

# Energy Efficient Broadcast in Wireless Ad hoc Networks with Hitch-hiking

Manish Agarwal, Joon Ho Cho, Lixin Gao  
 Dept. of Electrical and Computer Engineering  
 University of Massachusetts  
 Amherst, MA 01002, USA  
 Email: {magarwal,jhcho,lgao}@ecs.umass.edu

Jie Wu  
 Dept. of Computer Science and Engineering  
 Florida Atlantic University  
 Boca Raton, Fl 33431, USA  
 Email: jie@cse.fau.edu

**Abstract**—In this paper, we have proposed a novel concept called *Hitch-hiking* in order to reduce the energy consumption of broadcast application for wireless networks. Hitch-hiking takes advantage of the physical layer design that facilitates the combining of partial signals to obtain the complete information. The concept of combining partial signals using *maximal ratio combiner* [13] has been used to improve the reliability of the communication channel but has never been exploited to reduce energy consumption. We propose a centralized heuristic algorithm called *Wireless Multicast with Hitch-hiking (WMH)* to construct an energy efficient tree using Hitch-hiking and also present a distributed version of the heuristic. We also evaluate the proposed heuristics through simulation. Simulation results show that Hitch-hiking can reduce the transmission cost of broadcast by as much as 50%. Further, we propose and evaluate a protocol called *PSBT* that reduces energy consumption of broadcast by eliminating redundancy in receive operation. Finally, we propose an algorithm that takes advantage of both Hitch-hiking and PSBT in conserving energy.

**Index Terms**—Wireless ad hoc networks, broadcast algorithms, distributed algorithms, maximal ratio combiner, simulations

## I. INTRODUCTION

Wireless networks in the form of ad hoc networks and sensor networks have gained lots of attention in recent years. The rapidly increasing capabilities and declining costs of computing and communication devices have made it possible to use wireless networks in a wide range of applications that can improve quality of life, and even save lives. Sensor networks find their application in healthcare (e.g., health monitoring and coordination among doctors and nurses), aircraft flight control, weather forecasting, home appliance control, and protection against bioterrorism. Ad hoc networks can be used for communication in ad hoc settings such as in conferences or classrooms. One of the key challenges in the de-

ployment of wireless networks is how to prolong the lifetime of the networks. The lifetime of wireless networks is limited by the battery energy in wireless devices. Sensor networks will stress power sources because of their need for long operating lifetimes and high energy density [9], [6], [18]. Furthermore, the lifetime of batteries has not been improved as fast as processing speed of microprocessors. Therefore, energy efficiency is critical for the wide deployment of wireless networks.

Power saving techniques for ad hoc networks can be broadly classified into two categories: power saving protocols and power control for transmission. A *power saving protocol* [4], [16], [20] aims to put wireless nodes into periodical sleep state in order to reduce the power consumption in the idle listening mode. *Power control for transmission* [19], [14] manages energy consumption by adjusting transmission ranges. Our work deals with conserving power by employing power control for transmission.

Broadcast is a very important communication primitive used in wireless networks. Wireless networks, due to their ad hoc nature and mobile environment, make frequent use of broadcast primitives to adapt to network changes. Broadcast is also widely used in sensor networks to disseminate information about environmental changes to other nodes in the network. Therefore, it is essential to develop efficient broadcast protocols that are optimized for energy consumption.

In this paper, we investigate the problem of minimizing the total energy consumed in broadcasting data. Our key idea is to reduce energy consumption of the broadcast application by taking advantage of the physical layer design that facilitates the combining of partial information to obtain complete information. We refer to this as *Hitch-hiking*. By the effective use of partial signals, a packet can be delivered with less transmission power. The concept of combining partial signals using *maximal ratio combiner*

[13] has been used in physical layer design of wireless systems to increase reliability but has never been exploited to reduce energy consumption at the network layer.

Our contribution in this paper is to propose and analyze a centralized heuristic algorithm called Wireless Multicast with Hitch-hiking (WMH) and a distributed algorithm that can take advantage of Hitch-hiking to reduce the overall energy consumption of broadcast. The performance analysis shows that Hitch-hiking can reduce the energy consumption of the broadcast application by as much as 50%. Further, we propose and evaluate a protocol called Power Saving with Broadcast Tree (PSBT) to conserve the energy wasted by a node's receipt of the same packet many times in a broadcast. Finally, we propose an algorithm that takes advantage of both Hitch-hiking and PSBT in conserving energy.

The rest of the paper is organized as follows. In Section II, we give an overview of the related work concerning minimum-energy broadcast problem. In Section III, we describe the signal and system design behind Hitch-hiking. In Section IV, we propose heuristic algorithms that use Hitch-hiking to reduce energy consumption in broadcast application. In Section V, we propose and analyze a protocol called PSBT to reduce the effect of energy consumed in receiving a packet on the overall energy consumption of the broadcast. We also propose an algorithm that works with PSBT and takes advantage of Hitch-hiking to reduce the energy consumption. Finally, we conclude in Section VI.

## II. RELATED WORK

Both power saving protocols and power control for transmission have been studied in various settings. In [16], Singh et al. proposed a protocol called PAMAS that uses a second low power radio channel to detect activity from neighbors and turns on a node only when a neighbor communicates with the node. In Span [4], a small dominating set is selected locally and nodes outside the set are put into the sleep state. Power control has been studied for broadcast. In the source-independent approach, all nodes can be a source and are able to reach all other nodes by assigning appropriate ranges. The problem of minimizing the total transmission power consumption is NP-complete for both 2-D space [5] and 3-D space [10]. There are several heuristic solutions [12] for this problem. In the source-dependent approach, the source is given, but the problem is still NP-complete.

BIP [19] is a widely used heuristic approach to construct a minimum-energy broadcast tree rooted at the source node. BIP constructs the broadcast tree by starting with the source node as the only node in the tree and adds

one node at a time to the tree. Each time BIP has to add a node to the tree, it chooses the uncovered node which can be added to the tree at minimum additional cost. This process is continued until all the nodes in the network are added to the tree. It is important to understand that BIP takes advantage of *Wireless Multicast Advantage* referred to as WMA to some extent. WMA is the property of wireless channel through which a single transmission can be received by all those nodes that fall in the transmission range of the transmitting node. Several approximation methods with bounds have also been proposed [11],[17]. In [17], Wan et al. gave the analytical results for different algorithms to construct a minimum energy broadcast tree. They proved that the approximation ratio of MST is between 6 and 12 and that of BIP is between  $\frac{13}{3}$  and 12. Wan et al. also found that BIP fails to take full advantage of WMA for many cases, because it adds just one node at a time to the broadcast tree. BIP can be improved by allowing it to add more than one node at a time, but that process requires a different selection criteria than one proposed by the authors of BIP.

In [2], Cagalj et al. proposed a heuristic called EWMA that builds an energy efficient broadcast tree by improving upon an initial feasible solution using wireless multicast advantage. In EWMA, every forwarding node in the initial solution is given a chance to increase its power level if the overall cost of the tree decreases at the new power level. Each node finally chooses the power level at which the overall decrease in cost of the final tree is maximized. The authors of EWMA have also given a distributed version of the algorithm called DEWMA.

Recently, Cartigny et. al. proposed a localized energy-efficient broadcast [3] algorithm using relative neighborhood graph (RNG) as their base tree. In [1], Suman et al. proposed schemes for constructing energy efficient broadcast and multicast trees for reliable wireless communication.

All of the aforementioned studies assume that a node can only decode a signal whose signal strength is above a certain threshold and rest of the signals are ignored. In the following section, we describe a mechanism to even use signals with signal strength below a threshold to reduce the overall cost of the broadcast tree.

## III. BACKGROUND

In this section, we describe the signal and system design for the physical layer of the wireless ad hoc network and introduce related concepts that play crucial roles in our network layer design for Hitch-hiking.

### A. Packetization

We assume that messages are packetized. A packet consists of a preamble, a header, and a payload. A preamble is a sequence of pre-specified uncoded symbols assigned to facilitate timing acquisition, a header contains the error-control coded information sequence about the source/destination addresses and other control flags, and a payload contains the error-control coded message sequence.

We also assume that the header and the payload in a packet are the outputs of two different channel encoders, and that the two channel codes are used by all the nodes in the system. The separation of a header and a payload in channel coding enables a receiver to retrieve the information in a header without decoding the entire packet. The use of the same channel codes enables a receiver to enhance the signal-to-noise ratio at the input to the channel decoder by combining the payloads of multiple packets containing the same encrypted message. For the details of packet combining, see section III-D.

### B. Signal-to-Noise Ratio and Interference

One of the most important properties of a wireless channel is that a transmitted signal can reach any receiver, as long as the channel gain between the transmitter and the receiver is not zero. Hence, when the  $k$ th sensor transmits a packet to the  $l$ th sensor, the  $l'$ th sensor, with  $l'$  not necessarily equal to  $l$ , can receive the signal with the signal-to-noise ratio (SNR) per symbol given by

$$\gamma_{k,l'} = \frac{\alpha_{k,l'} E_k}{N_0}, \quad (1)$$

where  $\alpha_{k,l'}$  is the gain of the wireless link from the  $k$ th sensor to the  $l'$ th sensor,  $E_k$  is the energy of the transmitted signal by the  $k$ th sensor, and  $N_0$  is the noise density. The gain  $\alpha_{k,l'}$  includes the effects of propagation loss, antennas and amplifiers, and channel fading and shadowing [15, Chapters 3-4].

When there are  $K$  sensors transmitting packets at the same time, a receiver suffers the degradation in the SNR due to the interference. For example, if direct-sequence spread-spectrum (DS/SS) technique is employed as the modulation scheme and the conventional matched filter receiver is employed as the detector, the SNR of the packet transmitted by the  $k$ th sensor and received by the  $l$ th sensor is given by

$$\gamma_{k,l} = \frac{\alpha_{k,l} E_k}{N_0 + \frac{1}{N} \sum_{k'=1, k' \neq k}^K \alpha_{k',l} E_j} \quad (2)$$

where  $N$  is called the processing gain, which determines the level of interference suppression. However, when the

processing gain is very large or the scheduling algorithm is designed to avoid the collision of packets, the SNR (2) becomes the same as (1).

### C. Partial Reception

We introduce two<sup>1</sup> thresholds on the SNR (Signal-to-noise ratio): one is the threshold  $\gamma_p$  to successfully decode the payload and the other is the threshold  $\gamma_{acq}$  to successfully acquire the timing. We assume that the system is designed to have  $\gamma_{acq} < \gamma_p$  which is a reasonable assumption. Due to the discrepancy in the two thresholds, we observe the following three cases when a packet is received with the SNR per symbol  $\gamma$ : 1) When  $\gamma_{acq} < \gamma_p \leq \gamma$ , the node can successfully retrieve the payload message and this is called full reception, 2) When  $\gamma_{acq} \leq \gamma < \gamma_p$ , the node can successfully decode the header information, though it cannot successfully decode the payload message. This is called partial reception, and 3) When  $\gamma < \gamma_{acq} < \gamma_p$ , the node can neither decode the header or payload message and is called failed reception. *The key idea of Hitch-hiking is to take advantage of not only full reception but also partial reception of signals.*

When the  $k$ th node transmits a packet, the amount of reception by the  $l$ th node is quantified by the coverage of the  $l$ th node defined as

$$C \left( \frac{\gamma_{k,l}}{\gamma_p} \right) \quad (3)$$

where  $C(\beta)$  is the coverage function given by

$$C(\beta) = \begin{cases} 1, & \text{for } \beta > 1 \\ \beta, & \text{for } \frac{\gamma_{acq}}{\gamma_p} < \beta \leq 1 \\ 0, & \text{for } 0 < \beta \leq \frac{\gamma_{acq}}{\gamma_p}. \end{cases} \quad (4)$$

The  $l$ th node is fully covered by the  $k$ th node, if the coverage is equal to 1. Figure 1 shows an example of the coverage function when the difference of the thresholds  $\gamma_p$  and  $\gamma_{acq}$ , in dB, is 9, i.e.,  $\gamma_{acq}/\gamma_p \approx 1/8$ .

A channel gain is often modeled as a function of the distance such as a power of the distance. That is, the channel gain  $\alpha_{k,l}$  is given by  $\alpha_{k,l} = \frac{c}{(d_{k,l})^\alpha}$  with the path loss exponent  $2 \leq \alpha \leq 6$  [15], where  $d_{k,l}$  is the distance between the  $k$ th node and the  $l$ th node, and  $c$  is a constant. When the  $l$ th node is just 100% covered by the  $k$ th node, the coverage of the  $l'$ th node can be easily obtained as

$$C \left( \left[ \frac{d_{k,l}}{d_{k,l'}} \right]^\alpha \right) \quad (5)$$

<sup>1</sup>We assume that the threshold to successful decoding of a header is less than or equal to the threshold to successful timing acquisition. So, the header is successfully decoded whenever the timing is successfully acquired and, consequently, the threshold to successful decoding of a header does not play any role.

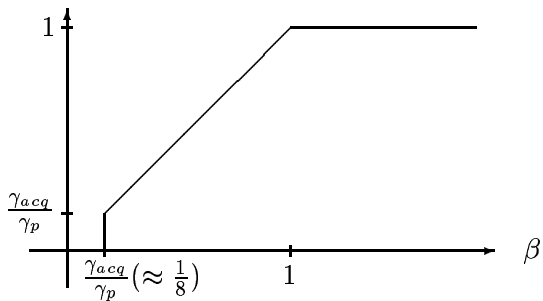


Fig. 1. The coverage function when  $\frac{\gamma_p}{\gamma_{acq}}$  is 9 (dB).

in terms of the distances, because  $\gamma_p = \frac{cE_k}{(d_{k,l})^\alpha N_0}$ . For example, when  $\gamma_{acq}/\gamma_p = 1/8$  and  $\alpha = 2$ , the coverage of the  $l$ 'th node is 0.25 if  $d_{k,l}/d_{k,l'} = 1/2$ , while it is 0 if  $d_{k,l}/d_{k,l'} = 1/3$ .

#### D. Combining Partial Receptions

Suppose that a node receives two packets containing the same information and the SNRs  $\gamma_1$  and  $\gamma_2$  of the packets satisfy  $\gamma_{acq} \leq \gamma_1 < \gamma_p$ ,  $\gamma_{acq} \leq \gamma_2 < \gamma_p$ , and  $\gamma_1 + \gamma_2 \geq \gamma_p$ . These two packets can be combined by a maximal ratio combiner [13] and can be successfully decoded as the resultant SNR  $\gamma_1 + \gamma_2$  at the output of the combiner is greater than or equal to the threshold  $\gamma_p$ .

The above idea of combining two partial receptions can be easily extended to combine multiple partial receptions, where the output SNR of the maximal ratio combiner is given by  $\gamma = \sum_{j=1}^J \gamma_j$  with  $J$  being the number of packets containing the same information, and with  $\gamma_j$  being the SNR of the  $j$ th packet, satisfying  $\gamma_j \geq \gamma_{acq}, \forall j$ .

This process of combining partially received packet to successfully decode complete packet is called *Hitch-hiking*.

In next section, we will develop a graph model and propose heuristic algorithms to construct energy efficient broadcast tree with Hitch-hiking.

## IV. ENERGY EFFICIENT BROADCAST WITH HITCH-HIKING

### A. Network Model

We assume a static ad hoc network with  $N$  nodes containing omnidirectional antennas. The nodes are assumed to be capable of receiving and combining partial signals to decode the message contained in the signal. Thus, nodes in the networks can take advantage of the Hitch-hiking model described in the last section. We represent a network by a directed graph  $G = (V, E)$  where  $V$  is the set of nodes corresponding to the wireless devices in the network and  $E$  is the set of edges corresponding to the com-

munication link between the devices. There exists a communication link between node  $i$  and node  $j$  if transmission from node  $i$  is received at node  $j$  with SNR greater than  $\gamma_{acq}$ . Associated with each node  $i \in V$  is a transmission power level of the node ( $p_i$ ) and with each edge  $(i, j) \in E$  is the coverage provided by the edge to the destination node.

We assume that  $\gamma_p = 1$  which implies that if SNR of a received signal is greater than or equal to 1, then the signal can be successfully decoded, which is a reasonable assumption. Thus, coverage of a node  $l$  by a transmission from a node  $k$  becomes  $C(\gamma_{k,l}) = C(\frac{cE_k}{d_{k,l}^\alpha N_0})$ . We further simplify the model by taking  $\frac{c}{N_0} = 1$  and thus making the coverage function as  $C(\frac{E_k}{d_{k,l}^\alpha})$  for the rest of the paper. In this paper, we have taken  $\alpha$  to be 2 and 4.  $\gamma_{acq}$  is taken to be 0 for the rest of the paper because  $\gamma_{acq}$  is practically so small that the partial coverage provided by signals having SNR below  $\gamma_{acq}$  does not contribute much in energy saving. Thus, we assume that the results presented in the paper are not affected much by the choice of  $\gamma_{acq} = 0$ . Nodes are assumed to be capable of having any power level between 0 and a maximum value determined by the hardware constraints of the node.

### B. Problem Definition

We assume a specified source node which has to broadcast a message to all other nodes in the network. Nodes that receive a message but do not retransmit it are called *leaf nodes*. Nodes, including the source node, that retransmit the message are called *forwarding nodes*.

*The objective function:* For the given source node  $S$ , we want to find a set of forwarding nodes and determine their power level such that the message sent by  $S$  is received by all the nodes in the network, and the total energy consumed for this task is minimized. In other words, for a given graph  $G$ , we want to determine  $p_i$  for all  $i \in V$  such that  $\sum_{i \in V} p_i$  is minimum and all the nodes are covered. This problem is called *Minimum-Energy Broadcast Problem* (MBP) [2].

For the same problem statement, we call the problem *Minimum-Energy Broadcast with Hitch-hiking* (MBH) when the nodes in the network are capable of receiving and combining partial signals as well.

### C. Hitch-hiking Advantage

In this section, we will demonstrate the advantage of Hitch-hiking to reduce the energy consumption in a broadcast through an example shown in Figure 2. In a broadcast, the same packet is transmitted many times by different nodes in order to cover the entire network. With Hitch-

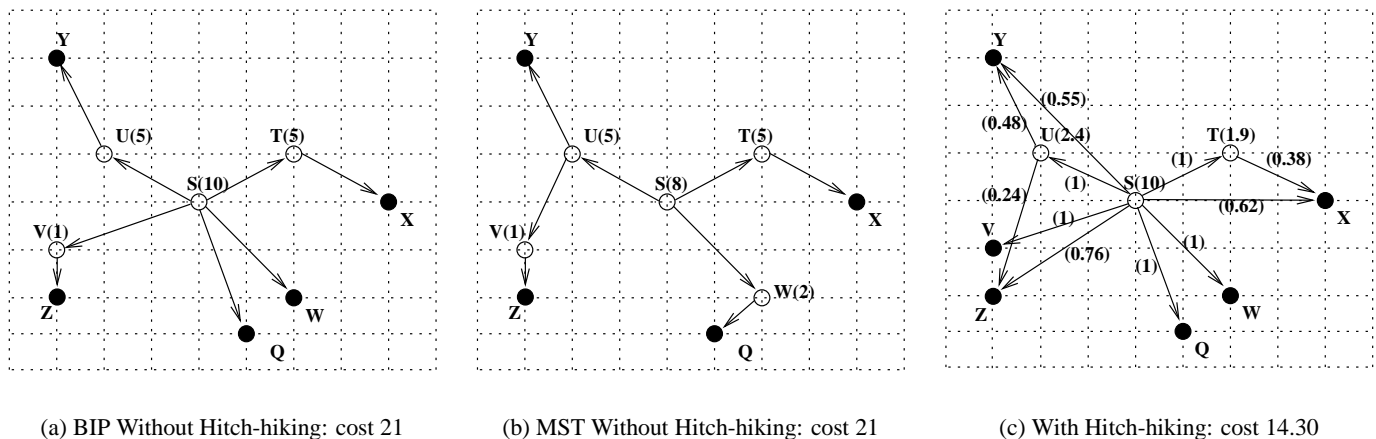


Fig. 2. Comparison of energy consumption for broadcast without Hitch-hiking and with Hitch-hiking. The number on each node indicates the power level used for broadcast. The number on each edge indicates the coverage provided by the edge to the destination node. (1) refers to full coverage while values less than 1 indicate the amount of partial coverage

hiking, we can take advantage of this multiple transmission of the same packet to effectively reduce the energy consumption. We assume that nodes can buffer the partial packets that they receive so that these can later be combined with other partial packets to decode the complete packet. In Figure 2, the power level used at each relay node is indicated at the node. Channel loss exponent  $\alpha$  is taken as 2 for this example. The black nodes are the non-forwarding nodes while white nodes are the relay nodes that forward packets to other nodes. Figure 2(a) and Figure 2(b) show the broadcast tree constructed with BIP and MST respectively without Hitch-hiking. Figure 2(c) shows the broadcast tree constructed using Hitch-hiking advantage. The energy consumption is reduced by Hitch-hiking due to the following two factors.

First, *Hitch-hiking can reduce the number of relay nodes needed.* For example in Figure 2(c), the broadcast from node  $S$  covers 76% of node  $Z$  and the broadcast from node  $U$  covers remaining 24% of node  $Z$ . Thus node  $Z$  is fully covered by combining the broadcasts from node  $S$  and node  $U$ . Node  $V$  no longer needs to relay to node  $Z$ . This reduces the number of relay nodes needed, in contrast to the broadcast without Hitch-hiking as shown in Figure 2(a). Second, *Hitch-hiking can reduce the power level of relay nodes.* For the example in Figure 2(c), since node  $S$  covers 55% of node  $Y$ , node  $U$  needs to cover only 45% of node  $Y$ . So node  $U$  can broadcast with reduced power level. Please note that due to Hitch-hiking, the cost of the broadcast tree was reduced from 21 to 14.30 in the given example.

In the next section, we will show that MBH is NP-complete.

#### D. Complexity of MBH

In [2], authors have given a formal proof of NP-completeness of general graph version and geometric version of the minimum energy broadcast problem (MBP). In order to prove that MBH problem is NP-complete, we will show that MBH belongs to NP and MBP is a special case of MBH.

*Theorem:* MBH problem is NP-complete

*Proof:* It is easy to see that MBH belongs to the NP class since it can be verified in polynomial time whether a given set of transmitting nodes cover all the nodes in the network and whether the cost of the final solution is less than a fixed value. Now we need to show that MBP is a special case of MBH.

Recall the thresholds  $\gamma_p$  and  $\gamma_{acq}$  described in section III for Hitch-hiking. When  $\gamma_p = \gamma_{acq}$ , we will have no case of partial reception of signals (section III). Thus the problem of MBH will be reduced to MBP where only full or failed signals exist. Hence, we can say that MBP is a special case of MBH for  $\gamma_p = \gamma_{acq}$ .

Since MBP is NP-complete and is a special case of the MBH problem, and because MBH belongs to NP class, we can therefore say that MBH problem is NP-complete.

As MBH is NP-complete, we propose a centralized heuristic and its distributed counterpart in the next subsections to construct energy efficient broadcast tree with Hitch-hiking. We assume static network so the tree has to be constructed only once. Thus the cost of constructing the tree is not considered.

#### E. Centralized Algorithm

First we give a brief overview of the proposed algorithm. The algorithm starts by constructing a Minimum

Spanning Tree (MST) for a given network using either Prim's or Kruskal's algorithm. Then, starting from the source node, at each step the algorithm picks a fully covered node say  $u$  whose power level has not been determined till then, and decides its power level. While deciding the power level of node  $u$ , only its child and grandchild nodes are considered, i.e the decision is based on local optimization. These nodes (including  $u$ ) form  $u$ 's local region. Node  $u$ 's final power level corresponds to the "maximum power reduction" in  $u$ 's local region. That is, the summation of  $u$ 's power level and  $u$ 's children power level should be minimized while still ensuring full coverage of  $u$ 's child and grandchild nodes. The reduction comes by the extra coverage (partial as well as full) provided by the increased power level of node  $u$  to its grandchild nodes in exchange for the decrease in power level of its child nodes. Finally, the coverage and power level of all uncovered nodes in the network are updated based on the coverage provided by the new power level of node  $u$ . Both centralized and distributed version of the algorithm is given. We call the proposed algorithm for improving upon the initial solution as WMH (Wireless Multicast with Hitch-hiking).

Before describing the algorithm in detail, we introduce some terminologies and attributes associated with each node in the network (Table I). An attribute called  $pc$  associated with each node stores the total coverage of the node at any instant of time. The value of  $pc$  for node  $i$  is represented as  $pc(i)$ . A node  $i$  is said to be *fully covered* if  $pc(i)$  becomes 100. In the beginning, only source node has its  $pc$  value as 100 and all other nodes have their  $pc$  value initialized to 0. Attribute called  $p_i$  keeps the transmission power level for node  $i$  and  $CH(i)$  is the set of child nodes of node  $i$  in the broadcast tree.

We start with link-based MST as the initial feasible solution. MST is chosen because of its good performance even as a final solution for MBP problem as shown in [2]. MST could be constructed using any well-known algorithm for constructing MST. So let us assume that we have the MST for a given network and that each node in the network has knowledge of its parent in the MST. Once the MST is constructed, each node is assigned a minimum power level such that it can reach all its neighboring nodes in the MST with that power level. To decrease the cost of the MST, we apply the WMH algorithm over it. As we go along with the explanation of WMH algorithm, we will also run it over the example given in Figure 2. The MST for the example is shown in Figure 2(b). Figure 3 gives the pseudocode of the WMH algorithm.

The WMH algorithm determines the final power level for each node by considering one node at each step. Once

TABLE I  
SOME ATTRIBUTES USED IN WMH

$pc(i)$	: Coverage of node $i$ in percent
$p_i$	: Transmission power level of node $i$
$CH(i)$	: Set of child nodes of node $i$
$F$	: Set of nodes whose final power level is decided
$PL(i)$	: Set of transmission power levels of node $i$
$g_i(p)$	: Gain of node $i$ at power level $p$
$p_j^{init}$	: Initial power level of node $j$
$p_j^{red}$	: Reduced power level of node $j$
$d_{j,k}$	: Distance between node $j$ and $k$
$N$	: Set of all nodes in the network

the final power level of a node is decided, it is not changed throughout the execution of the algorithm. A set  $F$  keeps all those nodes whose final power level has been decided by the algorithm. Starting with the source node, at each step the algorithm picks a node say  $i$  such that  $pc(i) = 100$ ,  $p_i > 0$  and  $i \notin F$  and determines its final power level. In other words, WMH picks the node that is fully covered, has transmission power greater than 0 and whose power level has not already been decided. If more than one node qualifies at the same time, the node with least node ID is picked by the algorithm. Since only the source node satisfies the condition in the beginning, WMH starts by deciding the power level of the source node.

In order to determine the final power level of a node say  $i$ , WMH calculates its gains for various power levels and assigns the power level to node  $i$  for which the gain is maximum. Since power level of node  $i$  can take arbitrary values over a range, WMH first constructs a discrete set of power levels for node  $i$  referred to as  $PL(i)$  and then determine the gain for each of the power level in  $PL(i)$ .  $PL(i)$  contains the current power level of node  $i$  and all those power levels at which node  $i$  covers all the child nodes of node  $j$  for at least one  $j \in CH(i)$ . In other words,  $PL(i)$  contains all those power levels at which node  $i$  can reduce the power level of any one of its child node to 0 by covering all the child nodes of that child node. In the example given in Figure 2, node  $S$  can reach nodes  $V$  and  $Q$  through power level 10, node  $X$  through power level 16 and node  $Y$  through power level 18. Thus,  $PL(S)$  is a set  $\{8, 10, 16, 18\}$ . It is easy to see that  $PL(i)$  can be populated in polynomial time by taking each child node  $j$  of node  $i$  one at a time and considering all the child nodes of node  $j$  to find a power level which covers them all. In the next step, WMH finds the gain for each power level in  $PL(i)$ .

The gain  $g_i(p)$  of node  $i$  is defined as the decrease in the total energy of the broadcast tree obtained by reduc-

ing the power level of some of the transmitting nodes in the MST, in exchange for the increase in node  $i$ 's transmission power level to  $p$ . In other words, when the power level of node  $i$  increases, it provides partial and full coverage to more nodes in the networks. Due to the increase in the partial or full coverage of the node say  $k$ , the parent node of node  $k$  can reduce its power level such that it provides less or no coverage to node  $k$ . The reduction in power level of some of the nodes reduces the overall energy consumption of the tree. To simplify the algorithm, the reduction in the power level of only the child nodes of node  $i$  is considered in the calculation of gain of node  $i$ . Gain  $g_i(p)$  is given by the following equation:

$$g_i(p) = \sum_{\forall j \in CH(i)} (p_j^{init} - p_j^{red}) - \delta \quad (6)$$

where  $p_j^{init}$  is the initial power level of node  $j$ ,  $p_j^{red}$  is the reduced power level of node  $j$  due to the increase in the power level of node  $i$  and  $\delta$  is the increase in power level of node  $i$ . In order to calculate  $p_j^{red}$ , WMH first calculates the coverage provided by node  $i$  at power level  $p$  to the child nodes of node  $j$ . If  $k$  is any child node of node  $j$ , the increased partial coverage of node  $k$  due to power level  $p$  of node  $i$ ,  $pc(k, p)$  is calculated by the following equation:

$$pc(k, p) = \min\left\{\left(\frac{p}{d_{i,k}^\alpha} \times 100 + pc(k)\right), 100\right\} \quad (7)$$

Please note that  $pc(k, p)$  is just a temporary variable required to calculate gain achieved by node  $i$  at power level  $p$  and the value of  $pc(k)$  is not changed till node  $i$  has decided its final power level.

In the example in Figure 2(b), while calculating the gain for node  $S$  at  $p = 10$ ,  $pc(Y, 10) = \left(\frac{10}{18} \times 100\right) = 55$ . This means that if node  $S$  transmits at power level 10, it will provide 55% coverage to node  $Y$ .

Based on the  $pc$  value of the child node  $k$  of node  $j$ , the reduced power level required by node  $j$  to fully cover node  $k$  ( $p_j(k)$ ) is calculated by the following equation:

$$p_j(k) = \left(1 - \frac{pc(k, p)}{100}\right) \times d_{j,k}^\alpha \quad (8)$$

where  $\left(1 - \frac{pc(k, p)}{100}\right)$  gives the remaining coverage required by node  $k$  to get fully covered. Finally,  $p_j^{red}$  is given by:

$$p_j^{red} = \max_{\forall k \in CH(j)} p_j(k) \quad (9)$$

In the example,  $p_U(Y) = (1 - 0.55) \times 5 = 2.25$ . Thus,  $p_U^{red} = 2.25$ . Note that, its only when node  $U$  calculates its gain at different power levels, it finds that maximum

gain can be achieved by increasing its power level to 2.4 where it can provide full coverage to node  $Z$  and thus increases its power level to 2.4 as shown in the figure.

$p_j^{red} \leq 0$  implies that all the child nodes of node  $j$  are already covered so  $p_j^{red}$  is made equal to 0. Once WMH has calculated  $p_j^{red}$  for all  $j \in CH(i)$ , it can calculate  $g_i(p)$  by putting the values in Equation (6).

In the example, node  $S$  at power level 10 provides 72% coverage to node  $Z$  thus reducing the required power level of node  $V$  from 1 to 0.24. Thus, gain of node  $S$  at power level 10 is :

$$g_S(10) = \sum_{x \in U, T, W, V} (p_x^{init} - p_x^{red}) - \delta = 2.75 + 3.1 + 2 + 0.76 - 2 = 6.61.$$

Please note that, it is only after node  $U$  decides its final power level, node  $Z$  gets full coverage by combining signals from node  $S$  and node  $U$  and node  $V$  reduces its power level to 0 as shown in Figure 2(b).

This completes the calculation of gain of node  $i$  at power level  $p$ . The process of finding gain is applied for each power level  $p \in PL(i)$ . Once the gain for all the power levels in  $PL(i)$  is determined, final power level of node  $i$  is chosen to be the one for which the gain is maximum. If for no power level  $p$ ,  $g_i(p) > 0$ ,  $p_i$  does not change. Once the final power level of node  $i$  is decided, WMH performs following steps in the given order:

- For all  $j \notin F$  and having  $pc(j) < 100$ , calculate  $pc(j, p_i)$  using Equation (7) and assign it to  $pc(j)$ . In the example, when node  $S$  has decided its final power level as 10,  $pc(Y, 10)$  is 55 so  $pc(Y)$  changes from 0 to 55. Similarly for nodes  $Z$  and  $X$ .
- Reduce the transmission power of all nodes  $j$  having  $p_j > 0$  and  $j \notin F$  based on the new  $pc$  value of their child nodes in the MST. The reduced transmission power is obtained using Equation (8) such that the nodes continue to provide full coverage to the child nodes even with the reduced power level. In the example, the power level of node  $T$  is reduced from 5 to 1.9 once node  $S$  is assigned power level 10. Similarly for nodes  $U$  and  $W$ .

The WMH algorithm continues till all the nodes in the network are fully covered. The algorithm is guaranteed to end because while deciding the power level of a node say  $i$ , it is ensured that all its child nodes are fully covered by the final power level of node  $i$ . Since in the initial MST, every node except the root node in the network had a parent node, thus every node is guaranteed to be covered after finite number of steps.

Next we show that the complexity of the WMH algorithm is polynomial in the total number of nodes  $n$ . To calculate  $g_i(p)$ , it takes  $O(n)$  time as it has to find the final power level for each of the child node of node  $i$ .

**Algorithm WMH**

```

1.  while (true)
2.      /* Stop if all the nodes are fully covered */
3.      if  $\forall i \in N, pc(i) \geq 100$ 
4.          break
5.      Find  $i \in N \mid (pc(i) \geq 100 \ \& \ p_i > 0 \ \& \ i \notin F)$ 
6.
7.      /* Find local optimal power level for node  $i$  */
8.      Find discrete power level set  $PL(i)$  for node  $i$ 
9.       $Gain(i)$ 
10.      $p_i \leftarrow$  power level of node  $i$  for which gain is maximum
11.      $F \leftarrow i$ 
12.
13.     /* Increase coverage value of other nodes based
14.     on the final power level of node  $i$  */
15.     for all  $j \in N$ 
16.         if ( $j \notin F \ \& \ pc(j) < 100$ )
17.              $pc(j) = \min\{pc(j) + (\frac{p_i}{d_{i,k}^\alpha} \times 100), 100\}$ 
18.
19.         /* Reduce power level of all the nodes due to
20.         change in the coverage value of the child nodes */
21.         for all  $j \in N$ 
22.             if ( $j \notin F \ \& \ p_j > 0$ )
23.                  $Reduce(i, p_i, j)$ 
24.
25.     Gain( $i$ )
26.     /* Find gain for all the power levels in  $PL(i)$  */
27.     for all  $p \in PL(i)$ 
28.         for all  $j \in CH(i)$ 
29.              $Reduce(i, p, j)$ 
30.              $g_i(p) = \sum_{j \in CH(i)} (p_j^{init} - p_j^{red}) - \delta$ 
31.
32.     Reduce( $i, p, j$ )
33.     /* Reduce the power level of node  $j$  on the basis of partial
34.     coverage provided by node  $i$  at power level  $p$  */
35.     for all  $k \in CH(j)$ 
36.          $pc(k, p) = \min\{pc(k) + (\frac{p}{d_{i,k}^\alpha} \times 100), 100\}$ 
37.          $p_j(k) = (1 - \frac{pc(k, p)}{100}) \times d_{j,k}^\alpha$ 
38.          $p_j^{red} = \max_{\forall k \in CH(j)} p_j(k)$ 

```

Fig. 3. Algorithm for WMH. Refer to Table I for meaning of notations.

Thus, in order to calculate gain for all  $p \in PL(i)$ , it takes  $O(n^2)$  time. Finally, it repeats all the above steps till all the nodes in the network are fully covered which will thus take  $O(n^3)$  time. So, the complexity of the algorithm is  $O(n^3)$ .

### F. Distributed WMH

In this section, we will extend the WMH algorithm to run in distributed fashion. We assume that the construc-

tion of the minimum-weight spanning tree can be done in a distributed manner by running the distributed algorithm proposed by Gallager et al[8]. Once the MST is constructed, each node knows the information about the cost of its two hop neighbors in the MST.

We assume that each node can determine its coverage by a transmission from a node say  $i$ , based on the SNR of a received signal from  $i$  using Equation (3). A node  $i$  can also determine when it can decide its final power level based on  $pc(i)$  and the conditions mentioned in Section IV-E. We call the node deciding its final power level as *Deciding node*. The main problem for a deciding node  $i$  in the distributed setting is to know the  $pc(j)$  and  $p_j$  where node  $j$  is within two hops neighbors of node  $i$ .

To solve this problem, each node maintains a table called PCTable containing the  $pc(j)$  and  $p_j$ , for all node  $j$  that are its two hop neighbors. Whenever  $pc(j)$  or  $p_j$  for a node  $j$  is changed, node  $j$  informs the change to its one hop neighbors. This information can be piggybacked on HELLO packets. The neighboring nodes further propagate the change to their neighbors so that each node within two hops of node  $j$  gets the information and can update the entry corresponding to node  $j$  in their PCTable. Also, on getting an update, all nodes in the MST whose power level is greater than 0 and who has not yet done decided its final power level, if possible reduces its power level based on new value of  $pc(j)$  using Equation (8).

Now each deciding node has sufficient information to calculate gain. When a node  $i$  becomes a deciding node, it starts a timer for fixed interval during which it waits for an update from its neighboring nodes. The timer value is equal to the HELLO packet interval. This is to ensure that the gain calculated by node  $i$  is based on the latest state of its neighbors. When the timer expires, the node starts deciding its final power level based on the entries in its PCTable. Decision steps are the same as in Section IV-E. After the node has decided its power level, the node backs off for certain period of time inversely proportional to its calculated gain. This allows the nodes with higher gain to broadcast first. If the node receives an update during this interval, it re-calculates its power level and repeats again. If the timer expires without any update, the node transmits the packet with the final power level.

The construction of the broadcast tree is started by the source node and as new nodes become fully covered, they decide on the their power level and make broadcast to cover more nodes. This is continued till all the nodes in the network are covered.



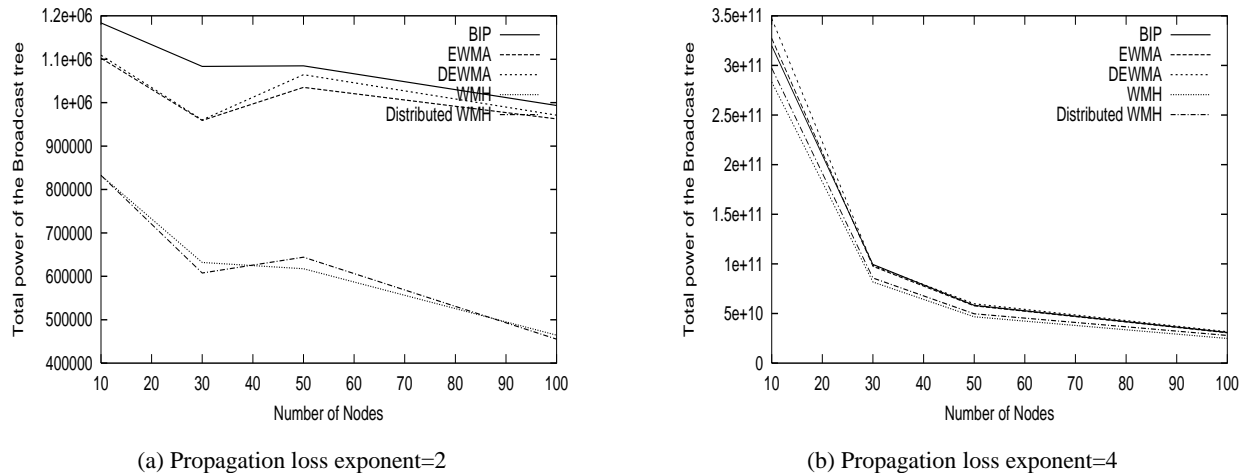


Fig. 4. Performance comparison of WMH with other protocols for propagation loss exponent 2 and 4

### G. Performance Evaluation

We performed a simulation study to evaluate the performance of our centralized algorithm (WMH) and its distributed version.

We compared our proposed algorithms with EWMA, BIP and MST algorithms for the same network settings. We performed the simulations for four different network sizes: 10, 30, 50 and 100 nodes similar to [2]. To fully cover distance  $d$ , the transmission power is taken to be  $d^\alpha$  with propagation loss constant  $\alpha$  taken as 2 and 4.

The nodes in the networks are distributed according to a spatial poisson distribution over the same deployment region. Thus the network density increases with increasing number of nodes. We ran 100 simulations for each simulation setup. The performance metric is the average of total power of the trees for a  $k$ -node network.

Figure 4(a) shows the performance of the proposed algorithm compared to BIP, EWMA and DEWMA for propagation loss constant 2. From the figure, we can see that power of the broadcast tree constructed using WMH and Distributed WMH is almost 50% less than that of BIP, EWMA and DEWMA. We can also see that the advantage of Hitch-hiking increases with the increase in the number of nodes. The distributed version of the algorithm performs almost as well as the centralized version.

Figure 4(b) shows that with the increase in propagation loss exponent  $\alpha$  from 2 to 4, the advantage of WMH and Distributed WMH over other protocols decreases. This is expected because with the increase in  $\alpha$ , energy contained in partial signals decreases faster with distance thus providing less advantage of Hitch-hiking.

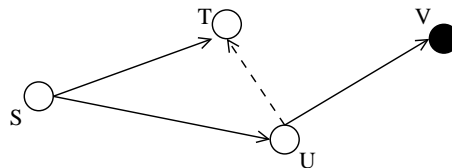


Fig. 5. An example showing multiple receive of same packet in a broadcast

### V. BROADCAST WITH POWER SAVING

Till now, we have considered only the transmission cost of the broadcast tree. An important observation about the broadcast application is that spatial overlapping of transmission zones may cause same message to be received multiple times by the nodes within the overlapping zones. For example in the broadcast tree shown in Figure 5, node  $T$  received the broadcast message from node  $S$  but when node  $U$  makes the broadcast of the same message to cover its child nodes, node  $T$  received the same message again. This wastes energy if the cost of receiving a packet is significant. Feeney et al. [7] have shown that the cost of receiving a packet is nearly one fifth the cost of sending the packet for the Lucent IEEE 802.11 WaveLAN PC Card which shows that the cost of receiving a packet is indeed significant. In this section, we propose and analyze a protocol called PSBT (Power Saving with Broadcast Tree) that saves energy by allowing the nodes to discard those packets that have already been received by the node. In [16], the authors gave an idea on how to extend PAMAS to conserve energy in the case of broadcast but have not given any analysis of the protocol for broadcast.

### A. The PSBT Protocol

PSBT assumes that the broadcast tree is already constructed. Thus, nodes are assumed to have the knowledge of their parent in the broadcast tree. The broadcast tree can be constructed by any algorithm and the choice of algorithm will not affect PSBT.

In PSBT, any node which has to make a transmission of broadcast message, first sends a small header packet containing the  $\langle sourceID \rangle$  associated with the broadcast message. The header packet also contains the duration of the transmission based on the available bandwidth and the size of the packet. As the energy consumed in receiving a packet is directly proportional to the size of the packet [7], and the size of the header packet is assumed to be very small as compared to the size of the data packet, we can ignore the energy consumed in sending and receiving the header packet. We assume that the cost of receiving a partial signal whose SNR is greater than  $\gamma_{acq}$  (section III) is the same as the cost of receiving a complete signal because the timing of partial signals can be acquired correctly. To prevent nodes from receiving unwanted partial signals, the header packets are transmitted with maximum power irrespective of the transmission power level of the transmitting node in the broadcast tree.

Every node in the network has the knowledge of its parent node in the broadcast tree corresponding to a source. When a node receives a header packet, it continues to listen to the ongoing transmission if the transmitting node is the parent node in the broadcast tree corresponding to the source of the broadcast message. Otherwise, the node discards the packet by going to sleep mode. Thus, each node receives a broadcast packet only from its parent in the broadcast tree.

For example shown in Figure 5, node  $T$  receives the message from node  $S$  as node  $S$  is the parent node of node  $T$  in the broadcast tree. When node  $T$  receives the header packet for the same message from node  $U$ , it goes to sleep mode for the duration mentioned in the header packet. Thus, node  $T$  saves the energy that would have been consumed in receiving the message from node  $U$ .

1) *Performance Evaluation of PSBT:* In this subsection, we evaluate the performance of PSBT through simulation. For the simulation purpose, the receive cost is taken to be  $\frac{d^2}{5}$  where  $d$  is the maximum possible range that can be covered by a transmission. This choice of receive cost is taken on the basis of the statistics provided by Feeney et al. in [7]. The rest of the simulation environment is the same as in section IV-G. Figure 6(a) compares the cost of a broadcast tree constructed using EWMA with and without PSBT running over the nodes in the network. The figure shows that PSBT is very effective in conserv-

ing energy and the advantage increases with an increase in the number of nodes. For less nodes, the energy saving obtained is around 10% with PSBT. For a large number of nodes, the energy saving is more than 50% with PSBT. The energy consumption without PSBT increases with the increase in the number of nodes in the network because as the number of nodes increases over the same deployment region, node density increases resulting in increased redundant messages being received by the nodes. With PSBT, since each message is received only once irrespective of the node density, the overall cost of the broadcast tree does not increase with the increase in the number of nodes.

Figure 6(b) compares the cost of the broadcast tree constructed using EWMA and WMH protocol with PSBT running on all the nodes in the network. It can be seen that the cost of the broadcast tree constructed using WMH is much higher than the cost of the tree constructed using EWMA. This is mainly because in EWMA, the cost of receiving a broadcast message is added only once for each node while in WMH, many partial packets are required to be received to decode one message adding many receive cost for each message. The number of times the cost of receiving a packet is added to the cost of the broadcast tree depends on the number of partial packets required by a node to decode the message completely.

Please note that the advantage of WMH as shown in section IV-G is still there if PSBT is not used. This is mainly because without PSBT, all the protocols will receive all the transmissions (partial as well as full), thus incurring equal receive cost overhead. So in section IV-G, only transmission cost was taken into consideration while calculating the cost of the tree. Thus, WMH was designed by taking only the transmission cost into consideration. When PSBT is used, we need to modify WMH so that it can take advantage of Hitch-hiking with minimum receive cost overhead. In the next section, we propose a modified version of WMH called WMHP (WMH with PSBT) which can take advantage of Hitch-hiking even in the presence of PSBT.

### B. Hitch-hiking with PSBT

Before going into the details of WMHP, we will look at the disadvantage of WMH in the presence of PSBT from a different perspective. It can be said that many of the partial signals received by a node, say  $n$ , had more cost in receiving them than the amount of energy saved by them, thus increasing the overall cost of the tree. As we have shown in section IV, a partial signal received by a node  $n$  can save energy by reducing the power of the parent node of node  $n$ . Thus, if the decrease in the power of the parent

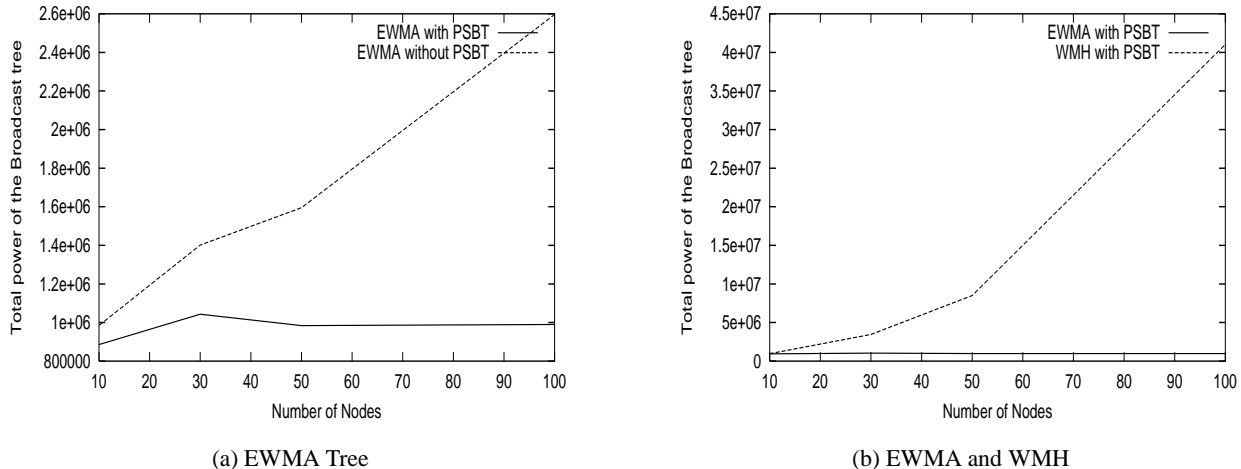


Fig. 6. Comparing energy consumption for broadcasts with and without PSBT for propagation loss exponent 2

node of node  $n$  due to a partial signal received by  $n$  is less than the cost of receiving the partial signal, then that partial signal will increase the cost of the tree and should be discarded by  $n$ . WMHP protocol takes this fact into consideration for constructing the broadcast tree.

WMHP is a distributed protocol which works very similar to distributed WMH except that a node does not accept all the partial signals it can receive to increase its coverage value. On receiving a partial signal  $s$  from a transmission, node  $i$  first calculates the function  $A_{i,s}$  that measures the advantage of receiving the partial signal given by the following equation:

$$A_{i,s} = \left(1 - \frac{C(s)}{100}\right) \times (p_j) - R \quad (10)$$

where  $j$  is the parent of node  $i$  in MST,  $C(s)$  is the coverage provided by signal  $s$  to node  $i$  and  $R$  is the fixed cost of receiving a signal.

If  $A_{i,s} > 0$ , node  $i$  will accept the partial signal  $s$  to increase its coverage. If  $A_{i,s} \leq 0$ , it will discard the partial signal  $s$ .

To minimize the overhead of receive cost, WMHP tries to combine minimum number of partial signals. Thus, each node maintains a table called CPTable that stores the coverage provided by the signals for which  $A_{i,s} > 0$  and the transmitting node of the signal. The entries in the table are sorted in the decreasing order of coverage. When the parent node of node  $i$  in MST has decided its final power level, node  $i$  chooses the first  $x$  signals from its CPTable such that

$$\sum_{\forall s \in x} C(s) \geq 100 \quad (11)$$

In other words, the node will take first  $x$  entries that can together provide full coverage to node  $i$ . As the signals are

sorted in the decreasing order of coverage, it is ensured that minimum number of partial signals are combined by a node to get full coverage.

The rest of the algorithm of WMHP is the same as that of distributed WMH. It is important to understand that the above steps are part of tree construction phase only and need to be executed only once. Once the broadcast tree has been constructed, a node using PSBT can decide on which signals to receive or discard based on the knowledge of the tree.

1) *Performance Evaluation of WMHP*: Figure 7 compares the performance of WMHP against BIP, EWMA and DEWMA with PSBT, for the same network setting as in section IV-E. The cost of receiving a packet is taken to be  $\frac{d^2}{5}$  where  $d$  is the maximum possible range that can be covered by a transmission. It can be seen that WMHP saves nearly 10% of the energy for less number of nodes in the network (lower node density). As the number of nodes increases, the advantage of WMHP over other protocols decreases. For number of nodes greater than 50, the performance of WMHP merges with that of EWMA for the given network setting. The reason behind the decrease in performance of WMHP with the increase in number of nodes is that as the node density increases, the average distance between parent and child nodes in MST decreases. Thus the average power required by transmitting nodes in MST to cover the entire network also decreases. In Equation (10), we saw that  $A_{i,s}$  is directly proportional to  $p_j$ . As  $R$  is fixed, the probability of finding  $A_{i,s} > 0$  decreases with increase in node density. Hence Hitch-hiking advantage decreases with increase in node density.

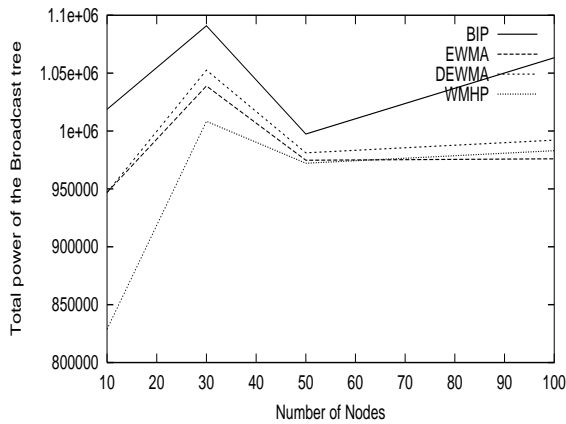


Fig. 7. Performance comparison of WMHP with other protocols for propagation loss exponent 2

## VI. CONCLUSION

We have proposed a novel concept called *Hitch-hiking* to reduce the overall cost of broadcast in an ad hoc network. We proposed a centralized algorithm and its distributed counterpart that can take advantage of the Hitch-hiking concept for energy efficient broadcasts. We showed through simulation that Hitch-hiking can reduce the cost of broadcast by as much as 50%.

We also analyzed the effect of cost of receiving a packet on the overall cost of broadcast in ad hoc networks. Then we proposed a protocol called PSBT that can save energy of broadcasts by preventing the nodes from receiving the same packet again. We also evaluated the amount of saving obtained by using PSBT through simulation.

Finally, we proposed and evaluated an algorithm to take advantage of Hitch-hiking with PSBT. The simulation result shows that with PSBT, the advantage of Hitch-hiking is limited to low density networks.

In the future, we want to extend the use of Hitch-hiking for reliable energy efficient broadcast. We would also like to investigate the use of Hitch-hiking in reducing energy consumption in the propagation of unicast traffic.

## ACKNOWLEDGMENT

This work was supported in part by NSF CAREER Award grant ANI-0196039, NSF grants CCR-0329794, CCR-0329741, ANI-0073736, EIA-0130806, and Alfred Sloan Fellowship. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## REFERENCES

- [1] S. Banerjee, A. Misra, J. Yeo, and A. Agarwala. Energy-efficient broadcast and multicast trees for reliable wireless communication. In *IEEE Wireless Communications and Networking Conference (WCNC), New Orleans, Louisiana*, March 2003.
- [2] M. Cagalj, J.-P. Hubaux, and C. Enz. Minimum-energy broadcast in all wireless networks: NP-completeness and distribution issues. In *The eighth ACM International Conference on Mobile Computing and Networking (MobiCom 2002)*, 2002.
- [3] J. Cartigny, D. Simplot, and I. Stojmenovic. Localized minimum-energy broadcasting in ad-hoc networks. *Proc. of IEEE INFOCOM 2003*, 2003.
- [4] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. SPAN: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *ACM Wireless Networks Journal*. 8, (5), 2002, 481-494.
- [5] A. Clementi, P. Penna, and R. Silvestri. The power range assignment problem in radio networks on the plane. *Proc. of the 17th Annual Symposium on Theoretical Aspects of Computer Science (STACS 2000)*, pages 651–660, 2000.
- [6] D. Estrin, R. Govindan, J. Heidmann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *MOBICOM*, 1999.
- [7] L. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in ad hoc networking environment. In *Proc. IEEE INFOCOM*, 2001.
- [8] R. Gallager, P. Humblet, and P. Spira. A distributed algorithm for minimum-weight spanning trees. In *ACM Transactions on Programming Languages and Systems*, 1983.
- [9] C. E. Jones, K.M. Sivalingam, P. Agrawal, and C. Chen. A survey of energy efficient network protocols in wireless networks. In *Wireless Networks*, 2001.
- [10] L. M. Kiousis, E. Kranakis, D. Krizanc, and A. Pelc. Power consumption in packet radio networks. *Proc. of the 14th Annual Symposium on Theoretical Aspects of Computer Science (STACS 1997)*, pages 363–374, 1997.
- [11] W. Liang. Constructing minimum-energy broadcast trees in wireless ad hoc networks. *Proc. of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002)*, pages 112–122, 2001.
- [12] E. L. Lloyd, R. Liu, M. V. Marathe, R. Ramanathan, and S. S. Ravi. Algorithmic aspects of topology control problems for ad hoc networks. *Proc. of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002)*, pages 123–134, 2001.
- [13] J. G. Proakis. *Digital Communications, 4th ed.* McGraw Hill, 2001.
- [14] R. Ramanathan and R. Hain. Topology control of multihop wireless networks using transmit power adjustment. *Proc. of IEEE INFOCOM'00*. 2, 2000, 404-413.
- [15] T. S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, 1996.
- [16] S. Singh, C. Raghavendra, and J. Stepanek. Power-aware broadcasting in mobile ad hoc networks. In *IEEE PIMRC*, September 1999.
- [17] P.-J. Wan, G. Calinescu, and X.-Y. Li. Minimum energy broadcast routing in static ad-hoc wireless networks. *Proc. of IEEE INFOCOM 2001*, pages 1162–1171, 2001.
- [18] X. Wang and I.F. Akilydiz. A survey on sensor networks. In *IEEE Communication Magazine*, 2002.
- [19] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. On constructing minimum spanning trees in  $k$ -dimensional spaces and related problems. *Proc. of IEEE INFOCOM'00*. 2000, 585-594.
- [20] W. Ye, J. Heidemann, and D. Estrin. An energy-efficient mac protocol for wireless sensor networks. In *Proc. of IEEE INFOCOM*, 2002.