# Algorithmic Aspects of Topology Control Problems for Ad hoc Networks*

Errol L. Lloyd [1] Rui Liu [1] Madhav V. Marathe [2] Ram Ramanathan [3]
S. S. Ravi [4]

**Abstract**

Topology control problems are concerned with the assignment of power values to the 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 discussed. 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 developed. 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, antenna gains at the sender and receiver, interference, noise, etc. [RR00].

*A preliminary version of this paper appeared in *Proc. Third ACM International Symposium on Mobile Ad Hoc Networking and Computing* (MobiHoc 2002), Lusanne, Switzerland, June 2002, pp. 123–134.

[1]Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716. Email: {ruliu, elloyd}@cis.udel.edu. Prepared through 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 not withstanding any copyright notation thereon.

[2] Los Alamos National Laboratory, MS M997, P.O. Box 1663, Los Alamos, NM 87545. Email: marathe@lanl.gov. Research supported by the Department of Energy under Contract W-7405-ENG-36.

[3]Internetwork Research Department, BBN Technologies, Cambridge, MA. Email: ramanath@bbn.com.

[4]Department of Computer Science, University at Albany - SUNY, Albany, NY 12222. Email: ravi@cs.albany.edu. Supported by NSF Grant CCR-97-34936.

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 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 for studying topology control problems is to make efficient use of available power at each node. In addition, using a 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 in 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 $\langle \mathbb{M}, \ \mathbb{P}, \ \mathbb{O} \rangle$. In such a specification, $\mathbb{M} \in \{\text{Dir}, \text{Undir}\}$ represents the graph model, $\mathbb{P}$ represents the desired graph property and $\mathbb{O}$ represents the minimization objective. For the problems considered in this paper $\mathbb{O} \in \{\text{MaxP}, \text{TotalP}\}$ (abbreviations of Max Power and Total Power). For example, consider the $\langle$Dir, Strongly Connected, MaxP$\rangle$ 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 $\langle$Undir, 2-Node Connected, TotalP$\rangle$ problem seeks to assign powers to the transceivers so that the resulting undirected graph has a node connectivity (see below for definition) 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 (edge) connectivity** of an undirected graph is the smallest number of nodes (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 greater than or equal to $k$, 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 $\rho$ if for every instance of the problem, the solution produced by the approximation algorithm is within the multiplicative factor of $\rho$ of the optimal solution. A **polynomial time approximation scheme** (PTAS) is an approximation algorithm that, given a problem instance and an accuracy requirement $\epsilon$, produces a solution that is within a factor $1 + \epsilon$ 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 $\langle$UNDIR, 1-NODE CONNECTED, MAXP$\rangle$ and $\langle$UNDIR, 2-NODE CONNECTED, MAXP$\rangle$. 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 $\langle$DIR, STRONGLY CONNECTED, TOTALP$\rangle$ problem [CH89, KK+97, CPS99, CPS00]. However, it is not difficult to see that their **NP**-hardness results as well as approximation algorithms also hold for the $\langle$UNDIR, 1-NODE CONNECTED, TOTALP$\rangle$ problem. 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 $\langle$DIR, STRONGLY CONNECTED, TOTALP$\rangle$ 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 ⟨DIR, STRONGLY CONNECTED, TOTALP⟩ 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 ⟨DIR, STRONGLY CONNECTED, TOTALP⟩ 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. Under a slightly different model, where there is an explicit relationship between the transmission power and distance, references [BLRS02, CMZ02] study topology control problems for connectivity properties. The complexity of several problems under this model is established in [BLRS02]. A $(1+\ln 2)$-approximation algorithm for the problem is presented in [CMZ02]. The approximation ratio is improved to 5/3 in a journal submission based on [CMZ02].

Additional related work may be found in [Hu93, RM99, WL+01, LH01, LHB+01, KKW+03].

### 3.2 Summary of Main Results

Throughout this paper, it is assumed that the power threshold values are symmetric. 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 ⟨UNDIR, $\mathbb{P}$, MAXP⟩ (⟨DIR, $\mathbb{P}$, MAXP⟩) 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. Our polynomial time algorithm can also be extended to graph properties specified by *proper functions*[5] [GW95].

2. We establish that there are non-monotone and efficiently testable properties (e.g. GRAPH IS A TREE) for which even determining whether there is a power assignment that can induce a graph with the specified property 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 properties 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, CN+03] are special cases of this general approach. As an illustration of our general approach, we present a constant factor approximation algorithm for the ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ problem. No approximation algorithm was previously known for this problem. In analyzing this approximation algorithm, we use some properties of critically 2-node connected graphs [Di67, Pl68, We96]. By a minor modification to this approximation algorithm, we also obtain a constant factor approximation algorithm for producing 2-edge-connected graphs. As in the case of minimizing maximum power, our general heuristic for approximating total power is also applicable to graph properties specified by proper functions.

[5] Given a graph $G(V, E)$, a function is $f : 2^V \rightarrow \{0, 1\}$ is *proper* if it satisfies the following two conditions: (1) $f(S) = f(V - S)$ for all $S \subseteq V$; and (2) If $A \cap B = \emptyset$, then $f(A) = f(B) = 0$ implies $f(A \cup B) = 0$.

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 graph property is both monotone and polynomial time testable. For a problem with $n$ transceivers, the algorithm uses $O(\log n)$ invocations of the algorithm to test the graph property. We also present a polynomial time approximation scheme which can, under certain circumstances, substantially reduce the number of invocations of the property testing algorithm. Next, we give an example of a non-monotone property for which the problem of minimizing the maximum power is **NP**-hard. Finally, we show that the additional requirement of minimizing the number of nodes that use the maximum power also renders the problem **NP**-hard, even for certain monotone properties. Note that both of the **NP**-hardness results utilize arbitrary power thresholds. The complexity of the problems in the geometric model (i.e., the power threshold is a function of the Euclidean distance) remains open.

## 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* $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ *and* $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$ *where the graph property* $\mathbb{P}$ *is 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 monotone and polynomial time testable graph property* $\mathbb{P}$*, the problems* $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ *and* $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$ *can be solved in polynomial time.*

**Proof:** We will present the proof for $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$. (The proof for $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ is virtually identical.)

Consider an instance of $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$. 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 $\mathbb{P}$ holds for a directed graph with $n$ nodes. Thus, the time needed to test whether property $\mathbb{P}$ 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 $\mathbb{P}$ 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 illustration of 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]), where the subset of nodes includes the sender and all the intended receivers. Connectedness of a specified subset of nodes can be seen to be a monotone property. Thus, the general approach presented above leads to a polynomial time algorithm for this property as well. In fact, the result extends to large class of network design problems that can be specified using *proper functions* [GW95, AKR95]. As noted in [GW95], the Steiner tree problem and the Steiner forest problem can be specified using this formalism. Given a network and a proper function specification, it is easy to test in polynomial time if the network satisfies the given proper function. Moreover, it is easy to see that any graph property specified using a proper function is a monotone property. Thus, our results apply to this class of network design problems as well.

We now present a polynomial time approximation scheme for $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ and $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$ problems. As a compensation for the slight deviation from the optimal value, this approach has the potential to reduce the running time substantially.

**Theorem 4.2** *Let $\mathbb{P}$ be a monotone graph property that can be tested for an $n$-node graph in time $F_{\mathbb{P}}(n)$. For any fixed $\epsilon > 0$, the problems $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ and $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$ can be approximated to within the factor $1+\epsilon$ in $O((n^2 + F_{\mathbb{P}}(n)) \log\log(max/min))$ time, where max and min are respectively the maximum and minimum power threshold values in the given problem instance.*

**Proof:** We will present the proof for $\langle$DIR, $\mathbb{P}$, MAXP$\rangle$. Since the number of power threshold values is $O(n^2)$, the values of $min$ and $max$ can be found in $O(n^2)$ time. Note that for any candidate power value (which is assigned to all the nodes), testing whether $\mathbb{P}$ holds for the induced graph can be done in $O(n^2 + F_{\mathbb{P}}(n))$ time.

Let $k$ be the smallest integer such that $(1+\epsilon)^k min \geq max$. Thus, $k = O(\log(max/min))$. Consider the following set of $k+1$ power values: $\{min, (1+\epsilon)min, (1+\epsilon)^2 min, \ldots, (1+\epsilon)^{(k-1)} min, max\}$. By doing a binary search on this set, we can determine the smallest integer $j$ such that the power value $(1+\epsilon)^j min$ causes the induced graph to have the property $\mathbb{P}$. The binary search uses $O(\log k) = O(\log\log(max/min))$ calls to the algorithm for testing $\mathbb{P}$. Thus, the running time of the algorithm is $O((n^2 + F_{\mathbb{P}}(n)) \log\log(max/min))$.

Further, since $j$ is the smallest value for which the power value $(1+\epsilon)^j min$ causes the induced graph to have the property $\mathbb{P}$, the optimal value must be at least $(1+\epsilon)^{(j-1)} min$. Thus, the solution found by the algorithm is within a factor $(1+\epsilon)$ of the optimal value. ■

When the ratio $max/min$ is substantially smaller than $2^n$, the above approximation scheme reduces the number of calls to the property testing algorithm to a value that is asymptotically smaller than $O(\log n)$.

### 4.2 Difficulty of Generalizing to Non-monotone Properties

We now show that there is a natural non-monotone graph property for which the problem of minimizing the maximum power is **NP**-hard. 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 **Exact Cover by 3-Sets** (X3C), which is known to be **NP**-complete [GJ79].

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

**Proof:** See Appendix A. ■

**Theorem 4.3** *There is a non-monotone property $\mathbb{P}$ for which* $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ *is **NP**-hard.*

**Proof:** Let $\mathbb{P}$ denote the property "$G$ IS A TREE". The **NP**-hardness of $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ follows from Lemma 4.2. ■

### 4.3 Difficulty of Minimizing the Number of Nodes of Maximum Power

An extension of $\langle$UNDIR, $\mathbb{P}$, MAXP$\rangle$ for monotone graph properties is explored in this section. 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**-hard even for certain monotone graph properties. A formal statement of the decision version of the problem is as follows.

**Max-power Users**

Instance: A positive integer $M$, a positive number $P$ (maximum allowable power value), a node set $V$, a power threshold value $p(u, v)$ for each pair $(u, v)$ of transceivers and a graph property $\mathbb{P}$.

Question: Is there 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 4.4** *There is a monotone and polynomial time testable property $\mathbb{P}$ for which the problem* **Max-power Users** *is **NP**-complete.*

**Proof:** See Appendix B. ■

## 5 A General Approach for Minimizing Total Power

### 5.1 Approximating Minimum Total Power

Topology control problems in which the minimization objective is 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

**Input:** An instance $I$ of $\langle$UNDIR, $\mathbb{P}$, TOTALP$\rangle$ 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 $\mathbb{P}$ 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 $E_c$ 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 $\mathbb{P}$ 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

section, we present a general outline for such an approximation algorithm for topology control problems of the form $\langle$UNDIR, $\mathbb{P}$, TOTALP$\rangle$. We observe that this general outline encompasses the approximation algorithm for $\langle$UNDIR, 1-NODE CONNECTED, TOTALP$\rangle$ presented in [KK+97]. Based on the general outline, we also develop an approximation algorithm with a constant performance guarantee for $\langle$UNDIR, 2-NODE CONNECTED, TOTALP$\rangle$. 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 it 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 properties such as 2-NODE CONNECTED, Step 2 cannot be done in polynomial time, unless $\mathbf{P} = \mathbf{NP}$ [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 $\langle$UNDIR, $\mathbb{P}$, TOTALP$\rangle$ where $\mathbb{P}$ 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 $\mathbb{P}$.*

*(ii) Consider the complete graph $G_c(V, E_c)$ constructed in 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 $\alpha > 0$ and $\beta > 0$ such that (a) $W(H) \leq \alpha\, OPT(I)$ and (b) $W(G') \leq \beta\, W(H)$. Then, $GTP(I) \leq 2\alpha\beta\, OPT(I)$. That is, 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 ⟨UNDIR, 1-NODE CONNECTED, TOTALP⟩ 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, $\alpha = \beta = 1$ for their approximation algorithm. Since 1-NODE-CONNECTED is a monotone property, it follows from Theorem 5.1 that the performance guarantee of 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 $\pi(u) \geq p(u, v)$ 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 $GTP(I) \leq 2\, W(G')$. 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 New Approximation Algorithms

This section presents two new approximation algorithms derived from the general approach outlined in Figure 1. These algorithms are for the two monotone properties 2-NODE CONNECTED and 2-EDGE CONNECTED respectively. The corresponding problems are denoted by ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ and ⟨UNDIR, 2-EDGE CONNECTED, TOTALP⟩.

### 5.2.1 An Approximation Algorithm for ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩

This section presents an approximation algorithm for the ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ problem. The **NP**-hardness of this problem is established in [CW03]. Our algorithm is derived from the general approach outlined in Figure 1. The following notation is used throughout this section. $I$ denotes the given instance of ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ 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 ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ 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 the instance $I$ 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 $(2 - 2/n)\, OPT(I)$. 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 $\langle$UNDIR, 2-NODE CONNECTED, TOTALP$\rangle$ 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 $\langle$UNDIR, 2-NODE CONNECTED, TOTALP$\rangle$ 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 a 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.

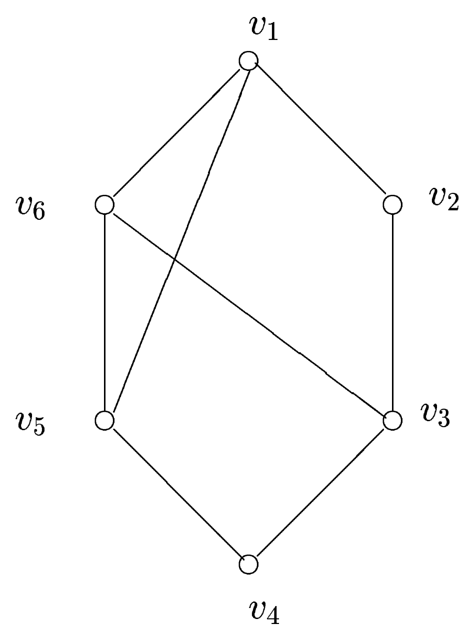



Figure 2: A simple cycle $\langle v_1, v_2, v_3, v_4, v_5, v_6, v_1\rangle$ with two chords $\{v_1, v_5\}$ and $\{v_3, v_6\}$.

**Definition 5.1** *Let $G(V, E)$ be an undirected graph. Suppose the node sequence $\langle v_1,\ v_2,\ v_3,\ \ldots,\ v_k,\ v_1\rangle$ 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[6] 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 $T$, called a **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 $E - E_T$, called a **back edge**, joins a node $u$ to an ancestor of $u$. 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 $u$. 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 $\langle$UNDIR, 2-NODE CONNECTED, TOTALP$\rangle$ 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 $w_1(u, v) = \pi^*(u)$.

[6] 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].

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

**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 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 $2\, OPT(I) - 2\, p^*$. 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* $\langle$Undir, 2-Node Connected, TotalP$\rangle$ *problem.* ■

### 5.2.2 An Approximation Algorithm for ⟨UNDIR, 2-EDGE CONNECTED, TOTALP⟩

A result analogous to Theorem 5.3 can also be obtained for ⟨UNDIR, 2-EDGE CONNECTED, TOTALP⟩ where the goal is to induce a graph that has the monotone property 2-EDGE CONNECTED. This problem has also been shown to be NP-complete in [CW03]. To obtain an approximation algorithm for this problem from the general framework, we use an approximation algorithm of Khuller and Vishkin [KV94]. Their approximation algorithm produces a 2-edge connected subgraph whose cost is at most twice that of a minimum 2-edge connected subgraph. In the notation of Theorem 5.1, we have $\beta \leq 2$. Again using the notation of Theorem 5.1, it is possible to show that $\alpha \leq (2 - 1/n)$. The proof of this result is almost identical to that for the 2-Node Connected case, except that we need an analog of Theorem 5.2. Before stating this analog, we have the following definition (which is analogous to Definition 5.2).

**Definition 5.3** *An undirected graph $G(V, E)$ is **critically** 2-EDGE-CONNECTED if it satisfies both of the following conditions. (i) $G$ is 2-EDGE-CONNECTED. (ii) For every edge $e \in E$, the subgraph of $G$ obtained by deleting the edge $e$ is* not *2-EDGE-CONNECTED.*

We can now state and prove the analog of Theorem 5.2 for critically 2-edge connected graphs.

**Lemma 5.5** *If a graph $G$ is critically 2-EDGE-CONNECTED then no cycle of $G$ has a chord.*

**Proof:** The proof is by contradiction. Suppose $G$ is critically 2-EDGE-CONNECTED but there is a cycle $C = \langle v_1, v_2, \ldots, v_r \rangle$, with $r \geq 4$, with a chord $\{v_i, v_j\}$. Consider the graph $G'$ obtained from $G$ by deleting the chord $\{v_i, v_j\}$. We will show that $G'$ is 2-EDGE-CONNECTED, thus contradicting the assumption that $G$ is critically 2-EDGE-CONNECTED.

To show that $G'$ is 2-EDGE-CONNECTED, it suffices to show that $G'$ cannot be disconnected by deleting any single edge. Consider any edge $\{x, y\}$ of $G'$, and let $G$" denote the graph created by deleting $\{x, y\}$ from $G'$. Since we deleted only one edge from $G'$, all the nodes of the cycle $C$ are in the same connected component of $G$". Thus, if we create the graph $G_1$ by adding the chord $\{v_i, v_j\}$ to $G$", the two graphs $G_1$ and $G$" have the same number of connected components. However, $G_1$ is also the graph obtained by deleting the edge $\{x, y\}$ from $G$. Since $G$ is 2-EDGE-CONNECTED, $G_1$ is connected. Thus, $G$" is also connected. We therefore conclude that $G'$ is 2-EDGE-CONNECTED, and this contradiction completes the proof of Lemma 5.5. ■

The remainder of the proof to show that $\alpha \leq (2 - 1/n)$ is identical to that for the 2-Node-Connected case. With $\alpha \leq (2 - 1/n)$ and $\beta \leq 2$, the following theorem is a direct consequence of Theorem 5.1.

**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 ⟨UNDIR, 2-EDGE CONNECTED, TOTALP⟩ problem.* ■

Our performance guarantee results are somewhat pessimistic since they are derived from a general framework. Using a different method of analysis, Calinescu and Wan [CW03] have shown recently that both of our heuristics provide a performance guarantee of 4.

As in the case of minimizing maximum power, our general framework for minimizing total power can also be used to obtain polynomial time approximation algorithms for topology control problems wherein the connectivity requirements are specified using *proper functions*. To obtain this result, we use the general method outlined in [GW95, AKR95] as the algorithm in Step 2 of our general heuristic. The method of [GW95, AKR95] gives a

2-approximation algorithm for network design problems specified using proper functions. Using the notation of Theorem 5.1, $\beta = 2$. It is also straightforward to show that the complete graph constructed in Step 1 of our heuristic has a required subgraph of weight at most the optimal solution value. In other words, $\alpha \leq 1$. Thus, we obtain a 4-approximation algorithm for the general class of problems defined in [GW95, AKR95]. An important example of a problem in this class is the Steiner variant of connectivity, where the goal is to assign power levels so as to connect only a specified subset of nodes of a graph rather than all the nodes. An approximation algorithm with a performance guarantee of $(1 + \ln \sqrt{3})$ is known for the Steiner tree problem in graphs [RZ00]. Thus, using this approximation algorithm, our approach yields a $(2 + \ln 3)$-approximation for the Steiner variant.

# 6 Experimental Results

In the preceding section, we showed that our algorithm for ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ provides a constant factor approximation. In this section, we report on the experimental performance of this algorithm. Since there are no existing approximation algorithms[7] specifically for ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩, in the experiments described here we compare the performance of our algorithm with Ramanathan and Rosales-Hain's algorithm in [RR00]. Recall that their algorithm finds an optimum solution for ⟨UNDIR, 2-NODE CONNECTED, MAXP⟩ in which the power level of each node is minimal. Our experiments were conducted using a customized implementation on both randomly generated networks and on networks derived from realistic data generated by the TRANSIMS project [TR03].

## 6.1 Randomly Generated Networks

### 6.1.1 Experimental Environment

The experimental setup used here is similar to the one described in [RR00]. The radio wave propagation model used is the *Log-distance Path Loss Model*:

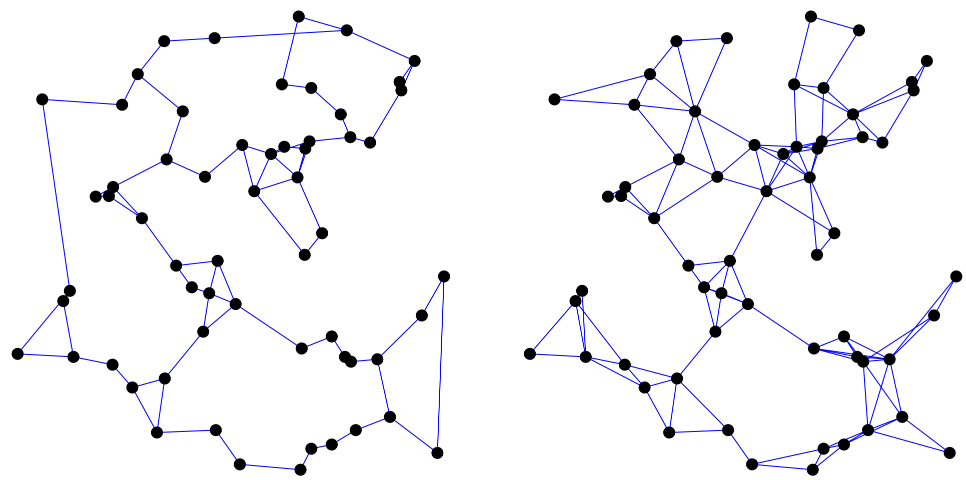
$$PL(d) \;=\; -10\log_{10}\left[\frac{G_t G_r \lambda^2}{(4\pi)^2 d_0^2}\right] + 10\eta\log_{10}\left[\frac{d}{d_0}\right]$$

where $\eta$ is the path loss exponent, $d_0$ is the close-in reference distance, $\lambda$ 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 spatial distribution of the nodes. In total there are 38 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 6.25 nodes/sq mile (10 nodes to 100 nodes) 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

[7] While this paper was under review, an algorithm that provides a constant factor approximation for the geometric version of ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩ was presented in [CW03].



(a) 2-Node Connected, Min Total (b) 2-Node Connected, Min Max

Figure 3: Examples of Network Topologies

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 approximation algorithm (Min Total) and the algorithm of [RR00] (Min Max) are run on the network consisting of those nodes. Each algorithm assigns powers to nodes 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 the nodes in the resulting network.

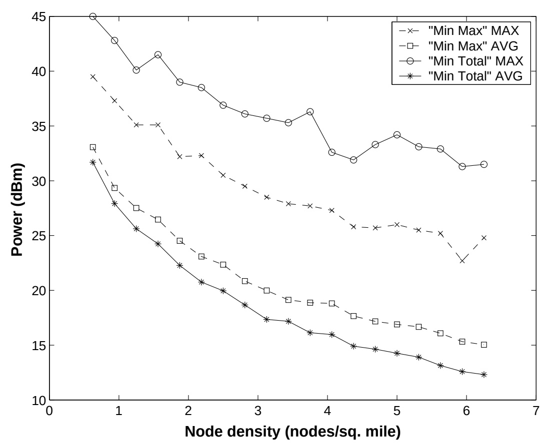
Prior to discussing the results, we first provide Figure 3 that shows the actual topologies for one simulated network with 60 nodes. Figures 3(a) and (b) are respectively the topologies resulting from our approximation algorithm (Min Total) and Ramanathan and Rosales-Hain's algorithm (Min Max).

### 6.1.2 Experimental Results and Discussion

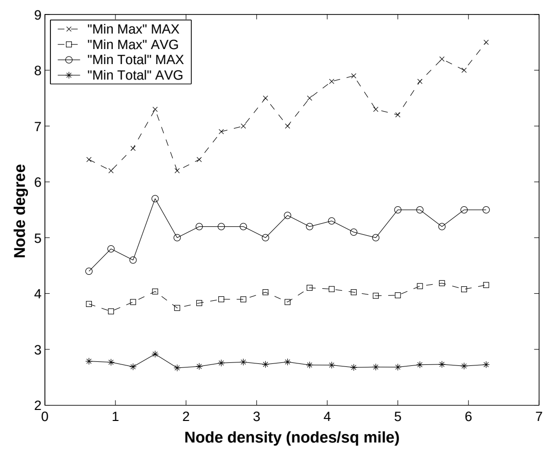
In reporting our experimental results, we plot four different quantities: (i) average power assigned to a node, (ii) maximum power assigned to any node, (iii) average degree of a node and (iv) maximum degree of all of the nodes. The experimental results on power and node degree are shown in Figure 4.

- In Figures 4 (a) and (c), "Min Max" AVG ("Min Max" MAX) and "Min Total" AVG ("Min Total" MAX) are the average (maximum) *power* using the Min Max and Min Total algorithms respectively.
- In Figures 4 (b) and (d), "Min Max" AVG ("Min Max" MAX) and "Min Total" AVG ("Min Total" MAX) are the average (maximum) *degrees* using the Min Max and the Min Total algorithms respectively.

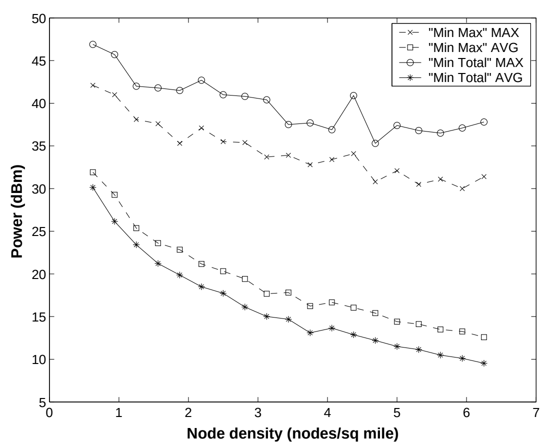
Figure 4 illustrates the results on power and node degree. In the cases where nodes are uniformly distributed, our Min Total algorithm consistently outperforms the Min Max algorithm in [RR00] in regard to average power by 5% -19%. This improvement increases as the density of the network increases. In contrast, the maximum power



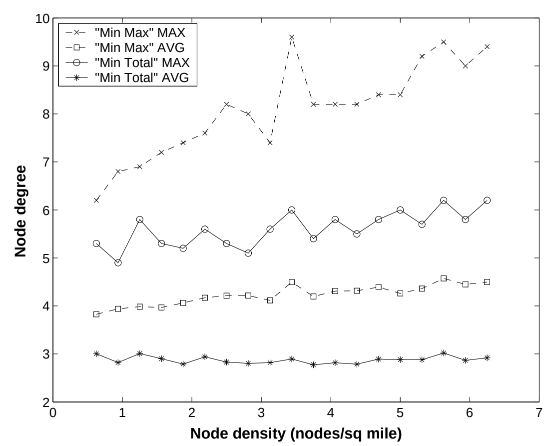

(a) Power in uniformly distributed network




(b) Degree in uniformly distributed network




(c) Power in skewed network




(d) Degree in skewed network

Figure 4: Experimental Results

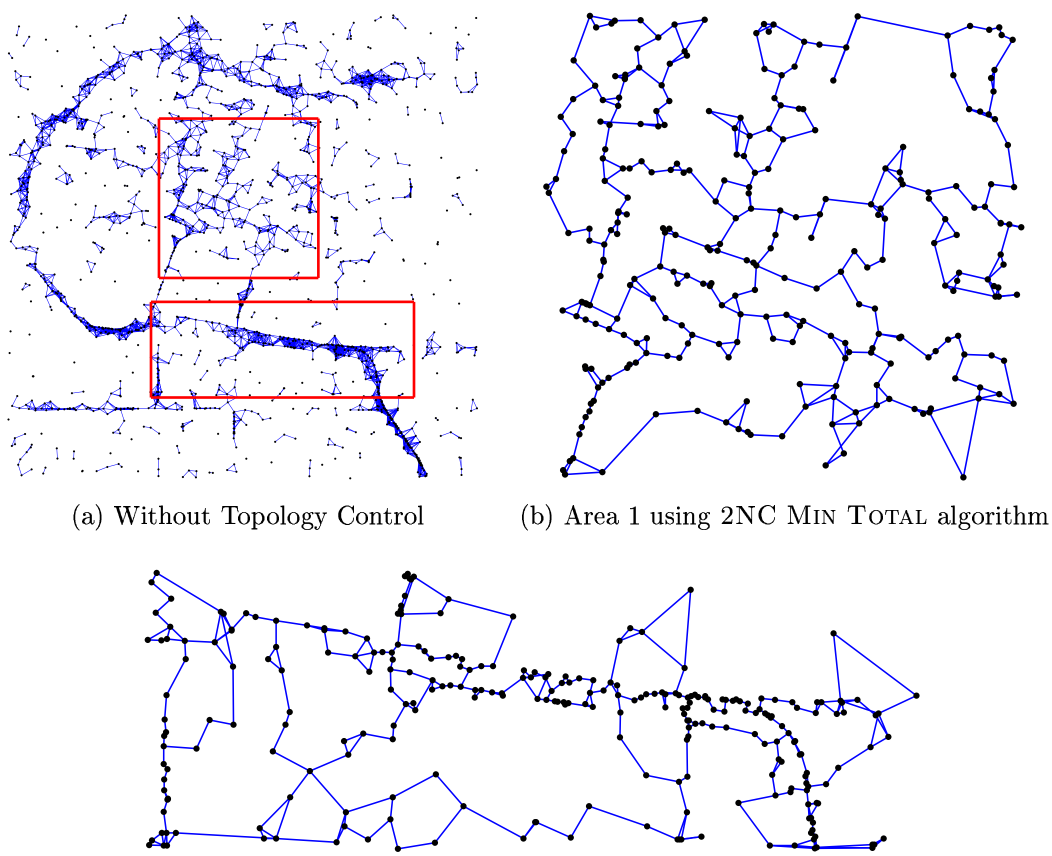
assigned by our algorithm is 14% -37% larger than that of [RR00]. The average power is about 60% -83% of the maximum power using the Min Max algorithm, and about 39% -70% using our algorithm. Those numbers decrease as the density of the network increases, which implies that the average power decreases faster than the maximum power, and a smaller percentage of nodes have the maximum power as the network density increases.

In skewed placements of nodes, our Min Total algorithm outperforms the Min Max algorithm with respect to average power by 6% -25%. We observe that the difference between average power and maximum power is larger in skewed placements than in uniform placements. The average power is about 40% -76% of the maximum power using Min Max algorithm of [RR00], and about 25% -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 the skewed network 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 (e.g. increases spatial spectrum reuse). 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 2.73 using our algorithm, which is very close to the smallest possible degree, 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.

## 6.2 The TRANSIMS Networks



(a) Without Topology Control

(b) Area 1 using 2NC MIN TOTAL algorithm

(c) Area 2 using 2NC MIN TOTAL algorithm

Figure 5: Topologies of the TRANSIMS network

In addition to the randomly generated networks, we also conducted an experimental study on a more realistic network obtained from the TRANSIMS Portland Study by the Los Alamos National Laboratory [TR03]. This data set contains locations of 1716 nodes over a 3 $km$ by 3 $km$ area. The locations were generated by carrying out a detailed simulation of the traffic in the Portland, OR, metropolitan area using the TRANSIMS simulation tool. Since the running time of our algorithm would be prohibitively high if run on all 1716 nodes, we selected two characteristic areas of this network and conducted experiments on those two areas. By so doing, the spatial effects of the network are preserved and the experimental results can be obtained in a reasonable time frame. Area 1 is a 1 $km$ by 1 $km$ square, where 284 nodes are somewhat uniformly distributed. Area 2 is a 600 meter by 1650 meter rectangle, where the majority of the 271 nodes are concentrated along a curve and the others are sparsely distributed over the remaining area. Similar to random networks, for each area, we conducted two experiments: One uses our approximation algorithm (MIN TOTAL) for ⟨UNDIR, 2-NODE CONNECTED, TOTALP⟩; the other uses Ramanathan and Rosales-Hain's algorithm (MIN MAX) for ⟨UNDIR, 2-NODE CONNECTED, MAXP⟩. However, instead of measuring transmission power, we measure the transmission range. That is because the nodes in the TRANSIMS data set are much more dense than our randomly generated networks. So, if one utilizes the propagation model we used in previous experiments, most nodes would use the minimum transmission power of 1 dbm. The results are presented in the following tables and figures.

| | Max Range | Average Range | Max Degree | Average Degree |
|---|---|---|---|---|
| MIN MAX | $158m$ | $67.75m$ | 12 | 4.80 |
| MIN TOTAL | $193m$ | $55.07m$ | 5 | 2.72 |

Table 1: Experimental Results in Area 1

| | Max Range | Average Range | Max Degree | Average Degree |
|---|---|---|---|---|
| MIN MAX | $153m$ | $73.59m$ | 28 | 7.94 |
| MIN TOTAL | $222m$ | $51.95m$ | 7 | 2.73 |

Table 2: Experimental Results in Area 2

- Table 1 and Table 2 present the experimental results for Area 1 and Area 2 respectively.
- Figure 5(a) shows the entire network of 1716 nodes given that every node has a 75 meter transmission range. The two selected areas are highlighted.
- Figures 5(b) and (c) illustrate the topologies of Area 1 and Area 2 respectively after using our algorithm (MIN TOTAL). Note that in Figure 5(b), several nodes appear not to be 2-node-connected (e.g. the node at the top middle part of the figure). The reason is that three nodes that are on or almost on a straight line, are connected to each other, and the edges between them overlap in the figure.

Our experiments with TRANSIMS data show that topology control can significantly reduce the average transmission power. In Figure 5(a), where each node has a transmission range of 75 meters, the induced graphs in Areas 1 and 2 are not even connected. After the application of MIN TOTAL algorithm, the induced graphs in both areas are 2-node-connected and the average range for the two areas is reduced to 55.07 meters and 51.95 meters respectively.

In Area 1, the average range assigned by our MIN TOTAL algorithm is 18.7% lower than that assigned by the MIN MAX algorithm, while the maximum transmission range of our algorithm is 22.2% higher than the MIN MAX algorithm. The induced maximum and average degrees are always smaller using the MIN TOTAL algorithm than using the MIN MAX algorithm. For Area 2, our MIN TOTAL algorithm assigns average range 29.4% lower, but 45%

higher maximum range. The contrast on the induced maximum and average degrees by using the two algorithms is even larger in Area 2.

These results are consistent with the experimental results on randomly generated networks. Our MIN TOTAL algorithm constantly outperforms the MIN MAX algorithm on average power (transmission range), and the margin is larger when the network is more skewed.

## 7 Directions for Future Research

Our work provides several directions for future research. First, it will be of interest to investigate whether approximation algorithms with performance guarantees better than 4 can be developed for inducing 2-node connected and 2-edge connected graphs. Second, it will be useful to consider topology control problems for other graph properties. In that direction, some complexity and approximation results for properties such as bounded diameter and lower bounds on node degrees under the objective of minimizing total power are presented in [KL+03]. A third direction is to investigate the behavior of topology control problems under the *asymmetric* power threshold model. Some results in that direction are also presented in [KL+03]. Finally, it will be of interest to develop distributed versions of algorithms for topology control problems. References [CW03, KKW+03, RR00] present some results along that direction.

**Acknowledgments:** We thank the reviewers for a careful reading of the manuscript and for providing valuable suggestions.

## References

[AKR95] A. Agrawal, P. Klein, R. Ravi, "When Trees Collide: An Approximation Algorithm for the Generalized Steiner Problem on Networks", *SIAM J. Computing*, Vol. 24, No. 3, 1995, pp. 440–456.

[BLRS02] D. M. Blough, M. Leoncini, G. Resta, and P. Santi, "On the Symmetric Range Assignment Problem in Wireless Ad Hoc Networks", *Proc. 2nd IFIP International Conference on Theoretical Computer Science*, Montreal, August 2002, pp. 71–82.

[CH89] W. Chen and N. Huang. "The Strongly Connecting Problem on Multihop Packet Radio Networks", *IEEE Trans. Communication*, Vol. 37, No. 3, Mar. 1989, pp. 293-295.

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

[CMZ02] G. Calinescu, I. Mandoiu, and A. Zelikovsky. "Symmetric Connectivity with Minimum Power Consumption in Radio Networks", *Proc. 2nd IFIP International Conference on Theoretical Computer Science*, Montreal, August 2002, pp. 119–130.

[CN+03] X. Cheng, B. Narahari, R. Simha, M. Cheng and D. Liu, *"Strong Minimum Energy Topology in Wireless Sensor Networks: NP-Completeness and Heuristics"*, IEEE Trans. on Mobile Computing, Vol. 2, No. 3, July-Sept. 2003, pp. 248–256.

[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, pp. 651–660.

[CW03] G. Calinescu and P-J Wan. "Symmetric High Connectivity with Minimum Total Power Consumption in Multihop Packet Radio Networks", *Proc. International Conference on Ad hoc and Wireless Networks* (ADHOC-NOW'03), Lecture Notes in CS, Vol. 2865, (Edited by S. Pierre, M. Barbeau and E. Kranakis), Montreal, Canada, Oct. 2003, pp. 235–246.

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

[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.

[GW95] M. Goemans, D. P. Williamson, "A General Approximation Technique for Constrained Forest Problems", *SIAM J. Computing*, Vol. 24, No. 2, 1995, pp. 296–317.

[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.

[KKW+03] M. Kubisch, H. Karl, A. Wolisz, L. Zhong and J. Rabaey, "Distributed Algorithms for Transmission Power Control in Wireless Sensor Networks", *Proc. IEEE Wireless Communications and Networking Conference* (WCNC 2003), New Orleans, LA, March 2003, pp. 558–563.

[KL+03] S. O. Krumke, R. Liu, E. L. Lloyd, M. V. Marathe, R. Ramanathan and S. S. Ravi, "Topology Control Problems Under Symmetric and Asymmetric Power Thresholds", *Proc. International Conference on Ad hoc and Wireless Networks* (ADHOC-NOW'03), Lecture Notes in CS, Vol. 2865, (Edited by S. Pierre, M. Barbeau and E. Kranakis), Montreal, Canada, Oct. 2003, pp. 187–198.

[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, pp. 264–273.

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

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

[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, pp. 857–861.

[RP01] E. M. Royer and C. Perkins, "Transmission Range Effects on AODV Multicast Communication", *ACM Mobile Networks and Applications (Monet)* (special issue on Multipoint Communication in Wireless Networks), 7(6), Dec. 2002, pp. 455–470.

[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.

[RZ00] G. Robins and A. Zelikovsky, "Improved Steiner Tree Approximation in Graphs", *Proc. 11th Ann. ACM-SIAM Symp. on Discrete Algorithms* (SODA 2000), San Francisco, CA, pp. 770–779.

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

[TR03] TRANSIMS. *http://transims.tsasa.lanl.gov/.*

[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.

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

[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.

# Appendix

## A Proof of Lemma 4.2

We first restate the lemma.

**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 $\langle$UNDIR, TREE, $*\rangle$ to denote this problem. The NP-hardness of this problem is established using a reduction from the X3C problem defined below.

**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 $|C_j| = 3$, $1 \leq j \leq m$.

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 $S$?

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

**Proof of Lemma 4.2:** In the $\langle$UNDIR, TREE, $*\rangle$ 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 $\langle$UNDIR, TREE, $*\rangle$ 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 $\langle$UNDIR, TREE, $*\rangle$ 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 $v$, $p(u, v) = p(v, u)$.)

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

For all other pairs of nodes, the power thresholds are set to 3. This completes the construction of the instance $I'$ of $\langle$UNDIR, TREE, $*\rangle$. 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 $\langle$UNDIR, TREE, $*\rangle$ 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 $\langle$UNDIR, 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 $v_1, v_2, \ldots, v_m$, 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 $C' = \{C_{j_k} \ : \ v_{j_k} \in X\}$ 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 $\langle$UNDIR, 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 adjacent to exactly 3 element nodes. No two set nodes can be adjacent to 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 adjacent to some set node. Further, no two element nodes can be adjacent to each other since the power thresholds between such nodes are 3. Thus, there are $3k + 1$ element nodes. This is a

contradiction since we know in this instance of $\langle$UNDIR, TREE, *$\rangle$, 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 must be adjacent to $R$, so one and only one of $u_i$ and $u_j$ is adjacent to a set node. Suppose it is $u_i$. We know from above that $3k$ element nodes are adjacent to 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 adjacent to $v_j$. Therefore, there are 4 nodes adjacent to $v_j$, 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. ■

# B Proof of Theorem 4.4

**Proof:** We use a reduction from SET COVERING (SC), a well-known **NP**-complete problem [GJ79].

SET COVERING (SC)

Instance: A set $S = \{x_1, x_2, \ldots, x_n\}$, a collection $C = \{C_1, C_2, \ldots, C_m\}$, where $C_i$ is a subset of $S$ $(1 \le i \le m)$, and a positive integer $K \le m$.

Question: Does there exist a subcollection $C' \subseteq C$, such that $|C'| \le K$ and the union of the sets in $C'$ is equal to $S$?

Let $\mathbb{P}$ be the property "THE DIAMETER OF $G$ IS LESS THAN OR EQUAL TO 6". This 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 [CLRS01]. Thus, **Max-power Users** is in **NP**. To prove the **NP**-hardness, we provide a reduction from SC.

Given an instance $I$ of SC, an instance $I'$ of **Max-power Users** is constructed as follows: For each element $x_i$ of $S$, create a node $u_i$ in $V$ and for each $C_i$ of $C$, create a node $v_i$ in $V$. Further, $V$ also contains four special nodes: $w$, $s_1$, $s_2$, $s_3$. The power threshold function $p$ is defined as follows. (It should be noted that the power thresholds are symmetric.)

$$
\begin{aligned}
&p(u_i, v_j) = 1 \quad \text{if } x_i \in C_j \\
&p(w, v_j) = P \quad (1 \le j \le m) \\
&p(w, s_1) = p(s_1, s_2) = p(s_2, s_3) = 1.
\end{aligned}
$$

For all other pairs 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 **Max-power Users**. It is clear that the construction can be done in polynomial time. Now, we show that there is a solution to the **Max-power Users** instance if and only if there is a solution to SC.

**If:** Suppose $C'$ is a solution to the instance of SC. We construct a power assignment $p'$ as follows.

$$
\begin{aligned}
p'(w) &= P \\
p'(v_i) &= P \quad \text{if } C_i \in C' \\
&\qquad\;\; \text{(Note: There are at most } K \text{ such nodes.)} \\
p'(x) &= 1 \quad \text{for any other node } x.
\end{aligned}
$$

We now argue that $p'$ is a solution to the instance of **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 $\mathbb{P}$

(i.e., the graph has diameter at most 6), we show that $w$ is within 3 hops of every other node. This follows from the following observations.

1. Nodes $s_1$, $s_2$, and $s_3$ are respectively 1, 2, and 3 hops away from $w$.
2. For any $C_i \in C$, if $C_i \in C'$, then the edge $\{v_i, w\} \in E$. Hence, node $v_i$ is only one hop away from $w$.
3. For any $x_i \in S$, node $u_i$ is 2 hops away from $w$, since $u_i$ is adjacent to some node $v_i$ that has an edge to $w$. (Otherwise, $C'$ doesn't cover the element $x_i$.)
4. For any $C_i \in C$, if $C_i \notin C'$, then $v_i$ is 3 hops away from $w$, since $v_i$ is adjacent to some $u_j$.

**Only if:** Suppose we have a power assignment $p'$ that is a solution to the instance of **Max-power Users**, and that $G(V, E)$ is the resulting graph. We construct a solution to the SC instance as follows. If there is an edge between $w$ and $v_i$ in $E$ (there are at most $M - 1$ such edges), then include set $C_i$ in $C'$. We claim that $C'$ is a solution to the instance of SC. Since $|C'| \leq M - 1 = K$, we need only show that $C'$ covers $S$. Since the diameter of $G(V, E) \leq 6$, $s_3$ is 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 $\{v_i, w\} \in E$, $v_i$ 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 there is an element $x_i \in S$ that is not in any set of $C'$. Then, $u_i$ is not adjacent to any node $v_j$ that is one hop away from $w$. Thus, $u_i$ must be adjacent to some node $v_j$ that is at least 3 hops away from $w$. Thus, node $u_i$ is at least 4 hops away from $w$ - a contradiction. This completes the proof of Theorem 4.4. ■