Characterizing Achievable Multicast Rates in Multi-Hop Wireless Networks

Randeep Bhatia
Bell Labs, Lucent Techologies
600 Mountain Ave.
Murray Hill, NJ 07974
randeep@research.bell-labs.com

Li (Erran) Li
Bell Labs, Lucent Techologies
101 Crawfords Corner Road
Holmdel, NJ 07733
erranlli@bell-labs.com

ABSTRACT

In this paper, we consider the multicast throughput optimization problem in multi-hop wireless networks. Given a source, and a set of receivers, we would like to find the set of multicast trees and a schedule such that the rate that the source can multicast to the receivers is maximized. We consider two transmission models: broadcast and unicast. In the broadcast model, a transmission is received by multiple downstream nodes in a multicast tree. In the unicast model, a separate transmission has to be sent to each downstream node. We consider the fundamental constraint that a node can not be involved in multiple communications at the same time. We consider two multicast models: a single multicast tree per session and multiple multicast tree per session. In the single multicast tree case, (1) for the unicast model, we show that the problem is NP-hard and it is not approximable to a factor better than 1.5; we then give a 1.5-approximation algorithm if all links have the same data rate, a 5-approximation algorithm if all nodes have the same transmission power and a 24-approximation algorithm for a realistic heterogeneous ad hoc network where nodes can have different transmission power. (2) for the broadcast model, we show that the problem is NP-hard and it is not approximable to a factor better than 2; we then give a simple 2-approximation algorithm to find the multicast tree and the transmission schedule. In the multiple multicast tree case, (1) for the unicast model, we show that the problem is APX-hard, and give a 1.5ρ-approximation where ρ is the best approximation ratio of the minimal cost Steiner tree problem; (2) for the broadcast model, our results indicate that the problem is hard, may not be approximable within a factor better than $log(n)$ where n is the number of multicast receivers. Our evaluation shows that the throughput achieved by our algorithms is much better than both the throughput achieved by using pruned shortest path tree and by using optimal unicast.

MobiHoc'05, May 25–27, 2005, Urbana-Champaign, Illinois, USA.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design; C.2.2 [**Computer-Communication Networks**]: Network Protocols

General Terms

Algorithms, Theory

Keywords

Mathematical programming, optimization, Graph theory, Combinatorics

1. INTRODUCTION

Multicasting serves as an efficient communication mechanisms for point to multi-point applications. It is particularly appealing for wireless networks due to the scarcity of spectrum and due to the broadcast nature of wireless communication. For example when omnidirectional antennas are used, every transmission by a node can be received by all nodes that lie within its communication range. Consequently, if the multicast group membership includes multiple nodes in the immediate communication vicinity of the transmitting node, a single transmission suffices for reaching all these receivers. This has resulted in the development of multicast functions in the 3G data network infrastructure, spurred on by group communication applications such as on-demand video streaming, group messaging and gaming through hand-held wireless devices. 3G standard bodies 3GPP and 3GPP2 have been actively standardizing multicast services [31, 30]. Multicasting is also an active area of research in multi-hop wireless networks [3, 13, 9, 25, 6].

Multicasting in wireless ad hoc network is however not free from the overhead of having to maintain the multicast tree the links of which may break due to node mobility. Also the lack of central coordination or authority that can keep track of the node mobility (and hence topology changes) results in communication and power overheads just for computing and establishing multicast trees. These overheads must be traded off against the spectrum efficiency and throughput gains that result from using multicast over unicast. Most of the previous work [3, 13, 9, 25] is for design of efficient protocols for multicasting in ad hoc networks to minimize the overheads outlined above. In another line of research that is present in the work of [33, 8, 27, 32] the main objective is to compute multicast trees that minimize the total energy consumption for each unit of data transmission. However, not much work has been done in characterizing the maximum throughput gains that can be achieved over unicast by using multicast in wireless ad hoc networks. In this paper we explore this fundamental throughput optimization problem.

A fundamental problem in multi-hop wireless networks is that the throughput per node in a multi-hop wireless networks with n nodes scales as $O(1/\sqrt{n})$ [17, 26]. Multicast is very appealing in enabling efficient communication between groups of nodes while minimizing the spectrum usage. In this paper, we consider the problem of optimizing the throughput for multicasting in a class of multi-hop wireless networks. The communication model we consider is the same as [1, 23, 2]. The main constraint imposed in this model is that a node can not be involved in multiple communications at the same time. We refer the violation of this constraint as *primary interference*. As a result, in this model, the problem of computing the multicast tree(s) for which the throughput is maximized must also explicitly deal with constraints imposed by link scheduling and hence is very different from the multicasting problems considered in the literature. A solution to our problem must include not just the multicast tree(s) but also link schedules that satisfy these constraints. In this paper, we consider a system free of secondary interference (if links which do not share a common node may not transmit at the same time, then we say, the system has secondary interference). For our future work, we will extend our solution to systems such as IEEE 802.11 which has secondary interference. Our results in this paper serves as upper bounds for systems with secondary interference.

The wireless media and MAC layer may or may not be capable of supporting broadcast. An example for a network that supports broadcast is where, each node transmits using a fixed frequency and omnidirectional antennas are used (in this case, a node is subject to secondary interference). An example for a network that does not support broadcast is where, each node is configured to receive at a particular CDMA code. If broadcast is supported then a single transmission suffices for sending a piece of data to all the downstream nodes in a multicast tree. Otherwise if only unicast is supported then a separate transmission is sent for each downstream node in a multicast tree for each data item. We consider both these possibilities in our model.

Given a particular source node, that wants to communicate with a given set of receivers, the objective of this paper is to determine the maximum rate at which the source can multicast to the receivers. We consider particular node deployment settings rather than random node deployment. Traditionally the source and the receivers are connected with a single multicast tree. Hence data arrives from the sender to each receiver on a single path. However, the multicast throughput can be improved if data from the sender to the receiver is split over multiple paths. This corresponds to splitting the data and sending it over multiple multicast trees. In this case the sender and receivers may be connected in more than one multicast tree. This has been considered in the overlay network context [5, 15]. We consider both these possibilities in our model.

Computing a throughput-optimal multicast tree can be done easily in wired networks (not the overlay setting). This is accomplished by adding links in order of non-increasing bandwidth until in the resultant network the source is connected to all receivers. Any spanning tree of the resulting network can be used as a multicast tree and achieves a throughput of f_{min} where f_{min} is the bandwidth of the last link added. This is because the rate of a multicast tree (without the scheduling constraints) is the smallest bandwidth link in the tree. Note that any optimal multicast tree cannot have higher than f_{min} throughput since by construction links of bandwidth strictly greater than f_{min} do not connect the source to all receivers. However, the problem is much harder in the wireless context due to the scheduling constraint where a node can not be involved in multiple communications at the same time. In this paper, we show that the problem is NP-hard in both the unicast and broadcast model, even for a geometric setting and where the link rates (bandwidth) are a function of the link distance. In particular we show that there does not exist a polynomial time algorithm that can always find a solution that achieves at least $2/3$ of the optimal rate in the unicast model and half of the optimal rate in the broadcast model. For the unicast model, we give a 5 approximation algorithm to compute a near optimal multicast tree where nodes have the same transmission power. We then show how we can extend to the case where nodes can have different transmission power. For the broadcast model, we give a simple 2-approximation algorithm. Note that this algorithm is the best possible polynomial time algorithm in terms of estimating the worst case throughput. We then consider the case where multiple multicast tree can be used for one session. We show that the problem for the unicast model is APX-hard and give a 1.5ρ-approximation algorithm where ρ is the best approximation ratio of the minimal cost Steiner tree problem. Our results for the broadcast model indicates that the problem may be hard to approximate within a factor better than $log(n)$. All our results except noted in the unsplittable unicast model, apply in the general setting where transmission power of nodes can be different. The focus of this paper is on a single multicast session; we assume only primary interference and all our algorithms are centralized.

The rest of the paper is organized as follows. We motivate our problem using simple examples in Section 2. We state our model and formulate our problem in Section 3. For the unsplittable case, we present hardness results and approximation algorithms for the unicast model and the broadcast model in Section 4 and 5 respectively. We study the splittable case in Section 6. We evaluate our algorithms in Section 7. We give related work in 8. We present our conclusions and discuss future work in Section 9.

2. MOTIVATION

For point to multi-point communication in wireline networks using multicast is evidently more efficient than the brute force approach of sending the same information from the source individually to each of the receivers. In this section we show that this is also the case for wireless ad hoc networks in spite of fundamental communication constraints, such as a node can not be involved in multiple communications (broadcast or unicast) at the same time. We show that these gains hold even if the underlying wireless media can only support unicast. We also show the advantage of having multiple multicast trees over single multicast trees.

Illustrating the benefit of multicast over unicast for wireless ad hoc networks-the single tree case: In Figure 1, suppose u wants to send the same data to

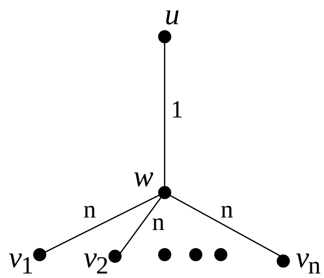

Figure 1: Benefit of Multicast Transmission Over Unicast Transmission

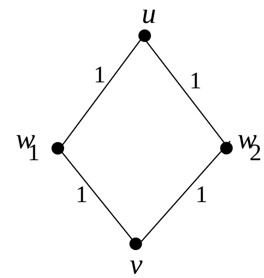

Figure 2: Benefit of Multiple Trees over a Single Tree

$v_1, v_2, \cdots, v_n$. Link (u, w) has a data rate of 1 unit. All other links have a data rate of n units. If unicast transmission is used to emulate multicast, then the same data has to traverse link (u, w) n times. The achievable (multicast) throughput between u and each v_i is therefore at most $1/n$.

If however even when the underlying wireless media does not support broadcast but when communication happens over a single multicast tree and the fundamental wireless communication constraints are imposed, the multicast session throughput is at least $1/2$. This may be achieved as follows. Consider a schedule over a sequence of time slots (any node is involved in at most one communication in each time slot) where only link (u, w) is active in a contiguous sequence of n slots followed by the next n time slots activating only the i-th link (w, v_i) in the i-th slot. Since each data item is sent only once over link (u, w) the claimed bound is obtained. When the underlying wireless media is capable of broadcast the multicast throughput can be as large as $n/(n+1)$. This is because now all the links (w, v_i), $\forall\ i = 1, 2, \cdots, n$ can be active simultaneously in a single slot.

A fundamental observation: The above example shows that the achievable unicast throughput can be arbitrarily worse than the multicast throughput for point to multipoint communications in wireless ad hoc networks. We now show that this observation holds in almost all uniform rate wireless ad hoc networks, even if unicast is allowed to split the flows between a sender and receiver over multiple paths. Consider a network in which all links have the same rate R. Due to the fundamental constraint that in the unicast mode of communication, only one outgoing link incident on the node can be active at any given time [23] the total flow going out of the links incident on the source node is at most R. Hence the total throughput out of the source is at most R. Since the source has to send one copy per receiver the optimal throughput of at least one receiver is upper bounded by R/n where n is the number of receivers. It can be seen that the multicast rate, even when the underlying wireless media only supports unicast, is at least R/d where d is the degree of any multicast tree. Note that for bounded degree networks this immediately implies a big gap between the achievable point to multipoint communication throughput in the two model. In unbounded degree networks, since there are a large number of links, it is highly likely that a multicast tree of small degree exists thus also implying a big gap[1].

[1] Note that the above simple relationship does not hold for non-uniform rate network. Nonetheless, we observe a big gap in our simulation in Section 7.

The above observation suggests that these big throughput gaps are likely, however one natural question is whether we can find efficient multicast trees that approximate the optimal throughput. Note that in the uniform link rate case this problem is one of minimizing the degree of the multicast tree, a well studied problem. However in non-uniform link rate case (especially when the rates are distance dependent as stipulated by the underlying wireless channel) some more innovations are needed both in determining bounds on how close it is possible to approximate the optimal solution and in designing efficient algorithms that approximate the optimal solution well. This work addresses these questions and others regarding the benefit of multiple multicast trees over single multicast tree.

Illustrating the benefit of multiple multicast tree over single multicast tree: We see that multicast has a clear advantage over unicast for point to multi-point communication. We now show the advantage of using multiple multicast trees. For simplicity, we use a single receiver. In Figure 2, u wants to send data to v. All links have a data rate of 1 unit. If we restrict multicast to be a single tree (in this case, it is a path), then the achievable throughput is at most $1/2$ unit. This is because if for instance path u, w_1, v is used for multicasting, then at least one of the links (u, w_1) and (w_1, v) is activated at most half of the time. If we allow multiple multicast trees, then the achievable throughput is at least 1 unit. This can be achieved as follows. The data to be sent is split into two portions of equal size and the portions are routed over two multicast trees (paths in this case): u, w_1, v and u, w_2, v. Consider a sequence of slots such that link (u, w_1) and link (w_2, v) are activated in odd slots; and link (u, w_2) and link (w_1, v) are activated in even slots. Hence the source is able to send data at the rate of one unit and the receivers also receive data at the same rate. It is easy to see that we can replicate the structure to create examples for more than one receivers where multiple multicast trees have a clear advantage.

3. MODEL AND PROBLEM FORMULATION

We consider a multi-hop wireless network with n nodes. The nodes communicate with each other via wireless links. Each node in the network can communicate directly with a subset of the other nodes in a network. Any time a node u can transmit *directly* to node v, we represent this fact by a directed edge (link) from node u to node v. We represent the nodes in the network and the possible communications among nodes by a directed graph $G = (V, E)$. Here V represents the set of nodes in the network and E the set of directed edges (links) in the network. We assume that there are m links in the network. We do not require that the links be bidirectional. We assume we are given a set of nodes $\Upsilon \subseteq V$ where one node $s \in \Upsilon$ is designated as the source and the other nodes in Υ are designated as receivers. We assume that system operates in a synchronous time-slotted mode. For a system that operates in an asynchronous mode, the results in this paper can provide upper bounds on its performance. We use $N(v)$ to denote the set of links incident on node v. Let $\sigma(e)$ represent the path loss on link e. This path loss depends on the distance $d(e)$ of the receiving end from the sending end of link e. Specifically $\sigma(e) = \frac{1}{d(e)^\beta}$ where $\beta \in [2, 4]$ [29]. Thus if $P(e)$ is the transmitted power used by the node at the sending end of link e then the received power at the node on the other end is $P = \frac{P(e)}{d(e)^\beta}$.

3.1 Interference Model

The interference model that we consider is a synchronous time-slotted joint TDMA/CDMA system where the nodes use unique signal sequences. This is one of the systems considered in [11, 18]. Another view of this system is to view the communication as taking place in a spread spectrum mode. We assume that the hopping sequences for different link transmissions in the same neighborhood is designed so that there is no interference between different link transmissions in the same time slot. However, we require that a given node can only be involved in at most one communication in any time slot. Note that this communication may be a broadcast. If a solution requires a node to participate in multiple communications in the same time slot then we say that the solution has a *primary conflict*. Therefore we are required to construct primary conflict-free schedules. We assume that a total bandwidth of W Hertz is available for use in this multi-hop network. Since we assume that two primary conflict-free link transmissions do not interfere with each other, we assume that each link transmission uses the entire bandwidth W. We also assume that each link can be modeled as an Additive White Gaussian Noise (AWGN) channel [10] with a noise spectral density N_0 so that the theoretical upper bound on the rate $R(e)$ obtained at the receiver for link e, where the sender for link e transmits at power e is given by

$$R(e) = W \log_2(1 + \frac{\sigma(e)P(e)}{N_0 W}) \text{ bits/second.}$$

Recall that $\sigma(e) = \frac{1}{d(e)^\beta}$. Thus

$$R(e) = W \log_2(1 + \frac{P(e)/d(e)^\beta}{N_0 W}) \text{ bits/second.}$$

3.2 Problem Formulation

The Multicast Throughput Optimization Problem (MTOP) is divided into 4 categories: based on whether the underlying physical (wireless) media supports broadcast or not and whether the solution is a single multicast tree or multiple multicast trees (the former refers to unsplittable multicast and the latter to splittable multicast). From here on we will refer to these problems by the abbreviation YXMTOP where X is B or U for broadcast or unicast and Y is U or S for unsplittable or splittable multicast. Thus for example UBMTOP refers to the problem of computing the best single multicast tree when the physical media supports broadcast. A solution to the MTOP problem is designated by a collection of tuples $(S_1, T_1), (S_2, T_2), \ldots (S_p, T_p)$. Here T_i is a multicast tree and each $s_i(t) \in S_i$ is a subset of links of T_i for every time slot $t = 0, 1, 2, \ldots$ such that in each slot t the set of links $\cup_i s_i(t)$ have no primary conflicts. Let $s(t) = \cup_i s_i(t)$ and S denote the link schedule over time: $s(0), s(1), s(2), \ldots$. We assume that S is cyclic with some bounded cycle length. Let γ denote the set of links in $\cup_i T_i$. Thus we can define for every link $e \in \gamma$ its frequency $f(e)$ in S, defined as the fraction of slots in which link e is present in S. For the UBMTOP or the UUMTOP problem (here $p = 1$) the maximum multicast throughput is then given by $\min_{e\in\gamma} f(e)R(e)$. Recall that $R(e)$ is the maximum possible rate for link e. Likewise for the SBMTOP or the SUMTOP problem we define the multicast rate R_i of a Tree T_i, $1 \le i \le p$ to be $R_i = \min_{e\in T_i} f_i(e)R(e)$, where $f_i(e)$ is the fraction of slots in which link e is present in S_i. Then for these problems the maximum multicast throughput is given by $\sum_i R_i$. This is because data from the source can arrive independently over these p trees at rates $R_1, R_2, \ldots R_p$ respectively to each receiver.

4. THE UNSPLITTABLE UNICAST MODEL

In this section we show that the UUMTOP cannot be approximated to a factor better than $\frac{3}{2}$ (66.66% of the optimal). When all the nodes of the ad hoc network use the same power we provide an efficient polynomial time algorithm for estimating the maximum multicast throughput to within a factor of 20% of the optimal. We then extend our algorithm to the case where nodes can have different transmission powers. In practice we believe that the performance of the algorithm is much better than its worst case behavior.

To illustrate our ideas we first show the hardness result in an abstract model where the link rates are independent of the distance and the transmit power. In this abstract model we also design an efficient approximation algorithm whose performance achieves the best possible bound, as determined by our hardness results. We then extend these results to the more realistic setting where the link rates are determined based on distances for an Additive White Gaussian Noise (AWGN) channel (as described in Section 3). For the same setting we then design a 5-approximation algorithm when all nodes have the same power. Finally, we show how we can extend our results to the case where nodes can have different transmission power. We consider a realistic heterogeneous ad hoc network setting as an example and give a 24-approximation algorithm.

Before we delve into the results, we first give some useful definitions and facts.

Definition 1. *Degree of a graph G is the maximum number of links incident on any node of G.*

Remark 1. **Undirected graph:** *In this section we assume that all nodes have the same power. All our approximation and hardness results apply under this restriction. Note that under this assumption the two directed links between any two nodes are symmetric (have the same rate). Hence in this section we assume that the underlying graph for the given ad hoc network is undirected.*

Remark 2. **Unit disk graphs [7]:** *Since all nodes are assumed to have the same power P they should all have the same range. Thus we can assume that there is a critical* *distance D such that there is a link $e \in E$ between two nodes if and only if they are at most D distance apart. Thus the set of ad hoc network considered here include all unit disk graphs for all values of D.*

Recall that a solution to the MTOP problem is designated by a collection of tuples $(S_1, T_1), (S_2, T_2), \ldots (S_p, T_p)$. Here T_i is a multicast tree and S_i is the schedule of its links. Note that in the unsplittable model $p = 1$. Hence in this section we will denote the solution by a single tuple (S, T).

REMARK 3. **Primary conflict free schedules in the unicast model:** *Note that if S is a primary conflict free schedule of links of a tree T of G over time (with $s(t) \in S$ the primary conflict free schedule for time slot t), then $s(t) \in S$ the schedule at time slot t can contain at most one link of T incident on any node v. Thus if $f(e)$ is the fraction of slots of S in which link e is present then for any node v we must have $\sum_{e \in N(v) \cap T} f(e) \leq 1$. Recall that $N(v)$ is the set of links incident on node v. In particular at least one link e must satisfy $f(e) \leq 1/\delta$ where δ is the degree of T.*

We will rely on the following result [24] about edge coloring of trees.

THEOREM 1. *[24] Any tree with degree δ can be edge colored with δ colors.*

The above result of [24] is more general and applies to any bipartite graph and even if there are parallel links.

REMARK 4. *The above result is relevant for the UUMTOP since it implies that when all the link rates are at least (equal to) R then for any multicast tree T the maximum multicast throughput is at least (exactly) R/δ, where δ is the degree of T. This is because by Remark 3 this rate is $\min_e Rf(e) \leq R/\delta$ if all link rates are R. Also if C is a δ edge coloring of T then a primary conflict free schedule S can be defined by scheduling every color class (the links therein) of C once every δ time slots. Note that in this schedule S, link e is present in at least $1/\delta$ fraction of the slots. Thus implying for this S the multicast throughput of T is at least R/δ.*

4.1 Tight Results for Single Rate Ad Hoc Networks

In this section, we consider an abstract model where all the link rates are equal to a common rate R. We show that there does not exist any polynomial algorithm that can estimate the multicast rate better than 2/3 of the optimal in the worst case. We then give an approximation algorithm that estimate the multicast rate to within 2/3 of the optimal in the worst case. Therefore, our results are tight for this special case.

LEMMA 1. *In the abstract model the maximum multicast throughput for the UUMTOP cannot be approximated to within 66.67% of the optimal multicast rate unless P=NP.*

PROOF. We show a reduction from the Hamiltonian Path problem which is known to be NP hard even for points in the plane and even for unit disk graphs [14, 19]. Given an instance $G = (V, E)$ of the Hamiltonian Path problem we create an instance of the UUMTOP in the abstract model by setting the rates of all links $e \in E$ to a common value R. We require a multicast session consisting of all the nodes $v \in V$, one of them is designated to be the sender and all others are designated as receivers. We claim that the optimal multicast throughput of this instance of the UUMTOP is $R/2$ if and only if there is a Hamiltonian path in G. In fact we can claim something stronger that in case there is no Hamiltonian path in G then this rate is at most $R/3$. This however implies that we cannot compute the maximum multicast throughput for the UUMTOP to within 66.67% of the optimal multicast rate (unless $P = NP$), because otherwise we will be able to tell whether G has a Hamiltonian path in polynomial time. We now establish our claims. Let G have a Hamiltonian path T. Then we can use T as the multicast tree for the UUMTOP instance. Note that T has degree 2. Thus by Remark 4 the multicast rate of T is $R/2$. If G does not have a Hamiltonian path then any multicast tree of G must have degree at least 3. Hence the maximum achievable multicast throughput is at most $R/3$, thus establishing the result. □

Now we show an algorithm that can compute the maximum multicast throughput for the UUMTOP in the abstract model to within 66.66% of the optimal multicast rate. Note that if we can find a multicast tree of the given wireless ad hoc network G of degree δ then we have a solution (see Remark 4) that achieves rate R/δ. Thus we need to find a multicast tree of minimum degree of G. Note that this is a hard problem but however it can be approximated to within an additive factor of plus one of the optimal. More specifically given a graph G and a set of terminals, a steiner tree of degree $\delta+1$ can be found in polynomial time where δ is the lowest degree of any steiner tree of G [12]. Using this algorithm we can estimate the maximum multicast throughput for this problem to within a fraction $\frac{\delta}{\delta+1} \geq 0.666$, since δ the degree of the optimal multicast tree has to be at least 2. Thus we can claim:

LEMMA 2. *In the abstract model the maximum multicast throughput for the UUMTOP can be approximated to within 66.66% of the optimal multicast rate.*

From Lemma 1 and Lemma 2, we know that the algorithm in [12] is the best possible polynomial time algorithm in terms of worst case performance unless $P = NP$.

4.2 Hardness Results for the AWGN Channel Model

We extend our hardness result to the Additive White Gaussian Noise (AWGN) channel model(as described in Section 3). In our proofs below we will assume that all nodes have power P. Note that this also implies that the underlying graphs are unit disk graphs (see Remark 2) for all possible values of D. Recall that if $d(e)$ is the length of a link (distance between its end-points) then the rate of this link $R(e)$ is given by $R(e) = W \log_2(1 + \frac{P/d(e)^\beta}{N_0 W})$. Since P, N_0, W are fixed we can write this as $R(e) = W \log_2(1 + C_1/d(e)^\beta)$ for some fixed value C_1. In the following we will use d_{max} and d_{min} to denote the largest and smallest link lengths ($\max_{e \in E} d(e)$ and $\min_{e \in E} d(e)$) respectively. In all our proofs we can safely assume that $d_{min} > 0$.

THEOREM 2. *In the AWGN model, the maximum multicast throughput for the UUMTOP cannot be approximated to within 66.67% of the optimal multicast rate unless P=NP.*

PROOF. We use the same reduction as in proof of Lemma 1 from the Hamiltonian Path problem for points in the plane (for the unit disk graph). Let the given graph have a Hamiltonian path T. Then when we use T as the multicast tree for the UUMTOP instance then the minimum multicast throughput we can get is at least

$$\frac{W}{2}\log_2(1+\frac{C_1}{d_{max}^{\beta}}).$$

This follows since in the worst case all links in T may have rate $W\log_2(1+\frac{C_1}{d_{max}^{\beta}})$ and since every node has degree 2 in T. On the other hand if the given graph does not have a Hamiltonian path then any multicast tree T must have degree 3 and hence the maximum multicast throughput is at most

$$\frac{W}{3}\log_2(1+\frac{C_1}{d_{min}^{\beta}}).$$

This is because in the best case all links of all nodes of degree 3 in T may have rate $W\log_2(1+\frac{C_1}{d_{min}^{\beta}})$. Note that if

$$\alpha\frac{W}{2}\log_2(1+\frac{C_1}{d_{max}^{\beta}}) \geq \frac{W}{3}\log_2(1+\frac{C_1}{d_{min}^{\beta}})$$

then any approximation algorithm that can estimate the maximum multicast throughput for the UUMTOP within μ fraction ($\mu > \alpha$) of the optimal throughput can be used to tell if there is a Hamiltonian path in G. Hence such an approximation algorithm cannot exist unless $P = NP$. We now show that it is possible to scale the distances in G such that $\alpha = \frac{2}{3}$. This therefore implies the claimed bound.

Note that one can scale all distances $d(e)$ to $kd(e)$ by scaling the coordinates of the location of all nodes $v \in V$ by multiplying them by k. We also adjust D the unit disk distance to kD (see Remark 2). Note that, the received signal strength and noise from other transmissions will also scale so that the signal to noise ratio will not change (here we assume that the additive white noise is negligible). Therefore, there is no change in the links in the graph. This scaling therefore does not affect any paths in the graph and in particular the Hamiltonian path. In the new graph the new values for d_{max} and d_{min} are kd_{max} and kd_{min} respectively. Let

$$k = \frac{C_1^{\frac{1}{\beta}} d_{min}^2}{\theta d_{max}^3}$$

for some large value θ. Note that then

$$\frac{\frac{W}{3}\log_2(1+\frac{C_1}{(kd_{min})^{\beta}})}{\frac{W}{2}log_2(1+\frac{C_1}{(kd_{max})^{\beta}})} = \frac{\log_2(1+p^{3\beta}\theta^{\beta})^2}{\log_2(1+p^{2\beta}\theta^{\beta})^3}$$

where $p = \frac{d_{max}}{d_{min}}$. For large values of θ this ratio approaches $4/6 = 2/3$ since

$$\frac{\log_2(p^{6\beta}\theta^{2\beta})}{\log_2(p^{6\beta}\theta^{3\beta})} = \frac{\log_2 p^{6\beta} + 2\beta\log_2\theta}{\log_2 p^{6\beta} + 3\beta\log_2\theta}.$$

From this it follows that $\alpha = \frac{2}{3}$. □

4.3 A 5-Approximation Algorithm For Ad Hoc Networks with Uniform Transmission Power

Now we design an approximation algorithm for the UUMTOP for unit disk graphs. Given a unit disk graph G, the algorithm runs a Kruskal like minimum spanning tree algorithm by sorting the links based on their lengths. It then considers the links in non-decreasing order of their length. It starts out with an empty tree T. At any time when considering link e of G, if link e does not form a cycle in T it adds e to T. The algorithm pauses at the first instance when all the terminals (multicast receivers and sender) are in one connected component of the tree T formed so far. Let the last link e to be added at this moment has length $d(e)$. The algorithm then continues the Kruskal algorithm for all remaining unconsidered links of length $d(e)$. At the point when the Kruskal algorithm stops, let $V' \subseteq V$ be the set of vertices in the connected component of the tree T that contains all the terminals. Let $G' = (V', E')$ be the subgraph of G that is induced by the vertices V' (contains all the links of G that are between pairs of nodes in V'). The algorithm then finds a degree 5 minimum spanning tree (cost is link length) of G', using a result of [28]. This tree is output as the multicast tree of G. We first show that the algorithm can be implemented in polynomial time and then we will show its performance bound.

LEMMA 3. *The algorithm outputs a multicast tree of G of degree at most 5 in polynomial time.*

PROOF. Consider the Euclidean graph over the vertices V' where there is a link between every pair of points $u, v \in V'$ of length equal to the Euclidean distance between u and v. Note that the links of G' are a subset of the links in this Euclidean graph. Recall that by construction all nodes (V') in this Euclidean graph can be connected into one component by just using the links of length $d(e)$ or less. Consider any (MST) minimum spanning tree (cost is link length) of this Euclidean graph. It cannot have any link of length exceeding $d(e)$, because if (u, v) is one such link then consider the two connected components of the MST obtained by removing link (u, v) from the MST. There must be a link of length at most $d(e)$ connecting these connected components in the Euclidean graph because by construction all nodes in V' can be connected into one component by using just the links of length $d(e)$ or smaller. Let (x, y) be one such link. Then by adding (x, y) to the MST and by removing link (u, v) in the resulting cycle we get another MST of less cost a contradiction. Thus any MST of the Euclidean graph must only include the links of G'. By the result of [28] a MST of any Euclidean graph of degree at most 5 exists and can be computed in polynomial time. Let T be one such tree. Then T is a spanning tree of G' and hence a multicast tree of G and has degree at most 5. Note that T is output by our algorithm.

□

THEOREM 3. *The multicast throughput of the tree output by the algorithm is at least 20% of the optimal multicast throughput.*

PROOF. Let $d(e)$ be the length of the last link added by the Kruskal algorithm at which point all the terminals (multicast receivers and sender) are in one connected component for the first time. This implies that it is not possible to connect all the terminals using only the set of links of length strictly less than $d(e)$. Hence the optimal multicast tree must use a link of length at least $d(e)$. Thus the optimal multicast rate is at most $R_{OPT} \leq W\log_2(1 + C_1/d(e)^{\beta})$.

Note that by Lemma 3 the multicast tree output by the algorithm has only links of length at most $d(e)$ and it has degree at most 5. Hence the rate of all links in this tree is at least $R \geq W \log_2(1 + C_1/d(e)^\beta)$. By Remark 4 this implies that the multicast throughput of the tree output by the algorithm is at least

$$\frac{W}{5} \log_2(1 + \frac{C_1}{d(e)^\beta}) \geq \frac{R_{OPT}}{5}$$

Note also that the method given in Remark 4 can be used to compute the associated primary conflict free schedule in polynomial time, thus establishing the result. □

4.4 Extension to Ad Hoc Networks with Non-uniform Transmission Power

For ease of explanation, we illustrate our algorithm using a realistic ad hoc network setting. We assume $\lambda \leq P_{max}/P_{min} \leq 16$ where P_{max} is the maximum transmission power and P_{min} is the minimal transmission power, and the path loss exponent $\beta = 4$. This power setting is realistic, e.g. cisco350 AP's maximum transmission power is 100mW and minimal transmission power for low-power setting is typically above 10mW.

For ease of presentation we omit the details and only present a sketch of our arguments for the algorithm and its analysis. Our algorithm has two steps. In the first step we use a binary search to find the minimal rate R_c such that the graph G'_c constructed below connects all the terminals of the multicast session. For a given R_c, let d_c be the distance such that $R_c = W \log_2(1+P_{min}/(N_0 W d_c^4))$. We construct a graph $G_c = (V, E_c)$ where an edge exists between two nodes iff the distance between them is no greater than d_c. If G_c connects all the terminals, then we complete the first step. If not, we add all the remaining edges such that the data rate of these edges is no smaller than R_c. Call the resulting graph G'_c. If G'_c connects all the terminals, we complete our binary search. In the second step, we run the 5-approximation algorithm for each connected component in G_c. As a result, we get a tree T_i for each connected component i. Let K be the number of connected components. We then add edges in G'_c to connect $T_i, \forall i = 1, 2, \cdots, K$ into one tree T. We observe that the optimal multicast rate $R_{OPT} \leq R_c$ since it is not possible to connect all the terminals using only links with rate larger R_c. We also observe that there are at most 19 neighboring components (with $\lambda \leq 16$ and $\beta = 4$, all edges in G'_c are of length less than $2d_c$ and nodes of any two distinct connected components are at least d_c apart. Z. Gaspar and T. Tarnai in [16] gives an upper bound 19.). Observe that G_c is a Euclidean graph, thus the maximal node degree of any T_i is at most 5. Therefore, the maximal degree of the resulting multicast tree T is at most 24. By Remark 4, the throughput of this algorithm is at least $R_c/24$. In other words, this algorithm is 24-approximation algorithm for this realistic setting of ad hoc networks. It is easy to see that the algorithm can be extended to other power settings. All that changes is the approximation factor.

5. THE UNSPLITTABLE BROADCAST MODEL

In this section we show that the UBMTOP cannot be approximated to a factor better than 2 (50% of the optimal). We also provide a matching upper bound by designing an efficient polynomial time algorithm for estimating the maximum multicast throughput to within a factor of 50% of the optimal. In practice we believe that the performance of the algorithm is much better than its worst case behavior. We show our results for the setting when the links may have arbitrary rate setting. Our results also hold in the more realistic setting where the link rates are distance dependent as in AWGN channel. However, for lack of space we omit the details. Unlike the unicast model where at most one link incident on a node can be active, in the broadcast model, all outgoing links of a node can be active. However, the same data must be sent to each of the downstream receivers at the same time. We assume a node can choose a subset of downstream receivers to receive a transmission. Other downstream nodes can receive a different transmission from a different sender at the same time. This capability can be achieved by beam-forming directional antenna or each node is scheduled to receive a transmission with a predetermined CDMA code. We remark that we do not claim that this model will be used in practice. However, results obtained from this model serve as a upper bound for practical wireless networks such as 802.11 multi-hop wireless networks when we consider primary interference only. For our future work, we would like to incorporate secondary interference in our model.

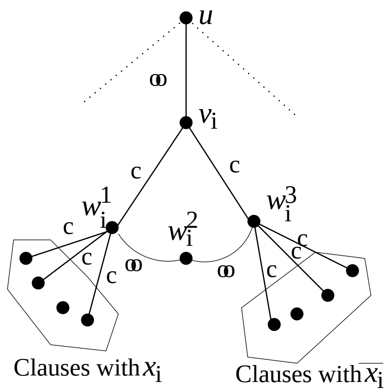

Figure 3: The UBMTOP problem is not approximable within a factor of 2

5.1 Hardness Results

THEOREM 4. *The throughput of the UBMTOP problem can not be approximated to a factor strictly better than 50% of the optimal multicast rate unless P=NP.*

PROOF. We show that the UBMTOP problem is NP-hard via a reduction from 3SAT. Given an instance of a 3SAT problem, we create a gadget as illustrated in Figure 3. There is a source node u (sender). For each variable x_i and its negation $\bar{x_i}$, we create a subtree rooted at v_i. We create three other nodes ${w_i}^1$, ${w_i}^2$, and ${w_i}^3$. We connect v_i to ${w_i}^1$ and ${w_i}^3$. We connect both ${w_i}^1$ and ${w_i}^3$ to ${w_i}^2$. We create a node for each clause and set these nodes to be receivers. For each clause node with variable x_i, we connect it to ${w_i}^1$. For each clause node with variable $\bar{x_i}$, we connect it to ${w_i}^3$. We connect the source node u to each v_i. All links have a rate of c except the three links $(u, v_i), ({w_i}^1, {w_i}^2)$ and $({w_i}^2, {w_i}^3)$. These three links have a rate of infinity (much larger than

c). We claim that a truth assignment to the given 3SAT formula exists if and only if there exists a multicast tree that achieves an optimal throughput of c. Let T be an optimal tree that achieves this rate. Let v_i be a node such that some clause node uses v_i to get its data from u. Note that in T node u must broadcast to node v_i using link (u, v_i). Also note that in T node v_i must broadcast to at least one node ${w_i}^1$ and ${w_i}^3$. In other words T must contain at least one of the two links $(v_i, {w_i}^1)$ and $(v_i, {w_i}^3)$. Note that if T contains link $(v_i, {w_i}^1)$ then node ${w_i}^1$ cannot broadcast to any clause nodes connected to it. This is because node ${w_i}^1$ can only be involved in at most one communication at any given time (either broadcasting or receiving data on link $(v_i, {w_i}^1)$ and hence the achieved multicast throughput would be $c/2 < c$. This implies that if T contains link $(v_i, {w_i}^1)$ then it does not contain any link that connects ${w_i}^1$ to any clause node with variable x_i. Likewise it can be shown that if T contains link $(v_i, {w_i}^3)$ then it does not contain any link that connects ${w_i}^3$ to any clause node with variable $\bar{x_i}$. Thus T contains exactly one of the two links $(v_i, {w_i}^1)$ and $(v_i, {w_i}^3)$. Note that if it contains link $(v_i, {w_i}^1)$ then it must also contain links $({w_i}^1, {w_i}^2)$ and $({w_i}^2, {w_i}^3)$ and node ${w_i}^3$ can cover (broadcast) all the clause nodes connected to it. This is equivalent to $\bar{x_i}$ being true. Likewise the other case corresponds to x_i being true. Hence given the tree T we can find a satisfying assignment. By reversing the process, given a satisfying assignment we can construct the tree T that achieves a multicast rate of c. For instance if in the satisfying assignment $\bar{x_i}$ is true then the tree T uses the path $u, v_i, {w_i}^1, {w_i}^2, {w_i}^3$ and in addition contain links joining node ${w_i}^3$ to some of the clause nodes connected to it (that are not covered by any other variable).

□

5.2 A 2-Approximation Algorithm

We sort the links by rate. We add one link at a time in non-increasing order until the source connects to all receivers. Denote this subgraph as G'. We denote the rate of the edge last added as R^*. We then find a tree T in G' by pruning all the unnecessary edges.

THEOREM 5. *The multicast throughput of the tree output by the algorithm is at least 50% of the optimal multicast throughput.*

PROOF. Let the source in the tree T be at level 0. A node is at level i if it is i hop away from the source in T. Let the maximal levels in T be h. In a primary conflict free schedule for the links of T nodes at alternate levels are scheduled to broadcast at the same time. Thus a set of links L_1 is active in one slot followed by the remaining links L_2 in T in the next slot, and this schedule is repeated forever. Thus in this primary conflict free schedule each link e is active in half of the slots ($f(e) = 0.5$). Let R_1 be the minimum rate among links in L_1 and R_2 the minimum rate among links in L_2. Note that by construction $R_1, R_2 \geq R^*$. Thus the multicast rate for this schedule is $\min_{e \in T} f(e)R(e) \geq R^*/2$. Note that the optimal multicat tree must contain a link of rate at most R^* since links with rate strictly more than R^* cannot connect source to all receivers. Thus the optimal multicast rate can not be better than R^*. Therefore, our algorithm is a 2-approximation algorithm. Given our hardness result, this algorithm is the best possible in terms of worse case. □

6. THE SPLITTABLE MODEL

In the splittable model we allow the data from the sender to the receiver to arrive on multiple paths. This is equivalent to the sender using multiple multicast trees to send its data. Recall that a solution to these problems is a collection of tuples $(S_1, T_1), (S_2, T_2), \ldots (S_p, T_p)$. Here T_i is a multicast tree and each $s_i(t) \in S_i$ is a subset of links of T_i for every time slot $t = 0, 1, 2, \ldots$ such that in each slot t the set of links $\cup_i s_i(t)$ have no primary conflicts. We let $s(t) = \cup_i s_i(t)$ and S denote the links schedule over time $s(0), s(1), s(2), \ldots$. Recall that in the splittable model we can define the multicast rate R_i of a Tree T_i, $1 \leq i \leq p$ to be $R_i = \min_{e \in T_i} f_i(e)R(e)$, where $f_i(e)$ is the fraction of slots in which link e is present in S_i. Then for these problems the maximum multicast throughput is given by $\sum_i R_i$. This is because data from the source can arrive independently over these p trees at rates $R_1, R_2, \ldots R_p$ respectively to each receiver.

We now give an equivalent definition of the problem in terms of the rates that will help us formulate a linear program for solving it. Let γ denote the set of all possible multicast trees over the terminals Υ. Obviously this is a large set. Let $x_T \geq 0$ be the data rate assigned by the solution to multicast tree $T \in \gamma$. Note that our objective is to maximize $\sum_{T \in \gamma} x_T$. Let S_T be the link schedule associated with tree T. Note that for an $e \in T$ its frequency $f_T(e)$ in S_T must be at least $x_T/R(e)$. Recall that $R(e)$ is the rate of link e.

In the case that unicast is used by the underlying wireless media, then at most one link incident on a node v can be active in S in any given time slot. This translates into the following linear constraint: ($N(v)$ is the set of links incident on node v in G)

$$\sum_{e \in N(v)} \sum_{T : e \in T} x_T/R(e) \leq 1, \ \forall v \in V. \tag{1}$$

In the case of broadcast, consider a node v (which is not the sender) in a tree $T \in \gamma$ (with $x_T > 0$) Let u be the parent node of v and let $u_1, u_2 \ldots u_k$ be the children of v in T . Then in order to avoid primary conflicts the link (u, v) cannot be active with any of the links $(v, u_1), (v, u_2) \ldots (v, u_k)$ in any given time slot when broadcast is used. Here there are two possible models: when node v broadcasts on tree T the rate at which data is sent on each outgoing link $(v, u_1), (v, u_2) \ldots (v, u_k)$ is the smallest rate among the rates $R((v, u_1)), R((v, u_2)) \ldots R((v, u_k))$ or each link (v, u_i) is still able to transmit data at rate $R((v, u_i))$ (for instance when an antenna array is used for broadcast). For ease of presentation we will only consider the latter model. Our results can be extended to the other model but for lack of space we leave the details for a full paper. In the following we denote $R_T^c(v)$ to be the minimum rate of the links to the children of node v in tree T thus $R_T^c(v) = \min_{(v,u) \in T} R((v, u))$. If no edge (v, u) exists in T then we set $R_T^c(v) = \infty$. Likewise we denote $R_T^p(v)$ to be the rate of the link to the parent of node v in tree T. If no edge (u, v) exists in T then we set $R_T^p(v) = \infty$.

Thus we have the following linear constraint for broadcast:

$$\sum_{T : v \in T} (x_T/R_T^c(v) + x_T/R_T^p(v)) \leq 1, \ \forall v \in V. \tag{2}$$

We now design efficient algorithms for the two models.

6.1 The Splittable Unicast Model

Here we are interested in finding a collection of multicast trees and their schedule so as to maximize the total multicast rate, when broadcast is not allowed by the underlying wireless media. Recall that Equation (1) is a necessary condition for a schedule to be primary conflict free. However it may not be sufficient. Although it can be shown [18] that if a set of x_T values satisfy this constraint then by scaling down these values by a factor of $2/3$ we are guaranteed that a schedule that is primary conflict free can be found. However this may yield a solution which may not be optimal but is guaranteed to be within 66.66% of the optimal. The resulting primary conflict free schedule is found by an edge coloring of the resulting graph [23] and we leave the details for a full paper. This is the approach we follow. We formulate the following linear program (LP):

$$Maximize \sum_{T \in \gamma} x_T \tag{3}$$

subject to

$$\sum_{e \in N(v)} \sum_{T: e \in T} x_T / R(e) \leq 1, \forall v \in V \tag{4}$$

$$x_T \geq 0, \forall T \in \gamma \tag{5}$$

The dual of the linear program is as follows.

$$Minimize \sum_{v \in V} y_v \tag{6}$$

subject to

$$\sum_{v \in V} \sum_{e \in N(v) \cap T} y_v / R(e) \geq 1, \forall T \in \gamma \tag{7}$$

$$y_v \geq 0, \forall v \in V \tag{8}$$

Note that a feasible solution to the dual must satisfy the constraint (7). Note that this constraint can be rewritten as:

$$\sum_{e=(u,v) \in T} (y_u + y_v)/R(e) \geq 1, \forall T \in \gamma.$$

Therefore, the dual LP is the following problem: Assign nonnegative weights y_v to the nodes of the network such that any steiner (multicast) tree of the network for terminals Υ has total cost at least 1, and a linear function of the node weight is minimized. Here the cost of a link $e = (u, v)$ is defined as $(y_u + y_v)/R(e)$. Given a ρ approximation algorithm of Steiner tree, we achieve an approximation algorithm of 1.5ρ for the UUMTOP.

Theorem 6. *The LP (3) can be solved within a factor ρ in polynomial time.*

Proof. We only give a proof sketch for lack of space. The proof is very much along the line of the proof of [20]. We present the proof for completeness. The idea is to solve the LP using the ellipsoid method [22]. We do this by first turning the dual LP into an LP where we are only interested in checking for feasibility. This is done using a binary search over the possible values for the objective function (6). Specifically, we add the inequality

$$\sum_{v \in V} y_v \leq R.$$

to the dual LP, remove the objective function (6) and look for the smallest value of R for which the modified dual LP is feasible. If we had an optimal algorithm A for computing the minimum cost steiner tree the feasibility check could have been done as follows. For a given set of y_v values we can compute the minimum cost steiner tree (using the edge costs described earlier) using A. If the cost of this steiner tree is at least one then the given y_v values are feasible. Otherwise we have a steiner tree (the minimum cost steiner tree) for which the constraints are violated. This tree then becomes a separation constraint for the ellipsoid algorithm or in other words A can be used as a separation oracle for the ellipsoid algorithm on the feasibility LP.

Unfortunately an optimal algorithm A for computing the minimum cost steiner tree is unlikely (unless P = NP). Now let us assume that A is the best possible ρ approximation algorithm for computing the minimum cost steiner tree. In this case when A says that the minimum cost steiner tree is of cost at least one, then the solution may not be feasible since there is a possibility that the actual cost of the minimum cost steiner tree is only $1/\rho$. However this means that if we replace every y_v by ρy_v then we have a feasible solution to the LP. Also note that if A says that the minimum cost steiner tree is of cost less than one then we are guaranteed that the solution is not feasible. Thus if R^* is the minimum value of R for which the algorithm decides that the linear program is feasible, then we know that the modified dual linear program is infeasible for $R^* - \epsilon$ (where ϵ depends on the precision of the binary search) , and is feasible for R^*. Therefore, the optimum solution of the dual program (6) is between R^* and ρR^*.

Note that the above algorithm lets us estimate the objective function of the primal LP (3) within a factor ρ, but it doesn't give us the trees over which this solution can be obtained. This is done by using a technique of [4]. Let χ be the set of all trees which are generated as part of running the ellipsoid algorithm (for which the constraints of the modified dual are violated). Since the ellipsoid algorithm runs in polynomial time χ has a polynomial size. It can be shown that the primal LP (3) doesn't change if we set all $x_T = 0$ for all $T \notin \chi$. Thus the modified primal LP (3) now has a polynomial size and can be solved in polynomial time (using any LP solver) to find the set of x_T values (for $T \in \chi$) for which the primal LP achieves an objective value of R^*. Note that the optimal objective value of the primal is guaranteed to be at most ρR^*. Thus we can find in polynomial time a set of multicast trees and their rates x_T that satisfy the constraints of the primal LP (3) and that approximate the optimum objective value to within a factor ρ. □

Theorem 7. *There is a 1.5ρ approximation algorithm for solving the UUTOP.*

Proof. As described before by scaling down all the x_T values, obtained by solving the LP (3), by a factor of $2/3$ we are guaranteed a solution that can be scheduled and in addition the schedule can be found by an edge coloring approach. Thus overall the algorithm achieves an approximation ratio of 1.5ρ. □

Theorem 8. *The UUTOP problem is NP-hard.*

We omit the proof of the theorem for lack of space. It is along the line of the proof in [20]. Moreover it can also be shown that it is even hard to develop a polynomial time approximation scheme for it. In other words, the problem is APX-hard.

6.2 Splittable Broadcast Model

Our results for the splittable broadcast model are along the same lines as those for the unicast model. However the resulting problems in the broadcast model are much harder since they require solving a node weighted steiner tree problem (here weights are associated with nodes rather than links). This problem can be shown to be $\Omega(\log n)$ hard to approximate by a reduction from set cover [14]. We omit the details for lack of space.

7. EVALUATION

In this section, we evaluate the performance of our approximation algorithms. We place 100 nodes randomly in a 1600 meter by 1600 meter square. We generate 5 such instances. Our results are averaged over the 5 topologies. We evaluate the throughput of a single multicast session. We vary the number of receivers from 5 to 30. The data rate of a link is distance dependent. In practice, it has a few discrete values, e.g. 802.11b has 4 different rates, 3G1xEV-DO has 10 different rates. We pick 802.11b's 4 different rates in the outdoor setting. According to many vendors advertised values, the threshold distance for the 4 rates (in Mbps) 11,5.5,2,1 is 250,350,400,500 meters respectively. Many multicast protocols typically compute a shortest path tree and then prune unnecessary edges. We refer to this algorithm as SPT. We compare our algorithm with SPT. We also compare with the throughput upper bound of unicast, $11/n$ in our setting where n is the number of receivers. Due to the complexity of the splittable case, we only evaluate the unsplittable case.

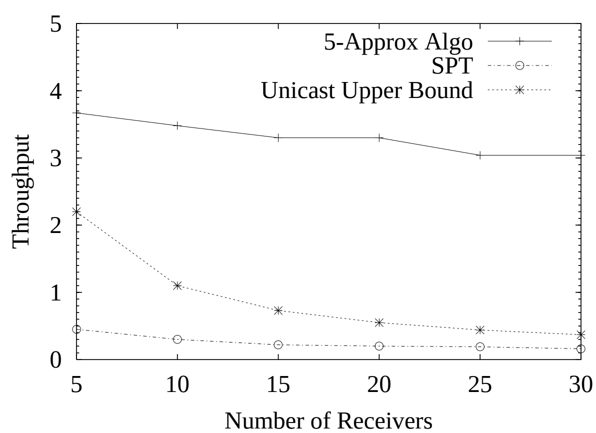

Figure 4: Comparison of the 5-approx. algorithm and SPT for unicast model and unicast upper bound

Figure 4 compares our 5-approximation algorithm and the SPT for the unicast model and the unicast upper bound. For all graphs, the throughput drops as the number of receivers increase. However, the throughput of our algorithm drops much more slowly because it tries to find low degree trees to take care of primary conflict. SPT is degree agnostic. So the maximum degree of SPT increases much faster as the number of receivers increase. Our algorithm is 7 time better than SPT when the number of receivers is 5, 18 times better when the number of receivers is 30. Our algorithm is also much better than the achievable throughput of unicast even with 5 receivers. For the broadcast model, our 2-approximation

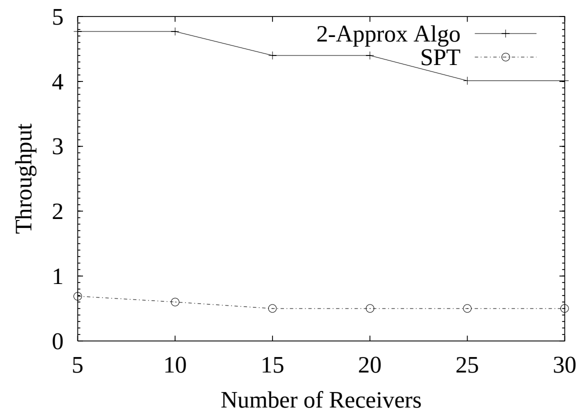

Figure 5: Comparison of the 2-approximation algorithm with SPT for the broadcast model

algorithm again performs much better than SPT as illustrated in Figure 5. When we compare the throughput of

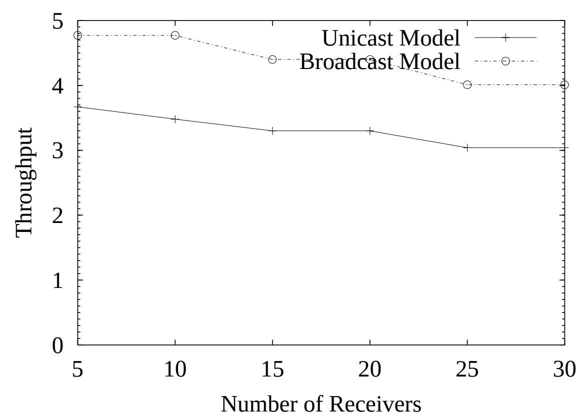

Figure 6: Comparison of the 5-approximation algorithm and the 2-approximation algorithm

the 5-approximation algorithm for the unicast model and the 2-approximation algorithm for the broadcast model (see Figure 6), we see that the latter is only 32% better. This shows that our 5-approximation algorithm is able to find multicast trees with very low degree.

8. RELATED WORK

The work that is most closely related to this paper is that of [23, 21]. Both of the two papers consider the achievable throughput for unicast transmissions between multiple source-destination pairs. Kodialam and Nandagopal [23] consider the same wireless communication constraint as ours. For a given unicast traffic pattern, they provide a polynomial time algorithm that computes routes and schedules such that the resulting throughput is at least 67% of the optimal. Jain et al. [21] consider more general wireless interference model where it can take into account interference from neighboring nodes, impact of directional antennas, availability of non-interfering channels, etc. However, they can only provide upper and lower bounds on the optimal throughput. To the best of our knowledge, we are the first to consider the problem of optimal multicast throughput in multi-hop wireless networks.

Optimal multicast throughput problem has recently been considered in the context of overlay networks [5, 15]. They increase the multicast throughput by utilizing multiple multicast trees. We also consider the case of multiple multicast tree. However, our problem is very different from theirs. In [15] a physical link of the network may be traversed multiple times by a single multicast packet since it may be contained in multiple logical links in the multicast tree. The bandwidth used for all these traversals has to be summed up and must not exceed the links capacity. This constraint does not apply in our context. However, the constraint that a node can not be involved in multiple communications does not apply to their context. As a result, the solution of the two problems are very different.

9. CONCLUSION AND FUTURE WORK

We believe we are the first to characterize the achievable throughput of multicast in multi-hop wireless networks. We consider primary interference in this paper. We consider both broadcast transmission model and unicast transmission model. We present hardness results and approximation algorithms for both the one multicast tree per session case and multiple tree per session case. Our evaluation shows that our algorithms perform very well when compared with pruned shortest path tree algorithm and the optimal unicast algorithm.

In our future work, we would like to tighten our analytical bounds. In particular, we want to investigate the hardness result for the multicast problem in the splittable broadcast model and design better approximation algorithm for it. We would also like to incorporate secondary interference in our algorithm. In addition, we want to explore the joint optimization problem when there are multiple active multicast sessions in the network.

10. REFERENCES

[1] D. Baker, J. Wieselthier, and A. Ephremides. A distributed algorithm for scheduling the activation of links in a self-organizing mobile radio networks. In *Proceedings of IEEE Int. Conference Communications*, pages 2F6.1–2F6.5, 1982.

[2] R. Bhatia and M. Kodialam. On power efficient communication over multi-hop wireless networks: Joint routing, scheduling and power control. In *Proceeding of IEEE INFOCOM*, 2004.

[3] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade. AMRoute: Ad hoc Multicast Routing protocol. Internet Draft, draft-talpade-manet-amroute-00.txt, 1998.

[4] R. Carr and S. Vempala. Randomized metarounding. In *Proceedings of the 32nd annual ACM symposium on Theory of computing*, pages 58–62, 2000.

[5] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh. Splitstream: high-bandwidth multicast in cooperative environments. In *Proceedings of the nineteenth ACM symposium on Operating systems principles (SOSP)*, pages 298–313, 2003.

[6] P. Chaporkar, A. Bhat, and S. Sarkar. An adaptive strategy for maximizing throughput in mac layer wireless multicast. In *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing(MOBIHOC)*, pages 256–267, 2004.

[7] B. N. Clark, C. J. Colbourn, and D. S. Johnson. Unit disk graphs. *Discrete Math.*, 86(1-3), 1990.

[8] A. E. F. Clementi, P. Crescenzi, P. Penna, G. Rossi, and P. Vocca. On the Complexity of Computing Minimum Energy Consumption Broadcast Subgraphs. In *Proceedings of 18th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 121–131, 2001.

[9] M. S. Corson and S. G. Batsell. A Reservation-Based Multicast (RBM) Routing Protocol for Mobile Networks: Initial Route Construciton Phase. *Wireless Networks*, 1(4):427–450, Dec. 1995.

[10] T. Cover and J. A. Thomas. *Elements of Information Theory*. John-Wiley and Sons, Inc., 1991.

[11] T. ElBatt and A. Ephremides. Joint scheduling and power control for wireless ad-hoc networks. In *Proceedings of IEEE INFOCOM*, 2002.

[12] M. Furer and B. Raghavachari. Approximating the minimum-degree steiner tree to within one of optimal. In *Third annual ACM-SIAM symposium on Discrete algorithms*, pages 409–423, 1994.

[13] J. J. Garcia-Luna-Aceves and E. L. Madruga. Multicast routing protocol for ad-hoc networks. In *Proceedings of IEEE INFOCOM*, 1999.

[14] M. R. Garey and D. S. Johnson. *Computers and Intractability—A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.

[15] N. Garg, R. Khandekar, K. Kunal, and V. Pandit. Bandwidth maximization in multicasting. In *Proceedings of European Symposium on Algorithms (ESA)*, pages 242–253, 2003.

[16] Z. Gaspar and T. Tarnai. Upper bound of density for packing of equal circles in special domains in the plane. *Per. Pol. Civil Eng.*, 44(1):13–32, 2000.

[17] P. Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Transactions on Information Theory*, IT-46(2):388–404, Mar. 2000.

[18] B. Hajek and G. Sasaki. Link Scheduling in Polynomial Time. *IEEE Transactions on Information Theory*, 34(5):910–917, 1988.

[19] A. Itai, C. H. Papadimitriou, and J. Szwarcfiter. Hamilton paths in grid graphs. *SIAM J. Comput.*, 1(4):676–686, 1982.

[20] K. Jain, M. Mahdian, and M. R. Salavatipour. Packing steiner trees. In *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 266–274, 2003.

[21] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 66–80, 2003.

[22] L. Khachian. A polynomial algorithm in linear programming. *Soviet Math. Dokl.*, 20:191–194, 1979.

[23] M. Kodialam and T. Nandagopal. Characterizing achievable rates in multi-hop wireless networks: the joint routing and scheduling problem. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 42–54, 2003.

[24] D. Konig. Graphok es alkalmazasuk a determinansok es a halmazok elmeletere [hungarian]. *Mathematikai es Termeszettudomanyi Ertesito*, 34:104–119, 1916.

[25] S.-J. Lee, W. Su, and M. Gerla. On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks. *Mobile Networks and Applications*, 7:441–453, 2002.

[26] J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris. Capacity of Ad Hoc Wireless Networks. In *Proceedings of ACM MOBICOM*, pages 61–59, 2001.

[27] W. Liang. Constructing minimum-energy broadcast trees in wireless ad hoc networks. In *Proceedings of 3rd ACM MOBIHOC*, pages 112–122, 2002.

[28] C. L. Monma, M. Paterson, S. Suri, and F. F. Yao. Computing euclidean maximum spanning trees. *Algorithmica*, 5(3):407–419, 1990.

[29] T. S. Rappaport. *Wireless Communication Principle and Practice*. Prentice Hall, 1996.

[30] The 3rd Generation Partnership Project 2(3GPP2). roadcast/Multicast Services Stage 1 . 3GPP2 TS-3GB-S.R0030-0v1.0, 2003.

[31] The 3rd Generation Partnership Project (3GPP). Multimedia Broadcast/Multicast Service (MBMS); Stage 2. 3GPP TR 23.846, 2003.

[32] J. E. Wieselthier, G. D.Nguyen, and A. Ephremides. On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks. In *Proceedings of IEEE INFOCOM*, 2000.

[33] M. Zagalj, J.-P. Hubaux, and C. Enz. Minimum-energy broadcast in all-wireless networks:: NP-completeness and distribution issues. In *Proceedings of 8th Annual international conference on Mobile computing and networking (MOBICOM)*, pages 23–28, 2002.