

# Delay Constrained Minimum Energy Broadcast in Cooperative Wireless Networks

Marjan Baghaie, and Bhaskar Krishnamachari

Department of Electrical Engineering, Viterbi School of Engineering  
University of Southern California, Los Angeles, California 90089

Email: {baghaiea, bkrishna}@usc.edu

**Abstract**—We formulate the problem of delay constrained energy-efficient broadcast in cooperative multihop wireless networks. We show that this important problem is not only NP-complete, but also  $o(\log(n))$  inapproximable. We derive approximation results and an analytical lower-bound for this problem. We break this NP hard problem into three parts: ordering, scheduling and power control. We show that when the ordering is given, the joint scheduling and power-control problem can be solved in polynomial time by a novel algorithm that combines dynamic programming and linear programming to yield the minimum energy broadcast for a given delay constraint. We further show empirically that this algorithm used in conjunction with an ordering derived heuristically using the Dijkstra’s shortest path algorithm yields near-optimal performance in typical settings. We use our algorithm to study numerically the trade-off between delay and power-efficiency in cooperative broadcast and compare the performance of our cooperative algorithm with a smart non-cooperative algorithm.

## I. INTRODUCTION

In a wireless network, a transmitted packet intended for one node is received not only by that node but also by other nodes in the nominal reception range of the transmitter. In a traditional point-to-point system, where there is only one intended recipient, this innate property of the wireless medium can be a hurdle, as the remaining neighboring nodes must treat that received packet as undesired interference. Nonetheless, broadcast and multicast systems can be designed to work cooperatively and thereby achieving potential performance gains by taking advantage of this property. As such, cooperative transmission in wireless networks has attracted a lot of interest not only from the research community in recent years [1], [2], [3], [4], [5], [6], [8] but also from industry in the form of the first practical cooperative mobile ad-hoc network system [10].

We focus on the problem of cooperative broadcast in this work, where a single node is sending a packet to the entire network. We consider a time-slotted system, in which the nodes that have received and decoded the packet are allowed to transmit it in future slots. Nodes are allowed to choose their transmit power dynamically. We assume that receiving nodes cooperatively combine the signal power received from multiple sources in an additive fashion. As discussed in prior work [2], [4], [3], [5], cooperative communication with such

energy accumulation can be implemented at receivers using maximal ratio combining (MRC) of orthogonal signals from source nodes that use orthogonal time/frequency channels, or spreading codes, or distributed space-time codes. We consider a memoryless system in which the MRC at the receiver is restricted to source transmissions from the present time slot.

A key tradeoff in this problem is between the total energy consumption<sup>1</sup> and the total delay measured in terms of the number of slots needed for all nodes in the network to receive the broadcast message. At one extreme, if we wish to minimize delay, each transmitting node should transmit at the highest power possible so that the maximum number of receivers can decode the message at each step (indeed, if there is no power constraint, then the source node could transmit at a sufficiently high power to reach all nodes in the first slot itself). On the other hand, reducing transmit power levels to save energy, may result in fewer nodes decoding the signal at each step, and therefore in a longer time to complete the broadcast. We therefore formulate the problem of performing this broadcast in such a way that the total transmission energy over all nodes is minimized, while meeting a desired delay constraint on the maximum number of slots that may be used to complete the broadcast. The design variable in this problem is to decide which nodes should transmit, when, and with what power.

The key contributions of our work are as follows:

- We formulate the problem of delay-constrained minimum energy broadcast in cooperative networks, going beyond the prior work in the literature on cooperative broadcast which has focused either on minimizing energy without delay constraints [2], [5], or on delay analysis without energy minimization [7]. Our extended problem formulation allows us to expose and investigate the energy-delay tradeoffs inherent in cooperative networking.
- We not only prove that the delay constrained minimum energy cooperative broadcast (DMECB) problem is NP-complete, but also that it is in fact  $o(\log(n))$ -inapproximable (i.e., unless  $P = NP$ , it is not possible to develop a polynomial time algorithm for this problem that can obtain a solution that is strictly better than a

This project was supported in part by NSF under award 0627028.

<sup>1</sup>As we consider unit time slot durations, we use the words energy and power interchangeably in this paper.

logarithmic-factor of the optimal). We are not aware of prior work on cooperative broadcast that shows such an inapproximability result.

- However, we are able to show that for any given ordering of the transmissions (which dictates that a node later in the ordering may not transmit before the nodes earlier in the ordering have decoded successfully), then the problem of joint scheduling and power allocation for DMECB can in fact be solved optimally in polynomial time using a combination of Dynamic Programming for the scheduling and Linear Programming for the power allocation.
- For small network instances, we compute the optimal solution through exhaustive search, and show empirically through simulations that our proposed joint scheduling and power control method works near-optimally when used in conjunction with an ordering provided by the Dijkstra tree construction.
- We also show through simulations the delay-energy tradeoffs and minimum energy performance for larger networks and demonstrate the significant improvements that can be achieved by our solution compared to non-cooperative broadcast.
- In addition to the above, we also present an analytical lower bound for the minimum energy required for a given delay constraint, and a polynomial-time algorithm for the general problem that is provably guaranteed to offer a  $O(n^\epsilon)$  approximation. This algorithm is based on the current best-known algorithm for the bounded diameter directed Steiner tree problem [15].

In summary, this paper makes several contributions that significantly enhance our understanding of complexity and algorithm design for cooperative broadcast in wireless networks, in the context of energy-delay tradeoffs.

The rest of the paper is organized as follows: section II places our work in the context of prior related work. Section III describes the system model. In section IV, we prove the inapproximability result by reduction from the set cover problem. A polynomial time algorithm for optimum delay constrained scheduling and power allocation is presented in section V, for the case when the ordering is given. We present an analytical lower bound for the optimum solution in section VI, and approximation results are discussed in section VII. Simulation results are presented in section VIII, where we suggest and evaluate several heuristics for the ordering. Section IX concludes the paper and suggests directions for future work.

## II. RELATED WORK

There are three main approaches for cooperative communications in networks. One is using *coherent signal synchronization*, in which a set of transmitters synchronize their transmissions at the signal level when transmitting to a single receiver [1]. While the benefits are potentially higher with this approach for many-to-one communications it can be difficult to implement such tight synchronization in practical systems, and the benefits in a broadcast environment where the goal is to reach multiple nodes are not clear when compared to

the following two approaches that allow for many-to-many cooperation. The second approach is *energy accumulation* [2], [4], [3], [5], in which a receiver can recover the original packet so long as the total received energy from multiple sources or successive transmissions exceeds a given threshold. Such an approach can be implemented using maximal ratio combining of orthogonal signals from multiple sources, e.g., through a Rake receiver in CDMA or distributed space-time codes. It has been shown that one can achieve significant saving in energy and/or transmission time when using an energy accumulation protocol, compared to traditional protocols [2], [5], [7]. If energy accumulation is achieved by transmitting the exact same packet from different relays or through successive re-transmissions, the scheme is shown to achieve capacity in an asymptotically wideband regime [2]. The third approach is *mutual information accumulation*, which can be achieved using rateless codes [8]. The two schemes have been shown to be equivalent at low signal-to-noise ratios (SNRs) [8]. We consider the energy accumulation approach in our work, but we believe the general algorithmic approach may be applicable to information accumulation protocols as well, though this remains to be seen in future work. We note that recently a commercially developed cooperative mobile ad hoc network system has been developed which utilizes a pragmatic cooperation method requiring minimal information exchange, based on a combination of phase dithering and turbo codes [9], [10]. It is shown in [9] that the performance of this pragmatic scheme is close to that of an ideal energy-accumulation approach based on space-time coding.

Many network protocols in mobile ad hoc and sensor networks need to operate in broadcast mode to disseminate certain control messages to the entire network (for instance, to initiate route requests, or to propagate a query). The subject of broadcast transmission in multi-hop wireless networks has attracted a lot of attention from the research community in both non-cooperative [11], [12], [14] and cooperative settings [2], [3], [5], [7], [6]. For traditional non-cooperative wireless networks, the work by Cagalj *et al.* shows that the problem of minimum energy broadcast is NP-hard. In [3], Mergen *et al.* show through a continuum analysis the existence of a phase transition in the behavior of cooperative broadcast: if the decoding threshold is below a critical value then the broadcast is successful, else only a fraction of the network is reached. In [5], Mergen and Scaglione, show that the problem of scheduling and power control for minimum energy broadcast is tractably solvable for highly dense (continuum) networks and show the gains obtained with respect to noncooperative broadcast. In [7], we examined the delay performance of cooperative broadcast and show that cooperation can result in extremely fast message propagation, scaling logarithmically with respect to the network diameter, unlike the linear scaling for non-cooperative broadcast.

The work by Maric and Yates [2] is closest in spirit to our work. They too address the minimum-energy cooperative broadcast problem under an energy accumulation assumption. However, in their work the system has memory, in that the

nodes can save soft information from all previous transmissions throughout time and use it to decode data later on. They prove that the problem is NP-complete in this case. In their setting, because of the memory, it is shown that it suffices to have each transmitter transmit only once; therefore there is no distinction between ordering and scheduling. This is no longer true in our memoryless setting where the energy from past transmissions cannot be accumulated. We therefore decouple the transmission ordering from joint scheduling and power control in our work and can provide a polynomial solution for the latter given an ordering. Beyond this, a key distinction in our work is that we consider delay constraints, whereas [2] focuses only on the minimum energy cooperative solution without delay constraints.

One prior work that discusses the power-delay tradeoff in a cooperative setting is the paper by Cui and Golsmith [13], however, the focus of that work is on space-time codes used for unicast forwarding, not broadcast.

### III. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a static wireless network with  $N$  nodes. Radio propagation is modeled by a given symmetric  $N$  by  $N$  channel matrix,  $H = \{h_{ij}\}$ , representing the gain on the channel between each pair of nodes  $i$  and  $j$ . Time is assumed to be discretized into fixed duration slots. We assume cooperative communication in which the received power at a given receiver at a time is additive in the attenuated powers received from each sender at that time. As described in [3], [5], [4], this kind of additive received power can be achieved via *maximal radio combining* under different scenarios including transmission using TDMA, FDMA channels, as well as with CDMA spreading codes and space-time codes. We assume appropriate coding is used so that each receiving node can decode the message so long as its received signal-to-noise (SNR) ratio is higher than a given threshold. Assuming the noise power is the same at all receivers, we can use a normalized received power threshold  $\tau$  to model this. We assume a memory-less model in which nodes do not accumulate energy from transmissions occurred in previous time slots. Thus the condition for successful decoding at some receiver node  $r$  at time  $t$  when a set of nodes  $S(t)$  is transmitting packets, with transmit power  $p_s(t) \forall s \in S(t)$  is:

$$p_r^{rcv}(t) = \sum_{s \in S(t)} p_s(t) h_{sr} > \tau \quad (1)$$

We consider the minimum-energy broadcast problem in a time-slotted wireless cooperative network of  $N$  nodes. We assume that the broadcast begins from a single source node. The aim is to get the message to all the nodes in the network with the minimum possible total energy within a time constraint,  $T_{max}$  (which can vary anywhere from 1 to  $N-1$ ). All nodes are assumed to operate in half-duplex mode, i.e. they cannot transmit and receive simultaneously. If used in transmission, the nodes operate based on a decode and

forward protocol. Therefore, they are not allowed to take part in transmission until they have fully decoded their message.

Let  $X_i(t)$  be an indicator variable that indicates for all nodes other than the source node if node  $i$  has decoded the message before time  $t$  as per equation (1), and for the source node 1, let  $X_1(t) = 1, \forall t$ . In essence, the optimization problem we have for delay constrained minimum energy broadcast is the following:

$$\begin{aligned} \min \quad & P_{total} = \sum_{t=1}^{T_{max}} \sum_{i=1}^N p_i(t) \quad (2) \\ \text{s.t.} \quad & X_i(T_{max} + 1) = 1, \quad \forall i \\ & X_i(t) = 0 \Rightarrow p_i(t) = 0, \quad \forall i \end{aligned}$$

The design variables we have in this optimization are  $p_i(t)$ , the transmit power for each node  $i$  at each time  $t$ , with the second constraint specifying that non-zero powers can only be allocated to nodes that have previously decoded the packet, since only these nodes can participate in transmission. (Note, however, that in the optimal solution,  $p_i(t)$  might be set to zero even for nodes such that  $X_i(t) = 1$ .) The first constraint imposes the delay requirement that we wish all nodes to have decoded the packet by the end of  $T_{max}$  slots.

The decision version of this problem, which we refer to in the following as DMECB (delay constrained minimum energy cooperative broadcast) can be defined correspondingly as follows: ‘‘Given some power bound  $C$ , does there exist an allocation of powers  $p_i(t)$  satisfying the constraints in 3 such that  $P_{total} \leq C$ ?’’ Note that an instance of the DMECB problem is defined by giving the symmetric  $N \times N$  matrix  $H$ , with a designated source node (vertex), a delay bound  $T_{max}$  and a power bound  $C$ . (Note: in the following sections we will also represent DMECB in the form of a graph  $G = (V, E)$  in which edges exist between any pair of nodes for which  $h_{ij} > 0$ , and the edges are labeled with a weight  $w_{ij} = h_{ij}^{-1}$  that corresponds to the transmit power needed at node  $i$  to exceed an SNR threshold of  $\tau = 1$  at the receiver  $j$ .)

### IV. HARDNESS

In this section we prove that not only finding an optimal solution for DMECB problem is NP-hard but finding any polynomial time algorithm that approximates the optimal solution within a factor of  $o(\log(n))$  is also NP-hard. We do so by an approximation-preserving reduction from the set cover problem.

Set cover problem is a classical problem in computer science. It is stated as follows: Given a universe  $U$  of  $n$  elements and a collection of subsets of  $U$ ,  $S = S_1, S_2, \dots, S_k$ , find a minimum subcollection of  $S$  that covers all elements of  $U$ . This problem is NP-complete and was shown, in [17], to be  $o(\log(n))$  inapproximable.

The set cover problem can be thought of as a bipartite graph  $G(V, E)$ , with  $|V| = k + n$ , representing the  $k$  sets and  $n$  elements and the edges are used to connect each set to its elements. This is shown in Figure 1 (a), where we assign a vertex for each set in the top part of the graph, and assign

a vertex for each element in the bottom part of the graph. We connect each set to its elements using an edge. Given an instance,  $G$  of the set cover problem, the optimal solution to the set cover problem  $OPT_{sc}$  would find the minimum subset of vertices in the top part of the graph, so that their transmission of a message can broadcast the message to all the vertices in the bottom part of the graph.

Given an instance,  $G$  of the set cover problem, with  $k$  sets and  $n$  elements, let us construct a new graph  $G'$  as follows: Assign a root node  $r$ , which is the source with the message at the starting time, call this level 0. Include  $k$  nodes in level 1, representing the  $k$  sets in the set cover problem, all connected to the root node with a small weight (say weight 1), as shown in Figure 1 (b). This is followed by the bipartite graph of  $G$ , which makes up level 2 and 3 of  $G'$ . Connect each of the  $k$  nodes in level 2 to their representative in level 1 and to all the other nodes in level 2 with low-weight edges. Notice the nodes in level 2 are also connected to their elements in level 3 of the graph, as shown in the Figure. We make all the weight on the edges arbitrarily small (say 1), with the exception of the edges in between the nodes in level 1 and 2. We make those edges arbitrarily large (say  $M$ , to be specified later).

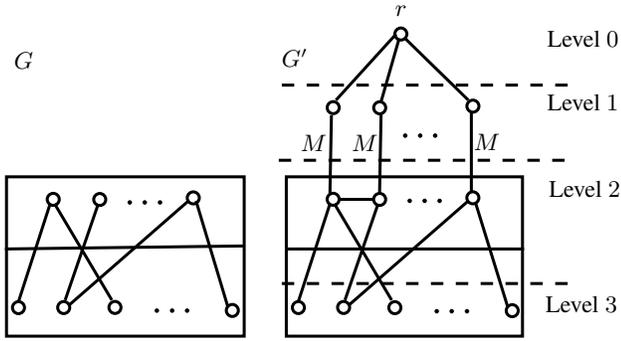


Fig. 1: Construction of  $G'$  for a given  $G$ , notice that not all the edges are shown (for clarity).

Assume the the weight on the edges represent the power needed for the message to be transmitted across that edge. If we were to run the optimal DMECB algorithm on  $G'$  with  $T_{max} = 3$  the algorithm would have to act as follows, to be able to cover all the nodes in the given time frame:

*Step 1:* Root transmits with power 1, turning on all its  $k$  neighbors on level 1.

*Step 2:* The algorithm picks a subset the  $k$  nodes on level 1 to transmit the message. This subset must be chosen to be as small as possible, given the large weight they have to endure to pass on the message on to the bipartite graph, and the fact that DMECB is trying to minimize the total weight. Yet it has to be large enough so that when the nodes in level 2 transmit, all the nodes in level 3 would receive the message. The optimal algorithm must be able to find such a subset.

*Step 3:* The nodes that receive the message in level 2, transmit the message in this step, turning on all the nodes in level 3 of the graph, as well as all the nodes in level 2 of the graph that were not selected for transmission, thus covering the whole

graph.

Let us call the solution of this optimal algorithm  $OPT_{DMECB}$ .

**Lemma IV.1**  $OPT_{DMECB} \leq M.OPT_{SC} + 1 + OPT_{SC}$

*Proof:* Consider an instance of SC (with graph  $G$ ), whose optimal solution is  $OPT_{SC}$ . Construct a graph  $G'$ , as explained and run the DMECB algorithm to get  $OPT_{DMECB}$ . The above inequality holds by construction of the graph. ■

**Lemma IV.2**  $OPT_{SC} \leq \frac{OPT_{DMECB}}{M}$

*Proof:* Consider an instance of DMECB on  $G'$  and its optimal solution  $OPT_{DMECB}$  for delay  $T = 3$ . Notice that if  $T > 3$ , we add additional single nodes (as virtual roots) to reduce the problem to the case where  $T = 3$ . Looking at  $G'$ , we observe that to meet the delay constraint, by end of step  $i$ , at least one node in level  $i$  must have heard the message - else it is impossible to get the message through to the rest of the levels in the time frame left. Let's say the root is on level 0. Consider the subset of level 1 that has come on at the end of time 1,  $s_1$  and from level 2 consider the set,  $s_2$ , that came on at the end of time step 2. We now want to show that  $s_2$  is a feasible solution for set cover. To do so, we make the following two claims: *Claim 1:* Nodes responsible for turning on  $s_2$  must be a subset of  $s_1$ . *Claim 2:*  $s_2$  is a feasible solution to set cover. Claim 1 holds because only nodes that have received the message by the end of time 1 can transmit the message at time 2. Not all of them might transmit though, so  $s_2$  is a subset of corresponding nodes in  $s_1$ . Claim 2 is true, because if there exists an element in level 3 that is not a corresponding node to anyone in  $s_2$ , it cannot come on by  $T = 3$ . Therefore,  $s_2$  is a feasible solution to set cover.  $OPT_{DMECB}$  must spend at least  $M$  for each element of  $s_2$  to come on, so  $OPT_{SC} \leq \frac{OPT_{DMECB}}{M}$ . ■

**Theorem IV.1** *The DMECB problem is  $o(\log(n))$  inapproximable, for  $T_{max} \geq 3$ .*

*Proof:* For an instance of set cover problem, with  $k$  being the total number of sets, lemma IV.1 can be re-written as  $OPT_{DMECB} \leq M.OPT_{SC} + 1 + k$ . We also know by lemma IV.2 that  $OPT_{SC} \leq \frac{OPT_{DMECB}}{M}$ . Therefore, for a sufficiently large  $M$ , we can write  $OPT_{SC} = \frac{OPT_{DMECB}}{M} + o(1)$ . Therefore, the reduction used in construction of the graph  $G'$  preserves the approximation factor. That is, if one can find an  $\alpha$ -approximation for DMECB, by extension there must exist an  $\alpha$ -approximation for set cover. We know, by [17], that the set cover problem is  $o(\log(n))$  inapproximable, thus DMECB must be  $o(\log(n))$  inapproximable. In other words, finding a polynomial time approximation algorithm that approximates  $OPT_{DMECB}$  with a factor of  $o(\log(n))$  is NP-hard. ■

The DMECB problem can be solved in polynomial time (trivially) for cases when  $T_{max} < 3$ . It is also trivial to verify the feasibility of a given power allocation, and verify whether or not it satisfies the decision version of DMECB given in

section III. Therefore, the problem belongs to the class of NP. Notice that the inapproximability result, given by Theorem IV, is stronger than NP-completeness result and implies NP-completeness. It is also worthy to note that without any delay constraint, the problem is still NP-complete and the proof can be obtained by following the approach in [2].

## V. OPTIMAL BROADCAST GIVEN ORDERING

In Section IV, we proved that the DMECB is NP-complete and  $o(\log(n))$  innapproximable. In this section, we breakdown this NP-complete problem into three subproblems, namely ordering, scheduling and power allocation, and given an ordering we propose an optimal polynomial time algorithm for joint scheduling and power allocation. We evaluate a heuristic for the ordering in Section VIII.

**Definition** An ordering, for a vector of  $n$  nodes, is an array of indices from 1 to  $n$ ; re-arranging the elements of that vector according to those indices would result in an ordered vector where each node, that has decoded the message, will only be allowed to take part in re-transmission when all its previous nodes have also decoded the message (and are thus allowed to take part in transmission).

Given an ordering, what remains to be solved is to figure out which nodes should take part in transmission, how much power they should transmit with and at what time slots, so we can achieve the minimum energy within the given delay constraint.

### A. Instantaneous optimal power allocation

If we know which nodes are transmitting the message and which nodes are receiving it, at any single time-slot, we can use a linear program (LP) to work out the optimal power allocation for that time slot. Consider an ordered vector of  $n$  nodes  $(1, \dots, k, \dots, i, \dots, n)$ , represented by their indices so that node  $i$  is the  $i$ th node in the array. Let us assume that by time slot  $t$ , node 1 to  $i$  have decoded the message and nodes  $i + 1$  to  $n$  are yet to decode. At time instance  $t$ , the optimal instantaneous power allocation for a set of transmitting nodes (say  $S(t) = (k, \dots, i)$ ) to turn on a set of receiving nodes nodes (say  $R(t) = (k + 1, \dots, n)$ ) can be calculated out using following LP:

$$\min \sum_{s \in S(t)} p_s(t) \quad (3)$$

such that

$$\sum_{s \in S(t)} p_s(t) h_{sr} > \tau \quad \forall r \in R(t) \quad (4)$$

We use the notation  $LP(\{[k, \dots, i], [k + 1, \dots, n]\}, \tau, H)$  to refer to solution of the above LP. As a notation,  $LP(\{[x, \dots, y], [z, \dots, \alpha]\}, \tau, H) = 0$ , if  $z \geq \alpha$ .

### B. Joint Scheduling and power allocation

Knowing the instantaneous optimal power allocation given the set of senders and receivers at each time slot, all that remains to be done is to determine these sets at each time slot, in order to minimize the overall power while meeting the delay constraint.

Let  $C(j, t)$  be the minimum energy needed to cover up to node  $j$  in  $t$  steps or less. The total minimum cost for covering  $n$  nodes by time  $T$  can be calculated using the following algorithm:

$$C(n, T) = \min [C(k, T - 1) + LP(\{1, \dots, k\}, \{k + 1, \dots, n\}, \tau, H)] \quad (5)$$

where  $\min$  is taken over  $k \in (1, \dots, n)$ ,  $C(k, 1) = LP(1, \{2, \dots, i\}, \tau, H)$ ,  $C(1, t) = 0 \quad \forall t$ , and  $C(2, t) = h_{12}^{-1} \quad \forall t$ . A pseudocode for the algorithm is presented below:

---

**Algorithm 1** Delay constrained minimum energy cooperative broadcast, given an ordering (DMECB\_go)

---

```

1: INPUT: an ordered array of nodes of size  $n$  (where node
    $i$  is the  $i$ th node in the array),  $T$  (delay),  $H$  (channel),  $\tau$ 
   (threshold).
2: OUTPUT:  $C$  (cost matrix)
3: Begin:
4: for  $i := 2$  to  $n$  do
5:    $C(i, 1) := LP([1, \{2, \dots, i\}], \tau, H)$ ;
6: end for
7: for  $t := 1$  to  $T$  do
8:    $C(1, t) := 0$ ;
9:    $C(2, t) := h_{12}^{-1}$ 
10: end for
11: for  $t := 2$  to  $T$  do
12:   for  $i := 1$  to  $n$  do
13:     for  $k := 1$  to  $i$  do
14:        $x(k) := C(k, t - 1) + LP(\{[1, \dots, k], [k + 1, \dots, i]\}, \tau, H)$ ;
15:     end for
16:      $C(i, t) := \min \mathbf{x}$ 
17:   end for
18: end for

```

---

The optimal scheduling and power allocation can be obtained by inspection of the above algorithm. Our Dynamic Programming algorithm invokes  $O(n^2 T_{max})$  calls to the LP solver, each of which takes polynomial time. Hence the DMECB\_go algorithm that does joint scheduling and power-control is a polynomial time algorithm.

Notice that a requirement for per-node maximum power can be trivially added to the LP formulation as additional constraints, we have left that out for simplicity. Should the maximum power be added, it should be large enough to ensure a feasible solution exists for the given connectivity and delay constraint. Also note that the delay constraint  $T$  can be made arbitrarily large, or removed entirely from the formulation, to allow for a solution to be found assuming there are no delay

constraints. In the latter case, the two-dimensional dynamic program proposed in (5), will reduce to a one-dimensional dynamic program:

$$C(n) = \min [C(k) + LP(\{1, \dots, k\}, \{k+1, \dots, n\})], \tau, H \quad (6)$$

where  $C(n)$  is the minimum cost of covering node  $n$  using our cooperative memoryless approach, starting from node 1 and  $C(1) = 0$ .

## VI. ANALYTICAL LOWER-BOUND

In this section we establish a lower-bound on the optimum solution to DMECB.

To get a better intuition for this lower-bound, let us start off by considering a unicast version of DMECB. As before, we have  $n$  nodes and a channel  $H$ , but this time the source  $s$  wants to transmit the message to a particular destination  $d$ , using the minimum energy within a given delay constraint  $T$ . The system is cooperative in that other nodes in the network, may be utilized as memoryless energy accumulating relays to help achieve the minimum energy goal. Let us call this problem *delay constrained minimum energy cooperative unicast*, DMECU.

**Theorem VI.1** *In DMECU, there exists a solution consisting of a simple path between source and destination, which is optimum.*

*Proof:* Let us prove by induction: *Claim:* In the DMECU problem with delay constraint  $T$ , there always exists a simple path from source  $s$  to any destination  $d$ , which is optimum. For  $T = 1$ , the claim is trivially true, as the optimal solution is direct transmission from  $s$  to  $d$ . Let us assume the claim is true for  $T = k - 1$ . To complete the proof, we need to show the claim holds for  $T = k$ . Pick any node in the network as the desired destination  $d$ . If the message can be transmitted from source  $s$  to  $d$  with minimum energy in a time frame less than  $k$ , then an optimal simple path exists by the induction assumption. So consider the case when it takes exactly  $T = k$  steps to turn on  $d$ . The system is memoryless, so  $d$  must decode by accumulating the energy transmitted from a set of nodes,  $\mathbf{v}$ , at time  $k$ . This can be represented as  $\sum_{v_i \in \mathbf{v}} p_{v_i}(k) h_{dv_i} \geq \tau$ . We observe that there must exist a node  $v_o \in \mathbf{v}$  whose channel to  $d$  is equal or better than all the other nodes in  $\mathbf{v}$ . Therefore, given  $h_{dv_o} \geq h_{dv_i}, \forall v_i \in \mathbf{v} - \{v_o\}$  then  $\sum_{v_i \in \mathbf{v}} p_{v_i}(k) h_{dv_o} \geq \sum_{v_i \in \mathbf{v}} p_{v_i}(k) h_{dv_i} \geq \tau$ . In another words, if we add the power from all nodes in  $\mathbf{v}$  and transmit instead from  $v_o$ , our solution cannot be worse.  $v_o$  must have received the message by time  $k - 1$ , to be able to transmit the message to  $d$  at time  $k$ . We know by the induction assumption that the optimal simple path solution exists from source to any node to deliver the message within  $k - 1$  time frame. Thus, for  $T = k$ , there exists a simple path solution between  $s$  and  $d$ , which is optimum. ■

Notice that the above theorem holds in the case where there is no delay constraint as well. The proof follows an straightforward modification of the above proof and is omitted for brevity.

**Corollary VI.1** *The Dijkstra's shortest path algorithm provides the optimal ordering in the case of minimum energy memoryless cooperative unicast, when there is no delay constraint.*

*Proof:* We have already established that an optimal minimum energy solution exists between source and destination, which is a simple path. The well-known Dijkstra's shortest path algorithm can find the minimum cost simple path between source and destination. Therefore, Dijkstra's algorithm provides the optimal ordering. ■

Using theorem VI.1 we know that the optimal unicast solution from source to any destination is given by a simple path. To find a lower-bound for the DMECB, we notice that the source has to cover all the nodes in the network. The cost paid by optimal DMECB to cover each node, cannot be lower than the cost paid by the optimal DMECU to cover that node - otherwise it contradicts the optimality of DMECU solution. Based on that observation we derive the following lower-bound,  $LB(T)$ , for the  $OPT_{DMECB}$  when the delay constraint is  $T$ :

$$LB(T) = \max_{i \in V - \{r\}} C(i, T) \quad (7)$$

where

$$C(i, T) = \min_{x \in Nr(i)} [C(x, T - 1) + w(x \rightarrow i)] \quad (8)$$

where  $V$  represents the set of all nodes in the network,  $r$  is the root,  $Nr(i)$  is the set that contains  $i$  and its neighboring nodes that have a non-zero edge to  $i$ ,  $w(x \rightarrow i)$  represents the power it takes for  $x$  to turn on  $i$  using direct transmission,  $C(i, T)$  is the minimum cost it takes for  $r$  to turn on  $i$ , possibly using relays, within at most  $T$  time slots. We set  $C(i, 1) = w(r \rightarrow i)$ . In words,  $C(i, T)$  calculates the minimum cost of optimal unicast DMECU to cover a given node  $i$ , starting from root under a delay constraint  $T$ .  $LB(T)$  takes the maximum of those costs and use it as lower-bound - since we know  $OPT_{DMECB}$  has to cover the costliest node and cannot do so any better than the  $OPT_{DMECU}$ . Computing this lower-bound incurs a running time of  $O(n^3)$ .

## VII. APPROXIMATION ALGORITHM

In section IV, we proved that DMECB is NP-complete and  $o(\log(n))$  inapproximable, therefore it is *hard* to approximate DMECB to a factor better than  $\log(n)$ . It is of theoretical interest to know how close we can get to the optimal solution, using a polynomial time algorithm. In this section we show that existing approximation algorithms for bounded diameter directed Steiner tree problem can be used to provide  $O(n^\epsilon)$  approximation for DMECB.

Steiner tree problem is a classic problem in combinatorial optimization [16]. We focus on a variation of this problem namely, bounded diameter directed Steiner tree, defined as follows. Given a directed weighted graph  $G(V, E)$ , a specified root  $r \in V$ , and a set of *terminal nodes*  $X \subseteq V$  ( $|X| = n$ ), the objective is to find minimum cost arborescence rooted

in  $r$  and spanning all vertices in  $X$ , subject to a maximum diameter  $T$ . Diameter refers to the maximum number of edges on any path in the tree. Notice that the tree may include vertices not in  $X$  as well, these are known as *Steiner nodes*. Directed Steiner tree problem is known to be *NP*-complete and  $O(\log(n))$  inapproximable [16]. In [15], the authors give the first non-trivial approximation algorithms for Steiner tree problems and propose approximation algorithms that can achieve an approximation factor of  $O(n^\epsilon)$  for any fixed  $\epsilon > 0$  in polynomial time. To the best of our knowledge this is currently the tightest approximation algorithms known for this problem.

In order to construct an instance of the Steiner tree problem from DMECB, we first restrict DMECB by not allowing multi-transmitter multi-receiver (multi-multi) transmissions. Notice that in the proof of theorem VI.1, we had established that multi-single transmissions, can be replaced with single-single transmissions without loss of optimality. Therefore, by not allowing multi-multi transmissions, we are limiting ourselves to single-single and single-multi transmissions. We call this an integral version of DMECB, DMECB-int. The integrality gap of the weighted set cover problem is shown to be  $\log(n)$  [16], it is straightforward to extend that result to show that DMECB-int also loses a factor of  $\log(n)$ , compared to optimal DMECB.

Consider an instance of DMECB-int,  $G(V, E)$ , with  $(|V| = n)$  and  $s \in V$  being the source node. To reduce this problem to an instance of directed Steiner tree problem, let us construct a new graph  $G'$ , consisting of  $n$  clusters,  $x'$ , each corresponding to each node in  $G$ . Let each cluster be a bipartite graph, with  $n$  nodes on the left (marked as “-”) and  $n$  nodes on the right (marked as “+”), as shown in Figure 2. The “-” nodes are intra-connected within a cluster with edges of weight 0. In each cluster,  $x' \in G'$  corresponding to node  $v \in G$ , the “+” and “-” nodes on each level,  $i$ , of the bipartite graph are connected to each other with an edge of weight  $w_i$ , representing the power needed by the corresponding node  $v \in G$  to turn on its  $i$  closest neighbors. The  $i_+$  node is then connected, with edges of weight 0, to all the “-” nodes in the corresponding neighbor clusters. We further add a single root node,  $r \in G'$ , and connect it via a zero-weight edge to all the “-” nodes in the cluster corresponding to  $s$ ,  $x'_s$ . We assign the root  $r$  and one desired “-” node from each cluster as terminal nodes and all other nodes in  $G'$  as Steiner nodes.

Let us look at an example of this construction, say node  $v_1 \in G$ , whose closest 3 neighbors are  $(v_2, v_4, v_6)$ . We have an equivalent cluster  $x'_1 \in G'$  corresponding to node  $v_1$ .  $x'_1$  has  $2n$  nodes, arranged in  $n$  levels. The weight between the two nodes in say level 3 is equivalent to the power it takes for  $v_1$  to turn on  $(v_2, v_4, v_6)$ . Furthermore, the node  $3_+$  in cluster  $x'_1$  is connected to the “-” nodes in clusters  $(x'_2, x'_4, x'_6)$  with edges of weight 0. This construction allows us to find a way to allow  $v_1$  to transmit with different power levels, without knowing what those powers might be in advance. We first add a single root node,  $r$ , and connect it via a zero-weight edge to all the “-” nodes in the cluster corresponding to  $s$ .

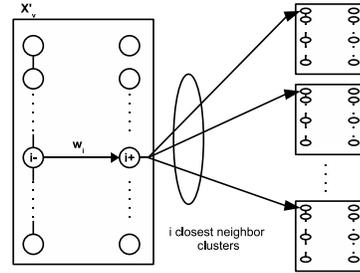


Fig. 2: A simplified example of how clusters are constructed in  $G'$ .

Run the directed Steiner tree algorithm on  $G'$  to obtain a solution. The solution must choose at least one node from each cluster, to meet the mandatory terminal nodes requirement. Recall that each cluster in  $G'$  corresponds to a node in  $G$  and that multi-multi was not allowed. To convert the solution of the Steiner tree algorithm on  $G'$  to a solution of DMECB-int on  $G$ , we look at the parent of each cluster, which is a “+” node in another cluster. Let’s say we want to see which node turns on  $v_6$  by looking at  $G'$ . We look at the parent of  $x'_6$  and see that it’s  $3_+ \in x'_1$ . So in  $G$ , we figure out that  $v_1$  must transmit with enough power to turn on 3 of its closest neighbor ( $w_3$ ), and it is as a result of this transmission that  $v_6$  comes on. Going through all the clusters and their parents, we can establish an ordering and transmission power for all the nodes that should take part in cooperation in  $G$ , and thus we have a solution for DMECB-int.

As mentioned, the directed Steiner tree is  $o(\log(n))$  inapproximable, and the best approximation algorithm currently available [15] give an  $O(n^\epsilon)$  approximation on the optimal solution. We had already lost  $O(\log(n))$  to convert DMECB to its integral form. The approximation algorithm proposed in [15] can approximate the optimal integral solution within  $O(n^\epsilon)$ . Therefore, using that algorithm we can approximate the optimal solution to DMECB within  $O(n^\epsilon \times \log(n))$ , which is asymptotically  $O(n^\epsilon)$ .

Therefore, using the construction given above, approximation algorithms given in [15] can be used to give an  $O(n^\epsilon)$  approximation for DMECB. Detailed discussions of the algorithms in [15] are beyond the scope of this paper. The running time of these algorithms are a function of  $\epsilon$ , thus the better the tradeoff the worse the running time. It remains of interest to see if tighter approximation algorithms can be constructed for DMECB. In the following section, we use computationally simple heuristics for ordering and use algorithm proposed in section V to calculate the optimal scheduling and power allocation given the ordering. We then compare these results against the lowerbounds given in section VI.

## VIII. PERFORMANCE EVALUATION

For the simulation purposes, we consider a network of  $N$  nodes uniformly distributed on a 15 by 15 square surface.

The transmission starts from a node, arbitrarily located, at the left center corner of the network, coordination (0, 7). We consider a static channel, with independent and exponentially distributed channel gains (corresponding to Rayleigh fading), where  $h_{ij}$  denotes the channel gain between node  $i$  and  $j$ . The mean value of the channel between two nodes,  $\overline{h_{ij}}$ , is chosen to decay with the distance between the nodes, so that  $\overline{h_{ij}} = d_{ij}^{-\eta}$ , with  $d_{ij}$  being the distance between nodes  $i$  and  $j$  and  $\eta$  being the path loss exponent. The corresponding distribution for channel gains, is then given by

$$f_{h_{ij}}(h_{ij}) = \frac{1}{h_{ij}} \exp\left(-\frac{h_{ij}(k)}{h_{ij}}\right)$$

Based on the intuition developed in Section VI, we use the Dijkstra's shortest path algorithm as our ordering heuristic. Simulations are repeated multiple times and average values are shown in the graphs. Notice that the minimum power calculated by different algorithms, shown on the y-axes of the graphs in this section, are normalized by value of  $\tau$ .

In Figure 3, we calculate the optimal ordering by brute-force for small number of nodes and compare the performance of our algorithm that uses Dijkstra's shortest path-based ordering with the optimal performance. As can be seen, Dijkstra's algorithm provides a good heuristic for ordering and will be used throughout this section.

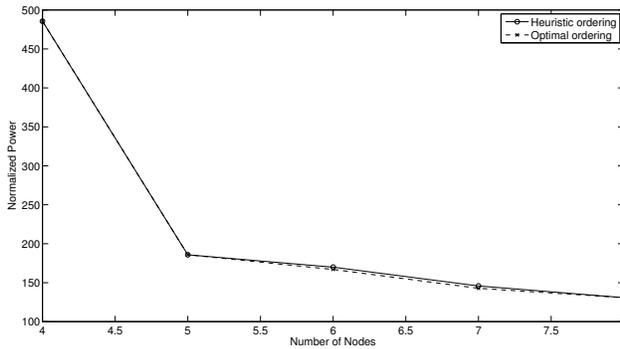


Fig. 3: Performance with optimal ordering vs Dijkstra's algorithm-based heuristic ordering

We next compare the performance of our cooperative algorithm with a smartly designed non-cooperative algorithm. Notice that in our cooperative algorithm we make use of the *wireless broadcast advantage (WBA)*, where transmission by one node can be received by multiple nodes and *cooperative advantage*, where a node can accumulate power from multiple transmitters. If an algorithm is using WBA, but not cooperative it can be thought of as an integral version of DMECB. This means, each node can receive the message from one transmitter only (and cannot accumulate from multiple transmitters), however one transmitter can transmit to multiple receivers. We had established in Section VII that DMECB-int is also NP Complete. It is however interesting to note that DMECB-int needs to solve a weighted set cover problem when

allocating powers as well, we know that set cover problem is  $o(\log(n))$  inapproximable [16], so the non-cooperative case is  $o(\log(n))$  inapproximable, even when ordering is provided. Greedy algorithms exist [16] that give  $O(\log(n))$  approximation for the weighted set cover problem, and thus providing a tight polynomial time approximation. Therefore, to simulate a smart non-cooperative algorithm, we use Dijkstra's algorithm-based ordering and the algorithm given in Section V, with the exception that instead of using an LP we use the greedy algorithm for power allocation.

The performance comparison between our proposed cooperative algorithm and the smart non-cooperative algorithm, for different values of  $N$  is shown in Figure 4 and power-delay tradeoff for cooperative and non-cooperative algorithms are presented in 5. As can be seen, the cooperative algorithm outperforms non-cooperative algorithm, and the advantage is more pronounced when a delay constraint is imposed.

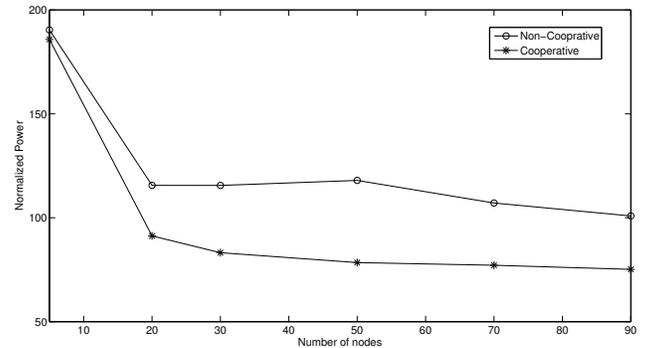


Fig. 4: Effect of cooperation in broadcast

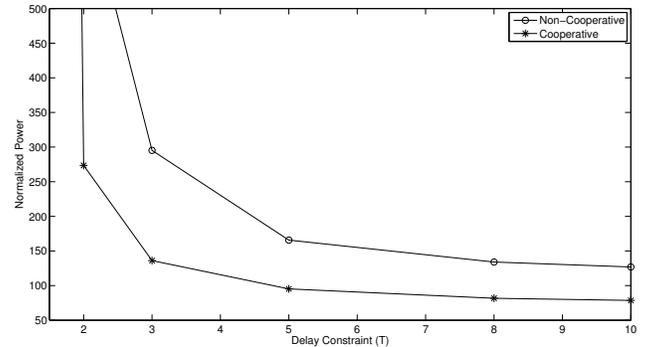


Fig. 5: Power-delay tradeoff in cooperative vs non-cooperative case

We study the power-delay tradeoff of the cooperative algorithm for different channel conditions and different values of  $\rho$  are given in Figure 6 and Figure 7, respectively, where  $\rho$  is network density (nodes/area). These figures highlight the sensitivity of the dense networks and those with poor channel conditions to delay constraints and the importance of having smart algorithms to minimize the energy consumption.

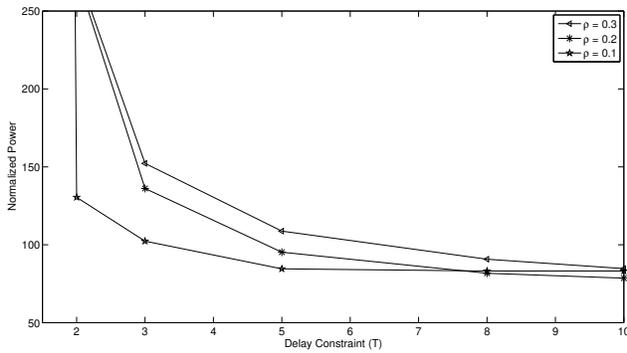


Fig. 6: Effect of network density on power-delay tradeoff

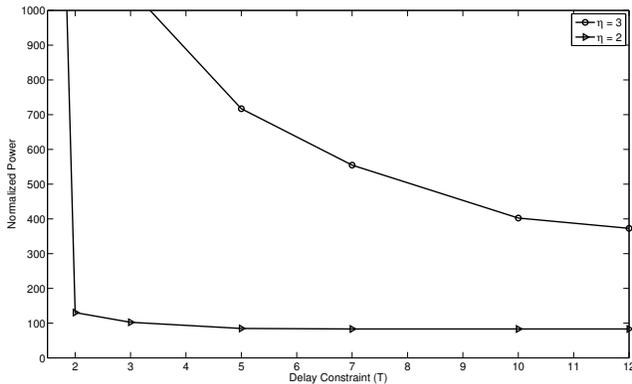


Fig. 7: Power-delay tradeoff for varying channel conditions

## IX. CONCLUSIONS

We have formulated the novel problem of delay constrained minimum energy cooperative broadcast (DMECB) in wireless networks. We have shown that this problem is  $o(\log n)$  approximable. For the general problem, we have developed an analytical lower bound and provable approximation results. Another key algorithmic contribution has been to show a polynomial time algorithm that can solve the problem optimally for a fixed transmission ordering. Our empirical results suggest that for practical settings, a near-optimal ordering can be obtained by using Dijkstra's shortest path algorithm.

There are a number of interesting directions for future work. In this paper we have focused on the static problem with full information, which allows for centralized decision making. In the future, we would like to explore distributed solutions to this problem which would be particularly suitable for more dynamic settings. From an analytical perspective, there is room for improvement in computing tighter lower bounds as well as further enhancements in the approximation results. Evaluating the proposed algorithms under more realistic settings (through more detailed simulations of physical layer implementation or through direct implementation on software radio platforms, and the use of more realistic energy models) would certainly help in moving this work towards practice. Finally, our work,

like most work in this domain of cooperative broadcasts, has focused on the single flow setting. It is of interest to study generalizations that allow for multiple simultaneous flows in the network.

## REFERENCES

- [1] A. Khandani, J. Abounadi, E. Modiano, L. Zhang, "Cooperative Routing in Wireless Networks," *Allerton Conference on Communications, Control and Computing*, October, 2003.
- [2] I. Maric and R. D. Yates, Cooperative Multihop Broadcast for Wireless Networks, *IEEE JSAC Special Issue on Fundamental Performance Limits of Wireless Sensor Networks*, Vol. 22, No. 6, Aug. 2004.
- [3] B. Sirkeci Mergen, A. Scaglione, G. Mergen, "Asymptotic Analysis of Multi-Stage Cooperative Broadcast in Wireless Networks," *Joint special issue of the IEEE Transactions on Information Theory and IEEE/ACM Trans. On Networking*, Vol. 52, No. 6, June 2006.
- [4] S.-H. Chen, U. Mitra, B. Krishnamachari, "Cooperative communication and routing over fading channels in wireless sensor networks," *IEEE International Conference on Wireless Networks, Communications, and Mobile Computing (WirelessCom)*, Maui, Hawaii, June 2005.
- [5] B. Sirkeci Mergen, A. Scaglione "On the power efficiency of cooperative broadcast in dense wireless networks," *IEEE Journal on Selected Areas in Communications (JSAC)*, Volume 25, Issue 2, February 2007.
- [6] G. Jakllari, S. V. Krishnamurthy, M. Faloutsos and P. Krishnamurthy, "On Broadcasting with Cooperative Diversity in Multi-hop Wireless Networks," *IEEE JSAC, Special Issue on Cooperative Communications and Networking*, Vol. 25, No. 2, February 2007.
- [7] M. A. Baghaie, B. Krishnamachari, "Fast Flooding using Cooperative Transmissions in Wireless Networks", *IEEE ICC 2009*.
- [8] S.C. Draper, L. Liu, A. F. Molisch, J. S. Yedidia, "Routing in Cooperative Wireless Networks with Mutual-Information Accumulation," *IEEE ICC 2008*.
- [9] D. K. Lee, K. M. Chugg, "A Pragmatic Approach to Cooperative Communication," *IEEE Military Communications Conference*, 2006.
- [10] T. Halford, K. Chugg, "Barrage Relay Networks," *UCSD ITA Workshop*, San Diego, 2010.
- [11] S.-Y.Ni, Y.-C. Tseng, Y.-S. Chen, J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking (MOBICOM)*, 1999.
- [12] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. Int. Symp. Mobile Ad Hoc Networking Computing (MobiHoc 2002)*, June 2002.
- [13] S. Cui, and A. J. Goldsmith, "Cross-layer design of energy-constrained networks using cooperative MIMO techniques", *Elsevier Signal Processing Journal*, 2006.
- [14] M. Cagalj, J.-P. Hubaux and C. Enz, "Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues" *Proceedings of ACM MobiCom 2002*, September 2002
- [15] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha and M. Li, "Approximation algorithms for directed Steiner tree problems", *Journal of Algorithms*, Vol. 33, pp. 73-91, 1998.
- [16] V. Vazirani, *Approximation Algorithms*, Springer-Verlag, Berlin, 2001.
- [17] R. Raz, and S. Safra, "A sub-constant error-probability low-degree test, and sub-constant error-probability PCP characterization of NP", *Proc. 29th Ann. ACM Symp. on Theory of Comp.*, ACM, 475-484, 1997.