Approximate Integer Decompositions for Undirected Network Design Problems

Chandra Chekuri* F. Bruce Shepherd†

July 14, 2008

Abstract

A well-known theorem of Nash-Williams and Tutte gives a necessary and sufficient condition for the existence of k edge-disjoint spanning trees in an undirected graph. A corollary of this theorem is that every $2k$-edge-connected graph has k edge-disjoint spanning trees. We show that the splitting-off theorem of Mader in undirected graphs implies a generalization of this to finding k edge-disjoint Steiner forests in Eulerian graphs. This leads to new 2-approximation rounding algorithms for constrained 0-1 forest problems considered by Goemans and Williamson. These algorithms also produce approximate integer decompositions of fractional solutions. We then discuss open problems and outlets for this approach to the more general class of 0-1 skew supermodular network design problems.

Keywords: network design, supermodular function, integer decomposition, approximation algorithm.

1 Introduction

In this article we consider the application of splitting-off techniques to obtain integer decomposition theorems and rounding algorithms for undirected network design problems such as the Steiner forest problem and others. A well-known theorem in graph theory is the following.

Theorem 1.1 (Nash-Williams, Tutte) *Given an undirected multi-graph $G = (V, E)$, there exist k edge-disjoint spanning trees $T_1, T_2, \ldots, T_k$ in G if and only if for every partition $V_1, V_2, \ldots, V_\ell$ of V the number of edges between the node sets of the partition is at least $k(\ell - 1)$.*

An easy corollary of the above is the following.

Corollary 1.2 *If G is $2k$-edge-connected, then there exist k edge-disjoint spanning trees in G.*

Let $\lambda_G(u, v)$ denote the connectivity between u and v in G. We consider packing Steiner forests instead of spanning trees and obtain the following generalization of Corollary 1.2 for Eulerian graphs.

Lemma 1.3 (The Forest Packing Lemma) *Given an Eulerian graph G and pairs of nodes $s_1t_1, s_2t_2, \ldots, s_\ell t_\ell$ such that for $1 \leq i \leq \ell$, $\lambda_G(s_i, t_i) \geq 2k$, there are k edge-disjoint forests, $F_1, F_2, \ldots, F_k$ such that in each F_j, s_i and t_i are connected for $1 \leq i \leq \ell$.*

*Dept. of Computer Science, University of Illinois, Urbana, IL 61801. Email: chekuri@cs.uiuc.edu.
†McGill University, 805 Sherbrooke West, Montreal QC. Email: bruce.shepherd@mcgill.ca.

We give a proof of this result (in Section 3) that relies on a simple application of Theorem 1.1 and the classical splitting-off technique of Mader [34]. A special case is proved in [19] where the goal is to pack Steiner "S-trees", i.e., trees that each contains a given subset S of the nodes. Splitting-off for packing Steiner trees in general graphs is also considered in [28]. While simple, the extension above to forests, rather than trees, is of interest in its own right and has already been of use in a related context [31].

Our algorithmic motivation for proving Lemma 1.3 actually arises from the following network design problem which is "dual" to the forest packing problem. In the *Steiner forest problem* (also called the generalized Steiner problem) we are given an edge weighted undirected graph $G = (V, E, w)$ and a set of pairs $s_1t_1, s_2t_2, \ldots, s_\ell t_\ell$. The goal is to find a minimum cost subgraph H of G such that for $1 \leq i \leq \ell$, s_i and t_i are connected in H. This problem has been studied intensively with some of the most general outcomes appearing in [23, 27]. Ultimately we seek results for packing more general classes of subgraphs, not just forests, in connection with network design arising from certain supermodular set functions. We outline this more general framework in the following subsections and state Theorem 1.5, a strict generalization of the forest packing lemma, that we prove in this paper.

1.1 Approximate Integer Decomposition Properties

In this article, we work exclusively with the standard cut-based linear programming (LP) *relaxation* for our network design problems. For $e \in E$ there is a variable $x_e \in [0, 1]$ that indicates if e is part of the subgraph. We seek to minimize $\sum_e w_e x_e$ subject to the constraint that for each $S \subset V$ that separates some pair s_it_i, $x(\delta(S)) \geq 1$. Primal-dual 2-approximation algorithms of Agrawal, Klein, and Ravi [3] and later Goemans and Williamson [22] show that the integrality gap of the cut-based LP is $(2 - 2/h)$ where h is the number of distinct terminals. We obtain an alternative proof of a gap of 2 and of more interest, we show the relaxation (and our 2-approximation algorithm) has a stronger "integrality" property. We can describe this now.

Let x be a solution to the LP and let k be an integer such that kx is integral. Consider the graph $G' = (V, E')$ obtained by taking $2kx_e$ copies of each edge $e \in E$. By Lemma 1.3 it follows that E' contains k edge-disjoint forests, each of which is a feasible solution to the Steiner forest problem. Thus the vector $2kx$ dominates a sum of k integral solutions. By convexity it follows that one of the k forests is of cost no more than $2w \cdot x$, in other words twice the cost of the original LP solution.

The above approach yields a 2-approximation algorithm with a stronger property than those from earlier methods in the following sense. It is always the case that if the integrality gap of an LP relaxation for a minimization problem is $\alpha \geq 1$, then αx dominates a convex (i.e., fractional) combination of integral solutions. However, in general, it does not follow that αkx, when kx is integral, dominates a sum (i.e., integral combination) of k integral solutions. If this stronger property holds for any feasible fractional solution x, we say the relaxation has the *α-approximate integer decomposition property*; more precisely, this is a property of the polyhedron consisting of feasible solutions for the relaxation. If $\alpha = 1$, the above decomposition property is called the *integer decomposition property* (IDP) and is well-studied, cf. [36]. Baum and Trotter [5, 6] show for instance, that a matrix A is totally unimodular if and only if $\{x : Ax \leq b, x \geq 0\}$ has the integer decomposition property for each integral b. Approximate integer decomposition for maximization problems, in particular packing problems, can be defined in a fashion similar to that for minimization problems. For a fractional solution x, one considers an integral vector kx, but seeks a decomposition (or cover) of kx into at most $\alpha k \geq k$ integer feasible solutions: $kx = \sum_{i=1}^{\lfloor \alpha k \rfloor} g_i$. Obviously, one of the g_i's is an integral solution whose weight (profit) is at least $\frac{1}{\alpha}$ times that of x.

This decomposition approach is perfectly natural and is often the technique used in the literature to establish an approximation ratio (first mention of a connection to the integer decomposition property seems to appear in [10]). Some well-known combinatorial problems have an integrality gap equal to their approximation ratio for integer decomposition. For instance, it is an exercise to show that the natural LP relaxation for the knapsack problem has the 2-approximate integer decomposition property. It is not always obvious, however, when such a property does hold. In this paper, we ask for example, whether recent celebrated 2-approximation results of Jain [27] can be extended to have the 2-approximate integer decomposition property (see Section 1.2.1).

We believe it is not only worthwhile to make the integer decomposition approach explicit (including its connections to traditional polyhedral results for IDP), but also that such stronger decomposition results are potentially important in their own right. For instance, the results of [10] for packing paths in trees provided the stronger integer decomposition property. These results were subsequently used in [14, 1, 2] where the integer decompositions corresponded to partitioning pairwise demands so that each class of demands could be routed with a distinct wavelength in an optical network. In some recent work, Fukunaga and Nagamochi [15] applied the approximate integer decomposition methodology to obtain algorithms for the set connector problem.

Before continuing with our main focus, approximations for network design, we give another application of integer decompositions, this time to result yield an approximation result due to Goemans and Williamson [22] for the prize collecting Steiner tree problem. Namely, we mention that their result can be alternatively derived from a result of Bang-Jensen, Frank, and Jackson [4] on packing arc-disjoint Steiner arborescences in directed graphs. We give some details below. In the *prize collecting Steiner tree problem* we are given an undirected edge-weighted graph $G = (V, E, c)$ and a root node $r \in V$. Each node v also has a non-negative penalty value $\pi(v)$. The objective is to find at tree $T = (V(T), E(T))$ rooted at r that minimizes $\sum_{e \in E(T)} c(e) + \sum_{v \notin V(T)} \pi(v)$. The first constant factor approximation algorithm for this problem was given in [7] and subsequently [22] gave a primal-dual algorithm that finds a tree T such that $\sum_{e \in E(T)} c(e) + 2 \sum_{v \notin V(T)} \pi(v) \leq 2\text{OPT}$, where OPT is the optimum value of the natural LP relaxation for the problem. This result has found use in other approximation algorithms, notably for the k-MST problem [8, 20] and several others. The result can be obtained from [4] as follows. Consider a fractional solution x to the LP relaxation: x_e is the value on edge e and $x(v)$ is the flow from r to v supported by x. We obtain a directed graph by bi-directing each edge e and placing a value of x_e on both of the resulting arcs. This clearly increases the cost of edges by a factor of 2. Now we apply the theorem in [4] to obtain a convex combination of arborescences rooted at r in which each v occurs in at least $x(v)$ arborescences. Picking the least cost arborescence yields the desired result. The remaining details are left to the interested reader.

1.2 Constrained Forest Problems and f-Connected Networks

Goemans and Williamson [22] obtain 2-approximation algorithms for a large class of network design problems that they refer to as *constrained forest problems*; they apply their primal-dual framework for this. Each of these problems is determined by an integer-valued function f that for each set $S \subseteq V$ gives a requirement value $f(S)$. (In some cases, we only require this for sets in a given family $\mathcal{F}$ – see Section 1.2.1.) A solution to the connectivity problem modeled by f is a collection of edges A such that at least $|A \cap \delta(S)| \geq f(S)$ for each $S \subseteq V$. Such a solution will be called *f-connected*, or an *f-connector*. The optimization problem is to find a minimum cost f-connector.

The most general class of functions for which the network design problem is known to have a constant factor approximation is the set of integer-valued skew supermodular functions. In establishing this result, Jain [27] introduced a new iterative rounding approach to obtain a 2-approximation for such skew supermodular problems, called *Steiner network design problems*. As we see in Section 1.2.1, many natural (NP-hard or otherwise) network design problems are modelled as minimum cost f-connector problems for a skew supermodular function f. In Section 5, we discuss a kind of inverse problem which we believe deserves further investigation. Given a requirement function, does it encode a natural class of network design problems? We give several results on when $\{0,1\}$-valued requirement functions encode certain connectivity augmentation design problems.

Jain's approach, based on the framework designed for submodular flows [13, 33], requires finding a *basic* solution to the cut LP relaxation for f-connected subgraphs. One of our motivations for studying primal rounding methods via a decomposition-based approach is to find a combinatorial rounding algorithm for the Steiner network problem. The LP for the Steiner network problem can be solved to any given precision using efficient combinatorial methods [21] and hence a rounding approach that works with *any* feasible primal solution would yield an efficient and combinatorial $(2+\epsilon)$-approximation for the problem. A second motivation is to determine whether the f-connected subgraph relaxation for the much larger class of skew supermodular functions f possess the 2-approximate integer decomposition property. Our main result, Theorem 1.5, provides some evidence that this may hold. Theorem 1.5, is a decomposition theorem and rounding algorithm that applies to some of the more general f-connector problems studied in [23].

1.2.1 Steiner Networks and Supermodular Functions: Results and Terminology

Let $G = (V, E)$ be an undirected graph. A family $\mathcal{F}$ of subsets of V is *skew crossing* if for each $A, B \in \mathcal{F}$ either $A - B, B - A \in \mathcal{F}$ or $A \cap B, A \cup B \in \mathcal{F}$ (or both). Let $f : \mathcal{F} \to \mathcal{Z}^+$ be an integer valued function. We call a subgraph H of G *f-connected* if for each $A \in \mathcal{F}$ we have that $|\delta_H(A)| \geq f(A)$. The main problem considered in this paper is that of finding a minimum cost f-connected subgraph for some interesting classes of functions f that capture natural network design problems. We present our arguments as though $\mathcal{F} = \mathcal{P}(V)$, but one easily verifies that results hold even in the case where f's domain is an arbitrary skew crossing set family.

We focus on the natural LP relaxation for this problem:

$$P(G, f) = \{x \in [0,1]^E : x(\delta(A)) \geq f(A) \text{ for each subset } A \in \mathcal{F}\}. \quad (1)$$

The f-connectivity problem asks to find an integer vector $x \in P(G, f)$ which minimizes $w \cdot x = \sum_e w_e x_e$. The most general class of functions we consider are *skew supermodular* functions [18] (also called *weakly supermodular* in [27])[1]. A function f is skew supermodular if for each pair of sets $A, B \in \mathcal{F}$ at least one of the following holds:

1. $A \cap B, A \cup B \in \mathcal{F}$ and $f(A) + f(B) \leq f(A \cap B) + f(A \cup B)$
2. $A - B, B - A \in \mathcal{F}$ and $f(A) + f(B) \leq f(B - A) + f(A - B)$.

The class of $\{0,1\}$ skew supermodular f-connectivity problems captures a variety of well-known combinatorial problems, many of which are outlined in the survey [23]. Let us reconsider a few special cases of this problem.

First if $f(A) = 1$ for every proper subset A of V, then this coincides with the minimum spanning tree problem. Given a set of terminals $T \subset V$ if we define f by $f(A) = 1$ if A splits T (that is, $A \cap T \neq \emptyset$ and $A \cap T \neq T$), then f captures the NP-hard Steiner tree problem. If $f(A) = 1$ for each $A = \{v\}$ and $f(A) = 0$ otherwise, then this is just the minimum node cover problem. Another case of interest is obtained as follows. Consider some pair of nodes $s, t \in V$. Define the following skew supermodular function f: $f(A) = 1$ for each subset A that separates s and t. Then f-connectivity is just asking for the minimum cost s-t path. Suppose that the maximum number of edge-disjoint s-t paths is k and define $f(A) = 1$ for each A that induces a minimum s-t cut.

[1] Andras Frank, at a workshop in Bertinoro, Italy, has convinced the authors that skew supermodular is a more appropriate name than weakly supermodular. He indicates that David Shmoys suggested this name in 1993.

Then the f-connectivity problem asks for a minimum cost subset of edges which if we duplicate, increases the connectivity from k to $k+1$. Call this the *s-t* connectivity augmentation problem.

For certain classes of functions f, the polytope $P(G, f)$ has integral extreme points. Examples include the shortest path and connectivity augmentation functions defined above. This is not always the case; for instance the NP-hard Steiner tree problem. It was shown in [38] that for all $\{0,1\}$ skew supermodular functions, the optimum over $P(G, f)$ is no better than a factor of 2 from the optimum over the integer hull of $P(G, f)$. This is proved via a primal-dual algorithm. Jain [27] generalized this to all integer-valued skew supermodular functions using a different approach of iterative rounding.

Encouraged by the decomposition results for Steiner forests in Lemma 1.3, we conjecture the following.

Conjecture 1.4 *For any graph G and $\{0,1\}$ skew supermodular f, if $x \in P(G, f)$ and kx is integral, then there exist f-connected integer vectors $h_1, h_2, \ldots, h_k$ such that $2kx \geq \sum_i h_i$.*

Indeed, we know of no reason why the statement could not hold for general integer-valued skew supermodular functions. Although we are unable to prove the above conjecture, our main theorem establishes some positive evidence by establishing it for certain classes of skew supermodular functions introduced by Goemans and Williamson [22]. We introduce these classes now.

A $\{0,1\}$ function is termed *maximal* if the following holds: for any disjoint subsets $A, B \subseteq V$, $f(A \cup B) \leq \max(f(A), f(B))$. Equivalently, if A and B are disjoint, then $f(A) = f(B) = 0$ implies that $f(A \cup B) = 0$. A function is *symmetric* if for each $A \subset V$, $f(A) = f(V - A)$. A $\{0,1\}$ function is *proper* if it is maximal, symmetric and $f(V) = 0$. Another special class of skew supermodular functions are *downward monotone functions* which satisfy the property that $f(A) \geq f(B)$ if $A \subset B$.

Theorem 1.5 *For any graph G and $\{0,1\}$ function f where f is either proper or downward monotone, if $x \in P(G, f)$ and kx is integral, then there exist f-connected integer vectors $h_1, h_2, \ldots, h_k$ such that $2kx \geq \sum_{i=1}^{k} h_i$. Moreover, given x we may find this decomposition in polynomial time.*

Note that the above theorem generalizes Lemma 1.3 since the Steiner forest problem is defined by a $\{0,1\}$ proper function. One consequence of the above theorem is a new polynomial-time 2-approximation algorithm for the minimum cost f-connectivity problem for proper and downward monotone functions. These new algorithms are primarily of theoretical interest since their running times are not competitive with the primal-dual algorithms [22].

In addition to proving Lemma 1.3 and Theorem 1.5, we consider the general question of whether f-connectivity problems arise in a natural way from other basic problems. We show in Section 5 that this is indeed the case for intersecting supermodular functions: they are effectively disguised connectivity augmentation problems in both the directed and undirected settings. We also characterize the (supermodular) functions which define Steiner forest problems. Negative results are given however for proper and skew supermodular functions.

1.2.2 Further Related Work

Our approach to finding approximate integer decompositions for $\{0,1\}$ network design problems amounts to packing forests each satisfying some connectivity requirement. As alluded to earlier, one special case of this has been considered more extensively in the literature. Given an undirected graph $G = (V, E)$ and set $S \subseteq V$ of terminals, find the maximum number of edge-disjoint S-Steiner trees in G. This problem has been studied from a polyhedral and computational point of view by Grötschel, Martin, and Weismantel [25, 26]. Their motivating application is routing in VLSI design. Kriesell [29] considered the same problem and conjectured that Corollary 1.2 generalizes to packing Steiner trees: that is if a set S is $2k$-edge-connected in G, then there are k edge-disjoint S-Steiner trees in G. As mentioned earlier, if G is *Eulerian*, Frank, Kiraly and Kriesell [19] show that if S is $2k$-edge-connected in G, then there are such disjoint Steiner trees. They also showed that if S is $3k$-edge-connected and $V-S$ is a stable set, then there are k edge-disjoint S-Steiner trees. In general graphs, Jain, Mahdian, and Salvatipour [28] showed that if S is k-edge-connected, then there are $\alpha_{|S|}k$ edge-disjoint Steiner trees where $\alpha_{|S|} \to \frac{4}{|S|}$. They also give results on fractional packing of Steiner trees and for this case they use the duality between fractional packing and approximation algorithms [9, 24]. As observed in [29, 28], the known results had not guaranteed two edge-disjoint Steiner trees even if S is k-edge-connected for any $k = o(n)$. Recently, Lau [30] showed that a k-packing of Steiner trees can in fact be found if S is $26k$-edge-connected, in the process obtaining the first constant factor approximation for integer packing of Steiner trees. In [31] Lau extended his ideas to the Steiner forest packing problem; given node pairs $s_1t_1, \ldots, s_\ell t_\ell$ such that $\lambda_G(s_it_i) \geq 32k$ for $1 \leq i \leq \ell$ then there are k edge-disjoint forests such that each s_it_i is connected in each of the k forests. This extension was partly motivated by our work in this paper.

2 Preliminaries

The central tool used in this article is that of *splitting-off* edges. We state Mader's Splitting-off Theorem, that was conjectured earlier by Lovász [32].

Theorem 2.1 (Mader [34]) *Let $G = (V \cup \{s\}, E)$ be an undirected multi-graph, where s has positive even degree, and s is not incident with a cut edge of G. Then s has two neighbours u and v such that the graph G' obtained from G by replacing su and sv by uv satisfies $\lambda_{G'}(x,y) = \lambda_G(x,y)$ for all $x, y \in V \setminus \{s\}$.*

In this paper we apply the above splitting-off theorem only for Eulerian graphs which do not have cut edges.

The following claim is standard.

Claim 2.2 *In an undirected graph G, for any three distinct nodes u, v, w, $\lambda_G(u,w) \geq \min\{\lambda_G(u,v), \lambda_G(v,w)\}$.*

Let S be a proper subset of the nodes V of an undirected graph $G = (V, E)$. We denote by $\delta(S)$ the *cut* induced by S; that is the subset of edges E with exactly one endpoint in S. For an edge vector $x : E \to \mathbf{R}$, and $E' \subseteq E$, we use $x(E')$ to denote the quantity $\sum_{e \in E'} x_e$. We say that a set X *splits* a set S, or is S*-splitting*, if both $X \cap S$ and $X - S$ are nonempty. We call a set of nodes S, an ℓ*-island*, if for any $u, v \in S$, $\lambda_G(u,v) \geq \ell$. From Claim 2.2, it follows that the maximal ℓ-islands are unique and disjoint. We also refer to S as being a *fractional* ℓ*-island* with respect to some edge vector x^*, if for any S-splitting set U, $x^*(\delta(U)) \geq \ell$.

For a vector $x \in \mathbf{R}^E_+$ we call a subset S' *deficient* if $x(\delta(S')) \leq 1$ and strongly deficient if the inequality is strict. Each *strongly* deficient set S' evidently satisfies $f(S') = 0$ if $x \in P(G,f)$. We make repeated use of the following lemma. It follows directly from the well-known fact that the function $\delta(S)$ is posi-modular[2], a definition introduced by Nagamochi and Ibaraki [35].

Lemma 2.3 *For any graph G and $x \in \mathbf{R}^E_+$, if S', S'' are deficient sets, then at least one of $S' - S'', S'' - S'$ is deficient.*

We need another simple lemma given below.

Lemma 2.4 *Let $x^* \in \mathbf{R}^E_+$ and let K be a minimal deficient set. Then K induces a fractional 1-island in the graph obtained by contracting $V - K$ to a single node.*

[2] A function $f : \mathcal{F} \to \mathbf{R}_+$ is posi-modular if $f(A) + f(B) \geq f(A-B) + f(B-A)$ for all $A, B \in \mathcal{F}$.

Proof: Let G^* be obtained by contracting $V - K$ to a single node v^*. If K is not a fractional 1-island, then there exists some subset Y' proper subset of K such that $x^*(\delta_{G^*}(Y')) = x^*(\delta_G(Y')) < 1$. But then Y' is strongly deficient for x^*, contradicting minimality of K. ∎

We give a corollary of Theorem 1.1 that is useful in subsequent sections.

Lemma 2.5 *Let $G = (V' \cup \{s\}, E)$ be such that V' is a $2k$-island in G and $|\delta_G(s)| \leq 2k$. Then the subgraph induced by V' has k edge-disjoint spanning trees.*

Proof: Let $G' = (V', E')$ be the subgraph of G induced by V'. Let $V_1, V_2, \ldots, V_\ell$ be any partition of V' in G'. We claim that $\sum_{i=1}^{\ell} |\delta_{G'}(V_i)| \geq 2k\ell - |\delta_G(s)| \geq 2k\ell - 2k$, hence the number of edges in G' between nodes of the partition is at least $k(\ell-1)$. Thus G' satisfies the conditions of Theorem 1.1 and hence has k edge-disjoint spanning trees. ∎

3 Packing Steiner Forests

In this section we prove Lemma 1.3. Recall that we are given an Eulerian graph $G = (V, E)$ and pairs of nodes $s_1t_1, s_2t_2, \ldots, s_kt_k$ such that $\lambda_G(s_i, t_i) \geq 2k$ for $1 \leq i \leq k$. Given G let $S_1, S_2, \ldots, S_h$ be the maximal $2k$-islands. In fact we prove the following theorem which can be easily seen to imply Lemma 1.3.

Theorem 3.1 *Let $G = (V, E)$ be an Eulerian graph and let $S_1, S_2, \ldots, S_h$ be the maximal $2k$-islands in G. Then, there are k edge-disjoint forests $F_1, F_2, \ldots, F_k$ in G such that in each F_j, and for $1 \leq i \leq h$, S_i is contained in a connected component of F_j. Given G and k, there is an algorithm that finds such a packing in time polynomial in n and $\log k$.*

Proof: The proof is by induction on $|V|$. The base cases with $|V| \leq 2$ are easy to see. We call $v \in V$ a *Steiner node* if v is a singleton island, otherwise it is a *terminal*. We reduce the problem to *basic* instances, defined as instances in which $\cup_j S_j = V$ and $|S_i| \geq 2$ for $1 \leq i \leq h$; in other words there are no Steiner nodes. We get rid of Steiner nodes by splitting-off the edges incident to them. Let s be a Steiner node. Since G is Eulerian $d(s)$ is even. From Theorem 2.1, there are edges su and sv incident to s such that su and sv can be split-off without affecting the connectivity of any pair of nodes not involving s. A solution to the problem on the modified graph can be extended to a solution to the original graph by replacing the edge uv by the path consisting of su and sv. Hence we can repeatedly split-off edges incident to s until the degree of s is 0. We can eliminate all Steiner nodes in this way and reduce the graph G to a basic instance.

We now assume that G is basic. If $h = 1$, then from Corollary 1.2, we can find k spanning trees and hence we are done. If $h \geq 2$, we may apply Lemma 2.4 to find a set $K := \cup_{i \in I} S_i$ such that $|\delta_G(K)| < 2k$ and contracting $V - K$ to a single node s produces a graph G' where K is a $2k$-island. To see this, simply choose x^* to be the edge vector with weight $1/2k$ on each edge and let K be a minimal deficient set. Since a deficient set cannot be S_i-splitting for any i, the S_i's inside K form our set I. From Lemma 2.5 we can find k edge-disjoint trees in $G[K]$ that do not use edges incident to s. Let these be $T_1, T_2, \ldots, T_k$. Now consider the graph G'' obtained by shrinking K in G to a single node s'. We can apply induction to G'' since it has fewer nodes than G (note that $|K| \geq 2$ since the instance is basic) to obtain edge-disjoint forests $F'_1, F'_2, \ldots, F'_k$ such that each S_i, $i \notin I$ is contained in a single component in each of the forests. We obtain the desired forests $F_1, F_2, \ldots, F_k$ in G as follows. To obtain F_i we replace s' in F'_i with T_i. Note that two nodes u and v which are connected in F'_i via s' will still be connected in F_i since T_i is spanning on K. This finishes the proof of the existence of the packing.

We now prove that the packing can be found in time polynomial in n and $\log k$. To obtain a time polynomial in $\log k$, the decomposition will be output in a compact form with some forests having integer multiplicities. We assume without loss of generality that the number of edges between any two pairs of nodes is at most $2k$, otherwise we can remove some edges without violating the connectivity requirements. We first observe that the maximum number of edge-disjoint spanning trees in a given graph can be found in time polynomial in n and $\log k$ (see Chapter 51, pages 887—889 in [37]). Therefore the trees in Lemma 2.5 can found in poly$(n, \log k)$. There are two non-trivial steps to verify polynomial running time.

First, we describe the implementation of the splitting-off step. Let v be a Steiner node with $v_1, v_2, \ldots, v_\ell$ as its neighbours and let $c(v, v_i)$ be the number of edges between v and v_i. Let $c(v) = \sum_i c(v, v_i)$. From Theorem 2.1, we can split-off edges incident to v in pairs. After we split-off all the edges incident to v let $c'(v_i, v_j)$ be the number of new edges generated between v_i and v_j. It follows that there exists a pair v_i, v_j such that $c'(v_i, v_j) \geq \max\{1, c(v)/(2\ell^2)\}$. For each pair v_i, v_j we can find the maximum number of edges that can be split-off at v to generate edges between v_i and v_j by doing a binary search in the range $[0, \min\{c(v, v_i), c(v, v_j)\}]$. Each search involves finding the edge connectivity between all pairs of nodes to ensure that the splitting-off is legal. Thus we can split-off edges incident to v in time polynomial in $\log k$ and n.

Second, when G is basic and $h \geq 2$, we need to find a minimal set K such that $|\delta_G(K)| < 2k$. This can be accomplished in polynomial time as follows. We compute the minimum-cut value $\lambda_G(s,t)$ for all node pairs s, t. Pick an arbitrary node u and let K be the set of all nodes v such that the $\lambda_G(u, v) \geq 2k$. From Claim 2.2 it is easy to see that K is a desired minimal set.

This finishes the proof. ■

4 Skew Supermodular Functions

We have essentially examined the problem of decomposing fractional solutions into forests so that any pair of nodes that were originally 1-connected (fractionally) are in the same component of each forest. In this section, we study this scheme for more general $\{0, 1\}$ connectivity functions. In particular we prove Theorem 1.5 on proper and downward monotone functions. For skew supermodular functions we describe a reduction to a special case.

4.1 Proper Functions

First, we ask if for such a function f, the following property holds. *For any fractional solution to the f-connector problem, a feasible integral solution is obtained from any forest which includes each maximal island in a common component.* We show that that this holds true for the class of $\{0, 1\}$ proper set functions. Hence Lemma 1.3 will imply our desired decomposition result.

Theorem 4.1 *Let x be a fractional f-connector of $G = (V, E)$ and let $X_1, \ldots, X_l$ be the maximal islands for x. If f is proper, then any forest F that includes each X_i in a common connected component is an f-connector of G.*

Proof: Note that the X_i's partition $V(G)$. It is sufficient to show that $f(S) = 1$ implies that there is an i such that S splits X_i. Suppose this is not the case and let S be a minimal such set. We may write S as the union of some of the islands. But then by repeated application of maximality of f, at least one of these islands X, must have $f(X) = 1$. Thus by minimality of S, and without loss of generality, we may assume that $S = X_1$. Now since X_1 is an island, we have that for each node $u \in X_1$ and each node $v \notin X_1$, there is a subset $S' \subseteq V - X_1$ containing v, such that $x(\delta(S')) < 1$, that is S' is strongly deficient. Note that $f(S') = 0$.

By Lemma 2.3, if S', S'' are strongly deficient sets and $S' - S'', S'' - S' \neq \emptyset$, then at least one of $S' - S'', S'' - S'$ is strongly deficient. Now to complete the proof, consider a minimal collection of strongly deficient sets that covers $V - S$; such a collection exists since each $v \in V - S$ is in a strongly deficient set as we argued above (recall that $S = X_1$). If for some pair S', S'' we have that $S' - S'', S'' - S'$ and $S' \cap S''$ are nonempty, then by our previous claim, we may assume that $S' - S''$ is also deficient. We may thus replace S' by the set $S' - S''$. Clearly we may repeat this process until the family of strongly deficient sets we obtain is a partition of $V - S$. By our assumption, $f(S) = 1$ and therefore by symmetricity, $f(V - S) = 1$. However, $V - S$ is the disjoint union of sets S' with $f(S') = 0$ contradicting maximality of f. This contradiction completes the proof. ■

Given a fractional solution $x \in P(G, f)$ let k be such that kx is integral. It follows that $2kx$ induces an Eulerian graph G^*. It is easy to see that the 1-islands induced by x in G are precisely the $2k$-islands in G^*. From Theorem 3.1 in G^* there are k edge-disjoint forests $F_1, F_2, \ldots, F_k$ such that each island is connected in each of the F_i. Thus, from Theorem 4.1 each F_i is an f-connector. This establishes that $2kx$ can be decomposed into k f-connectors when f is a $\{0, 1\}$ proper function. Further the decomposition can be found in time polynomial in n and $\log k$ as shown by Theorem 3.1.

4.2 Downward Monotone Functions

We now consider the the class of $\{0, 1\}$ downward monotone functions. Recall that f is downward monotone implies that $f(A) \geq f(B)$ if $A \subset B$. Whereas for proper functions one can apply the forest-packing lemma directly, one must do more work in the case of downward monotone functions. We identify a collection of subproblems for which we apply Lemma 2.5 and collectively these will give the desired f-connected forests. Thus the second claim of Theorem 1.5 will be established.

Let $x \in P(G, f)$ and k be an integer such that kx is integral. We denote by G^* the Eulerian graph induced by $2kx$. Suppose there is no strongly deficient set in G. Then G^* has k edge-disjoint spanning trees each of which is a f-connector and we are done. Otherwise let $\mathcal{S} = \{S_1, S_2 \ldots, S_\ell\}$ be the minimal strongly deficient sets for x. Lemma 2.3 implies that these sets are disjoint. Let $S = V \setminus (\cup_i S_i)$. Note that S could be the empty set. We observe some useful properties. First, (i) for each i, $f(S_i) = 0$ since S_i is strongly deficient; (ii) by downward monotonicity, $f(Y) = 0$ if $S_i \subseteq Y$. Second, if $S \neq \emptyset$, for any $Y \subseteq S$, Y is not strictly deficient; otherwise $\mathcal{S}$ would not be the set of all minimal strongly deficient sets. Each S_i is a minimal strongly deficient set and hence from Lemmas 2.4 and 2.5 we can find k disjoint spanning trees in $G^*[S_i]$. Let $\mathcal{T}_i = \{T_{i,1}, \ldots, T_{i,k}\}$ be such a set of trees.

First we consider the case that $S = \emptyset$ which implies that $S_1, S_2, \ldots, S_\ell$ partition V. Let $\mathcal{F} = \{F_1, \ldots, F_k\}$ be a collection of k edge-disjoint forests where $F_j = \cup_{i=1}^{\ell} E(T_{i,j})$. It is easy to see that the F_j are edge-disjoint. Each F_j is a f-connector by remark (ii) above.

We now consider the case that $S \neq \emptyset$. Obtain a graph G_1^* from G^* by shrinking $V \setminus S$ into a single node s. We note that S is a $2k$-island in G_1^* since neither S nor any of its subsets was strongly deficient in G. Therefore, for any $u, v \in S$, $\lambda_{G_1^*}(u, v) \geq 2k$. Let the degree of s in G_1^* be $2k'$. Since S was not strongly deficient in G, $k' \geq k$. We modify G_1^* by splitting-off edges incident to s while preserving the connectivity of nodes in S, until the degree of s is exactly $2k$. Let G_2^* be the resulting graph. Using Lemma 2.5, there are k edge-disjoint spanning trees $T_1, \ldots, T_k$ in $G_2^*[S]$. Let E_i be the edge set of T_i. An edge $e \in E_i$ is either an original edge from G^* or is an edge that is obtained by splitting-off two edges e', e'' incident to s. In the latter case, note that e' and e'' also correspond to original edges from G^* (possibly incident to distinct nodes in $\cup_i S_i$). Let $E_i' \subseteq E(G^*)$ be the set of edges obtained from E_i by replacing each split-off edge $e \in E_i$ by its corresponding edges e', e''. We remark that E_i' may no longer induce a connected component on S in the graph G^*. Finally, let $e_1, e_2, \ldots, e_{2k}$ be the edges incident to s in G_2^*. We also associate these edges to their original edges in G^*. We obtain the desired edge-disjoint f-connectors $\mathcal{F} = \{F_1, \ldots, F_k\}$ in G^* as follows. We set $F_j = \{e_j\} \cup E_j' \cup (\cup_{i=1}^{\ell} E(T_{i,j}))$. By construction, the F_j's are edge-disjoint.

Lemma 4.2 *For $1 \leq j \leq k$, F_j is a f-connector.*

Proof: Recall that we already argued the the case when $S = \emptyset$. Let $Y \subset V$ such that $f(Y) = 1$. Note that Y cannot contain S_i for any i for otherwise $f(Y) = 0$ since $f(S_i) = 0$. In addition, if Y "properly" intersects some S_i, then there is an edge $e \in E(T_{i,j})$ that crosses Y (that is, $e \in \delta_{G^*}(Y)$). Therefore it is sufficient to restrict attention to those sets Y such that $Y \subseteq S$. Note that $e_j \in \delta_{G^*}(S)$ and $e_j \in F_j$; therefore e_j satisfies S if $f(S) = 1$. So suppose Y is a proper subset of S. Since T_j is a spanning tree in $G_2^*[S]$, there is an edge e in E_j that crosses Y. If e is an edge from G^*, then $e \in E'_j$ and hence $e \in F_j$. Otherwise e is an edge obtained in the splitting-off process at s and we replace e by e' and e'' in E'_j. Since at least one of e' and e'' crosses Y in G^*, the proof is complete. ■

We have thus shown the existence of k f-connectors in G^*. It remains to argue that these f-connectors can be found in time polynomial in n and $\log k$. We observe that the only non-trivial part in converting the existence proof into an algorithmic proof is the splitting-off step at s when $S \neq \emptyset$, and in the use of Lemma 2.5 to find spanning trees in $G^*[S_i]$ for $1 \leq i \leq \ell$, and in $G_2^*[S]$. The arguments in the proof of Theorem 3.1 can be used identically here to implement these steps in polynomial time.

4.3 Reduction to Split Instances

We now consider arbitrary $\{0, 1\}$ skew supermodular functions. We describe a reduction of Conjecture 1.4 to a restricted class of instances that we next define. Given a function f and a feasible fractional solution $x \in P(G, f)$ we call (G, f, x) a *split instance* if $x \in P(G, f)$ and there is a subset of nodes $S \subset V$ such that

- for every $A \subseteq S$, $x(\delta(A)) \geq 1$, that is no subset of S is strongly deficient for x, and
- for every $A \subset V \setminus S$, $f(A) = 0$.

Theorem 4.3 *Let f be a $\{0, 1\}$ skew supermodular function and $x \in P(G, f)$ such that kx is integral. Given G, x, k there is an algorithm that obtains a split instance (G', f', x') such that (i) f' is a $\{0, 1\}$ skew supermodular function, (ii) $x' \in P(G', f')$ and (iii) $2kx$ is decomposable into k f-connectors in G if $2kx'$ is decomposable into k f'-connectors in G'.*

If the following conjecture is true, so is Conjecture 1.4.

Conjecture 4.4 *Let f be $\{0, 1\}$ skew supermodular function on G and let (G, f, x) induce a split instance. If kx is integral, then there exist f-connected integer vectors $h_1, h_2, \ldots, h_k$ such that $2kx \geq \sum_i h_i$.*

Theorem 4.3 does not claim polynomial time for the algorithm that reduces a given instance to a split instance. We give a sketch of the proof of Theorem 4.3. In the following we assume that kx is integral and that G^* is the Eulerian graph induced by $2kx$ and G. The algorithm starts with an instance (G, f, x) and loops between two phases: a *deficient shrinking phase* and a 1-*set shrinking phase* and it stops once it produces a split instance. In each phase some non-singleton subset of nodes Y is shrunk into a single node y and the connectivity function is modified for the new graph G'. More precisely if f is the original function on G then we obtain a new $\{0, 1\}$ function f' in G' as follows: (i) $f'(\{y\}) = f(Y)$, (ii) for $A \subset V \setminus Y$, $f'(A) = f(A)$, and (ii) for $A \supset Y$, $f'(\{y\} \cup (A \setminus Y)) = f(A)$. It is easy to check that for any Y, the function f' is skew supermodular if f is.

Deficient Shrinking Phase: This phase is similar to the first step in Section 4.2 on downward monotone functions. Given $x \in P(G, f)$, let $\mathcal{S} = \{S_1, S_2 \ldots, S_\ell\}$ be the minimal strongly deficient sets for x. Lemma 2.3 implies that these sets are disjoint. Also, by Lemma 2.5, for $1 \leq i \leq \ell$, we can find k edge-disjoint spanning trees in the graph $G_i^* = G^*[S_i]$.

Consider the problem G', f' obtained by shrinking each S_i to a single node, and defining f' as the restriction of f to this modified graph. Let $E' \subseteq E(G')$ and E'' be a subset of edges from the G_i^*'s. If E' induces an f'-connected graph in the smaller instance G', f', and E'' includes a spanning tree for each G_i^*, then $E' \cup E''$ induces an f-connected subgraph of G. Moreover, for any f-connected subgraph $H = (V, F)$ we must have that $F \cap E(G')$ induces an f'-connected graph in G'. Thus it suffices to focus on the reduced problem for G', f'.

If we have a split instance after the deficit shrinking step, we stop the procedure. Otherwise, we continue to a 1-set shrinking phase.

1-set shrinking phase: Such a phase begins with an instance G', f' and a subset S (possibly empty, and arising from the deficient shrinking phase in G, f with $S = V(G) - (\cup_i S_i)$), such that (1) for each $v \in V(G') - S$, we have $f'(v) = 0$ and (2) for each subset $Y \subseteq S$, Y is not strongly deficient. Note that (1) follows from our processing because $f(S_i) = 0$ since S_i was strongly deficient, (2) follows since otherwise, Y would contain a minimal strongly deficient set and so would have been one of the S_i's in the deficient shrinking phase.

We now consider any minimal $A \subseteq V(G') \setminus S$ such that $f'(A) = 1$. One notes that the minimal sets of this type are node-disjoint by skew supermodularity of f'. Also, if there is no such set, we have a split instance and so we would have terminated. Note also that since each $v \in V(G') \setminus S$ satisfies $f'(v) = 0$, we have that $|A| \geq 2$ for any such A. We shrink A without affecting feasibility, since any set Y with $f'(Y) = 1$ is not A-splitting, for otherwise skew supermodularity would imply a proper subset of A has $f'(A) = 1$ which contradicts the minimality of A. Since $|A| \geq 2$, such a shrinking operation reduces the size of the graph. Upon completion of 1-set shrinking we return to deficient shrinking.

This completes the description of the procedure. After every pair of phases, we shrink some nontrivial subset, and hence after at most n iterations, we obtain a split instance. This finishes the proof sketch of Theorem 4.3.

Recall that we do not claim that the the reduction to a split instance can be carried out in polynomial time. The bottleneck is the step in the 1-set reduction that requires us to find a minimal set $A \subseteq V(G') \setminus S$ such that $f'(A) = 1$.

5 What Problem is f-connectivity Solving?

Given a specific $\{0, 1\}$ skew supermodular function f, it is natural to ask *what problem is f-connectivity solving*? In other words, in which cases does a supermodular function f encode a problem of more natural combinatorial significance? To make this more concrete, we give several (positive and negative) results related to this agenda.

In each case, we may have a graph $G = (V, E)$ and a skew supermodular function f. Our goal is to build minimum cost networks that are f-connected. Throughout this section, we refer to a set A as *good* if $f(A) = 1$; otherwise it is *bad*. As usual, G represents where we may install capacity and so it does not play a central role in this section. Instead, we explore whether certain functions f can be interpreted as a connectivity problem in a related network.

5.1 Connectivity Augmentation

We first examine two instances where f-connectivity is encoding a *connectivity augmentation problem* in some graph $G' = (V, E')$. In other words, where there is a set of edges E' and list of terminal pairs $s_i t_i$ for some $i = 1, 2 \ldots, k$ such that a set $S \subseteq V$ is good (for f) if and only if $\delta_{G'}(S) = \emptyset$ and there is some pair with $s_i \in S, t_i \notin S$.

5.1.1 Fully Supermodular Functions

A set function f is *fully supermodular* if $f(V) = f(\emptyset) = 0$ and for all A, B we have $f(A) + f(B) \leq f(A \cup B) + f(A \cap B)$. Again, more generally these may be defined in terms of an intersecting family $\mathcal{F}$ of sets, but we only present our arguments in the case where all sets are in the family. Such functions can be used to generalize a number of classical results in combinatorial optimization, including Edmonds' Disjoint Branching Theorem. This was proposed by Frank [16] who actually introduced the more general class of *intersecting supermodular* functions, that only require the inequality above for A, B with nonempty intersection.

We show that f-connectivity network design for $\{0, 1\}$ fully supermodular functions f arise as a *connectivity augmentation problem.* Namely, we show that there is a set of "auxiliary" edges E' such that in the graph $G' = (V, E')$ there exist nodes s, t and $f(S) = 1$ if and only if S is $\{s, t\}$-splitting and $\delta(S) \cap E' = \emptyset$. Thus finding a minimum cost f-connected graph is the same as finding a minimum cost set $F \subseteq E$ of edges such that s, t are connected in $G[V, E' \cup F]$. We mention that the following argument applies equally well to directed network design problems.

For any pair of good sets, A, B we have $A \cap B, A \cup B$ are also good. Thus there is a unique maximal good set M and a unique minimal good set S. Since $M \neq V, S \neq \emptyset$, we may choose an arbitrary $s \in S$ and an arbitrary $t \in V - M$ and so every good set contains s and not t.

We obtain the above claimed G' by starting with the empty graph $G^0 := (V, \emptyset)$ and adding edges in an iterative fashion. In iteration i we find a single edge e^i that we add to G^i to obtain G^{i+1}. This is done as follows. Suppose there is some bad set X such that $\delta_{G^i}(X) = \emptyset$. (If there is no such set then $G' = G^i$ and the procedure terminates.) We show that there is some edge $e^i = uv$ such that $u \in X, v \notin X$, and e^i is not contained in any $\delta(A)$ for a good set A. Suppose this is not the case, then for each $u \in X$ and $v \in V - X$, there is a good set Y_{uv} containing u but not v. Fix some $v \in V - X$ and note that $Y(v) = \cup_{u \in X} Y_{uv}$ is also good and $X \subseteq Y(v)$. But then $\cap_{v \in V - X} Y(v)$ is also good and evidently this set is just X, a contradiction. Thus after some $\ell \leq \binom{n}{2}$ iterations, we have that in G^ℓ, a cut $\delta_{G^\ell}(A)$ is empty if and only if it is good. A similar proof yields an analogous result for directed graphs.

5.1.2 Fastidious Functions

Recall that a $\{0, 1\}$ proper function f is a symmetric set function $f : V \to \{0, 1\}$ such that $f(A \cup B) \leq \max\{f(A), f(B)\}$. In the next section, we see that not all proper functions arise from connectivity augmentation. In this section, we consider a subclass of proper functions that arise from Steiner forest problems. A symmetric function is *fastidious* if no good set is the union of bad sets. (For proper functions, no good set is the *disjoint* union of bad sets.) We show that *fastidious functions are precisely those that encode Steiner forest problems.* One direction is clear: a Steiner forest problem obviously gives rise to a fastidious function. We now show the converse.

Given a fastidious f, we define a graph $H = (V, E(H))$ where

$$E(H) = \{uv : \text{every } \{u, v\}\text{-splitting set } S \text{ is good (that is } f(S) = 1)\}.$$

Let $S_1, S_2, \ldots, S_\ell$ be the connected components of H. We now claim that a set X is good according to f if and only if it splits some S_i, which would give us the desired result. Suppose this is not the case, and let X be a minimal good set such that $\delta_H(X) = \emptyset$. That is for any $u \in X$ and $v \in V - X$, there is a bad set Y_{uv} such that $u \in Y_{uv}$ and $v \notin Y_{uv}$. Consider two cases. Suppose first that for some pair, such a set exists with $X - Y_{uv}$ nonempty. Then since f is proper, either $X - Y_{uv}$ or $Y_{uv} \cap X$ is good. By minimality of X, there is some edge zw of H with $z \in X - Y_{uv}$ and $w \in Y_{uv} \cap X$. But any such edge must lie in $\delta_H(Y_{uv})$, contradicting the fact that $f(Y_{uv}) = 0$.

In the second case, for every pair uv with $u \in X, v \in V \setminus X$, we have $X \subseteq Y_{uv}$. But then the $(V - Y_{uv})$'s are a collection of bad sets whose union is the good set $V - X$, a contradiction.

5.2 Skew Supermodular Functions and Embedded Connectivity

We have seen several cases of supermodular functions encoding an underlying (or hidden) connectivity problem. We cannot expect to be as lucky for general skew supermodular functions. Consider the node cover problem that arises from the skew supermodular function $f : V \rightarrow \{0,1\}$ where a set S is good if and only if S is a singleton. One may deduce that there is no graph $G' = (V, E')$ for which f encodes a connectivity augmentation problem on G'. However, we may cast the problem in this form if we embed in a larger graph and allow higher connectivity requirements: take $H = (V + s, \{sv : v \in V\})$. Consider the problem of adding edges E'' to H so that each node v is 2-edge-connected to s in $H + E''$. The good sets for f now correspond precisely to the "deficit cuts" for this 2-connectivity problem. In general, we may define an *embedded connectivity problem* consists of a pair of graphs $G = (V, E), H' = (V', E')$ with $V \subseteq V'$, ℓ node pairs $s_1t_1, \ldots, s_\ell t_\ell$ where $s_i, t_i \in V'$ for each i, and integers k_i for $i = 1, 2 \ldots, \ell$. For such an instance, we call a subset $S \subseteq V'$ a *target set* if for some i, S is $\{s_i, t_i\}$-splitting and $|\delta_{H'}(S)| < k_i$, the target connectivity for s_i, t_i. The following is easily shown.

Fact 5.1 *Any embedded connectivity problem gives rise to a skew supermodular function.*

We believe that analyzing which functions encode embedded connectivity problems is a potentially fruitful direction for further study. Such embedding problems would seem, however, of limited use unless the size of H' is polynomially bounded in G (and somehow f). The focus should thus be on *p-bounded* problems, where in addition $|V'| \leq p(|V|)$, for some polynomial p.

6 Conclusions

We have shown how splitting-off combined with Theorem 1.1 yields decomposition and rounding algorithms for a large class of 0-1 network design problems. Several open problems remain. First, it would be interesting to resolve Conjecture 1.4. If the decomposition algorithm can be generalized to integer-valued skew supermodular functions, it would yield an alternative algorithm to that of Jain [27]. It would also yield a combinatorial rounding algorithm for the Steiner network problem. Second, the inverse f-connectivity questions raised in Section 5 are of interest in their own right, and may also prove useful in resolving Conjecture 1.4.

Acknowledgement: This work was done while both authors were at Lucent Bell Labs and their work was partially funded by a basic research grant number N00014-03-M-0141 from ONR to Lucent Bell Labs. We thank the anonymous referees for useful comments that greatly improved the paper.

References

[1] M. Andrews and L. Zhang. Bounds on fiber minimization in optical networks with fixed fiber capacity. *Proc. of IEEE INFOCOM*, 409–419, 2005.

[2] M. Andrews and L. Zhang. Complexity of wavelength assignment in optical network optimization. *Proc. of IEEE INFOCOM*, 2006.

[3] A. Agrawal, P. Klein, and R. Ravi. When trees collide: an approximation algorithm for the generalized Steiner problem in networks. *SIAM J. on Computing*, 24:440–456, 1995.

[4] J. Bang-Jensen, A. Frank and B. Jackson. Preserving and increasing local edge-connectivity in mixed graphs. *SIAM J. on Discrete Math*, 8:155–178, 1995.

[5] S. Baum and L. E. Trotter Jr. Integer rounding and polyhedral decomposition for totally unimodular systems. *Optimization and Operations Research* (Proceedings Workshop Bad Honnef, 1977; R. Henn, B. Korte, W. Oettli, eds.) [Lecture Notes in Economics and Mathematical Systems 157], Springer, Berlin, 15–23, 1978.

[6] S. Baum and L. E. Trotter Jr. Integer rounding for polymatroid and branching optimization problems. *SIAM J. on Algebraic and Discrete Methods*, 2, 416–425, 1981.

[7] D. Bienstock, M.X. Goemans, D. Simchi-Levi and D. Williamson. A Note on the Prize Collecting Traveling Salesman Problem. *Mathematical Programming*, 59:413–420, 1993.

[8] A. Blum, R. Ravi and S. Vempala. A Constant-Factor Approximation for the k-MST Problem. *J. of Computer and System Sciences*, 58(1), 101–108, 1999.

[9] R. Carr and S. Vempala. Randomized meta-rounding. *Random Structures and Algorithms*, 20(3):343–352, 2002.

[10] C. Chekuri, M. Mydlarz and F.B. Shepherd. Multicommodity Demand Flow in a Tree and Packing Integer Programs. *ACM Trans. on Algorithms*, 3(3), 2007.

[11] J. Edmonds. Minimum partition of a matroid into independent sets. *J. Res. Nat. Bur. of Standards Sect.*, 869:67–72, 1965.

[12] J. Edmonds. Edge-disjoint branchings. Combinatorial Algorithms (B. Rustin, ed.), Acad. Press, New York, pages 91–96, 1973.

[13] J. Edmonds and R. Giles, A min-max relation for submodular functions on graphs, *Annals of Discrete Math.*, **1**: 185–204, 1977.

[14] T. Erlebach, A. Pagourtzis, K. Potika and S. Stefanakos. Resource allocation problems in Multifiber WDM Tree Networks. *Proc. of Workshop on Graph Theoretic Concepts in Computer Science*, LNCS 2880, Springer-Verlag, 218–229, 2003.

[15] T. Fukunaga and H. Nagamochi. The Set Connector Problem in Graphs *Proc. of IPCO*, 484–498, 2007.

[16] A. Frank. Kernel systems of directed graphs. *Acta Scientiarum Mathematicarum*, 41: 63–76, 1979.

[17] A. Frank Connectivity augmentation problems in network design. *Mathematical Programming: State of the Art*, 1994, eds., J.R. Birge and K.G. Murty, The University of Michigan Press, pp. 34-63.

[18] A. Frank. Applications of relaxed submodularity. *Proc. of the International Congress of Mathematics*, Berlin, 1998, Vol. III: Invited Lectures, Documenta Mathematica, Extra Volume ICM 1998, eds. G. Fischer and U. Rehmann, pp. 343-354.

[19] A. Frank, T. Kiraly, and M. Kriesell. On decomposing a hypergraph into k connected subhypergraphs. *Discrete Applied Mathematics*, 131(2):373–383, 2003. Also available as Technical Report TR-2001-02, Egervary Research Group on Combinatorial Optimization, 2001.

[20] Naveen Garg. A 3-Approximation for the Minimum Tree Spanning k Vertices. *Proc. of IEEE FOCS*, 302–309, 1996.

[21] N. Garg and R. Khandekar. Fast Approximation Algorithms for Fractional Steiner Forest and Related Problems. *Proc. of IEEE FOCS*, 500-509, 2002.

[22] M. Goemans and D. Williamson. A general approximation technique for constrained forest problems. *SIAM J. on Computing*, 24:296–317, 1995.

[23] M. Goemans and D. Williamson. The Primal-Dual Method for Approximation Algorithms and its Application to Network Design Problems. Book chapter in *Approximation Algorithms for NP-Hard Problems*, D. Hochbaum, ed., PWS Publishing Co., 1997.

[24] M. Grötschel, L. Lovász and A. Schrijver. Geometric Algorithms and Combinatorial Optimization. Springer-Verlag, 1988.

[25] M. Grötschel, A. Martin and R. Weismantel. Packing Steiner trees: polyhedral investigations. *Math. Programming A*, 72(2):101–123, 1996.

[26] M. Grötschel, A. Martin and R. Weismantel. Packing Steiner Trees: Separation Algorithms. *SIAM J. on Discrete Mathematics*, 9(2): 233–257, 1996.

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

[28] K. Jain, M. Mahdian, and M. R. Salvatipour. Packing Steiner Trees. *Proc. of ACM-SIAM SODA*, 266–274, 2003.

[29] M. Kriesell. Local spanning trees in graphs and hypergraph decomposition with respect to edge connectivity. Technical Report 257, University of Hannover, 1999.

[30] L. C. Lau. An approximate max-Steiner-tree-packing min-Steiner-cut theorem. *Combinatorica*, 27(1): 71–90, 2007.

[31] L. C. Lau. Packing Steiner forests. *Proc. of IPCO*, 362–276, 2005.

[32] L. Lovász. [Unsolved Problems] in: *Proceedings of the Fifth British Combinatorial Conference* (Aberdeen, 1975; C.St.J.A. Nash-Williams, Win-Sheehan, eds.) [Congressun Numerantium XV], Utilitas Mathematica, Winnipeg, Manitoba, 1976, 638–685.

[33] C. Lucchesi and D. Younger, A minimax theorem for directed graphs, *J. London Math. Society (2)*, **17**: 369–374, 1978.

[34] W. Mader. A reduction method for edge-connectivity in graphs. In *Advances in Graph Theory* (Proceedings of Cambridge Combinatorial Conference, Cambridge, 1977; B. Bollobas, ed.) [Annals of Discrete Mathematics 3], North-Holland, Amsterdam, 1978, 145–164.

[35] H. Nagamochi and T. Ibaraki. Polyhedral structure of submodular and posi-modular systems. *Discrete Applied Math.*, 107(1):165–189, 2000.

[36] A. Schrijver. Theory of Linear and Integer Programming. John Wiley and Sons, 1987.

[37] A. Schrijver. Combinatorial Optimization: Polyhedra and Efficiency. Springer-Verlag, 2003.

[38] D. P. Williamson, M. X. Goemans, M. Mihail, and V. Vazirani. A primal-dual approximation algorithm for generalized Steiner network problems. *Combinatorica*, 15(3):435–454, 1995.