Network Lifetime and Power Assignment in ad hoc Wireless Networks

Gruia Calinescu[1], Sanjiv Kapoor[2], Alexander Olshevsky[3], and Alexander Zelikovsky[4]

[1] Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616
calinesc@iit.edu
[2] Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616.
kapoor@iit.edu
[3] Department of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332
gte864k@prism.gatech.edu
[4] Computer Science Department, Georgia State University, Atlanta, GA 30303
alexz@cs.gsu.edu

Abstract. Used for topology control in ad-hoc wireless networks, Power Assignment is a family of problems, each defined by a certain connectivity constraint (such as strong connectivity) The input consists of a *directed* complete weighted graph $G = (V, c)$. The *power* of a vertex u in a directed spanning subgraph H is given by $p_H(u) = \max_{uv \in E(H)} c(uv)$. The *power* of H is given by $p(H) = \sum_{u \in V} p_H(u)$, Power Assignment seeks to minimize $p(H)$ while H satisfies the given connectivity constraint. We present asymptotically optimal $O(\log n)$-approximation algorithms for three Power Assignment problems: Min-Power Strong Connectivity, Min-Power Symmetric Connectivity (the undirected graph having an edge uv iff H has both uv and vu must be connected) and Min-Power Broadcast (the input also has $r \in V$, and H must be a r-rooted outgoing spanning arborescence).

For Min-Power Symmetric Connectivity in the Euclidean with efficiency case (when $c(u,v) = ||u,v||^\kappa / e(u)$, where $||u,v||$ is the Euclidean distance, κ is a constant between 2 and 5, and $e(u)$ is the transmission efficiency of node u), we present a simple constant-factor approximation algorithm. For all three problems we give exact dynamic programming algorithms in the Euclidean with efficiency case when the nodes lie on a line.

In Network Lifetime, each node u has an initial battery supply $b(u)$, and the objective is to assign each directed subgraph H satisfying the connectivity constraint a real variable $\alpha(H) \geq 0$ with the objective of maximizing $\sum_H \alpha(H)$ subject to $\sum_H p_T(u)\alpha(H) \leq b(u)$ for each node $u \in V$. We are the first to study Network Lifetime and give approximation algorithms based on the PTAS for packing linear programs of Garg and Könemann. The approximation ratio for each case of Network Lifetime is equal to the approximation ratio of the corresponding Power Assignment problem with non-uniform transmission efficiency.

G. Di Battista and U. Zwick (Eds.): ESA 2003, LNCS 2832, pp. 114–126, 2003.

1 Introduction

Energy efficiency has recently become one of the most critical issues in routing of ad-hoc networks. Unlike wired networks or cellular networks, no wired backbone infrastructure is installed in ad hoc wireless networks. A communication session is achieved either through single-hop transmission or by relaying through intermediate nodes otherwise. In this paper we consider a static ad-hoc network model in which each node is supplied with a certain number of batteries and an omnidirectional antenna. For the purpose of energy conservation, each node can adjust its transmitting power, based on the distance to the receiving node and the background noise. Our routing protocol model assumes that each node periodically retransmit the hello-message to all its neighbors in the prescribed transmission range.

Formally, let $G = (V, E, c)$ be a weighted directed graph on network nodes with a *power requirement* function $c : E \to R^+$ defined on the edges. Given a *power assignment* function $p : V \to R^+$, a directed edge (u, v) is *supported* by p if $p(u) \geq c(u, v)$. The supported subgraph (sometimes called in the literature "transmission graph) H of G consists of supported edges. We consider the following network connectivity constraints (sometimes called in the literature "topology requirements") Q for the graph H: (1) strong connectivity, when H is strongly connected; (2) symmetric connectivity, when the undirected graph having an edge uv iff H has both uv and vu must be connected (3) broadcast (resp. multicast) from a root $r \in V$, when H contains a directed spanning tree rooted at r (resp. directed Steiner tree for given subset of nodes rooted at r). In this paper we start by considering the following generic optimization formulation [1,2].

Power Assignment problem. Given a power requirement graph $G = (V, E, c)$ and a connectivity constraint Q, find power assignment $p : V \to R^+$ of the minimum total power $\sum_{v \in V} p(v)$ such that the supported subgraph H satisfies the given connectivity constraint Q.

For simplicity of exposition, we use mostly the following equivalent definition of the Power Assignment problem: Given a directed spanning subgraph H, define the *power* of a vertex u as $p_H(u) = \max_{uv \in E(H)} c(uv)$ and the *power* of H as $p(H) = \sum_{u \in V} p_H(u)$. To see the equivalence, note that an optimal power assignment supporting directed spanning subgraph H never has $p(v) > \max_{uv \in E(H)} c(uv)$. Then the Power Assignment problem becomes finding the directed spanning subgraph H satisfying the connectivity constraint with minimum $p(H)$. Specifying the connectivity constraint, we obtain the following problems: Min-Power Strong Connectivity, Min-Power Symmetric Connectivity, Min-Power Broadcast, and Min-Power Multicast.

Although the Power Assignment problem formulation is quite relevant to the power-efficient routing it disregards possibly different number of batteries initially available to different nodes and, more importantly, the possibility of dynamic readjustment of the power assignment. In this paper we introduce a new power assignment formulation with a more relevant objective of maximizing the time period the network connectivity constraint is satisfied.

Formally, we assume that each node $v \in V$ is initially equipped with a battery supply $b(v)$ which is reduced by amount of $t \cdot p(v)$ for each time period t during which v is assigned power $p(v)$. A *power schedule* PT is a set of pairs (p_i, t_i), $i = 1, \ldots, m$, of power assignments $p_i : V \to R^+$ and time periods t_i during which the power assignment p_i is used. We say that the power schedule PT is *feasible* if the total amount of energy used by each node v during the entire schedule PT does not exceed its initial battery supply $b(v)$, i.e., $\sum_{i=1}^{m} t_i \cdot p_i(v) \le b(v)$.

Network Lifetime problem. Given a power requirement graph $G = (V, E, c)$, a battery supply $b : V \to R^+$ and a connectivity constraint Q, find a feasible power schedule $PT = \{(p_1, t_1), \ldots, (p_n, t_m)\}$ of the maximum total time $\sum_{i=1}^{m} t_i$ such that for each power assignment p_i, the supported subgraph H satisfies the given connectivity constraint Q.

Using the equivalent formulation, Network Life problem becomes the following linear programming problem: each directed subgraph H satisfying the connectivity constraint is assigned a real variable $\alpha(H) \ge 0$ with the objective of maximizing $\sum_H \alpha(H)$ subject to $\sum_H p_T(u)\alpha(H) \le b(u)$ for each node $u \in V$. We note that an solution with only $|V|$ non-zero variables $\alpha(H)$ exists, show that Network Life is NP-hard under several connectivity constraints, and give the first approximation algorithms for Network Life based on the PTAS for packing linear programs of Garg and Könemann [3].

The related problem considered by Cardei et al [4] has uniform unadjustable power assignments with the objective to maximize number of disjoint dominating sets in a graph. The drawback of this formulation is that dominating sets are required to be disjoint while dropping this requirement will give better solution for the original problem. S. Slijepcevic and M. Potkonjak [5] and Cardei and Du [4] discuss the construction of disjoint set covers with the goal of extending the lifetime of wireless sensor networks. The sets are disks given by the sensor unadjustable range, and the elements to be covered are a fixed set of targets. A similar problem but in a different model has been studied by Zussman and Segall [6]. They assume that the most of energy consumption of wireless networks comes from routing the traffic, rather than routing control massages. They look for the best traffic flow routes for a given set of traffic demands using concurrent flow approaches [7] for the case when nodes do not have adjustable ranges.

Besides the general case of the given power requirements graph G, we consider the following important special cases : (1) symmetric case, where $c(u, v) = c(v, u)$; 2) Euclidean case, where $c(u, v) = d(u, v)^\kappa$, where $d(u, v)$ the Euclidean distance between u and v and κ is the signal attenuation exponent, which is assumed to be in between 2 and 5 and is the same for all pairs of nodes; (3) single line case, which is the subcase of Euclidean case when all nodes lie on a single line.

We also consider the following very important way of generating an asymmetric power requirement graph G' from a given symmetric power requirement graph G. Let $e : V \to R^+$ be the *transmission efficiency* defined on nodes of G, then power requirements with non-uniform transmission efficiency $G' = (V, E, c')$ are defined as $c'(u, v) = c(u, v)/e(u)$. This definition is motivated by possible co-existence of heterogenous nodes and by our solution method for Network Lifetime. We also consider the three special cases above with non-uniform transmission efficiency, while the asymmetric power requirements case is not changed by the addition of non-uniform transmission efficiency.

Table 1. Table of upper bounds (UB) and lower bounds (LB) on the Power Assignment complexity. New results are bold. Marked by * are the folklore results, while references preceded by ** indicate the result is implicit in the respective papers.

Complexity of the Power Assignment problem						
power requirements	asymmetric		Euclidean+eff.		symmetric	
Conn. Constraints	UB	LB	UB	LB	UB	LB
Strong Conn.	**3 + 2 ln (n-1)**	**SCH**	**3 + 2 ln (n-1)**	NPH	2 [8,9]	MAX-SNP*
Broadcast	**2 + 2 ln (n-1)**	SCH	**2 + 2 ln (n-1)**	NPH	**2 + 2 ln (n-1)**	SCH [11,1]
Multicast	DST*	**DSTH**	DST*	NPH	$O(\ln n)$** [12]	SCH** [11,1]
Symmetric Conn.	**2 + 2 ln (n-1)**	**SCH**	**11.73**	NPH	$\frac{5}{3} + \epsilon$ [13]	MAX-SNPH*

We present most of our new results on Power Assignment in Table 1, together with some of the existing results. For a more comprehensive survey of existing results, we refer to [15]. We omit the case of a single line – then all enlisted problems can be solved exactly in polynomial time. More precise, without efficiency, the algorithms were folklore or appeared in [9], and with efficiency we claim polynomial time algorithms.

SCH is used to mean as hard as Set Cover; based on the Feige [16] result there is no polynomial-time algorithm with approximation ratio $(1 - \epsilon) \ln n$ for any $\epsilon > 0$ unles $P = NP$. DST means that the problem reduces (approximation-preserving) to Directed Steiner Tree and DSTH means Directed Steiner Tree reduces (approximation-preserving) to the problem given by the cell. Best known approximation ratio for Directed Steiner Tree is $O(n^\epsilon)$ for any $\epsilon > 0$ and finding a poly-logarithmic approximation ratio remains a major open problem in approximation algorithms.

Liang [22] considered some asymmetric power requirements and presented, among other results, the straightforward approximation-preserving reduction (which we consider folklore, and is implicit, for example, in [12]) of Min-Power Broadcast and Min-Power Multicast to Directed Steiner Tree. We improve the approximation ratio for Min-Power Broadcast to $2+2\ln(n-1)$. Min-Power Symmetric Connectivity and Min-Power Strong Connectivity were not considered before with asymmetric power requirements. For Min-Power Broadcast with symmetric power requirements we improve the approximation ratio from $10.8 \ln n$ of [12] to $2+2\ln(n-1)$. We remark that the method of [12] also works for Multicast with symmetric power requirements, giving a $O(\ln n)$ approximation ratio, while with asymmetric power requirements, the problem appears to be harder - it is DSTH to be precise.

The rest of the paper is organized as follows. In Section 2 we use methods designed for Node Weighted Steiner Trees to give $O(\ln n)$ approximation algorithms for Min-Power Broadcast, and Min-Power Strong Connectivity, all with asymmetric power requirements (Min-Power Symmetric Connectivity is omitted due to space limitations). In Section 3 we give constant-factor approximations for symmetric connectivity in the Euclidean with efficiency case. Section 4 deals with the Network Lifetime problem. Section 5 lists extensions of this work and some remaining open problems for Power Assignment. Due to space limitations we omit our results on lower bounds on the approximation complexity of the Power Assignment problem and dynamic programming algorithms for the case of a single line with efficiency.

2 Algorithms for Asymmetric Power Requirements

In this section we assume the power requirements are asymmetric and arbitrary. We present the algorithm for Min-Power Broadcast with an asymptotically optimal $2(1+\ln(n-1))$ approximation ratio, where n is the cardinality of the vertex set. The algorithm is greedy and we adopt the technique used for Node Weighted Steiner Trees by [17], which in turn is using an analysis of the greedy set cover algorithm different than the standard one of Chvatal [18]. The algorithm attempts to reduce the "size" of the problem by greedily adding structures.

The algorithm starts iteration i with a directed graph H_i, seen as a set of arcs with vertex set V. The strongly connected components of H_i which do not contain the root and have no incoming arc are called *unhit components*. The algorithms stops if no unhit components exists, since in this case the root can reach every vertex in H_i. Otherwise, a weighted structure which we call *spider* (details below) is computed such that it achieves the biggest reduction in the number of unhit components divided by the weight of the spider. The algorithm then adds the spider (seen as a set of arcs) to H_i to obtain H_{i+1}. For an arc $uv \in E(G)$, we use *cost* to mean $c(uv)$, the power requirement of the arc.

Definition 1. *A* spider *is a directed graph consisting of one vertex called* head *and a set of directed paths (called* legs*), each of them from the head to a (vertices called)* feet *of the spider. The definition allows legs to share vertices and arcs. The* weight *of the spider S, denoted by $w(S)$, is the maximum cost of the arcs leaving the head plus the sum of costs of the legs, where the cost of a leg is the sum of the costs of its arcs without the arc leaving the head.*

See Figure 1 for an illustration of a spider and its weight. The weight of the spider S can be higher than $p(S)$ (here we assume S is a set of arcs), as the legs of the spider can share vertices, and for those vertices the sum (as opposed to the maximum) of the costs of outgoing arcs contributes to $w(S)$. From every unhit component of H_i we arbitrarily pick a vertex and we call it a *representative.*

Definition 2. *The* shrink factor *$sf(S)$ of a spider S with head h is either the number of representatives among its feet if h is reachable (where, by convention, a vertex is reachable from itself) from the root or if h is not reachable from any of its feet, or the number of representatives among its feet minus one, otherwise.*

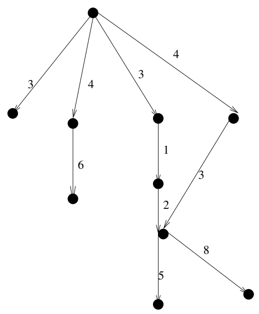

Fig. 1. A spider with four legs, weight $\max\{3, 4, 3, 4\} + 6 + (1 + 2 + 5) + (3 + 8) = 29$ and power 25.

Input: A complete directed graph $G = (V, E)$ with power requirement function $c(u, v)$ and a root vertex
Output: An directed spanning graph H (seen as a set of arcs, with $V(H) = V$) such that in H there is a path from the root to every vertex of V.

---

(1) Initialize $H = \emptyset$
(2) While H has at least one unhit component
  (2.1) Find the spider S which minimizes $w(S)/(sf(S))$ with respect to H
  (2.2) Set $H \leftarrow H \cup S$

Fig. 2. The Greedy Algorithm for Min-Power Broadcast with asymmetric power requirements

Our algorithm appears in Figure 2. We describe later the detailed implementation of Step 2.1 of the algorithm. Let $u(H)$ be the number of unhit components of direct graph H. Due to space limitations, we omit the proof of the next lemma:

Lemma 1. *For a spider S (seen as a set of arcs), $u(H_i \cup S) \leq u(H_i) - sf(S)$.*

Fact 1 *Given a spider S (seen as a set of arcs), $p(H_i \cup S) \leq p(H_i) + w(S)$.*

Next we describe how to find the spider which minimizes its weight divided by its shrink factor. In fact, we search for *powered* spiders, which besides head h and legs have a fixed power $p(h)$ associated with the head. The weight of the powered spider S', denoted by $w(S')$, equals $p(h)$ plus the sum of costs of the legs (where as before the cost of a leg is the sum of the costs of its arcs without the arc leaving the head). Given a spider one immediately obtains a powered spider of the same weight, while given a powered spider S', the spider S obtained from S' by keeping only the edges of S' (thus ignoring the fixed power of the head) satisfies $w(S) \leq w(S')$.

We try all possible heads h, and all possible discrete power for the head (there are at most n such discrete power values - precisely the values $c(hu)$ for every $u \in G$, where $c(hh) = 0$ by convention). Define the *children* of the head to be the vertices within its power value - where the head is also considered a child. For each representative r_i, compute the shortest path P_i from a child of h to r_i. If h is not reachable from the root, partition the representatives in two sets - R_1 which cannot reach h and R_2 which can reach h; otherwise let $R_1 = R$ and $R_2 = \emptyset$. Sort R_1 and R_2 such that the lengths of the paths P_i are in nondecreasing order. Then the best spider with head h and the given power value can be obtained by trying all $0 \leq j_1 \leq |R_1|$ and $0 \leq j_2 \leq |R_2|$ and taking the paths P_i leading to the first j_1 representatives of R_1 and the first j_2 representatives of R_2.

The following lemma shows the existence of a good spider; it is a counterpart of Lemma 4.1 and Theorem 3.1 of [17]. Let OPT denote the value of the optimum solution.

Lemma 2. *Given any graph H_i and set of representatives obtained from H_i, there is a spider S such that $\frac{w(S)}{sf(S)} \leq 2\frac{OPT}{u(H_i)}$.*

Proof. Let T be the optimum arborescence outgoing from the root and R the set of representatives obtained from H_i; $|R| = u(H_i)$. Traverse T in postorder and whenever a vertex v is the ancestor of at least two representatives (where by default every vertex is an ancestor of itself) define a spider with head v and legs given by the paths of T from v to the representatives having v as an ancestor. Remove v and its descendents from T, and repeat. The process stops if the number of remaining representatives is less than two. If there is one representative left, define one last spider with the head the root and one leg to the remaining representative. Let S_i, for $1 \leq i \leq q$ be the spiders so obtained.

It is immediate that $w(S_1) + w(S_2) + \ldots + w(S_q) \leq OPT$. If $r(S_i)$ is the number of representatives in spider S_i, we have that $r(S_1)+r(S_2)+\ldots+r(S_q) = |R|$. Note that $r(S_i) \leq 2sf(S_i)$, as except for the spider with the root as its head (for which $r(S_i) = sf(S_i)$) $2 \leq r(S_i) \leq sf(S_i) + 1$. We conclude that $2(sf(S_1) + sf(S_2) + \ldots + sf(S_q)) \geq |R| = u(H_i)$. The spider with highest ratio among S_j, $1 \leq j \leq q$, has $\frac{w(S_j)}{2sf(S_j)} \leq \frac{OPT}{u(H_i)}$.

Theorem 1. *The algorithm described in this subsection has approximation ratio $2(1 + \ln(n-1))$ for Min-Power Broadcast with asymmetric power requirements.*

Proof. Let q_i be the number of unhit components of H_i (where H_0 is the initial graph with no edges), S_i be the spider picked to be added to H_i, $d_i = sf(S_i)$, and $w_i = w(S_i)$.

From Lemma 1, we have: $q_{i+1} \leq q_i - d_i$. Since the algorithm is greedy, by Lemma 2, $\frac{w_i}{d_i} \leq \frac{2OPT}{q_i}$. Plugging equation the above equations into each other and rearranging the terms , it follows that $q_{i+1} \leq q_i - d_i \leq q_i(1 - \frac{w_i}{2OPT})$. Assuming there are m steps, this implies that $q_{m-1} \leq q_0 \prod_{k=0}^{m-2}(1 - \frac{w_k}{2OPT})$ Taking natural logarithm on both sides and using the inequality $\ln(1 + x) \leq x$, we obtain that $\ln \frac{q_0}{q_{m-1}} \geq \frac{\Sigma_{k=0}^{m-2} w_k}{2OPT}$ However, $q_{m-1} \geq 1$ and $q_0 = n-1$ so that $2OPT \ln(n-1) \geq \Sigma_{k=0}^{m-2} w_k$

The weight of the last spider can be bounded as $w_{m-1} \leq 2OPT$ from Lemma 2. Finally, since $APPROX \leq \sum_{k=0}^{m-1} w_k$, which follows from Fact 1, we have that $APPROX \leq 2(1 + \ln\,(n-1))OPT$.

2.1 Min-Power Strong Connectivity with Asymmetric Power Requirements

In this subsection we use the previous result to give an approximation algorithm for Min-Power Strong Connectivity with asymmetric power requirements. Let v be an arbitrary vertex. An optimum solution of power OPT contains an outgoing arborescence A_{out} rooted at v (so $p(A_{out}) \leq OPT$) and an incoming arborescence A_{in} rooted at v (so $c(A_{in}) = p(A_{in}) \leq OPT$).

The broadcast algorithm in the previous subsection produces an outgoing arborescence B_{out} rooted at v with $p(B_{out}) \leq 2(1+\ln(n-1))p(A_{out})$. Edmonds' algorithm produces a minimum cost arborescence B_{in} rooted at v with $c(B_{in}) \leq c(A_{in})$. Then $p(B_{out} \cup B_{in}) \leq p(B_{out}) + c(B_{in}) \leq 2(1 + \ln(n-1))p(A_{out}) + c(A_{in}) \leq (2\ln(n-1)+3)OPT$. Therefore we have

Theorem 2. *There is a* $2\ln(n-1)+3$*-approximation algorithm for Strong Connectivity with asymmetric power requirements.*

We mention that Min-Power Unicast with asymmetric power requirements is solved by a shortest paths computation. Min-Power Symmetric Unicast (where the goal is to obtain the minimum power undirected path connecting two given vertices) with asymmetric power requirements can also be solved in $O(n^2 \log n)$ by a shortest paths computation in a specially constructed graph described in Section 4 of [2]. Algorithms faster than $O(n^2)$ are not known for Min-Power Symmetric Unicast even in the simplest Line case.

3 Min-Power Symmetric Connectivity in the Euclidean-with-Efficiency Case

In this section we present a constant-ratio algorithm for Min-Power Symmetric Connectivity when power requirements are in the Euclidean-with-efficiency model: $c(uv) = d(u,v)^\kappa / e(u)$, where d is the Euclidean distance and $2 \leq \kappa \leq 5$.

The algorithm is very simple: for any unordered pair of nodes uv define $w(u,v) = c(u,v) + c(v,u)$ and compute as output a minimum spanning tree M in the resulting weighted undirected graph.

We prove the algorithm above (which we call the *MST* algorithm) has constant approximation ratio using only the fact that d is an arbitrary metric (as for example in the three dimensional Euclidean case).

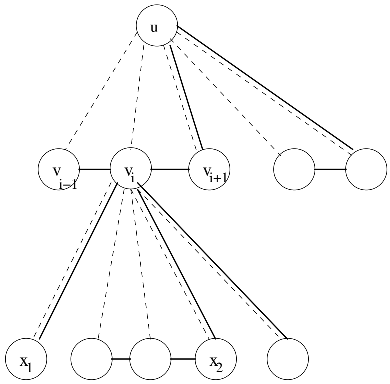

Fig. 3. An illustration of the transformation from T (the dotted lines) to T_r (given by solid lines).

For any tree T, let $w(T) = \sum_{(u,v)\in T} w(u,v)$. Note that

$$w(T) = \sum_{v\in V} \sum_{y:(v,y)\in T} c(v,y) \geq \sum_{v\in V} \max_{y:(v,y)\in T} c(v,y) = p(T).$$

Let T be an arbitrary spanning tree of G. We arbitrarily pick a root for T. For each node u with $k(u)$ children, we sort the children $v_1, v_2, \ldots, v_{k(u)}$ such that $d(u,v_i) \geq d(u,v_{i+1})$. With a fixed parameter $r > 1$ (to be chosen later), we modify T in a bottom-up manner by replacing, for each $1 \leq i < k(u)$, each edge (u,v_i) with (v_i,v_{i+1}) if $d(u,v_i) \leq r \cdot d(u,v_{i+1})$ (see Figure 3). We denote by T_r the rooted resulting tree. Our main lemma (whose proof we omit due to space constraint) below relates the weight of T_r to the power of T:

Lemma 3. *For any rooted tree T, $w(T_r) \leq \left(2^\kappa + (r+1)^\kappa + \frac{r^\kappa}{r^\kappa - 1}\right) p(T)$*

Note that $p(MST) \leq w(MST) \leq w(T_r) \leq \left(2^\kappa + (r+1)^\kappa + \frac{r^\kappa}{r^\kappa-1}\right) p(T)$, where T is the minimum power tree.

Theorem 3. *The approximation ratio of the MST algorithm is at most $\min_{r>1}\{2^\kappa + (r+1)^\kappa + \frac{r^\kappa}{r^\kappa-1}\}$*

Numerically obtained, this approximation ratio is (i) 11.73 for $\kappa = 2$, achieved at $r = 1.32$ (ii) 20.99 for $\kappa = 3$, achieved at $r = 1.15$; (iii)38.49 for $\kappa = 4$, achieved at $r = 1.08$ (iv) 72.72 for $\kappa = 5$, achieved at $r = 1.05$.

4 Network Lifetime

In this section we first show that the Network Lifetime problem is NP-Hard for symmetric power requirements and each considered connectivity constraint:

strong connectivity, symmetric connectivity and broadcast. Then we show how the Garg-Köneman PTAS [3] PTAS can be used for reducing Network Lifetime to Power Assignment. In the following we drop mentioning the specific connectivity constraint when the discussin applies to all possible connectivity constraints.

Recall that the Network Lifetime problem has as input a power requirement graph $G = (V, E, c)$ and a battery supply vector $b : V \to R^+$. A set $\mathcal{S}$ of directed spanning subgraphs of G is given implicitly by the connectivity constraints. In general, $|\mathcal{S}|$ is exponential in $|V|$. Then Network Lifetime is the following packing linear program: Maximize $\sum_{H\in\mathcal{S}} x_H$ subject to $\sum_{H\in\mathcal{S}} p_H(v)x_H \leq b(v),\ \forall v \in V$, $x_H \geq 0, \forall H \in \mathcal{S}$.

We note that an optimum vertex solution only uses $|V|$ non-zero variables x_H. With potentially exponential number of columns, it is not surprising the following theorem, whose proof uses an idea from [4] and is ommited due to space limitations, holds:

Theorem 4. *Even in the special case when all the nodes have the same battery supply, the Network Lifetime for Symmetric Connectivity (or Broadcast or Strong Connectivity) problem is NP-hard in the symmetric power requirements case.*

The Network Lifetime linear program above is a packing LP. In general, a packing LP is defined as

$$\max\{c^T x | Ax \leq b, x \geq 0\} \tag{1}$$

where A, b, and c have positive entries; we denote the dimensions of A as nxl. In our case the number of columns of A is prohibitively large (exponential in number of nodes) and we will use the $(1 + \epsilon)$-approximation Garg-Köneman algorithm [3]. The algorithm assumes that the LP is implicitly given by a vector $b \in R^n$ and an algorithm which finds the column of A minimizing so-called length. The *length* of column j with respect to LP in Equation (1) and non-negative vector y is defined as $length_y(j) = \frac{\Sigma_{i=1}^n A(i,j)y(i)}{c(j)}$.

We cannot directly apply the Garg-Köneman algorithm because, as we notice below, the problem of finding the minimum length column is NP-Hard in our case, and we can only approximate the minimum length column. Fortunately, it is not difficult to see that when the Garg-Köneman $(1 + \epsilon)$-approximation algorithm uses f-approximation minimum length columns it gives an $(1 + \epsilon)f$-approximation solution to the packing LP (1) [19][1].

The Garg-Köneman algorithm with f-approximate columns is presented in Figure 4. When applied to the Network Lifetime LP, it is easy to see that the problem of finding the minimum length column, corresponds to finding the minimum power assignment with transmission efficiencies inverse proportional to the elements of vector y, i.e., for each node $i = 1, \ldots, n$, $e(i) = 1/y_i$. This implies the following general result.

[1] Although this complexity aspect has not been published anywhere in literature, it involves only a trivial modification of [3] and will appear in its journal version [19].

Input: A vector $b \in R^n$, $\epsilon > 0$, and an f-approximation algorithm F for the problem of finding the minimum length column $A_{j(y)}$ of a packing LP $\{\max\ c^T x | Ax \le b, x \ge 0\}$
Output: A set of S of columns of A: $\{A_j\}_{j\in S}$ each supplied with the value of the corresponding variable x_j, such that x_j, for $j \in S$, are all non-zero variables in a feasible approximate solution of the packing LP $\{\max\ c^T x | Ax \le b, x \ge 0\}$

---

(1) Initialize: $\delta = (1+\epsilon)((1+\epsilon)n)^{-1/\epsilon}$, for $i = 1, \ldots, n$ $y(i) \leftarrow \frac{\delta}{b(i)}$, $\quad D \leftarrow n\delta$, $S \leftarrow \emptyset$.
(2) While $D < 1$
  Find the column A_j $(j = j(y))$ using the f-approximate algorithm F.
  Compute p, the index of the row with the minimum $\frac{b(i)}{A_j(i)}$
  if $j \notin S$ $\quad x_j \leftarrow \frac{b(p)}{A_q(p)}$ else $\quad x_j \leftarrow x_j + \frac{b(p)}{A_q(p)}$
  $S \leftarrow S \cup \{j\}$
  For $i = 1, \ldots, n$, $y(i) \leftarrow y(i)\left(1 + \epsilon \frac{b(p)}{A_j(p)} / \frac{b(i)}{A_j(i)}\right)$, $\quad D \leftarrow b^T y$.
(3) Output $\{(j, \frac{x_j}{\log_{1+\epsilon} \frac{1+\epsilon}{\delta}})\}_{j\in S}$

Fig. 4. The Garg-Köneman Algorithm with f-approximate minimum length columns

Theorem 5. *For a connectivity constraint and a case of the power requirements graph, given an f-approximation algorithm F for Power Assignment with the given connectivity constraint and the case of the power requirements graph with added non-uniform efficiency, there is a $(1+\epsilon)f$-approximation algorithm for the corresponding Network Lifetime problem.*

The above theorem implies approximation algorithms for the Network Lifetime problem in the cases for which we developed approximation algorithms for the Power Assignment problem with nonuniform efficiency (see Table 1).

5 Conclusions

We believe the following results hold, but their exposition will complicate this long paper too much:

1. Min-Power Steiner Symmetric Connectivity with asymmetric power requirements, in which a given set of terminals must be symmetrically connected, can also be approximated with a $O(\log n)$ ratio using a spider structure similar to the one used for broadcast, but with a "symmetric" weight, and a greedy algorithm.
2. The algorithms for Node Weighted Steiner Tree of Guha and Khuller [20] can also be adapted (but in a more complicated way, as they are more complicated than [17]) to obtain, for any $\epsilon > 0$, algorithms with approximation ratio of $(1.35+\epsilon)\ln n$ for Min-Power Symmetric Connectivity, Min-Power Steiner Symmetric Connectivity, Min-Power Broadcast, and Min-Power Strong Connectivity with asymmetric power requirements.

We leave open the existance of efficient exact or constant factors algorithm for Min-Power Broadcast or Min-Power Strong Connectivity in the Euclidean with efficiency case. We also leave open the NP-Hardness of Network Life in Euclidean cases.

Another special case is when nodes have non-uniform "sensitivity" $s(v)$. Even in the Line-with-sensitivity case, when $c(u, v) = ||u, v||^{\kappa}/s(v)$, we do not know algorithms better than the general $O(\log n)$ algorithms from Section 2. Adding non-uniform sensitivity to symmetric power requirements results in Power Assignment problems as hard as set cover.

References

1. P. Wan, G. Calinescu, X.-Y. Li, and O. Frieder, "Minimum energy broadcast in static ad hoc wireless networks," *Wireless Networks*, 2002.
2. G. Calinescu, I. Mandoiu, and A. Zelikovsky, "Symmetric connectivity with minimum power consumption in radio networks," in *Proc. 2nd IFIP International Conference on Theoretical Computer Science,(TCS 2002)*, R. Baeza-Yates and U. Montaniri and N. Santoro (eds.), Kluwer Academic Publ., August 2002, 119–130.
3. N. Garg and J. Könemann, "Faster and simpler algorithms for multicommodity flow and other fractional packing problems," in *Proceedings of FOCS*, 1997.
4. D.-Z. D. M. Cardei, "Improving wireless sensor network lifetime through power aware organization," *Submitted to ACM Wireless Networks.*
5. S. Slijepcevic and M. Potkonjak, Power Efficient Organization of Wireless Sensor Networks, IEEE International Conference on Communications (ICC), Helsinki, June 2001, pp. 472–476.
6. G. Zussman and A. Segall, "Energy efficient routing in ad hoc disaster recovery networks," in *IEEE INFOCOM'03*, 2003.
7. F. Leighton and S.Rao, "Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms," *Journal of the ACM*, vol. 6, pp. 787–832, 1999.
8. W. Chen and N. Huang, "The strongly connecting problem on multihop packet radio networks," *IEEE Transactions on COmmunications*, vol. 37, pp. 293–295, 1989.
9. L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc, "Power consumption in packet radio networks," *Theoretical Computer Science*, vol. 243, pp. 289–305, 2000, preliminary version in STACS'97.
10. A. E. Clementi, P. Penna, and R. Silvestri, "On the power assignment problem in radio networks," *Electronic Colloquium on Computational Complexity*, vol. Report TR00-054, 2000, preliminary results in APPROX'99 and STACS'2000.
11. A. Clementi, P. Crescenzi, P. Penna, G. Rossi, and P. Vocca, "On the complexity of computing minimum energy consumption broadcast subgraphs," in *18th Annual Symposium on Theoretical Aspects of Computer Science, LNCS 2010*, 2001, pp. 121–131.
12. I. Caragiannis, C. Kaklamanis and P. Kanellopoulos, "New results for energy-efficient broadcasting in wireless networks," in *ISAAC'2002*, 2002, pp. 332–343.
13. E. Althaus, G. Calinescu, I. Mandoiu, S. Prasad, N. Tchervenski and A. Zelikovsky, "Power efficient range assignment in ad-hoc wireless networks," *WCNC'03*, 2003, pp. 1889–1894.

14. D. Blough, M. Leoncini, G. Resta, and P. Santi, "On the symmetric range assignment problem in wireless ad hoc networks," in *2nd IFIP International Conference on Theoretical Computer Science (TCS 2002).* Kluwer Academic Publishers, 2002, pp. 71–82.
15. A. Clementi, G. Huiban, P. Penna, G. Rossi, and Y. Verhoeven, "Some recent theoretical advances and open questions on energy consumption in ad-hoc wireless networks," in *Proc. 3rd Workshop on Approximation and Randomization Algorithms in Communication Networks (ARACNE)*, 2002.
16. U. Feige, "A threshold of $\ln n$ for approximating set cover," *Journal of the ACM*, vol. 45, pp. 634–652, 1998.
17. P. Klein and R.Ravi, "A nearly best-possible approximation algorithm for node-weighted steiner trees," *Journal of Algorithms*, vol. 19, pp. 104–115, 1995.
18. V. Chvatal, "A greedy heuristic for the set covering problem," *Mathematics of Operation Research*, vol. 4, pp. 233–235, 1979.
19. J. Könneman, "Personal communication."
20. S. Guha and S. Khuller, "Improved methods for approximating node weighted steiner trees and connected dominating sets," *Information and Computation*, vol. 150, pp. 57–74, 1999.
21. M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* New York: Freeman, 1979.
Numerische Mathematik, vol. 1, pp. 269–271, 1960.
22. W. Liang, "Constructing Minimum-Energy Broadcast Trees in Wireless Ad Hoc Networks" *MOBIHOC'02*, 112–122, 2002.