Approximating Maximum Integral Flows in Wireless Sensor Networks via Weighted-Degree Constrained k-Flows

Zeev Nutov
The Open University of Israel
nutov@openu.ac.il

ABSTRACT

We consider the Maximum Integral Flow with Energy Constraints problem: given a directed graph $G = (V, E)$ with edge-weights $\{w(e) : e \in E\}$ and node battery capacities $\{b(v) : v \in V\}$, and two nodes $r, s \in V$, find a maximum integral rs-flow f so that for every node v its energy consumption $\sum_{vu \in E} f(vu)w(vu)$ is at most $b(v)$. Let k be the maximum integral flow value. We give a polynomial time algorithm that computes a flow of value at least $\lfloor k/16 \rfloor$. As checking whether $k \geq 1$ can be done in polynomial time, this gives an approximation algorithm with ratio that approaches $1/16$ when k is large, and is not worse than $1/31$. This is the first constant ratio approximation algorithm for this problem, which solves an open question from [2]. This result is based on a bicriteria approximation algorithm for a more general problem, in which we seek a minimum cost set of k pairwise edge-disjoint rs-paths (that is, a k-flow) subject to weighted degree constraints. We give a polynomial time algorithm that computes a flow of value k and violates the weighted degrees by a factor at most 4. This result is of independent interest.

Categories and Subject Descriptors

G.2.2 [**Mathematics of Computing**]: Discrete Mathematics—*Graph Theory, Network problems*

General Terms

Algorithms, Design

1. INTRODUCTION

1.1 Problem definition and motivation

In many Network Design problems we seek a subgraph H with prescribed properties that minimizes/maximizes a certain objective function. Such problems are vastly studied in Combinatorial Optimization and Approximation Algorithms. Some known examples are Max-Flow, Min-Cost k-Flow, Maximum b-Matching, Minimum Spanning/Steiner Tree, and many others. See, e.g., [17, 9].

Many of these problems are motivated by applications for *wired networks*, but recently problems arising from *wireless networks* received a lot of attention due to their extensive applications. A *wireless sensor* contains a battery whose energy is used to transmit messages to other sensors. As the battery capacity of a sensor is limited, it is crucial to use a strategy that maximizes the lifetime of the network. We study the situation where sensors are deployed in the field to gather data (e.g., military, medical, traffic, etc., c.f., Zhao and Guibas [21]) and then relay the data packets via other sensors back to a base station s. It is desirable to get as many data packets as possible from the sensors to the base station, before some of the sensor batteries are depleted.

Many papers considered *fractional flows* when splitting of packets into fractional portions is allowed. This version admits an easy polynomial time algorithm via linear programming, c.f., [3, 5, 10, 12, 16, 20]. As data packets are usually quite small, there are situations where splitting of packets into fractional ones is not desirable nor practical. We consider a model where data packets are considered as units that cannot be split, i.e., when the packet flows are of *integral* values only.

We can augment the network with a super source node r and connect it to all the source nodes with zero energy. We can then view the network as having a single source r and a single sink s. The goal is to maximize the rs-flow subject to the battery energy constraints. Formally, we obtain the following problem, which is the "wireless variant" of the classic Max-Flow problem. For a graph $H = (V, I)$ and a node $v \in V$, let $\delta_H(v) = \delta_I(v)$ denote the set of edges leaving v in H, and let $\delta_H^{in}(v) = \delta_I^{in}(v)$ denote the set of edges entering v in H. An rs-flow in a graph $G = (V, E)$ is a function f that assigns to every edge $e \in E$ a nonnegative real number $f(e)$ so that the flow conservation constraints hold:

$$\sum_{e \in \delta_E^{in}(v)} f(e) = \sum_{e \in \delta_E(v)} f(e) \quad \text{for all } v \in V - \{r, s\} .$$

Namely, the flow entering v equals the flow leaving v. Assuming no flow is entering r or leaving s, the value of a flow f is the total amount $\sum_{e \in \delta_E(r)} f(e)$ of flow leaving r, which equals the total amount $\sum_{e \in \delta^{in}(s)} f(e)$ of flow entering s. A flow is integral if $f(e)$ is integral for all $e \in E$. Henceforth we consider integral flows only, hence "flow" means "integral flow". It is well known that any (integral) rs-flow f of value k can be decomposed into k rs-paths so that $f(e)$ is at least the number of path containing e.

DialM-POMC'08, August 22, 2008, Toronto, Ontario, Canada.

We consider the following problem:

Maximum Integral Flow with Energy Constraints
Instance: A directed graph $G = (V, E)$ with edge-weights $\{w(e) : e \in E\}$, battery capacities $\{b(v) : v \in V\}$, and two nodes $r, s \in V$.
Objective: Find a maximum (integral) rs-flow f that satisfies the energy constraints

$$\sum_{e \in \delta_E(v)} f(e)w(e) \leq b(v) \quad \text{for all } v \in V \ . \tag{1}$$

We observe that for unit weights the problem is easily reduced to a Max-Flow problem with node-capacities, and thus is solvable in polynomial time. Furthermore, if for every node v all the edges in $\delta_G(v)$ have the same cost, then the problem is easily reduced to the case of unit weights, and thus is also solvable in polynomial time. However, as was shown in [2], the problem becomes APX-hard even if the weights can take only two values. Furthermore, it was also shown in [2] that the problem is strongly NP-hard for geometric configurations on the real line, where $w(uv)$ is the square of the distance between u and v. Hence it seems that a constant ratio approximation algorithm is the best one can expect, namely, the problem is unlikely to admit a PTAS.

Maximum Integral Flow with Energy Constraints belongs to a class of Maximum Network Lifetime problems. In these problems, every node v has a limited battery capacity $b(v)$, and a transmission energy $w(vu)$ to any other node u is known. In transmission round i, we choose a subnetwork H_i with given properties, and every node transmits one message to each one of its neighbors in H_i; in many applications, each H_i is an arborescence (see [4]), and in our case each H_i is an rs-path. The goal is to maximize the *lifetime* of the network, that is to find a maximum length feasible sequence $H_1, H_2, \ldots, H_k$ of subnetworks; feasibility means that every graph H_i satisfies the required properties, and that for every node v the total transmission energy during all rounds is at most $b(v)$. This is the *Multiple Topology* version of the problem. In the *Single Topology* variant, all the networks H_i are identical, c.f., [4] for more details.

1.2 Our results

It was shown in [2] that Maximum Integral Flow with Energy Constraints is APX-hard. We give the first constant ratio algorithm for the problem, thus solving an open problem from [2].

THEOREM 1.1. *For an instance of* Maximum Integral Flow with Energy Constraints *problem let k be the maximum integral flow value. There exist a polynomial time algorithm that checks if $k \geq 1$ and finds a flow of value at least $\lfloor k/16 \rfloor$. Consequently, the problem admits an approximation algorithm with ratio $\frac{1}{k}$ for $k \leq 16$, and $\frac{1}{16}(1 - \frac{(k \bmod 16)}{k}) \geq \frac{1}{31}$ for any $k \geq 17$; the ratio approaches $1/16$ when k is large, and is not worse than $1/31$.*

Theorem 1.1 is based on a bicriteria approximation algorithm for a more general problem, in which we seek a minimum cost set of k pairwise edge-disjoint rs-paths (that is, a k-flow) subject to weighted degree constraints. This problem is of independent interest. Given a graph $H = (V, F)$ with edge weights $\{w(e) : e \in F\}$, the *weighted degree* of a node $v \in V$ is $w(\delta_H(v)) = \sum_{e \in \delta_H(v)} w(e)$.

Weighted-Degree Constrained Min-Cost k-Flow
Instance: A directed graph $G = (V, E)$ with edge-costs $\{c(e) : e \in E\}$, edge-weights $\{w(e) : e \in E\}$, degree bounds $\{b(v) : v \in V\}$, two nodes $r, s \in V$, and an integer k.
Objective: Find a minimum cost subgraph $H = (V, F)$ of G that contains k pairwise edge-disjoint rs-paths and satisfies the weighted degree constraints

$$w(\delta_H(v)) \leq b(v) \quad \text{for all } v \in V \ . \tag{2}$$

Let τ^* denote the optimal value of the natural LP-relaxation for Weighted Degree Constrained Min-Cost k-Flow that seeks to minimize $c \cdot x$ over the following polytope P_k:

$$\begin{array}{rcll} x(\delta_E^{in}(S)) & \geq & k & \text{for all } s \in S \subset V - r \\ \sum_{e \in \delta_E(v)} x(e)w(e) & \leq & b(v) & \text{for all } v \in V \\ 0 \leq x(e) & \leq & 1 & \text{for all } e \in E \end{array}$$

THEOREM 1.2. Weighted-Degree Constrained Min-Cost k-Flow *admits a polynomial time algorithm that either correctly establishes that the polytope P_k is empty, or computes a subgraph H of G with k pairwise edge-disjoint rs-paths of cost $c(H) \leq \tau^*$ so that $w(\delta_H(v)) \leq 4b(v)$ for all $v \in V$.*

The Weighted Degree Constrained Min-Cost k-Flow problem belongs to a class of Degree Constrained Network Design problems. In these problems, one seeks the cheapest subgraph H of a given graph G that satisfies both prescribed connectivity requirements and degree constraints. One such type are the matching/edge-cover problems, which are solvable in polynomial time, c.f., [17]. For many other degree constrained problems, even checking whether there exists a feasible solution is NP-complete, hence one considers bicriteria approximation when the degree constraints are relaxed. See [7, 11, 8, 13, 14, 18, 1, 19, 6, 15] for literature on this type of problems.

2. PROOF OF THEOREM 1.1

Here we prove Theorem 1.1 based on Theorem 1.2. We may assume that we know the maximum flow value k, by applying binary search in the range $0, \ldots, nq$, where

$$q = \max_{v \in V} \frac{b(v)}{\min\{w(e) : e \in \delta_E(v), w(e) > 0\}} \ .$$

Indeed, if G contains an rs-path of weight 0, then k is infinite. Otherwise, every flow path contains a node v that uses an edge $e \in \delta_G(v)$ with $w(e) > 0$, which implies the bound $k \leq nq$. As an edge of G may be used several times, add $k - 1$ copies of each edge of G. Equivalently, we may assign to every edge capacity k, and consider the corresponding "capacited" problems; this will give a polynomial algorithm, rather than a pseudo-polynomial one. For simplicity of exposition, we will present the algorithm in terms of multigraphs, but it can be easily adjusted to capacited graphs.

It is easy to see that checking whether $k \geq 1$ can be done in polynomial time. We prove a slightly stronger statement, that also shows that the Single Topology version of the Maximum Integral Flow with Energy Constraints problem, when all the flow should be routed through a single rs-path P, can be solved in polynomial time.

Lemma 2.1. *Given an instance of* Maximum Integral Flow with Energy Constraints *and an integer ℓ, one can check in linear time if there exists a path P so that a flow of value at least ℓ can be routed via P. Consequently, an optimal path that enables routing a maximum amount of flow can be found in $O(|E|\log q)$ time.*

Proof. Let G' be obtained from G by deleting every edge $vu \in E$ with $w(vu) > b(v)/\ell$. It is easy to see that G contains a path P as in the Lemma if, and only if, G' contains an rs-path. Clearly, constructing G' and checking if G' contains an rs-path can be done in linear time.

We now describe how to find an optimal path P in time $O(|E|\log q)$. Let k be an optimal flow through a path. Note that $k \leq q$ or $k = \infty$. Indeed, if G contains an rs-path of weight 0, then k is infinite. Otherwise, there is a node v that uses an edge $e \in \delta_G(v)$ with $w(e) > 0$, which implies the bound $k \leq q$. The algorithm verifies that k is finite, and if so, applies binary search in the range $0, \ldots q$, to find the largest integer ℓ for which a flow of value ℓ can be routed through a single path. For each candidate ℓ, the time required is linear. Hence the total time complexity is as claimed. $\square$

Now we observe that Theorem 1.2 implies the following "pseudo-approximation" algorithm:

Corollary 2.2. *For* Maximum Integral Flow with Energy Constraints *there exists a polynomial time algorithm that either correctly establishes that the polytope P_k is empty, or finds an integral flow f of value at least k that violates the energy constraints by a factor at most 4, namely*

$$\sum_{e\in\delta_E(v)} f(e)w(e) \leq 4b(v) \quad \text{for all } v \in V \ . \tag{3}$$

Proof. Use binary search in the range $0, \ldots, nq$ to find the largest integer ℓ for which the algorithm as in Theorem 1.2 (with costs ignored) returns an ℓ-flow f that satisfies (3). $\square$

The algorithm in Theorem 1.1 is as follows:

1. Apply the algorithm as in Lemma 2.1 to determine whether there exists a path P so that at least 1 flow unit can be routed via P.
   If no such P exists, then return "$k = 0$" and STOP.
2. Set $b(v) \leftarrow b(v)/4$ for all $v \in V$ and compute a flow f using the algorithm as in Corollary 2.2.
3. If the value of f is at least 1, then return f;
   Else return P.

For the approximation ratio, all we need to prove is that if the original instance admits a k-flow, then the new instance with bounds $b(v)/4$ admits a flow of value $\lfloor k/16\rfloor$.

Let $\lambda(r, s; H)$ be the maximum number of pairwise edge-disjoint rs-paths in a graph H. Clearly, if H_k is a graph with $\lambda(r, s; H_k) \geq k$ then, for any $\ell \leq k$, H_k has a subgraph H_ℓ with $\lambda(r, s; H_\ell) \geq \ell$ so that $c(H_\ell) \leq c(H_k) \cdot (\ell/k)$. We prove that there exists H_ℓ so that the weighted degree of every node v in H_ℓ is at most 4 times the "expected" weight $w(\delta_{H_k}(v)) \cdot (\ell/k)$; namely, H_ℓ has both low weighted degrees and low cost. This result is of independent interest, but for the proof of Theorem 1.1 we need only the existence of H_ℓ with weighted degrees $\leq 4w(\delta_{H_k}(v)) \cdot (\ell/k)$ (namely, we ignore the costs).

Lemma 2.3. *Let $H_k = (V, F)$ be a graph with edge-costs $\{c(e) : e \in E\}$ and edge-weights $\{w(e) : e \in E\}$, so that $\lambda(r, s; H_k) \geq k$. Then for any integer $\ell \leq k$ the graph H_k contains a subgraph H_ℓ with $\lambda(r, s; H_\ell) \geq \ell$ so that $c(H_\ell) \leq c(H_k) \cdot (\ell/k)$ and $w(\delta_{H_\ell}(v)) \leq 4w(\delta_{H_k}(v)) \cdot (\ell/k)$ for all $v \in V$.*

Proof. Consider the Weighted Degree Constrained Min-Cost ℓ-Flow problem on $G = H_k$ with weighted degree bounds $b(v) = w(\delta_H(v)) \cdot (\ell/k)$. Clearly, $x(e) = \ell/k$ for every $e \in F$ is a feasible solution of cost $c(H_k)\cdot(\ell/k)$ to the LP-relaxation $\min\{c \cdot x : x \in P_\ell\}$. In particular, P_ℓ is non-empty, and thus the algorithm as in Corollary 2.2 computes a subgraph H_ℓ as required. $\square$

Substituting $\ell = \lfloor k/16\rfloor$ in Lemma 2.3 and ignoring the costs we obtain:

Corollary 2.4. *Let H be a graph with $\lambda(r, s; H) \geq k$ and with weights $\{w(e) : e \in E\}$. Then H contains a subgraph H' with $\lambda(r, s; H') \geq \lfloor k/16\rfloor$ so that $w(\delta_{H'}(v)) \leq w(\delta_H(v))/4$ for all $v \in V$.*

Theorem 1.1 is now easily deduced from Lemma 2.1 and Corollaries 2.2 and 2.4.

3. PROOF OF THEOREM 1.2

3.1 Weighted degree constrained network design

The connectivity requirements can specified by a set function h on subsets of V, as follows.

Definition 3.1. *For an edge set or a graph H and node set S let $\delta^{in}_H(S)$ denote the set of edges in H entering S. Given a set-function h on V and a graph $H = (V, F)$ we say that H is h-*connected *if $|\delta^{in}_H(S)| \geq h(S)$ for all $S \subseteq V$.*

Directed Weighted Degree Constrained Network (DWDCN)

Instance: A directed graph $G = (V, E)$ with edge-costs $\{c(e) : e \in E\}$, set-function h on V, edge-weights $\{w(e) : e \in E\}$, degree bounds $\{b(v) : v \in V\}$.

Objective: Find a minimum cost h-connected subgraph H of G that satisfies (2).

We assume that h admits a polynomial time evaluation oracle. In many cases, even checking whether there exists a feasible solution is NP-complete, thus we consider bicriteria approximation algorithms. An (α, β)-approximation algorithm for DWDCN either computes an h-connected subgraph $H = (V, F)$ of G of cost $\leq \alpha \cdot \tau$ that satisfies $w(\delta_H(v)) \leq \beta \cdot b(v)$ for all $v \in V$, where τ is the optimal solution value, or correctly determines that the problem has no feasible solution. Note that even if the problem does not have a feasible solution, the algorithm may still return a subgraph that violates the weighted degree constraints (2) by a factor of β. Several types of h are considered in the literature, among them the following known one:

Definition 3.2. *A set function h on V is* intersecting supermodular *if for any $X, Y \subseteq V$, $X \cap Y \neq \emptyset$*

$$h(X) + h(Y) \leq h(X \cap Y) + h(X \cup Y) \ . \tag{4}$$

h is a ring function *(or an s-*ring function*) if h is intersecting supermodular and there exists $s \in V$ so that $h(S) > 0$ implies $s \in S$.*

For an edge set I, let $x(I) = \sum_{e \in I} x(e)$. Let τ^* denote the optimal value of the natural LP-relaxation for DWDCN that seeks to minimize $c \cdot x$ over the following polytope:

$$
\begin{array}{llcll}
P_h : & x(\delta_E^{in}(S)) & \geq & h(S) & \text{for all } \emptyset \neq S \subset V \\
& \sum_{e \in \delta_E(v)} x(e)w(e) & \leq & b(v) & \text{for all } v \in V \\
& 0 \leq x(e) & \leq & 1 & \text{for all } e \in E
\end{array}
$$

Fukunaga and Nagamochi [6] and the author [15] were the first to consider weighted-degree constrained network design problems. Fukunaga and Nagamochi [6] considered *undirected* graphs, and gave a $(1,4)$-approximation algorithm for minimum spanning trees and a $(2,7)$-approximation algorithm for weakly supermodular h. For *directed* graphs, the best known ratio for DWNC with intersecting supermodular h [15] computes a solution of cost $\leq 2 \cdot \tau^*$, so that the weighted degree of every $v \in V$ is at most $6b(v)$.

The Weighted Degree Constrained Min-Cost k-Flow problem is a special case of DWDCN obtained by setting

$$
h(S) = \begin{cases} k & \text{if } s \in S \subseteq V - r \\ 0 & \text{otherwise} \end{cases}
$$

This h is an s-ring function. Thus Theorem 1.2 will be proved if we can prove:

Theorem 3.1. *DWDCN with ring function h admits a polynomial time algorithm that either correctly establishes that P_h is empty, or computes an h-connected graph H of cost $\leq \tau^*$ so that the weighted degree in H of every $v \in V$ is at most $3b(v)$.*

Note again that in this case, h is not only intersecting supermodular, but it is also an s-ring function; this enables to obtain a better algorithm than the one given in [15].

3.2 The algorithm

During the algorithm, F denotes the partial solution, I are the edges to add to F, and B is the set of nodes on which the outdegree bounds constraints are still present. The algorithm starts with $F = \emptyset$, $B = V$ and performs iterations. In any iteration, we work with the "residual problem" polytope $P_h(I, F, B)$ ($\alpha \geq 1$ is a fixed parameter, which we eventually set to $\alpha = 1$):

$$
\begin{array}{lcll}
x(\delta_I^{in}(S)) & \geq & h(S) - |\delta_F^{in}(S)| & \text{for all } \emptyset \neq S \subset V \\
\sum_{e \in \delta_I(v)} x(e)w(e) & \leq & b(v) - w(\delta_F(v))/\alpha & \text{for all } v \in B \\
0 \leq x(e) & \leq & 1 & \text{for all } e \in I
\end{array}
$$

Recall some facts from polyhedral theory. Let x belong to a polytope $P \subseteq R^m$ defined by a system of linear inequalities; an inequality is *tight* (for x) if it holds as equality for x. $x \in P$ is a *basic solution* for (the system defining) P if there exist a set of m tight inequalities in the system defining P such that x is the unique solution for the corresponding equation system; that is, the corresponding m tight equations are linearly independent. It is well known that if $\min\{c \cdot x : x \in P\}$ has an optimal solution, then it has an optimal solution which is basic, and that a basic optimal solution for $\{c \cdot x : x \in P_h(I, F, B)\}$ can be computed in polynomial time, c.f., [1].

Definition 3.3 ([6]). *$P_h(I, F, B)$ is (α, Δ)-sparse for $\alpha, \Delta \geq 1$ if any basic solution $x \in P_h(I, F, B)$ has an edge $e \in I$ with $x(e) = 0$, or satisfies at least one of the following:*

$$
\begin{array}{lcll}
x(e) & \geq & 1/\alpha & \text{for some } e \in I \qquad (5) \\
|\delta_I(v)| & \leq & \Delta & \text{for some } v \in B \qquad (6)
\end{array}
$$

The following general statement was proved in [15]:

Theorem 3.2 ([15]). *If for any I, F so that $P_h(I, F, B)$ is non-empty the polytope $P_h(I, F, B)$ is (α, Δ)-sparse, and if $P_h = P(E, \emptyset, V)$ is nonempty, then DWDCN admits an $(\alpha, \alpha+\Delta)$-approximation algorithm w.r.t. the LP-relaxation $\min\{c \cdot x : x \in P_h(E, \emptyset, V)\}$.*

Thus to prove Theorem 3.1, it is sufficient to prove the following statement:

Theorem 3.3. *For any ring set function h, if $P_h(I, F, B)$ is non-empty, then it is $(1,3)$-sparse.*

3.3 Proof of Theorem 3.3

Note that if $x \in P_h(I, F, B)$ is a basic solution so that $0 < x(e) < 1$ for all $e \in I$, then every tight equation is induced by either:

- *cut constraint* $x(\delta_I^{in}(S)) \geq h(S) - |\delta_F^{in}(S)|$ defined by some set $\emptyset \neq S \subset V$ with $h(S) - |\delta_F^{in}(S)| \geq 1$.
- *degree constraint* $\sum_{e \in \delta_I(v)} x(e)w(e) \leq b(v) - w(\delta_F(v))/\alpha$ defined by some node $v \in B$.

A family $\mathcal{F}$ of sets is *laminar* if for every $S, S' \in \mathcal{F}$, either $S \cap S' = \emptyset$, or $S \subseteq S'$, or $S' \subseteq S$. We use the following statement observed in [15].

Lemma 3.4 ([15]). *Let h be an intersecting supermodular set function. For any basic solution x to $P_h(I, F, B)$ with $0 < x(e) < 1$ for all $e \in I$, there exist a laminar family $\mathcal{L}$ on V and $T \subseteq B$ such that x is the unique solution to the linear equation system:*

$$
\begin{array}{lcll}
x(\delta_I^{in}(S)) & = & h(S) - |\delta_F^{in}(S)| & \text{for all } S \in \mathcal{L} \\
\sum_{e \in \delta_I(v)} x(e)w(e) & = & b(v) - w(\delta_F(v))/\alpha & \text{for all } v \in T
\end{array}
$$

where $h(S) - |\delta_F^{in}(S)| \geq 1$ for all $S \in \mathcal{L}$. In particular, $|\mathcal{L}| + |T| = |I|$ and the characteristic vectors of $\delta_I^{in}(S)$ for all $S \in \mathcal{L}$ are linearly independent.

Let $\mathcal{L}$ and T be as in Lemma 3.4 In the particular case when f is a ring function, $\mathcal{L}$ must be *nested*, namely, there exists an ordering $S_1, S_2, \ldots, S_\ell$ of $\mathcal{L}$ so that $S_1 \subset S_2 \subset \cdots \subset S_\ell$. Define a child-parent relation on the members of $\mathcal{L} + T$ as follows. For $S \in \mathcal{L}$ or $v \in T$, its parent is the inclusion minimal member of $\mathcal{L}$ properly containing it, if any. Note that if $v \in T$ and $\{v\} \in \mathcal{L}$, then $\{v\}$ is the parent of v, and that no members of T has a child. Since $\mathcal{L}$ is nested, every member of $\mathcal{L}$ except the smallest one has *exactly* one child in $\mathcal{L}$ (but may have several children in T).

We prove that if $x \in P_h(I, F, B)$ is a basic solution so that $0 < x(e) < 1$ for all $e \in I$, then there exists $v \in B$ with $|\delta_I(v)| \leq 3$. Suppose to the contrary that this is not so. Then we must have:

- $|\delta_I^{in}(S)| \geq 2$ for all $S \in \mathcal{L}$.
- $|\delta_I(v)| \geq 4$ for all $v \in T$.

Assign one token to each endnode of an edge in I. The number of tokens is thus $2|I|$. A token contained in S is an S-*token*. Assuming Theorem 3.3 is not true, we obtain the contradiction $|I| > |\mathcal{L}| + |T|$ by showing that given $S \in \mathcal{L}$, we can assign the S-tokens so that:

The 2-Scheme:
S and every descendant of S gets 2 S-tokens.

The contradiction $|I| > |\mathcal{L}| + |T|$ is obtained as follows. Any member of $\mathcal{L} + T$ gets 2 tokens, and we exhibit at least one more token as follows. If there is $v \in T$ that is not contained in the maximal member of $\mathcal{L}$, then v gets 3 tokens, which gives one spare token. Otherwise, the tail tokens of edges entering a maximal set in $\mathcal{L}$ are not assigned, and in this case we have at least 2 spare tokens. Thus all we need to prove is that the 2-Scheme above is feasible.

Initial assignment:
For every $v \in T$, we assign some 3 tail-tokens of the edges in $\delta_I(v)$.

The rest of the proof is by induction on the number of descendants of $S \in \mathcal{L}$. If S has no children/descendants, it contains at least $|\delta_I^{in}(S)| \geq 2$ head-tokens, as claimed.

Lemma 3.5. *Suppose that* $0 < x(e) < 1$ *for all* $e \in E$, *and let* $S \in \mathcal{L}$ *with child* $R \in \mathcal{L}$. *Then*

$$|(\delta_I^{in}(S) - \delta_I^{in}(R)) \cup (\delta_I^{in}(R) - \delta_I^{in}(S))| \geq 2 .$$

Proof. We cannot have $\delta_I^{in}(R) = \delta_I^{in}(S)$ as this contradicts the linear independence in Lemma 3.4. Hence at least one of the edge-sets $\delta_I^{in}(S) - \delta_I^{in}(R), \delta_I^{in}(R) - \delta_I^{in}(S)$ is nonempty. If one of these sets is empty, say $\delta_I^{in}(S) - \delta_I^{in}(R) = \emptyset$, then $x(\delta_I^{in}(R)) - x(\delta_I^{in}(S))$ must be a *positive* integer. Thus $|\delta_I^{in}(R) - \delta_I^{in}(S)| \geq 2$, as otherwise there is an edge $e \in \delta_I^{in}(R) - \delta_I^{in}(S)$ with $x(e) = 1$. The case $\delta_I^{in}(R) - \delta_I^{in}(S) = \emptyset$ is identical. □

Lemma 3.6. *The* 2-Scheme *is feasible.*

Proof. If $(S-R) \cap T = \emptyset$ (namely, if S has no child in T) then S can get 2 S-tokens not assigned to R, by Lemma 3.5. If $(S - R) \cap T \neq \emptyset$ (namely, if S has a child $v \in T$) then S can get 2 tokens from v. In both cases, we get an assignment as claimed. □

The proof of Theorem 3.3 is complete.

4. REFERENCES

[1] N. Bansal, R. Khandekar, and V. Nagarajan. Additive gurantees for degree bounded directed network design. In *STOC*, pages 769–778, 2008.

[2] H. L. Bodlaender, R. B. Tan, T. C. van Dijk, and J. van Leeuwen. Integer maximum flow in wireless sensor networks with energy constraint. TR UU-CS-2008-005. To appear in SWAT 2008.

[3] J. Chang and L. Tassiulas. Maximum lifetime routing in wireless sensor networks. *IEEE/ACM Transaction on Networking*, 12(4):609–619, 2004.

[4] M. Elkin, Y. Lando, Z. Nutov, M. Segal, and H. Shpungin. Novel algorithms for the network lifetime problem in wireless settings. To appear in AdHoc-NOW 2008.

[5] P. Floréen, P. Kaski, J. Kohonen, and P. Orponen. Exact and approximate balanced data gathering in energy-constrained sensor networks. *Theor. Comp. Sci.*, 344(1):30–46, 2005.

[6] T. Fukunaga and H. Nagamochi. Network design with weighted degree constraints. TR 2006–012, Kyoto University, 2008.

[7] M. Furer and B. Raghavachari. Approximating the minimum-degree steiner tree to within one of optimal. *Journal of Algorithms*, 17(3):409–423, 1994.

[8] M. X. Goemans. Minimum bounded degree spanning trees. In *FOCS*, pages 373–282, 2006.

[9] T. F. Gonzalez, editor. *Handbook on Approximation Algorithms and Metaheuristics*. Chapman & Hall/CRC, 2007.

[10] B. Hong and V. K. Prasanna. Maximum lifetime data sensing and extraction in energy constrained networked sensor systems. *J. Parallel and Distributed Computing*, 66(4):566–577, 2006.

[11] K. Jain. A factor 2 approximation algorithm for the generalized Steiner network problem. *Combinatorica*, 21(1):39–60, 2001.

[12] K. Kalpakis, K. Dasgupta, and P. Namjoshi. Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks. *Computer Networks*, 42(6):697–716, 2003.

[13] J. Könemann and R. Ravi. A matter of degree: Improved approximation algorithms for degree bounded minimum spanning trees. *SIAM Journal on Computing*, 31(3):1783–1793, 2002.

[14] L. C. Lau, J. Naor, M. R. Salavatipour, and M. Singh. Survivable network design with degree or order constraints. In *STOC*, pages 651–660, 2007.

[15] Z. Nutov. Approximating directed weighted-degree constrained networks. To appear in APPROX 2008.

[16] F. Ordonez and B. Krishnamachari. Optimal information extraction in energy-limited wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1121–1129, 2004.

[17] A. Schrijver. *Combinatorial Optimization Polyhedra and Efficiency.* Springer-Verlag Berlin Heidelberg New York, 2004.

[18] M. Singh and L. C. Lau. Approximating minimum bounded degree spanning trees to within one of optimal. In *STOC*, pages 661–670, 2007.

[19] M. Singh and L. C. Lau. Additive approximation for bounded degree survivable network design. In *STOC*, pages 759–768, 2008.

[20] Y. Xue, Y. Cui, and K. Nahrstedt. Maximizing lifetime for data aggregation in wireless sensor networks. *Mobile Networks and Applications*, 10:853–864, 2005.

[21] F. Zhao and L. Guibas. *Wireless Sensor Networks: An Information Processing Approach.* Morgan Kaufman, 2004.