Title:

ALGORITHMIC ASPECTS OF TOPOLOGY CONTROL PROBLEMS FOR AD HOC NETWORKS

Author(s):

Rui "NMI" Liu, University of Delaware, Newark, DE
Errol L. Lloyd, University of Delaware, Newark, DE
Madhav V. Marathe, D-2
Ram "NMI" Ramanathan, Internetwork Design Department, Cambridge, MA
S. S. Ravi, University at Albany - SUNY

Submitted to:

The Third ACM International Symposium on Mobile Ad **Hoc** Networking & Computing
Lausanne, Switzerland
June 9-11, 2002

Los Alamos
NATIONAL LABORATORY

Algorithmic Aspects of Topology Control Problems for Ad hoc Networks[1]

RUI LIU   ERROL L. LLOYD
Department of Computer and Information Science
University of Delaware, Newark, DE 19716
Email: {ruliu, elloyd}@cis.udel.edu.

MADHAV V. MARATHE
Los Alamos National Laboratory
MS M997, P.O. Box 1663
Los Alamos, NM 87545
Email: marathe@lanl.gov

RAM RAMANATHAN
Internetwork Design Department
BBN Technologies (A Division of GTE)
Cambridge, MA
Email: ramanath@bbn.com

S. S. RAVI
Department of Computer Science
University at Albany - SUNY
Albany, NY 12222
Email: ravi@cs.albany.edu.

Note: The research was supported by the Department of Energy under Contract W-7405-ENG-36, NSF Grant CCR-97-34936 and by the collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

Algorithmic Aspects of Topology Control Problems for Ad hoc Networks

Abstract

Topology control problems are concerned with the assignment of power values to nodes of an ad hoc network so that the power assignment leads to a graph topology satisfying some specified properties. This paper considers such problems under several optimization objectives, including minimizing the maximum power and minimizing the total power. A general approach leading to a polynomial algorithm is presented for minimizing maximum power for a class of graph properties, called **monotone** properties. The difficulty of generalizing the approach to properties that are not monotone is pointed out. Problems involving the minimization of total power are known to be NP-complete even for simple graph properties. A general approach that leads to an approximation algorithm for minimizing the total power for some monotone properties is presented. Using this approach, a new approximation algorithm for the problem of minimizing the total power for obtaining a 2-node-connected graph is obtained. It is shown that this algorithm provides a constant performance guarantee. Experimental results from an implementation of the approximation algorithm are also presented.

1 Introduction

1.1 Motivation

An ad hoc network consists of a collection of transceivers, All communication among these transceivers is based on radio propagation. For each ordered pair (u, v) of transceivers, there is a **transmission power threshold,** denoted by $p(u, v)$, with the following significance: A signal transmitted by the transceiver u can be received by v only when the transmission power of u is at least $p(u, v)$. The transmission power threshold for a pair of transceivers depends on a number of factors including the distance between the transceivers, the direction of the antenna at the sender, interference, noise, etc. [RR00].

Given the transmission powers of the transceivers, an ad hoc network can be represented by a directed graph. The nodes of this directed graph are in one-to-one correspondence with the transceivers. A directed edge (u, v) is **in** this graph if and only if the transmission power of u is at least the transmission power threshold $p(u, v)$.

The main goal of **topology control** is to assign transmission powers to the transceivers so that the resulting directed graph satisfies some specified properties. Since the battery power of each transceiver is an expensive resource, it is important to achieve the goal while minimizing a given function of the transmission powers assigned to the transceivers. Examples of desirable graph properties are connectivity, small diameter, etc. Examples of minimization objectives considered in the literature are the maximum power assigned to a transceiver and the total power of all transceivers (the latter objective is equivalent to minimizing the average power assigned to a transceiver).

As stated above, the primary motivation to study topology control problems is to make efficient use of available power at each node. In addition, using minimum amount of power at each node to achieve a given task is also likely to decrease the MAC layer interference between adjacent radios. We refer the reader to [LHB+01, RMM01, WL+01, RR00, RM99, TK84] for a thorough discussion of the power control issues for ad hoc networks.

1.2 Formulation of Topology Control Problems

Topology control problems have been studied under two graph models. The discussion above corresponds to the **directed graph model** studied in [RR00]. The **undirected graph model** proposed in

[KK+97] represents the ad hoc network as an undirected graph in the following manner. First, the directed graph model for the network is constructed. Then, for any pair of nodes u and v, whenever both the directed edges (u, v) and (v, u) are present, this pair of directed edges is replaced by a single undirected edge $\{u, v\}$. All of the remaining directed edges are deleted. Under this model, the goal of a topology control problem is to assign transmission powers to nodes such that the resulting undirected graph has a specified property and a specified function of the powers assigned to nodes is minimized. Note that the directed graph model allows two-way communication between some pairs of nodes and one-way communication between other pairs of nodes. In contrast, every edge in the undirected graph model corresponds to a two-way communication.

In general, a topology control problem can be specified by a triple of the form $\langle \mathbb{M}, \mathbb{P}, \mathbf{O} \rangle$. In such a specification, M E {DIRECTED, UNDIRECTED} represents the graph model, $\mathbb{P}$ represents the desired graph property and **O** represents the minimization objective. For the problems considered in this paper **O** E {MAX POWER, TOTAL POWER}. For example, consider the (DIRECTED, STRONGLY CONNECTED, MAX POWER) problem. Here, powers must be assigned to transceivers so that the resulting directed graph is strongly connected and the maximum power assigned to a transceiver is minimized. Similarly, the (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER) problem seeks to assign powers to the transceivers so that the resulting undirected graph has a node connectivity of (at least) **2** and the sum of the powers assigned to all transceivers is minimized.

2 Additional Definitions

This section collects together the definitions of some graph theoretic and algorithmic terms used throughout this paper.

Given an undirected graph $G(V, E)$, an **edge subgraph** $G'(V, E')$ of G has all *of* the nodes of G and the edge set E' is a subset of E. Further, if G is an edge weighted graph, then the weight of each edge in G' is the same as it is in G.

The **node connectivity** of an undirected graph is the smallest number of nodes that must be deleted from the graph so that the resulting graph is disconnected. The **edge connectivity** of an undirected graph is the smallest number of edges that must be deleted from the graph so that the resulting graph is disconnected. For example, a tree has node and edge connectivities equal to 1 while a simple cycle has node and edge connectivities equal to **2.** When the node (edge) connectivity of a graph is k, the graph is said to be **k-node connected** (**k-edge connected).** Given an undirected graph, polynomial algorithms are known for finding its node and edge connectivities [va90].

The main results of this paper use the following definition.

Definition 2.1 *A property* $\mathbb{P}$ *of the (directed or undirected) graph associated with an ad hoc network is* **monotone** *if the property continues to hold even when the powers assigned to some nodes are increased while the powers assigned to the other nodes remain unchanged,*

Example: For any $k \geq 1$, the property k-NODE CONNECTED for undirected graphs is monotone since increasing the powers of some nodes while keeping the powers of other nodes unchanged may only add edges to the graph. However, properties such as ACYCLIC or BIPARTITE are not monotone.

Some of the topology control problems considered in this paper are NP-complete. For such problems, we study approximation algorithms. In this context, an approximation algorithm provides a **performance guarantee** of ρ if for every instance of the problem, the solution produced by the approximation algorithm is within the multiplicative factor of ρ of the optimal solution. A **polynomial time approximation scheme** (PTAS) is an approximation algorithm that, given a problem instance and an accuracy requirement ϵ, produces a solution that is within a factor $1 + \text{E}$ of the optimal solution.

3 Previous Work and Summary of Results

3.1 Previous Work

The form of topology control problems considered in this paper was proposed by Ramanathan and Rosales-Hain [RR00]. They presented efficient algorithms for two topology control problems, namely (UNDIRECTED, 1-NODE CON-

NECTED, MAX POWER) and (UNDIRECTED, 2-NODE CONNECTED, MAX POWER). After determining the minimum value for the objective, their algorithms also reduce the power assigned to each transceiver such that each power level is minimal while maintaining the desired graph property. In addition, they presented efficient distributed heuristics for these problems.

Several groups of researchers have studied the (UNDIRECTED, 1-NODE CONNECTED, TOTAL POWER) problem [CH89, KK+97, CPS99, CPS00]. Reference [CH89] proves that the problem is **NP**-hard and presents an approximation algorithm with a performance guarantee of 2. The other references consider a geometric version of the problem along with a *symmetry* assumption concerning transmission power thresholds. More precisely, these references assume the following: (a) Each transceiver is located at some point of d-dimensional Euclidean space. (b) For any pair of transceivers u and v, $p(u,v) = p(v,u) =$ the Euclidean distance between the locations of u and v. For a justification of this model, see Kirousis et al [KK+97]. They show that the (UNDIRECTED, 1-NODE CONNECTED, TOTAL POWER) problem is **NP**-hard when transceivers are located in 3-dimensional space. They also present an approximation algorithm with a performance guarantee of 2 for the problem in any metric space. In addition, they provide some results for the 1-dimensional version of the (UNDIRECTED, 1-NODE CONNECTED, TOTAL POWER) problem where there is an additional constraint on the diameter of the resulting undirected graph. Clementi et al [CPS99] show that the 2-dimensional version of the (UNDIRECTED, 1-NODE CONNECTED, TOTAL POWER) problem remains **NP**-hard. They also show that the 2-dimensional version with a diameter constraint can be efficiently approximated to within some constant factor and that the 3-dimensional version does not have a polynomial time approximation scheme.

Researchers have also addressed other versions of topology control problems. Hu [Hu93] proposed a distributed algorithm based on Delaunay triangulation to maintain connectivity. However, that paper does not address the issue of assigning transmission powers to nodes. Radoplu and Meng [RM99] present a distributed protocol for maintaining strong connectivity in a network with mobile nodes. The networks generated by their protocol include minimum-energy paths (i.e., paths that allow messages to be transmitted using a minimum amount of energy) from each node to a designated master node. Wattenhofer et al [WL+01] discuss a cone-based distributed algorithm for topology control; their algorithm generates a power assignment which ensures that the size of the node set that remains connected under this power assignment is the same as the one in which every node is assigned the full power. Li and Halpern [LH01] improve upon the protocol of [RM99] by proposing another protocol; given a network G, the new protocol creates a subnetwork G' such that whenever there is a path between a pair of nodes in G, there is a minimum-energy path between them in G'. Li et al [LHB+01] provide a more detailed analysis of the protocol of [WL+01] and establish a precise bound on the angle of the cone that ensures connectivity. They also establish several properties of the protocol in [WL+01].

3.2 Summary of Main Results

The main results of this paper are the following.

1. We show that for any monotone graph property $\mathbb{P}$ that can be tested in polynomial time for undirected (directed) graphs, the problem (UNDIRECTED, $\mathbb{P}$, MAX POWER) ((DIRECTED, $\mathbb{P}$, MAX POWER)) can be solved in polynomial time. This generalizes some of the results in [RR00] where efficient algorithms were presented for two monotone properties, namely 1-NODE CONNECTED and 2-NODE CONNECTED.

2. We show that there are non-monotone and efficiently testable properties (e.g. GRAPH IS A TREE) for which the problem of minimizing the maximum power is NP-complete. This result shows that, in general, if the monotonicity condition is eliminated, then obtaining an efficient algorithm for minimizing maximum power may not be possible.

3. As mentioned above, for any monotone and efficiently testable property $\mathbb{P}$, a solution that minimizes the maximum power can be obtained in polynomial time, However, if we introduce the additional requirement that the number of nodes that use the maximum power must also be minimized, we show that there are monotone prop-

erties for which the resulting problem is NP-complete.

4. We present a general approach for developing approximation algorithms for **NP**-hard topology control problems under the **Total Power** minimization objective. The approximation results of [CH89, **KK+97**] are special cases of this general approach. As an illustration of our general approach, we present a constant factor approximation algorithm for the **(Undirected, 2-Node Connected, Total Power)** problem. **No** approximation algorithms **was** previously known for this problem. In analyzing this approximation algorithm, we use some properties of critically 2-node connected graphs [Di67, P168, We96]. By a minor modification to this approximation algorithm, we also obtain a constant factor approximation algorithm for producing 2-edge-connected graphs.
5. Finally, we present experimental results obtained from an implementation of the above approximation algorithm and compare its performance with an algorithm discussed in [RR00].

4 Results for Minimizing Maximum Power

In this section, we present our results for the **Max Power** objective. We begin with a general algorithm for the topology control problem where the the graph property is both monotone and polynomial time testable. Next, we give an example of a non-monotone property for which the problem of minimizing the maximum power is NP-complete. Finally, we show that the additional requirement of minimizing the number of nodes that use the maximum power also renders the problem NP-complete even for certain monotone properties.

4.1 An Algorithm for Monotone and Efficiently Testable Properties

We begin with a simple lemma that points out the usefulness of monotonicity.

Lemma 4.1 *For any instance of* **(Undirected, 𝔽, Max Power)** *and* **(Directed, ℙ, Max Power)** *where the graph property 𝔽 as monotone, there is an optimal solution in which all of the nodes are assigned the same power value.*

Proof: Consider an optimal solution to the given instance where the nodes don't necessarily have the same power values. Let Q denote the maximum power assigned to any node. Since the graph property is monotone, for any node whose power value is less than Q, we can increase it to Q without destroying the property. ∎

Theorem 4.1 *For any graph property* ℙ *that is monotone and that can be tested in polynomial time, the problems* **(Undirected, ℙ, Max Power)** *and* **(Directed, ℙ, Max Power)** *can be solved in polynomial time.*

Proof: We will present the proof for instances of **(Directed, ℙ, Max Power).** (The proof for **(Undirected, ℙ, Max Power)** is virtually identical.)

Consider an instance of **(Directed, ℙ, Max POwer)**. By Lemma **4.1**, there is an optimal solution in which every transceiver is assigned the same power value. We can estimate the number of candidate optimal power values as follows. Let T denote the set of all transceivers in the system and let $|T| = n$. Consider any transceiver $u \in T$. The number of different power values that need to be considered for u is at most $n - 1$, since at most one new power value is needed for each transceiver in $T - \{u\}$. Therefore, **for** all of the n transceivers, the total number of candidate power values to be considered is $n(n - 1) = O(n^2)$.

For each candidate power value, the corresponding directed graph can be constructed in $O(n^2)$ time. Let $F_{\mathbb{P}}(n)$ denote the time needed to test whether property ℙ holds for a directed graph with n nodes. Thus, the time needed to test whether property ℙ holds for each candidate solution value is $O(n^2 + F_{\mathbb{P}}(n))$. An optimal solution can be obtained by sorting the $O(n^2)$ candidate solution values and using binary search to determine the smallest value for which property 𝔽 holds. Since the number of candidate solution values is $O(n^2)$, the time taken by the sorting step is $O(n^2 \log n)$. The binary search would try $O(\log n)$ candidate solution values and the time spent for testing each candidate is $O(n^2 + F_{\mathbb{P}}(n))$. Thus, the total running time of this algorithm is

$O((n^2 + F_{\mathbb{P}}(n)) \log n)$. Since $F_{\mathbb{P}}(n)$ is a polynomial, the algorithm runs in polynomial time. ■

As an example to illustrate the above theorem, let $\mathbb{P}$ denote the property 2-NODE CONNECTED for undirected graphs. It is known that this property can be tested in $O(n^2)$ time for a graph with n nodes [va90]. For this property, the general algorithm outlined in the proof of Theorem 4.1 yields an algorithm with a running time of $O(n^2 \log n)$. This running time matches the time of the algorithm given in [RR00]. However, it should be noted that the algorithm in [RR00] not only finds an optimal solution but also reduces the power of each transceiver so that the power levels are minimal. There is no increase in their asymptotic running time.

Instead of requiring the entire graph to be connected, one may require connectivity only for a specified subset of the nodes. Such a requirement arises in the context of multicasting (see for example [RP01]). Connectedness of a specified subset of nodes can be seen to be a monotone property. So, the general approach presented above leads to a polynomial time algorithm for this property as well.

4.2 Difficulty of Generalizing to Nonmonotone Properties

We now show that there is a natural non-monotone graph property for which the problem of minimizing the maximum power is NP-complete. As mentioned earlier, this result points out that if the monotonicity requirement is omitted, then an efficient algorithm for minimizing maximum power may not be possible.

The property that we use for this purpose is "G IS A TREE". Surprisingly, we show that this property makes the topology control problem **NP**-complete even without any minimization objective. The proof of Lemma 4.2 utilizes a reduction from the following problem, which is known to be **NP**-complete [GJ79].

Exact Cover by 3-Sets (X3C)

Instance: A set $S = \{x_1, x_2, \ldots, x_n\}$ of elements, where $n = 3r$ for some integer r; a collection $C = \{C_1, C_2, \ldots, C_m\}$ of subsets of S such that $|C_j| = 3$, $1 \le j \le m$.

Question: Does C contain an ***exact cover*** for S, that is, is there a subcollection C' of C such that the sets in C' are pairwise disjoint and their union is equal to S?

We note that whenever there is a solution to an instance of **X3C,** the number of sets in the solution is exactly r (i.e. $n/3$).

Lemma 4.2 ***To find a power assignment such that the resulting undirected graph G is a tree is NP-complete.***

By abuse of terminology, we use (UNDIRECTED, TREE, *) to denote this problem.

Proof: In the (UNDIRECTED, TREE, *) problem, we are given a collection of nodes, and a (symmetric) power threshold $p(u, v)$ for each pair of nodes. The question is whether there exists a power assignment such that the graph induced by the power assignment is a tree.

It is easy to see that (UNDIRECTED, TREE, *) is in **NP** since one can guess a power assignment and verify in polynomial time that the resulting graph is a tree. We prove the **NP**-hardness of the problem by a reduction from **X3C** (defined above).

Given an instance **I** of **X3C** consisting of a set S with n elements and a collection C of m subsets, we construct an instance **I'** of the (UNDIRECTED, TREE, *) problem as follows. The node set V of **I'** contains a total of $n + m + 1$ nodes: There is one node (called an **element node)** u_i corresponding to each element x_i of S (thus, there are totally **3r** element nodes), one node (called a **set node)** v_j corresponding to each set C_j of C (thus, there are totally m set nodes), and a special node (called the **root node)** denoted by R. The power thresholds are chosen as follows. (The reader should bear in mind that the power thresholds are symmetric; that is, for any pair of nodes u and v, $p(u, v) = p(v, u)$.)

$$\begin{array}{rcll} p(R, v_j) & = & 1 & (1 \le j \le m). \\ p(u_i, v_j) & = & 2 & \text{if } x_i \in C_j,\ 1 \le i \le n,\ 1 \le j \le m. \end{array}$$

For all other pairs of nodes, the power thresholds are set to **3.** This completes the construction of the instance **I'** of (UNDIRECTED, TREE, *). It is easy to verify that the construction can be carried out in polynomial time. We now argue that there is a solution to the (UNDIRECTED, TREE, *) instance if and only if there is a solution to the **X3C** instance.

If: Suppose the X3C instance has a solution C'. We choose the following power assignment: $p(R) = 1$, $p(u_i) = 2$ $(1 \leq i \leq n)$, $p(v_j) = 2$ if C_j is in C' and $p(v_j) = 1$ otherwise $(1 \leq j \leq m)$. It can be seen that the graph G resulting from this power assignment contains only the following edges:

(a) The edge $\{R, v_j\}$, for each j, $1 \leq j \leq m$.

(b) For each node v_j whose corresponding set C_j is in C', there are edges from v_j to the three nodes corresponding to the elements in C_j.

By choosing R as the root and using the fact that C' is an exact cover, it can be verified that G is a tree: The root node R is adjacent to each of the set nodes; and, each element node appears as one of the three children of a set node corresponding to a subset in the collection C'.

Only if: Now, suppose the $\langle$UNDIRECTED, TREE, $*\rangle$ instance has a solution. Let $p(x)$ denote the power assigned to node x and let G denote the graph induced by the power assignment.

We first observe that $p(R) \geq 1$; otherwise, R would be an isolated node and thus G cannot be a tree. Similarly, $p(v_j) \geq 1$ for every set node v_j and $p(u_i) \geq 2$ for every element node u_i. As a consequence, the root node R is adjacent to each of the set nodes v_1, v_2, ..., v_m, and the maximum power assigned is at least **2**. Therefore, there are two cases to consider:

Case 1. The maximum power assigned is **2**.

Let $X = \{v_{j_k} : p(v_{j_k}) = 2\}$. We claim that the collection $C' = \{C_{j_k} : v_{j_k}$ E $X\}$ is an exact cover for S. We prove this by first showing that each element x_i appears in some subset of C'. To see this, we note that the graph G is connected (since it is a tree). Thus, there is at least one edge from the element node u_i (corresponding to element x_i) to some other node of G. Since the maximum power assigned to any node is **2** and the power threshold for the element node u_i to have an edge to R or an edge to any other element node is **3**, u_i must be adjacent to a set node v_j. Further, because the threshold values are symmetric, $p(v_j) = 2$. Thus, $v_j \in$ X and the corresponding subset C_j is in C'. Hence, each element appears in some subset in the collection C'.

We now show that the subsets in the collection C' are pairwise disjoint. Suppose some pair of subsets C_a and C_b in C' have a common element x_i. By our choice of C', the power values assigned to the corresponding set nodes v_a and v_b are both **2**. Further, the power assigned to node u_i is also **2**. Thus, in the graph G, u_i is adjacent to both v_a and v_b. As observed earlier, the root node R is adjacent to both v_a and v_b. Now, the four edges $\{R, v_a\}$, $\{v_a, u_i\}$, $\{u_i, v_b\}$ and $\{v_b, R\}$ create a cycle in G. This contradicts the assumption that G is a tree. So, the subsets in C' are pairwise disjoint, and C' is indeed an exact cover for S.

Case 2. The maximum power assigned is **3**.

First, note that at most two nodes can have power **3**, since if three nodes have power **3**, then they are mutually adjacent, and thus G is not a tree.

Second, if the power assignment is as in the following cases, we argue that there is an equivalent assignment in which the maximum power is **2**. These cases are: only one node has power **3**; R and one set node v_i have power **3**; and, one element node u_i and one set node v_j have power **3** where $x_i \in C_j$. In any of these cases, the resulting graph G has no edge with power threshold **3**, so an assignment with maximum power **2** can be obtained by reducing the power level of the nodes with power **3** while keeping the assignments to all of the other nodes unchanged. The induced graph doesn't change. Thus, the new assignment is a solution with maximum power **2** to the instance of $\langle$UNDIRECTED, TREE, $*\rangle$. Following the argument in Case **1**, a solution to X3C can be constructed.

Finally, we claim that there are no such valid power assignments in the remaining cases (i.e. R and u_i have power **3**; v_i and v_j have power **3**; u_i and u_j have power **3**; or, u_i and v_j have power **3** where $x_i \notin C_j$). The reasons are the following:

1. If two set nodes v_i and v_j have power **3**, then the edges $\{R, v_i\}$, $\{R, v_j\}$ and $\{v_i, v_j\}$ form a cycle.

2. If the root node R and one element node u_i have power **3**, the edge $\{R, u_i\}$ is in G. Therefore, edge$\{u_i, v_j\}$, $1 \leq j \leq m$, is not in G, otherwise R, u_i, and v_j form a cycle. Recall that $p(u_i) \geq 2$ for every element node u_i, therefore each v_j with power **2** is linked with exactly **3** element nodes. No two set nodes can linked with the same element node, otherwise those three nodes and R form a cycle. Hence, totally $3k$ (where k is the number of set nodes with power **2**) element nodes are linked to some set node. Further, no element nodes can link with each other since the power thresholds between such nodes are

3. This implies that there are $3k+1$ element nodes. There is a contradiction because we know in this instance of (UNDIRECTED, TREE, *), the number of element nodes is a multiple of **3.**

3. If two element nodes u_i and u_j have power **3,** the edge $\{u_i, u_j\}$ is in G. Recall that all set nodes have to be linked with R, so one and only one of u_i and u_j is linked with a set node. Suppose it is u_i. We know from above that $3k$ element nodes are linked with some set node. So, together with u_j, there are $3k+1$ element nodes - a contradiction.

4. If one element node u_i and one set node v_j have power **3,** where $x_i \notin C_j$, then u_i is linked with v_j. Therefore, there are **4** nodes linked with v_j, which are u_i and three element nodes whose corresponding elements are in set C_j. Hence, there are totally $3k+1$ element nodes - a contradiction.

This completes the proof of the case **2** as well as that of Lemma, **4.2.** ■

Theorem 4.2 *There is a non-monotone property* $\mathbb{P}$ *for which* (UNDIRECTED, $\mathbb{P}$, MAX POWER) *is **NP**-complete.*

Proof: Let $\mathbb{P}$ denote the property "G IS A TREE". It is clear that (UNDIRECTED, $\mathbb{P}$, MAX POWER) is in **NP** because given a power assignment the required conditions can be verified in polynomial time. Further, the **NP**-hardness follows from Lemma **4.2.** ■

4.3 Difficulty of Minimizing the Number of Nodes of Maximum Power

This section explores an extension of the (UNDIRECTED, $\mathbb{P}$, MAX POWER) problem for monotone graph properties. While such problems can be solved efficiently, our algorithm in Section **4.1** assigns the maximum power value to all of the nodes. From a practical point of view, it is important to reduce the number of nodes with maximum power without affecting the required property. In this section, we show that this additional requirement renders the problem NP-complete even for certain monotone graph properties. **A** formal statement of the problem is as follows.

Minimizing Max-power Users

Instance: A positive integer M; a positive number P (max-power); a node set V; a power threshold function p associated with V ($\forall u, v \in V$, $p(u,v)$ is the power threshold); and a graph property $\mathbb{P}$.

Question: Dues there exist a power assignment where the power assigned to each node is at most P and the number of the nodes that are assigned power P is at most M, such that the resulting undirected graph G satisfies $\mathbb{P}$?

Theorem 4.3 *There is* **a** *monotone property* $\mathbb{P}$ *that can be tested an polynomial time* **for** *which the problem* **Minimizing Max-power Users** *is NP-complete.*

Proof: We use a reduction from **Set-covering** a well-known NP-complete problem [GJ79].

Set-covering

Instance: A positive integer K; a set $A = \{a_1, a_2, \ldots, a_n\}$; a set $F = \{f_1, f_2, \ldots, f_m\}$, where f_i, $1 \le i \le m$ is a subset of A and $\forall a_i \in A$, $\exists f_j \in F$, such that $a_i \in f_j$. Without loss of generality, suppose all f_i, $1 \le i \le m$ are non-empty.

Question: Does there exist a set $S \subseteq F$, with $|S| \le K$, where $\forall a_i$ **E** A, 3s $\in S$, such that $a_i \in s$?

Let $\mathbb{P}$ be the property "THE DIAMETER OF G IS LESS THAN OR EQUAL TO **6**". The property implies that in G, each node is at most 6 hops away from any other node. Obviously, $\mathbb{P}$ is monotone, and can be tested in $O(N^3)$ time by using the Floyd-Warshall algorithm, where N **is** the number of nodes in the graph G [CLRS01]. We now show that **Minimizing Max-power Users** is NP-complete.

'It is obvious that **Minimizing Max-power Users** is in NP since $\mathbb{P}$ can be tested in polynomial time. To prove the **NP**-hardness we provide a reduction from **Set-covering.**

Given an instance I of **Set-covering,** map each element a_i of A to a node u_i and map each f_i of F to a node v_i. Further, our instance **I'** of **Minimizing Max-power Users** also contains four special nodes: w, s_1, s_2, s_3. The power threshold function p is defined as follows (note that the power thresholds are symmetric):

$$p(u_i, v_j) = 1 \quad \text{if } a_i \in f_j$$
$$p(w, v_j) = P \quad (1 \le j \le m)$$
$$p(w, s1) = p(s_1, s_2) = p(s_2, s_3) = 1.$$

For any other pair of nodes x and y, $p(x,y) = P + 1$.

The value of M is set to $K + 1$. This completes the construction of an instance I' of **Minimizing Max-power Users.** It's clear that the construction can be done in polynomial time. Now, we show that there is a solution to the **Minimizing Max-power Users** instance if and only if there is a solution to **Set-covering.**

If: Suppose S is a solution to an instance of **Set-covering,** We construct **a** power assignment S' as follows.

$$\begin{array}{lcll} S'(w) & = & P & \\ S'(v_i) & = & P & \text{if } f_i \text{ E } S \\ & & & \text{(there are at most } K \text{ such nodes)} \\ S'(x) & = & 1 & \text{for any other node } x. \end{array}$$

We now argue that S' is a solution to the instance of **Minimizing Max-power Users.** Obviously, the maximum power assigned is P and at most M (i.e. $K + 1$) nodes have power P. To establish that the resulting graph $G(V, E)$ satisfies $\mathbb{P}$ we show that w is within **3** hops of each other node (hence the graph diameter is at most **6).** That this is so follows from these observations:

1. Nodes s_1, s_2, and s_3 are (respectively) 1,**2,** and **3** hops away from w.
2. $\forall f_i \in F$, if f_i E S, then edge$\{v_i, w\} \in E$, hence node v_i is **1** hop away from w.
3. $\forall a_i$ E A, node u_i is **2** hops away from w, since u_i is linked with some v_i that has an edge to w, otherwise S doesn't cover a_i.
4. $\forall f_i \in F$, if $f_i \notin S$, then v_i is **3** hops away from w, since v_i is linked to some u_j.

Only if: Suppose we have a power assignment S' that is a solution to the instance of **Minimizing Max-power Users,** and that $G(V, E)$ is the resulting graph. If there is an edge between w and v_i in E (there are at most $M - 1$ such edges), then place set f_i into set S. We claim that S is a solution to the instance of **Set-covering,** Since $|S| \leq M - 1$ (i.e. K), we need only show that S covers A. Since the diameter of $G(V, E) \leq$ **6,** v_3 has to be at most 6 hops away from any other node. It follows that w must be within **3** hops of every other node. For each v_i, if edge$\{v_i, w\}$ E E, v_i is one hop away from w. However, if edge$\{v_i, w\} \notin$ E, v_i is at least **3** hops away from w. Now suppose $\exists a_i$ E **A** that is not in any set $f \in S$. Then, u_i is not linked with any node v_j that is one hop away from w. Thus, u_i has to be linked with some v_j that is at least **3** hops away from W, which means node u_i is at least **4** hops away from w - a contradiction. This completes the proof of Theorem **6.3.** ■

5 A General Approach for Minimizing Total Power

5.1 Approximating Minimum Total Power

Topology control problems in which the minimization objective is the total power tend to be computationally intractable. For example, the problem is **NP**-hard even for the (simple) property 1-NODE-CONNECTED [KK+97]. **A** common way of coping with such problems is to develop polynomial time approximation algorithms for them. In this section, we present a general outline for such an approximation algorithm for topology control problems of the form (UNDIRECTED, $\mathbb{P}$, TOTAL POWER). We observe that this general outline encompasses the approximation algorithm for (UNDIRECTED, 1-NODE CONNECTED, TOTAL POWER) presented in [KK+97]. Based on the general outline, we also develop an approximation algorithm with a constant performance guarantee for (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER). A slight modification of this approximation algorithm yields an approximation algorithm for the problem of obtaining a 2-edge-connected graph while minimizing total power.

In presenting our general scheme, we assume (as done in Section **4.1)** that the property $\mathbb{P}$ to be satisfied by the graph is monotone and that the property can be tested in polynomial time. We also assume *symmetric* power thresholds as in [KK+97, CPS99, CPS00]; that is, for any pair of transceivers u and v, the power thresholds $p(u, v)$ and $p(v, u)$ are equal.

An outline for our general approximation algorithm (called Heuristic GEN-TOTAL-POWER) is shown in Figure **1.** Note that Steps 1 and **3** of the outline can be implemented in polynomial time. The time complexity of Step **2** depends crucially on the property $\mathbb{P}$. For some properties such as 1-NODE CONNECTED, Step **2** can be done in polynomial time. For other

Input: An instance I of (UNDIRECTED, $\mathbb{P}$, TOTAL POWER) where the property $\mathbb{P}$ is monotone and polynomial time testable.

Output: A power value $\pi(u)$ for each transceiver u such that the graph induced by the power assignment satisfies property P and the total power assigned to all nodes is as small as possible.

Steps:

1. From the given problem instance, construct the following undirected complete edge weighted graph $G_c(V, E_c)$. The node set V is in one-to-one correspondence with the set of transceivers. The weight of every edge $\{u, v\}$ in E, is equal to the power threshold value $p(u, v)$ (which is also equal to $p(v, u)$ by the symmetry assumption).
2. Construct an edge subgraph $G'(V, E')$ of G_c such that G' satisfies property P and the total weight of the edges in E' is minimum among all edge subgraphs of G_c satisfying property $\mathbb{P}$.
3. For each node (transceiver) u, assign a power value $\pi(u)$ equal to the weight of the largest edge incident on u.

Figure 1: Outline of Heuristic GEN-TOTAL-POWER for Approximating Total Power

properties such as 2-NODE CONNECTED, Step *2* cannot be done in polynomial time unless P = NP [GJ79]. In such cases, an efficient algorithm that produces an approximately minimum solution can be used in Step **2.** The following theorem proves the correctness of the general approach and establishes its performance guarantee as a function of some parameters that depend on the property $\mathbb{P}$ and the approximation algorithm used in Step *2* of the general outline.

Theorem 5.1 *Let* I *be an instance of* (UNDIRECTED, $\mathbb{P}$, TOTAL POWER) *where* P *is a monotone property. Let* $OPT(I)$ *and* $GTP(I)$ *denote respectively the total power assigned to the nodes in an optimal solution and in a solution produced by Heuristic* GEN-TOTAL-POWER *for the instance* I.

(i) The graph G'' *resulting from the power assignment produced by the heuristic (i.e. step 3) satisfies property* $\mathbb{P}$.

('ai) Consider the complete graph $G_c(V, E,)$ *constructed an Step 1* **of** *the heuristic. Let* $H(V, E_H)$ *be an edge subgraph* **of** G_c *with minimum total edge weight satisfying property* $\mathbb{P}$ *and let* $W(H)$ *denote the total edge weight* **of** H*. Let Step* **2 of** *the heuristic produce an edge subgraph* $G'(V, E')$ **of** G *with total edge weight* $W(G')$*. Suppose there are quantities* a! > 0 *and* $\beta > 0$ *such that*

(a) $W(H) \leq \alpha\, OPT(I)$ *and*

(b) $W(G') \leq \beta\, W(H)$,

then $GTP(I) \leq 2\alpha\beta\, OPT(I)$*. In other words, Heuristic* GEN-TOTAL-POWER *provides a performance guarantee* **of** $2\alpha\beta$.

Before proceeding to the proof of this result, we illustrate its use by discussing how the 2-approximation algorithm presented in **[KK+97]** for the (UNDIRECTED, 1-NODE CONNECTED, TOTAL POWER) problem can be derived from the above general outline. In Step **2** they use an efficient algorithm for constructing a minimum spanning tree of G_c. They also show that the total power assigned by any optimal solution is at least the weight of a minimum spanning tree of G_c. Thus, using the notation of Theorem **5.1,** a! $= \beta = 1$ for their approximation algorithm. Since 1-NODE-CONNECTED is a monotone property, it follows from Theorem **5.1** that the performance guarantee provided by their algorithm is **2.**

Proof of Theorem 5.1:

Part (i): The edge subgraph $G'(V, E')$ constructed in Step **2** of the heuristic satisfies property $\mathbb{P}$. We show that every edge in E' is also in the subgraph G'' induced by the power assignment produced in Step **3.** Then, even if G'' has other edges, the monotonicity of $\mathbb{P}$ allows us to conclude that G'' satisfies $\mathbb{P}$.

Consider an edge $\{u, v\}$ with weight $p(u, v)$ in E'. Recall that $p(u, v)$ is the minimum power threshold for the existence of edge $\{u, v\}$ and that the power thresholds are symmetric. Since Step **3** assigns to each node the maximum of the weights of edges incident on that node, we have *n (~>* $p(u, v)$ and $\pi(v) \geq p(u, v)$. Therefore, the graph G'' induced by the power assignment also contains the edge $\{u, v\}$ and this completes the proof of Part (i).

Part (ii): By conditions (a) and (b) in the statement of the theorem, we have $W(G') \leq \alpha\beta\, OPT(I)$. We observe that $GTP(I) \leq 2W(G')$. This is because in Step **3** of the heuristic, the weight of any edge is assigned to at most two nodes (namely, the endpoints of the edge). Combining the two inequalities, we get $GTP(I) \leq 2\alpha\beta\, OPT(I)$, and this completes the proof of Theorem **5.1.** ■

5.2 A New Approximation Algorithm

This section presents an approximation algorithm for the **(UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER)** problem. This algorithm is derived from the general approach outlined in Figure **1.** We note that the property **2-NODE-CONNECTED** is monotone. The following notation is used throughout this section. I denotes the given instance of (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER) with n transceivers. For each transceiver u, $\pi^*(u)$ denotes the power assigned to u in an optimal solution. Further, $OPT(I)$ denotes the sum of the powers assigned to the nodes in an optimal solution.

We obtain an approximation algorithm for the **(UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER)** problem from the outline of Figure **1** by using an approximation algorithm from [KR96] for the minimum weight **2-NODE-CONNECTED** subgraph problem in Step **2** of the outline. This approximation algorithm provides a performance guarantee of $(2 + 1/n)$. Using the notation of Theorem **5.1,** we have $\beta \leq (2 + 1/n)$.

We also show (see Lemma **5.1** below) that for the complete edge weighted graph $G_c(V, E_c)$ constructed from I in Step **1** of the outline, there is an edge subgraph $G_1(V, E_1)$ such that G_1 is **2-NODE-CONNECTED** and the total weight $W(G_1)$ of the edges in G_1 is at most $(2 - 2/n)\,OPT(I)$. Again, using the notation of Theorem **5.1,** this result implies that $\alpha \leq (2 - 2/n)$.

Thus, once we establish Lemma **5.1,** it would follow from Theorem **5.1** that the performance guarantee of the resulting approximation algorithm for the **(UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER)** problem is $2\,(2-2/n)\,(2+1/n)$, which approaches **8** asymptotically from below. The remainder of this section is devoted to the formal statement and proof of Lemma **5.1.**

Lemma 5.1 *Let I denote an instance of the* (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER) *problem with* **n** *transceivers. Let $OPT(I)$ denote the total power assigned to the transceivers in an optimal solution to I. Let $G_c(V, E_c)$ denote the complete graph constructed in Step 1 of Heuristic* GEN-TOTAL-POWER. *There is an edge subgraph $G_1(V, E_1)$ of G_c such that G_1 is* 2-NODE-CONNECTED *and the total weight $W(G_1)$ of the edges in G_1 is at most $(2 - 2/n)\,OPT(I)$.*

Our proof of Lemma **5.1** begins with an optimal power assignment to instance I and constructs the graph G_1 satisfying the properties mentioned in the above statement. This construction relies on several definitions and known results from graph theory. We begin with the necessary definitions.

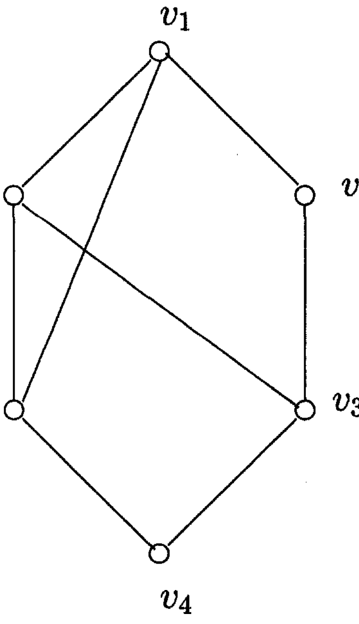

Figure 2: A simple cycle $\langle v_1, v_2, v_3, v_4, v_5, v_6,$ VI) with two chords $\{v_1, v_5\}$ and $\{v_3, v_6\}$.

Definition 5.1 *Let $G(V, E)$ be an undirected graph. Suppose the node sequence* (VI, v_2, v_3, ..., v_k, $v_1\rangle$ *forms a simple cycle C of length at least 4 in G. Any edge $\{v_i, v_j\}$ of G $(1 \leq i \neq j \leq k)$ which is not in C is a chord.*

Figure **2** shows a simple cycle of length **6** with two chords.

Definition *5.2 An undirected graph $G(V, E)$ is critically* 2-NODE-CONNECTED *if it satisfies both of the following conditions.*

(i) G is 2-NODE-CONNECTED.

(ii) For every edge $e \in E$, the subgraph of G *obtained by deleting the edge e* is not 2-NODE-CONNECTED,

For example, a simple cycle on three or more nodes is critically 2-NODE-CONNECTED. This is because such a cycle is 2-NODE-CONNECTED, and deleting any edge of the cycle yields a simple path which is not 2-NODE-CONNECTED.

A number of properties of critically 2-NODE-CONNECTED graphs have been established in the literature (see for example, [Di67, Pl68, We96]). We use the following property in proving Lemma 5.1.

Theorem 5.2 *If a graph G is critically* 2-NODE-CONNECTED *then no cycle of G has a chord.* ■

For a proof of the above' theorem, see [Di67, Pl68].

We also use some terminology associated with **Depth-First-Search** (DFS) [CLRS01]. When DFS is carried out on a connected undirected graph $G(V, E)$, a spanning tree $T(V, E_T)$ is produced. Each edge in T is called a **tree edge.** Each tree edge joins a child to its parent. An **ancestor** of a node u in T is a node which is not the parent of u but which is encountered in the path from u to the root of T. Each edge in $E - E_T$ is called a **back edge.** Each back edge joins a node u to an ancestor of u in T. The following lemma establishes a simple property of back edges that arise when DFS is carried out on a critically 2-NODE-CONNECTED graph.

Lemma 5.2 *Let $G(V, E)$ be a critically* 2-NODE-CONNECTED *graph and let $T(V, E_T)$ be a spanning tree for* G *produced using DFS. For any node u, there is at most one back edge from u to an ancestor of u in T.*

Proof: The proof is by contradiction. Suppose a node u has two or more back edges. Let v and w be two ancestors of u in T such that both $\{u, v\}$ and $\{u, w\}$ are back edges. Note that these two edges are in G. Without loss of generality, let w be encountered before v in the path in T from the root to u in T. The path from w to u in T together with the edge $\{u, w\}$ forms a cycle in G. By our choice of w, this cycle also includes the node v. Therefore, the edge $\{u, v\}$ is a chord in the cycle. This contradicts the assumption that G is critically 2-NODE-CONNECTED since by Theorem **5.2**, no cycle in G can have a chord. The lemma follows. ■

We now prove several additional lemmas that are used in our proof of Lemma 5.1. Consider the given instance I of the (UNDIRECTED, 2-NODE CONNECTED, TOTAL **POWER)** problem and let V denote the set of transceivers. **Fix** an optimal solution to the instance I and let p^* denote the maximum power value assigned to a node in this optimal solution. Let the chosen optimal power assignment induce the graph $G^*(V, E^*)$. Note that G^* is 2-NODE-CONNECTED. Let $G_1^*(V, E_1^*)$ be an edge subgraph of G^* such that G_1^* is critically 2-NODE-CONNECTED. (Such a subgraph can be obtained by starting with G^* and repeatedly removing edges until no further edge deletion is possible without violating the 2-NODE-CONNECTED property.) For each edge $\{u, v\}$ of G_1^*, we assign a weight $w_1(u, v)$ as follows.

1. Let r be a node such that $\pi^*(r) = p^*$. Using r **as** the root, perform a DFS of G_1^*. Let $T(V, E_T)$ be the resulting spanning tree. Thus, each edge of G_1^* is either a tree edge or a back edge.

2. For each tree edge $\{u, v\}$ where v is the parent of u, let $w_1(u, v) = \pi^*(u)$.

3. For each back edge $\{u, v\}$ where v is an ancestor of u, let $w_1(u, v) = \pi^*(u)$.

The following lemma bounds the total weight $W_1(G_1^*)$ of all the edges in G_1^* under the edge weight function w_1 chosen above.

Lemma *6.3* $W_1(G_1^*) \leq (2 - 2/n)OPT(I)$.

Proof: **As** mentioned above, each edge of G_1^* is either a tree edge or a back edge. Consider the tree edges first. For each tree edge $\{u, v\}$, where v is the parent of u, $w_1(u, v) = \pi^*(u)$. Thus, the weight $\pi^*(u)$ is assigned to at most one tree edge (namely, the edge that joins u to the parent of u if any in T). The power value of the root r in the optimal solution, namely p^*, is not assigned to any tree edge (since the root has no parent). Thus, the total weight of all of the tree

[1] It should be noted that the graph theoretic terminology used in [Di67, Pl68] is different from ours. The statement of Theorem 5.2 given above is from [We96].

edges under the weight function w_1 is bounded by $OPT(I) - p^*$.

Now consider the back edges. For each back edge $\{u,v\}$, where v is an ancestor of u, $w_1(u,v) = \pi^*(u)$. Since G_1^* is critically 2-NODE-CONNECTED, by Lemma *5.2,* each node has at most one back edge to an ancestor. Thus, the weight $\pi^*(u)$ is assigned to at most one back edge. Again, the power value p^* of the root r in the optimal solution is not assigned to any back edge. Thus, the total weight of all of the back edges under the weight function w_1 is also bounded by $OPT(I) - p^*$.

Therefore, the total weight $W_1(G_1^*)$ of all of the edges in G_1^* under the edge weight function w_1 is at most $2\,OPT(I) - 2p^*$. Since p^* is the largest power value assigned to a node in the optimal solution, p^* is at least $OPT(I)/n$. Hence, $W_1(G_1^*)$ is bounded by $(2 - 2/n)\,OPT(I)$ as required. ■

The following lemma relates the weight $w_1(u,v)$ of an edge $\{u,v\}$ to the power threshold $p(u,v)$ needed for the existence of the edge.

Lemma 5.4 *For any edge* $\{u,v\}$ in G_1^*, $p(u,v) \leq w_1(u,v)$.

Proof: Consider any edge $\{u,v\}$ in G_1^*. Since G_1^* is an edge subgraph of G^* (the graph induced by the chosen optimal power assignment), $\{u,v\}$ is also an edge in G^*. Also, recall that the minimum power threshold values are symmetric. Therefore, $\pi^*(u) \geq p(u,v)$ and $\pi^*(v) \geq p(u,v)$. Hence $\min\{\pi^*(u), \pi^*(v)\} \geq p(u,v)$. The weight assigned to the edge $\{u,v\}$ by the edge weight function w_1 is either $\pi^*(u)$ or $\pi^*(v)$. Therefore, $w_1(u,v) \geq \min\{\pi^*(u), \pi^*(v)\}$. It follows that $w_1(u,v) \geq p(u,v)$. ■

We are now ready to complete the proof of Lemma *5.1.*

Proof of Lemma 5.1: Starting from an optimal power assignment to the instance I, construct the graph $G_1^*(V, E_1^*)$ as described above. Since the graph G_c constructed in Step 1 of the heuristic (Figure 1) is a complete graph, every edge in G_1^* is also in G_c. Consider the edge subgraph $G_1(V, E_1)$ of G_c where $E_1 = E_1^*$. Since G_1^* is 2-NODE-CONNECTED, so is G_1. By Lemma *5.4,* for each edge $\{u,v\}$ in E_1, $p(u,v) \leq w_1(u,v)$. Therefore, the total weight $W(G_1)$ of all of the edges in G_1 under the edge weight function p is at most $W_1(G_1^*)$. By Lemma *5.3,* $W_1(G_1^*)$ is bounded by $(2 - 2/n)\,OPT(I)$. Therefore, $W(G_1)$ is also bounded by $(2-2/n)\,OPT(I)$. In other words, the edge subgraph $G_1(V, E_1)$ is 2-NODE-CONNECTED and the total weight of all its edges is at most $(2 - 2/n)\,OPT(I)$. This completes the proof of Lemma *5.1.* ■

The following is a direct consequence of the above discussion.

Theorem 5.3 *There is a polynomial time approximation algorithm with a performance guarantee of* $2\,(2 - 2/n)\,(2 + 1/n)$ *(which approaches 8 asymptotically from below) for the* (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER) *problem.*

Finally, we complete this section with an analogous result for two *edge* connectivity:

Theorem 5.4 *There is a polynomial time approximation algorithm with a performance guarantee of* $8(1 - 1/n)$ *(which approaches 8 asymptotically from below) for the* (UNDIRECTED, 2-EDGE CONNECTED, TOTAL POWER) *problem.* ■

The proof (omitted here) is similar to that for Theorem *5.3* and utilizes a 2-approximation algorithm of [KV94] for the minimum cost 2-EDGE-CONNECTED SUBGRAPH problem.

5.3 Experimental Results

5.3.1 Overview

In the preceding sections, we established that our algorithm for (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER) has an approximation ratio of 8. In this section, we report on the experimental performance of this algorithm. Since there are no existing approximation algorithms specifically for (UNDIRECTED, 2-NODE CONNECTED, TOTAL POWER), in the experiments described here we compare the performance of our algorithm with Ramanathan and Rosales-Hain's algorithm in [RR00]. Recall that the algorithm finds an optimum solution for the (UNDIRECTED, 2-NODE CONNECTED, MAX POWER) in which the power level of each node is minimal.

5.3.2 Experimental Environment

Our experiments were conducted using a customized implementation with an experimental setup similar to the one described in [RR00]. In these experiments, the radio wave propagation model used is the ***Log-distance Path Loss Model:***

$$PL(d) = -10\log_{10}\left[\frac{G_t G_r \lambda^2}{(4\pi)^2 d_0^2}\right] + 10\eta\log_{10}\left[\frac{d}{d_0}\right]$$

where η is the path loss exponent, d_0 is the close-in reference distance, λ is the radio wavelength, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, and d is the separation distance between transmitter and receiver (see [Ra96] for detailed descriptions of these parameters). All of the parameters are chosen to emulate a 2.4 GHz wireless radio, and if d is less than a certain threshold, the transmission power is set to the minimum transmission power of 1 dBm.

The experiments are conducted by varying the density of the network and the geographical distribution of the nodes. In total there are 22 sets of experiments, and 10 trials are run on each set. Each of the results we cite is the average over the 10 trials.

The node density varies from 0.625 node/sq mile to 4.06 nodes/sq mile in a 4 mile by 4 mile area. The experiments are conducted using two node distributions: one uniform and one skewed. Specifically, in the uniformly distributed networks, all nodes are placed using a random uniform distribution. In the networks with a skewed distribution, the network area is equally divided into a 2 by 2 grid, with 80% of the nodes uniformly distributed in two diagonal squares, and the other 20% of the nodes uniformly distributed in the other two diagonal squares.

In each experiment, after generating a placement of the nodes, both our Min Total algorithm and the Min Max algorithm of [RR00] are run on the network consisting of those nodes. Each algorithm assigns power to each node such that the resulting network is 2-NODE CONNECTED. For each algorithm we measure both the maximum and average power assigned, as well as the maximum and average degrees of nodes in the resulting network.

The experimental results are shown in Figures 3, 4, 5, and 6. In Figures 3 and 5, MM AVG (MM MAX) and MT AVG (MT MAX) are the average (maximum) power using the Min Max and the Min Total algorithms respectively; likewise in Figures 4 and 6, MM AVG (MM MAX) and MT AVG (MM MAX) are the average (maximum) degrees using the Min Max and the Min Total algorithms respectively.

5.3.3 Observations

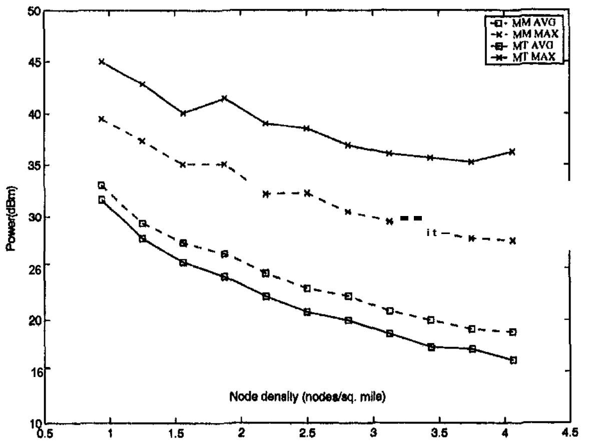

Figure 3: Power in uniformly distributed network

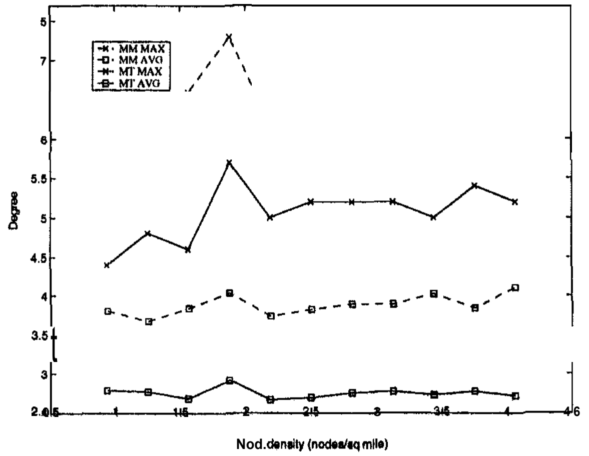

Figure 4: Degree in uniformly distributed network

In this section we discuss the results both in regard to the node power assignments and in regard to the node degrees.

First, with respect to power in the cases where nodes are uniformly distributed, our Min Total algorithm consistently outperforms the Min Max algorithm in [RR00] in regard to average power by 5% -15%. In contrast, the maximum power assigned by our algorithm is 14% -31% larger than that of [RR00]. The average power is about 68% -83% of the maximum power using the Min Max algorithm, and about 44% -70% using our algorithm. Those numbers decrease as the density of the network increases, which implies that the average power decreases faster than

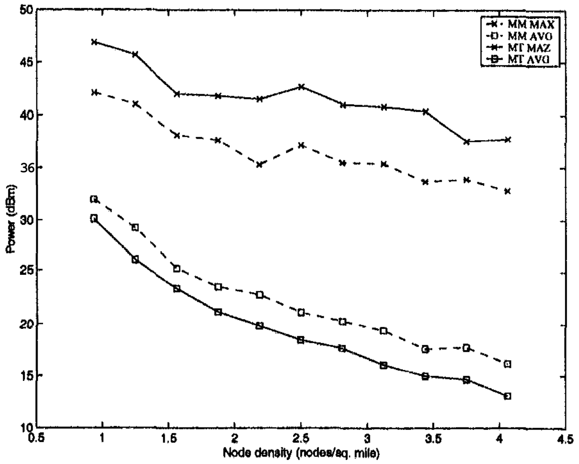

Figure 5: Power in skewed network

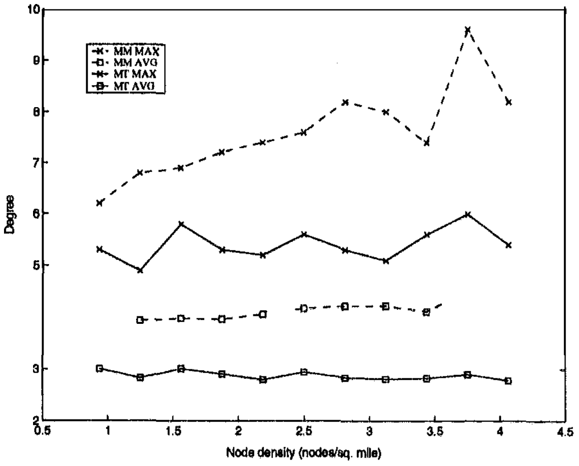

Figure 6: Degree in skewed network

the maximum power, and a smaller percentage of nodes have the maximum power as the network density increases.

In the skewed placements of nodes, our Min Total algorithm outperforms the Min Max algorithm with respect to average power by 6% -19%. We observe that the difference between average power and maximum power is larger than in uniform placements. The average power is about 50%-76% of the maximum power using Min Max algorithm of [RR00], and about 35%-64% using our algorithm. In other words, for a given average node density, the maximum power in a skewed network is higher than that in a uniformly distributed network, while the average power in skewed is lower. The reason is that in a skewed network the node density varies significantly from region to region. With a larger number of nodes in a smaller area, the average distance between two nodes is less, hence the required power levels are, on the average, smaller.

As a general rule, smaller is better in regard to node degrees in the network induced by the power assignments. In that context, in the case where nodes are uniformly distributed, the average (maximum) degree of the network with power assigned by our Min Total algorithm is consistently smaller than the average (maximum) degree of the network with power assigned by the Min Max algorithm in [RR00]. When using either of the algorithms, the average degree doesn't vary much as the network density changes. Specifically, the average degree is around 2.75 using our algorithm, which is very close to the smallest degree possible since in a 2-node-connected graph, the degree of each node must be at least 2.

The results in regard to node degrees under the skewed node distribution are similar to those for the uniform case, and are omitted due to lack of space.

6 Current Work

We are currently working on extending these results to other graph properties such as bounded node degree, bounded diameter, etc. We are also considering properties that involve more than one graph parameter (e.g. 2-Node-connected and bounded maximum degree). In the case of problems involving MAX POWER, we are working on the development of efficient algorithms for minimizing the power assigned to each transceiver after a solution that minimizes the maximum power has been obtained.

References

[CH89] W. Chen and N. Huang. "The Strongly Connecting Problem on Multihop Packet Radio Networks", ***IEEE Trans. Communication,*** Vol. 37, No. 3, Mar. 1989.

[Di67] G. A, Dirac. "Minimally 2-Connected Graphs", ***J. fur Reine und Angewandte Mathematik,*** Vol. 228, 1967, pp. 204–216.

[CLRS01] T. Cormen, C. Leiserson, R. Rivest and C. Stein. ***Introduction to Algorithms*** (second edition), MIT Press and McGraw-Hill, Cambridge, MA, 2001.

[CPS99] A. E. F. Clementi, P. Penna and R. Silvestri. “Hardness Results for the Power Range Assignment Problem in Packet Radio Networks”, *Proc. Third International Workshop on Randomization and Approximation in Computer Science* (APPROX 1999), Lecture Notes in Computer Science Vol. 1671, Springer-Verlag, July 1999, pp. 195–208.

[CPS00] A. E. F. Clementi, P. Penna and R. Silvestri. “The Power Range Assignment Problem in Packet Radio Networks in the Plane”, *Proc. 17th Annual Symposium on Theoretical Aspects of Computer Science* (STACS 2000), Feb. 2000.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*, W. H. Freeman and Co., San Francisco, CA, 1979.

[Hu93] L. Hu. “Topology Control for Multi-hop Packet Radio Networks”, *IEEE. Trans. Communications,* 41(10), 1993, pp. 1474-1481.

[KK+97] L. M. Kirousis, E. Kranakis, D. Krizanc and A. Pelc. “Power Consumption in Packet Radio Networks”, *Proc. 14th Annual Symposium on Theoretical Aspects of Computer Science* (STACS 97), Lecture Notes in Computer Science Vol. 1200, Springer-Verlag, Feb. 1997, pp. 363–374.

[KR96] S. Khuller and B. Raghavachari. “Improved Approximation Algorithms for Uniform Connectivity Problems”, *J. Algorithms*, Vol. 21, 1996, pp. 434–450.

[KV94] S. Khuller and U. Vishkin, “Biconnectivity Approximations and Graph Carvings”, *J. ACM,* Vol. 41, 1994, pp. 214–235.

[LH01] L. Li and J. Y. Halpern, “Minimum Energy Mobile Wireless Networks Revisited”, *Proc. IEEE Conference on Communications* (ICC’01), June 2001, pp. 278–283.

[LHB+01] L. Li, J. Y. Halpern, P. Bahl, Y. Wang and R. Wattenhofer, “Analysis of Cone-Based Distributed Topology Control Algorithm for Wireless Multi-hop Networks”, *Proc. ACM Principles of Distributed Computing Conference* (PODC’01), Aug. 2001.

[Pl68] M. D. Plummer. “On Minimal Blocks”, *Trans. AMS,* Vol. 134, Oct.-Dec. 1968, pp. 85–94.

[RM99] V. Radoplu and T. H. Meng. “Minimum Energy Mobile Wireless Networks”, *IEEE J. Selected Areas in Communications,* 17(8), Aug. 1999, pp. 1333-1344.

[RMM01] E. M. Royer, P. Melliar-Smith and L. Moser, “An Analysis of the Optimum Node Density for Ad hoc Mobile Networks”, *Proc. IEEE Intl. Conf. on Communication* (ICC’01), Helsinki, Finland, June 2001.

[RP01] E. M. Royer and C. Perkins, “Transmission Range Effects on AODV Multicast Communication”, *To appear in ACM Mobile Networks and Applications* (special issue on Multipoint Communication in Wireless Networks).

[RR00] R. Ramanathan and R. Rosales-Hain. “Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment”, *Proc. IEEE INFOCOM 2000,* Tel Aviv, Israel, March 2000, pp. 404–413.

[va90] J. van Leeuwen. “Graph Algorithms”, Chapter 10 in *Handbook of Theoretical Computer Science,* Vol. A, Edited by J. van Leeuwen, MIT Press and Elsevier, Cambridge, MA, 1990.

[WL+01] R. Wattenhofer, L. Li, P. Bahl and Y. Wang. “Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks”, *Proc. IEEE INFOCOM 2001,* Anchorage, Alaska, April 2001, pp. 1388–1397.

[We96] D. B. West. *Introduction to Graph Theory,* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1996.

[RW97] R. Ravi and D. P. Williamson, “An Approximation Algorithm for Minimum-Cost Vertex-Connectivity Problems”, *Algorithmica,* 18: 21-43, 1997.

[Ra96] T. S. Rappaport. *Wireless Communications: Principles and Practice,* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1996.

[TK84] H. Takagi, and L. Kleinrock. “Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals”, *IEEE Transactions on Communications,* Vol. COM-32, No. 3, pp. 246-257, March 1984. Also appears in *Multiple Access Communications, Foundations for Emerging Technologies,* Norman Abramson (Ed.), IEEE Press, pp.342-353, 1992.