algorithm. This is identical to BLU, except that unicast paths are established only to

⁶ In denser networks, and in networks in which the Source node is not centrally located, the optimal tree is rarely a hub configuration.

the desired destinations rather than to all network nodes. The multicast tree consists of the superposition of the appropriate unicast paths.

Multicast Link-based MST (MLiMST) algorithm. A broadcast tree is formed using BLiMST. As with MIP, to obtain the multicast tree, the broadcast tree is pruned by eliminating all transmissions that are not needed to reach the members of the given multicast group.

7. Performance results

We have evaluated the performance of the three algorithms for many network examples. Networks with a specified number of nodes (typically 10 or 100) are randomly generated within a square region (e.g., 5×5 , as in figure 2), i.e., the location of each node is randomly generated. One of the nodes is randomly chosen to be the Source. Multicast groups of a specified size are chosen randomly from the overall set of nodes. Since broadcasting is simply a special case of multicasting in which all of the nodes are included in the multicast group, we discuss our results for broadcasting and multicasting together and refer solely to the multicasting versions of the algorithms. No restrictions are placed on the maximum transmitter power (i.e., $p_{\max} = \infty$). The transmitter power actually used (r^α) depends on the distance (r) to the farthest neighbor to which a node is transmitting. We have considered propagation loss exponents of $\alpha = 2$ and $\alpha = 4$. In all cases (i.e., for a specified network size, multicast group size, and tree algorithm), our results are based on the performance of 100 randomly generated networks.

Our performance metric is the total power of the multicast (or broadcast) tree. To facilitate the comparison of our algorithms over a wide range of network examples, we present our results in terms of the normalized power for each network example. For each individual network example, say network m , we specify the locations of the nodes, the Source, and the destinations, and we compute the power associated with the multicast tree generated by each of the algorithms. To determine a benchmark for each network instance, we define

$$Q_i(m) = \text{total power of multicast tree for network } m, \text{ generated by algorithm } i, \quad (2)$$

and

$$Q_{\text{best}}(m) = \min\{Q_i(m), i \in I\}, \quad (3)$$

where I is the set of algorithms. Thus, $Q_{\text{best}}(m)$ is the power of the lowest-power tree among the set of algorithms (for the particular network instance m). For our larger (100-node network) examples, the set I consists of algorithms 1, 2, and 3. For our small (ten-node network) examples, we also have results for a fourth algorithm, namely the recursive exhaustive search algorithm [14], which provides the optimal solution; thus $Q_{\text{best}}(m)$ is the true value of the power of the optimal

tree in such cases. We then define the *normalized power* associated with algorithm i to be

$$Q'_i(m) = \frac{Q_i(m)}{Q_{\text{best}}(m)}. \quad (4)$$

This metric provides a measure of how close each algorithm comes to providing the lowest-power tree. In all cases, we have used the performance values obtained after a single application of the Sweep algorithm, which was discussed in section 5.

Note. Consider a 5×5 square unit region in which nodes are randomly distributed (such as figure 2). It is important to note that the units of distance used in this study are arbitrary. For example, the doubling of all distances in the 5×5 region while the nodes maintain their relative positions (resulting in nodes being distributed in a 10×10 square unit region), results in link power levels that increase by a factor of 2^α . Since the resulting trees produced by the algorithms studied here remain the same under any scaling of the distance, the overall tree power levels also increase by the same factor of 2^α . Therefore, any ratio comparisons of the tree powers produced by the algorithms are independent of the distance scaling factor. In other words, the normalized power of equation (4) does not depend on the size of the region, even though the absolute power does.

Table 1 summarizes performance for the three algorithms for networks with ten nodes, various multicast-group sizes, and for a propagation constant value of $\alpha = 2$. Each entry represents the results for 100 randomly generated networks. The entries in the table are mean and variance, respectively, of the normalized tree power $Q'_i(m)$. The multicast-group size includes the source node in addition to the destination nodes. For the case of a single destination (group size = 2), MLU provides the optimal solution because this is simply the unicast routing problem. For a group size of 5, MLU still provides the best performance in terms of mean tree power; MIP provides a slightly higher mean value, but a smaller variance. For the case of broadcasting (group size = 10), MIP provides the best performance, both in terms of mean and variance. In fact, MIP performs better than MLiMST for all group sizes.

Table 2 provides similar results for $\alpha = 4$. MIP again provides improved performance, as compared to the other two algorithms, for group sizes of 5 and greater.

Performance results for 100-node networks are shown in tables 3 and 4. As noted above, the normalization is taken with respect to the best of the three algorithms under study because the true optimal values are not available. As in the case of the smaller networks, MLU performs better than the other two algorithms when multicast groups are small. This is not surprising, since this algorithm provides a minimum-cost path to each individual destination. By contrast, the other two algorithms are based on the formation of a minimum-cost tree that reaches all nodes (with subsequent pruning to eliminate paths to unneeded destinations), and therefore may provide long, circuitous paths to some of the destinations. For $\alpha = 2$,

Table 1
Mean and variance of normalized tree power for 100 network instances:
10-node networks, $\alpha = 2$.

Group size	MLU	MLiMST	MIP
2	1.0000; 0.0000	1.1256; 0.1244	1.0583; 0.0201
5	1.1040; 0.0245	1.1852; 0.0415	1.1055; 0.0193
10	1.1388; 0.0270	1.1987; 0.0471	1.1049; 0.0145

Table 2
Mean and variance of normalized tree power for 100 network instances:
10-node networks, $\alpha = 4$.

Group size	MLU	MLiMST	MIP
2	1.0000; 0.0000	1.0381; 0.0315	1.0285; 0.0209
5	1.0401; 0.0120	1.0430; 0.0124	1.0283; 0.0091
10	1.0473; 0.0098	1.0474; 0.0128	1.0229; 0.0050

Table 3
Mean and variance of normalized tree power for 100 network instances:
100-node networks, $\alpha = 2$.

Group size	MLU	MLiMST	MIP
5	1.0410; 0.0072	1.2711; 0.0757	1.1892; 0.0440
10	1.0680; 0.0088	1.1496; 0.0153	1.0729; 0.0161
25	1.1060; 0.0097	1.0831; 0.0057	1.0135; 0.0010
50	1.1495; 0.0103	1.0801; 0.0020	1.0025; 0.0001
75	1.1502; 0.0047	1.0683; 0.0019	1.0009; 0.0000
100	1.1456; 0.0044	1.0676; 0.0019	1.0010; 0.0000

Table 4
Mean and variance of normalized tree power for 100 network instances:
100-node networks, $\alpha = 4$.

Group size	MLU	MLiMST	MIP
5	1.0623; 0.0178	1.1211; 0.0309	1.1058; 0.0223
10	1.1123; 0.0246	1.0570; 0.0064	1.0409; 0.0051
25	1.1243; 0.0154	1.0359; 0.0027	1.0116; 0.0006
50	1.1531; 0.0128	1.0253; 0.0006	1.0030; 0.0001
75	1.1673; 0.0153	1.0198; 0.0005	1.0027; 0.0001
100	1.1615; 0.0105	1.0232; 0.0006	1.0016; 0.0000

MIP provides the best performance for group sizes of 25 or greater; for $\alpha = 4$, MIP is best for group sizes of 10 or more. MIP again provides better performance than MLiMST in all cases, as it did in the ten-node network examples.

Our performance results indicate that our proposed algorithm, MIP, provides better performance than MLiMST over the complete range of network examples that we have studied, based on the criteria of mean tree power as well as variance. We attribute this improved performance to the fact that MIP exploits the node-based wireless multicast advantage property, whereas MLiMST ignores this property as it forms trees on the basis of link-based costs. Both MIP and MLiMST provide better performance than MLU when the size of the multicast groups is moderate to large. When multicast groups are large, the structure obtained by first establishing a broadcast tree is highly beneficial. However, when multicast groups are small, many energy-inefficient paths are established; such behavior would be expected even for truly optimal broadcast trees, and is a consequence of the suboptimal nature of the

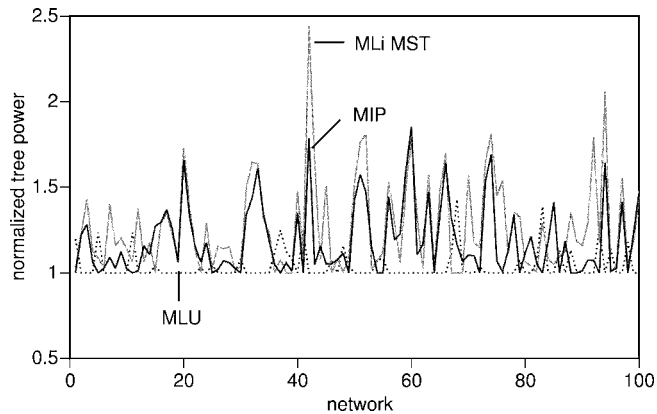


Figure 6. Normalized tree power for the three algorithms for 100 network instances; 100-node networks, multicast group size = 5, $\alpha = 2$.

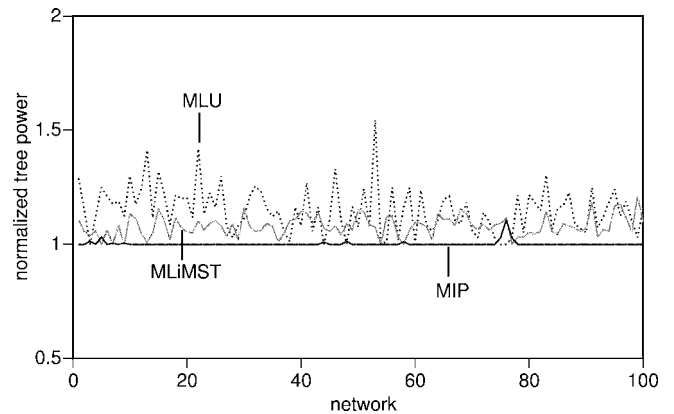


Figure 8. Normalized tree power for the three algorithms for 100 network instances; 100-node networks, multicast group size = 50, $\alpha = 2$.

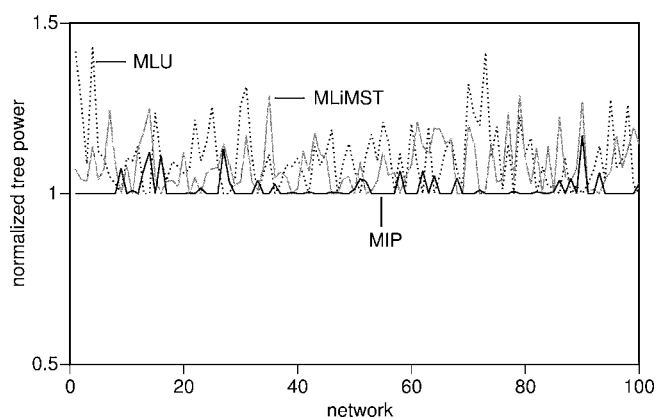


Figure 7. Normalized tree power for the three algorithms for 100 network instances; 100-node networks, multicast group size = 25, $\alpha = 2$.

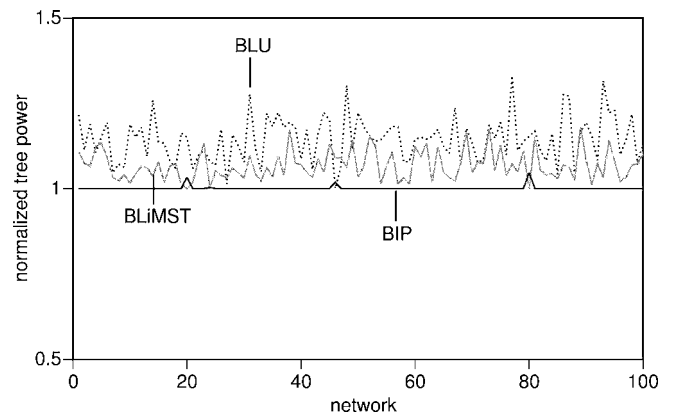


Figure 9. Normalized tree power for the three algorithms for 100 network instances; 100-node networks, broadcast case (i.e., multicast group size = 100), $\alpha = 2$.

pruning operation. Our observations apply also for the case of $\alpha = 4$. As α increases, the penalty for using longer links increases; thus trees may consist of a larger number of shorter links, but this fact does not change the relative behavior of the algorithms.

Figures 6–9 illustrate graphically the relative performance of the algorithms we have studied. The horizontal axis represents the network ID (m ranges between 1 and 100), and the vertical axis is the normalized tree power $Q'_i(m)$. These results, which correspond to a subset of those presented in table 3, permit us to evaluate the relative performance of the algorithms for a set of 100 network instances. MLU (which is represented by the dotted curves) often provides the best performance for small multicast groups, but this advantage decreases and eventually disappears as the size of the multicast group increases. Its performance is usually the worst of the three algorithms for large multicast group sizes, although there is one case in which it provides the best performance for the broadcast case.

The lighter solid curve represents MLI MST, and the darker solid curve represents MIP. For most network instances, MIP performs better than MLI MST. For large multicast group sizes, MIP provides the best performance in almost all network instances.

We remarked in section 1 that, although mobility is not explicitly addressed in this paper, our model allows some “elasticity” because transmitter power is controlled as part of the tree construction process. Adjusting transmitter power in response to node motion (without reconstructing the tree) certainly can result in trees that are not as energy efficient as those that would be obtained by re-executing MIP. However, simple power adjustment of this type does provide a robust way to maintain connectivity. Our future studies will address ways to incorporate mobility into the tree construction process.

Thus far, we have discussed only the normalized values of tree power. This metric is useful for the comparison of the relative performance of our algorithms. Additional insight into the properties of the algorithms can be obtained by looking at the actual values of tree power. Table 5 shows the mean value of the tree power for the same set of network instances summarized in table 3 for a 5×5 region. As noted earlier, the actual communication ranges (rather than their relative values) are relevant to the evaluation of absolute values of power, but not to the normalized values. Also shown, in parentheses below the mean value, are the minimum and maximum values of tree power observed over the set of 100 network instances for each of the three algorithms. Table 6 shows the corresponding

Table 5

Mean tree power (and minimum and maximum values) for 100 network instances: 100-node networks, 5×5 region, $\alpha = 2$ (see table 3).

Group size	MLU	MLiMST	MIP
5	3.983 (1.716, 8.017)	4.818 (1.775, 7.735)	4.511 (1.737, 7.618)
25	8.819 (6.157, 11.014)	8.631 (6.437, 10.626)	8.096 (6.162, 9.868)
50	10.769 (8.360, 13.683)	10.135 (7.696, 12.603)	9.415 (7.118, 11.792)
100	12.348 (10.149, 14.926)	11.514 (9.762, 13.323)	10.802 (9.004, 12.592)

Table 6

Mean tree power (and minimum and maximum values) for 100 network instances: 100-node networks, 5×5 region, $\alpha = 4$ (see table 4).

Group size	MLU	MLiMST	MIP
5	1.129 (0.269, 2.398)	1.180 (0.337, 2.389)	1.164 (0.337, 2.361)
25	2.552 (1.534, 5.611)	2.352 (1.410, 4.473)	2.300 (1.393, 4.472)
50	3.137 (1.975, 5.214)	2.797 (1.709, 4.584)	2.738 (1.645, 4.461)
100	3.792 (2.530, 9.436)	3.351 (2.239, 9.373)	3.282 (2.188, 9.325)

results for the examples of table 4. The network that results in the minimum (or maximum) value of tree power is typically different for the three algorithms.

The mean tree power increases as the multicast group size increases. This is certainly expected, since a greater number of destinations must be reached. The ratio of maximum to minimum values decreases as the group size increases. This is also expected, because when the group size is small, the minimum power tree depends not only on the random locations of the complete set of 100 nodes, but also on which nodes are included in the group.

The values of tree power depend on the units of range. Recall that the transmitted power is defined to be r^α , where we have arbitrarily defined the units of range such that all nodes are randomly located with a 5×5 square region; scaling the range to a different set of units would result in considerably different values. In our examples, the values of tree power are lower for $\alpha = 4$ than for $\alpha = 2$. This is a consequence of the fact that most of the communication ranges are less than 1 (recall that the 100 nodes are randomly located in a 5×5 region). Also, there is a significantly greater variation in the ratio of maximum to minimum values for $\alpha = 4$. This results from the greater cost of longer links for higher values of α .

8. Conclusions

In this paper, we have identified some of the fundamental issues associated with energy-efficient broadcasting and multicasting in infrastructureless wireless networks, and we have presented preliminary algorithms for the solution of this problem. Our studies show that improved performance can be ob-

tained when exploiting the properties of the wireless medium; i.e., networking schemes should reflect the node-based nature of wireless communications, rather than simply adapt link-based schemes originally developed for wired networks. In particular, the Broadcast Incremental Power (BIP) and Multicast Incremental Power (MIP) algorithms, which exploit the wireless multicast advantage, provide better performance than the other algorithms we have studied over a wide range of network examples.

Furthermore, the fact that improved performance can be obtained by jointly considering physical layer issues and network layer issues suggests that novel approaches to wireless networking, which incorporate the vertical integration of protocol layer functions, may provide advantages over traditional network architectures. A major challenge, and a topic of continued research, is the development of distributed algorithms that provide the benefits that have been demonstrated in this paper. Furthermore, it is important to study the impact of limited bandwidth and transceiver resources, as well as to develop mechanisms to cope with node mobility.

Appendix A. Pseudocode for the Broadcast Incremental Power (BIP) algorithm

The following pseudocode describes the Broadcast Incremental Power (BIP) algorithm, which is implemented in the same manner as the straightforward version of the Prim algorithm, with the exception of the addition of a single code line, as indicated below.

```

Broadcast_Incremental_Power_Algorithm( )
{
  S = {source};
  TREE[source's_nearest_neighbor] = source;
  for (i = 1; i <= N, i++)
    for (j = 1; j <= N, j++)
      link_cost_matrix[i][j] = P[i][j];
  while (|S| < N)
  {
    (I,J) = compute_nearest_link
      (TREE, link_cost_matrix);
    S = S U {J};
    TREE[J] = I;
    /*- the following step is not needed -*/
    /*- in Prim's algorithm -*/
    while (j is not in S)
      link_cost_matrix[I][j] = P[I][j] - P[I][J];
  }
}

```

```

compute_nearest_link(TREE, link_cost_matrix)
{
  link_cost_min = infinity;
  for (all i on TREE)
    for (all j outside TREE)
      if (link_cost_matrix[i][j] < link_cost_min)
      {
        link_cost_min = link_cost_matrix[i][j];
        I = i;
      }
}

```

```

    J = j;
  }
return (I,J);
}

```

Appendix B. Pseudocode for the sweep operation

The input to the sweep routine is a tree, such as that produced by BIP or an alternative method. In such a tree, all nodes (except the Source) have a single “father” node. All nodes (except leaf nodes) have one or more “son” nodes. The sweep operation consists of the following steps, which (in the version presented here) are executed at each transmitting node in increasing order of node ID.

- At Node i : A list is made of all nodes reached by Node i at its current transmission power level, excluding upstream nodes (i.e., nodes between the Source and Node i in the current version of the tree); this list of nodes includes not only son nodes, but also other nodes within communication range. The two versions of the sweep differ in how they process this list of Node i 's neighbors. In **sweep (TREE)**, a node becomes a son of Node i (instead of its previous father node) only if doing so permits the reduction of transmitter power at its previous father node. In **simplified_sweep (TREE)**, all nodes in this list become sons of Node i (instead of their previous father nodes), whether or not such changes in the tree result in reduced transmitter power at their father nodes. In both versions, when a node has lost one or more sons, its transmitter power is reduced wherever possible, i.e., to a level that is sufficient to reach its sons in the new tree (after nodes are assigned to new fathers).
- Repeat at Node $i + 1$.
- Continue until all transmitting nodes have been examined.

At each step, the tree structure is maintained. Also, at each step, the cost of the tree may decrease, but will never increase.

sweep (TREE)

```

{
tree_cost_min = cost (TREE);
SWEPT_TREE = TREE;
  for (i = 1; i <= N; i++)
    if (i == transmitting node)
      {
TEST_TREE = SWEPT_TREE;
update upstreams of i;
for (j=1; j <= N; j++)
  {
    if (j is within the transmission range
      of i and j is not an upstream of i)
      TEST_TREE[j] = i;
  }
tree_cost = cost (TEST_TREE);
if (tree_cost < tree_cost_min)
  {
tree_cost_min = tree_cost;
SWEPT_TREE = TEST_TREE;

```

```

    }
  }
return SWEPT_TREE;
}

simplified_sweep (TREE)
{
for (i = 1; i <= N; i++)
  if (i == transmitting node)
    {
update upstreams of i;
for (j = 1; j <= N; j++)
  {
    if (j is within the transmission
      range of i and j is not an upstream
      of i) TREE[j] = i;
  }
}
return TREE;
}

```

References

- [1] R.K. Ahuja, T.L. Magnanti and J.B. Orlin, *Network Flows* (Prentice-Hall, Englewood Cliffs, NJ, 1993).
- [2] M.H. Ammar, G.C. Polyzos and S.K. Tripathi (eds.), Special issue on network support for multipoint communication, *IEEE Journal on Selected Areas in Communications* 15(3) (April 1997).
- [3] A.J. Ballardie, P.F. Francis and J. Crowcroft, Core based trees, in: *Proceedings of ACM SIGCOMM'93*, San Francisco, CA (1993).
- [4] D. Bertsekas and R. Gallager, *Data Networks* (Prentice-Hall, Englewood Cliffs, NJ, 1992).
- [5] C.-C. Chiang, M. Gerla and L. Zhang, Forwarding group multicast protocol (FGMP) for multihop, mobile wireless networks, *Cluster Computing* 1(2) (December 1998) 187–196.
- [6] C. Diot, W. Dabbous and J. Crowcroft, Multipoint communication: A survey of protocols, functions, and mechanisms, *IEEE Journal on Selected Areas in Communications* 15(3) (1997) 277–290.
- [7] K. Makki, N. Pissinou and O. Frieder, Efficient solutions to multicast routing in communication networks, *Mobile Networks and Applications* 1 (1996) 221–232.
- [8] M. Nagy and S. Singh, Multicast scheduling algorithms in mobile networks, *Cluster Computing* 1(2) (December 1998) 177–185.
- [9] NSF, Research Priorities in Wireless and Mobile Communications and Networking, Report of a Workshop (March 1997).
- [10] C.D. Rais and S.K. Tripathi, Studying vertical dependence to improve NFS performance in wireless networks, *Cluster Computing* 1(2) (December 1998) 225–235.
- [11] L. Viennot, Complexity results on election of multipoint relays in wireless networks, Report RR-3584, INRIA (December 1998).
- [12] J.E. Wieselthier, G.D. Nguyen and A. Ephremides, Algorithms for bandwidth-limited energy-efficient wireless broadcasting and multicasting, in: *Proceedings of MILCOM 2000*, Los Angeles, CA (October 2000) 485–490.
- [13] J.E. Wieselthier, G.D. Nguyen and A. Ephremides, Energy-efficient wireless multicast of session traffic, in: *Proceedings of Hawaii International Conference on System Sciences (HICSS-34)*, Maui, Hawaii (January 2001).
- [14] J.E. Wieselthier, G.D. Nguyen and A. Ephremides, Algorithms for energy-efficient multicasting in static ad hoc wireless networks, *Mobile Networks and Applications* 6(3) (June 2001) 251–263.
- [15] L. Zhang, S. Deering, D. Estrin, S. Shenker and D. Zappala, RSVP: A new resource reservation protocol, *IEEE Network Magazine* (September 1993) 8–18.



Jeffrey E. Wieselthier was born in Brooklyn, NY, on March 14, 1949. He received the S.B. degree from the Massachusetts Institute of Technology, Cambridge, in 1969, the M.S. degree from the Johns Hopkins University, Baltimore, MD, in 1971, and the Ph.D. degree from the University of Maryland, College Park, in 1979, all in electrical engineering. He was employed at the Naval Surface Warfare Center, White Oak, Silver Spring, MD, from 1969 to 1979. Since 1979 he has been with the Information Technology Division of the Naval Research Laboratory, Washington, DC. He was Technical Program Co-Chair of the Third IEEE Symposium on Computers and Communications in Athens, Greece, in 1998 and Treasurer of the 1991 IEEE International Symposium on Information Theory in Budapest, Hungary. He won the IEEE Fred W. Ellersick Award for the best unclassified paper at MILCOM 2000. He has studied a variety of communication networking problems, including multiple access and routing in spread-spectrum networks and the use of neural networks and other approaches for network performance evaluation, optimization and control. His current interests include wireless communication networks, with an emphasis on issues relating to energy-efficient operation. Dr. Wieselthier is a member of Eta Kappa Nu and Sigma Xi, and is a Senior Member of IEEE.
E-mail: wieselthier@itd.nrl.navy.mil



Gam D. Nguyen received the Ph.D. in electrical engineering from the University of Maryland, College Park, MD in 1990. He has been at the Naval Research Laboratory, Washington, DC, since 1991. He won the IEEE Fred W. Ellersick Award for the best unclassified paper at MILCOM 2000. His research interests include communication systems, computer communication networks, and information processing.
E-mail: nguyen@itd.nrl.navy.mil



Anthony Ephremides received his B.S. degree from the National Technical University of Athens (1967), and M.S. (1969) and Ph.D. (1971) degrees from Princeton University, all in electrical engineering. He has been at the University of Maryland since 1971, and currently holds a joint appointment as Professor in the Electrical and Computer Engineering Department and the Institute for Systems Research (ISR). He is co-founder of the NASA Center for Commercial Development of Space on Hybrid and Satellite Communications Networks established in 1991 at Maryland as an off-shoot of the ISR. He was a Visiting Professor in 1978 at the National Technical University in Athens, Greece, and in 1979 at the EECS Department of the University of California, Berkeley, and at INRIA, France. During 1985–1986 he was on leave at MIT and ETH in Zurich, Switzerland. He was the General Chairman of the 1986 IEEE Conference on Decision and Control in Athens, Greece and of the 1991 IEEE International Symposium on Information Theory in Budapest, Hungary. He was the Technical Program Co-Chair of the IEEE INFOCOM in New York City in 1999 and of the IEEE International Symposium on Information Theory in Sorrento, Italy in 2000. He has also been the Director of the Fairchild Scholars and Doctoral Fellows Program, an academic and research partnership program in Satellite Communications between Fairchild Industries and the University of Maryland. He has received several awards, including the IEEE Donald E. Fink Prize Paper Award (1992) and the Sigmobility Award of the ACM for contributions to wireless communications (1997). He has been the President of the Information Theory Society of the IEEE (1987), and has served on the Board of the IEEE (1989 and 1990). He is a Fellow of IEEE. Dr. Ephremides' interests are in the areas of communication theory, communication systems and networks, queueing systems, signal processing, and satellite communications.
E-mail: tony@eng.umd.edu