On Generalizations of Network Design Problems with Degree Bounds

Nikhil Bansal[1], Rohit Khandekar[1], Jochen Könemann[2],
Viswanath Nagarajan[1], and Britta Peis[3]

[1] IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
[2] University of Waterloo
[3] Technische Universität Berlin

Abstract. Iterative rounding and relaxation have arguably become the method of choice in dealing with unconstrained and constrained network design problems. In this paper we extend the scope of the iterative relaxation method in two directions: (1) by handling more complex degree constraints in the minimum spanning tree problem (namely *laminar* crossing spanning tree), and (2) by incorporating 'degree bounds' in other combinatorial optimization problems such as *matroid intersection* and *lattice polyhedra*. We give new or improved approximation algorithms, hardness results, and integrality gaps for these problems.

1 Introduction

Iterative rounding and relaxation have arguably become the method of choice in dealing with unconstrained and constrained network design problems. Starting with Jain's elegant *iterative rounding* scheme for the generalized Steiner network problem in [14], an extension of this technique (iterative *relaxation*) has more recently lead to breakthrough results in the area of constrained network design, where a number of linear constraints are added to a classical network design problem. Such constraints arise naturally in a wide variety of practical applications, and model limitations in processing power, bandwidth or budget. The design of powerful techniques to deal with these problems is therefore an important goal.

The most widely studied constrained network design problem is the *minimum-cost degree-bounded spanning tree* problem. In an instance of this problem, we are given an undirected graph, non-negative costs for the edges, and positive, integral degree-bounds for each of the nodes. The problem is easily seen to be NP-hard, even in the absence of edge-costs, since finding a spanning tree with maximum degree two is equivalent to finding a Hamiltonian Path. A variety of techniques have been applied to this problem [5,6,11,17,18,23,24], culminating in Singh and Lau's breakthrough result in [27]. They presented an algorithm that computes a spanning tree of at most optimum cost whose degree at each vertex v exceeds its bound by at most 1, using the *iterative relaxation* framework developed in [20,27].

The iterative relaxation technique has been applied to several constrained network design problems: spanning tree [27], survivable network design [20,21], directed graphs with intersecting and crossing super-modular connectivity [20,2]. It has also been applied to degree bounded versions of matroids and submodular flow [15].

F. Eisenbrand and B. Shepherd (Eds.): IPCO 2010, LNCS 6080, pp. 110–123, 2010.

In this paper we further extend the applicability of iterative relaxation, and obtain new or improved bicriteria approximation results for minimum crossing spanning tree (MCST), crossing matroid intersection, and crossing lattice polyhedra. We also provide hardness results and integrality gaps for these problems.

Notation. As is usual, when dealing with an undirected graph $G = (V, E)$, for any $S \subseteq V$ we let $\delta_G(S) := \{(u, v) \in E \mid u \in S,\ v \notin S\}$. When the graph is clear from context, the subscript is dropped. A collection $\{U_1, \cdots, U_t\}$ of vertex-sets is called *laminar* if for every pair U_i, U_j in this collection, we have $U_i \subseteq U_j$, $U_j \subseteq U_i$, or $U_i \cap U_j = \emptyset$. A $(\rho, f(b))$ approximation for minimum cost degree bounded problems refers to a solution that (1) has cost at most ρ times the optimum that satisfies the degree bounds, and (2) satisfies the relaxed degree constraints in which a bound b is replaced with a bound $f(b)$.

1.1 Our Results, Techniques and Paper Outline

Laminar MCST. Our main result is for a natural generalization of bounded-degree MST (called Laminar Minimum Crossing Spanning Tree or *laminar MCST*), where we are given an edge-weighted undirected graph with a laminar family $\mathcal{L} = \{S_i\}_{i=1}^m$ of vertex-sets having bounds $\{b_i\}_{i=1}^m$; and the goal is to compute a spanning tree of minimum cost that contains at most b_i edges from $\delta(S_i)$ for each $i \in [m]$.

The motivation behind this problem is in designing a network where there is a hierarchy (i.e. laminar family) of service providers that control nodes (i.e. vertices). The number of edges crossing the boundary of any service provider (i.e. its vertex-cut) represents some cost to this provider, and is therefore limited. The laminar MCST problem precisely models the question of connecting all nodes in the network while satisfying bounds imposed by all the service providers.

From a theoretical viewpoint, cut systems induced by laminar families are well studied, and are known to display rich structure. For example, *one-way cut-incidence matrices* are matrices whose rows are incidence vectors of directed cuts induced by the vertex-sets of a laminar family; It is well known (e.g., see [19]) that such matrices are totally unimodular. Using the laminar structure of degree-constraints and the iterative relaxation framework, we obtain the following main result, and present its proof in Section 2.

Theorem 1. *There is a polynomial time $(1, b + O(\log n))$ bicriteria approximation algorithm for laminar MCST. That is, the cost is no more than the optimum cost and the degree violation is at most additive $O(\log n)$. This guarantee is relative to the natural LP relaxation.*

This guarantee is substantially stronger than what follows from known results for the general *minimum crossing spanning tree* (MCST) problem: where the degree bounds could be on arbitrary edge-subsets $E_1, \ldots, E_m$. In particular, for general MCST a $(1, b + \Delta - 1)$ [2,15] is known where Δ is the maximum number of degree-bounds an edge appears in. However, this guarantee is not useful for laminar MCST as Δ can be as large as $\Omega(n)$ in this case. If a multiplicative factor in the degree violation is allowed, Chekuri et al. [8] recently gave a very elegant $\left(1, (1 + \epsilon)b + O(\frac{1}{\epsilon} \log m)\right)$ guarantee (which subsumes the previous best $(O(\log n), O(\log m)\, b)$ [4] result). However, these results also cannot be used to obtain a small additive violation, especially if b is large. In particular, both the results [4,8] for general MCST are based on the natural LP relaxation, for which there is an integrality gap of $b + \Omega(\sqrt{n})$ even without regard to costs and when $m = O(n)$ [26] (see also [3]). On the other hand, Theorem 1 shows that a purely additive $O(\log n)$ guarantee on degree (relative to the LP relaxation and even in presence of costs) is indeed achievable for MCST, when the degree-bounds arise from a laminar cut-family.

The algorithm in Theorem 1 is based on iterative relaxation and uses two main new ideas. Firstly, we drop a carefully chosen *constant fraction of degree-constraints* in each iteration. This is crucial as it can be shown that dropping one constraint at a time as in the usual applications of iterative relaxation can indeed lead to a degree violation of $\Omega(\Delta)$. Secondly, the algorithm does not just drop degree constraints, but in some iterations it also *generates new degree constraints*, by merging existing degree constraints.

All previous applications of iterative relaxation to constrained network design treat connectivity and degree constraints rather asymmetrically. While the structure of the connectivity constraints of the underlying LP is used crucially (e.g., in the ubiquitous uncrossing argument), the handling of degree constraints is remarkably simple. Constraints are dropped one by one, and the final performance of the algorithm is good only if the number of side constraints is small (e.g., in recent work by Grandoni et al. [12]), or if their structure is simple (e.g., if the 'frequency' of each element is small). In contrast, our algorithm for laminar MCST exploits the structure of degree constraints in a non-trivial manner.

Hardness Results. We obtain the following hardness of approximation for the *general MCST* problem (and its matroid counterpart). In particular this rules out any algorithm for MCST that has additive constant degree violation, even without regard to costs.

Theorem 2. *Unless $\mathcal{NP}$ has quasi-polynomial time algorithms, the MCST problem admits no polynomial time $O(\log^{\alpha} m)$ additive approximation for the degree bounds for some constant $\alpha > 0$; this holds even when there are no costs.*

The proof for this theorem is given in Section 3, and uses a a two-step reduction from the well-known *Label Cover* problem. First, we show hardness for a *uniform* matroid instance. In a second step, we then demonstrate how this implies the result for MCST claimed in Theorem 2.

Note that our hardness bound nearly matches the result obtained by Chekuri et al. in [8]. We note however that in terms of *purely* additive degree guarantees, a large gap remains. As noted above, there is a much stronger lower bound of $b + \Omega(\sqrt{n})$ for LP-based algorithms [26] (even without regard to costs), which is based on discrepancy. In light of the small number of known hardness results for discrepancy type problems, it is unclear how our bounds for MCST could be strengthened.

Degree Bounds in More General Settings. We consider crossing versions of other classic combinatorial optimization problems, namely *matroid intersection* and *lattice polyhedra*. We discuss our results briefly and defer the proofs to the full version of the paper [3].

Definition 1 (Minimum crossing matroid intersection problem). *Let $r_1, r_2 : 2^E \to \mathbb{Z}$ be two supermodular functions, $c : E \to \mathbb{R}$ and $\{E_i\}_{i \in I}$ be a collection of subsets of E with corresponding bounds $\{b_i\}_{i \in I}$. Then the goal is to minimize:*

$$\{c^T x \mid \quad x(S) \geq \max\{r_1(S), r_2(S)\}, \forall\, S \subseteq E; \quad x(E_i) \leq b_i,\ \forall\, i \in [m]; \quad x \in \{0,1\}^E\}.$$

We remark that there are alternate definitions of matroid intersection (e.g., see Schrijver [25]) and that our result below extends to those as well.

Let $\Delta = \max_{e \in E} |\{i \in [m] \mid e \in E_i\}|$ be the largest number of sets E_i that any element of E belongs to, and refer to it as *frequency*.

Theorem 3. *Any optimal basic solution x^* of the linear relaxation of the minimum crossing matroid intersection problem can be rounded into an integral solution $\hat{x}$ such that $\hat{x}(S) \geq \max\{r_1(S), r_2(S)\}$ for all $S \subseteq E$ and*

$$c^T \hat{x} \leq 2c^T x^* \quad \textit{and} \quad \hat{x}(E_i) \leq 2b_i + \Delta - 1 \quad \forall i \in I.$$

The algorithm for this theorem again uses iterative relaxation, and its proof is based on a 'fractional token' counting argument similar to the one used in [2].

An interesting special case is for the *bounded-degree arborescence* problem (where $\Delta = 1$). As the set of arborescences in a digraph can be expressed as the intersection of partition and graphic matroids, Theorem 3 readily implies a $(2, 2b)$ approximation for this problem. This is an improvement over the previously best-known $(2, 2b + 2)$ bound [20] for this problem.

The bounded-degree arborescence problem is potentially of wider interest since it is a relaxation of ATSP, and it is hoped that ideas from this problem lead to new ideas for ATSP. In fact Theorem 3 also implies an improved $(2, 2b)$-approximation for the *bounded-degree arborescence packing* problem, where the goal is to pack a given number of arc-disjoint arborescences while satisfying degree-bounds on vertices (arborescence packing can again be phrased as matroid intersection). The previously best known bound for this problem was $(2, 2b + 4)$ [2]. We also give the following integrality gap.

Theorem 4. *For any $\epsilon > 0$, there exists an instance of unweighted minimum crossing arborescence for which the LP is feasible, and any integral solution must violate the bound on some set $\{E_i\}_{i=1}^m$ by a multiplicative factor of at least $2 - \epsilon$. Moreover, this instance has $\Delta = 1$, and just one non-degree constraint.*

Thus Theorem 3 is the best one can hope for, relative to the LP relaxation. First, Theorem 4 implies that the multiplicative factor in the degree cannot be improved beyond 2 (even without regard to costs). Second, the lower bound for arborescences with costs presented in [2] implies that no cost-approximation ratio better than 2 is possible, without violating degrees by a factor greater than 2.

Crossing Lattice Polyhedra. Classical *lattice polyhedra* form a unified framework for various discrete optimization problems and go back to Hoffman and Schwartz [13] who proved their integrality. They are polyhedra of type

$$\{x \in [0,1]^E \mid x(\rho(S)) \geq r(S), \quad \forall S \in \mathcal{F}\}$$

where $\mathcal{F}$ is a *consecutive submodular* lattice, $\rho : \mathcal{F} \to 2^E$ is a mapping from $\mathcal{F}$ to subsets of the ground-set E, and $r \in \mathbb{R}^{\mathcal{F}}$ is supermodular. A key property of lattice polyhedra is that the uncrossing technique can be applied which turns out to be crucial in almost all iterative relaxation approaches for optimization problems with degree bounds. We refer the reader to [25] for a more comprehensive treatment of this subject.

We generalize our work further to *crossing lattice polyhedra* which arise from classical lattice polyhedra by adding "degree-constraints" of the form $a_i \le x(E_i) \le b_i$ for a given collection $\{E_i \subseteq E \mid i \in I\}$ and lower and upper bounds $a, b \in \mathbb{R}^I$. We mention that this model covers several important applications including the crossing matroid basis and crossing planar mincut problems, among others.

We can show that the standard LP relaxation for the general crossing lattice polyhedron problem is weak; details are deferred to the full version of the paper in [3]. For this reason, we henceforth focus on a restricted class of crossing lattice polyhedra in which the underlying lattice $(\mathcal{F}, \le)$ satisfies the following monotonicity property

$$(*) \quad S < T \implies |\rho(S)| < |\rho(T)| \qquad \forall\, S, T \in \mathcal{F}.$$

We obtain the following theorem whose proof is given in [3].

Theorem 5. *For any instance of the crossing lattice polyhedron problem in which $\mathcal{F}$ satisfies property $(*)$, there exists an algorithm that computes an integral solution of cost at most the optimal, where all rank constraints are satisfied, and each degree bound is violated by at most an additive $2\Delta - 1$.*

We note that the above property $(*)$ is satisfied for matroids, and hence Theorem 5 matches the previously best-known bound [15] for degree bounded matroids (with both upper/lower bounds). Also note that property $(*)$ holds whenever $\mathcal{F}$ is ordered by inclusion. In this special case, we can improve the result to an additive $\Delta - 1$ approximation if only upper bounds are given.

1.2 Related Work

As mentioned earlier, the basic bounded-degree MST problem has been extensively studied [5,6,11,17,18,23,24,27]. The iterative relaxation technique for degree-constrained problems was developed in [20,27].

MCST was first introduced by Bilo et al. [4], who presented a randomized-rounding algorithm that computes a tree of cost $O(\log n)$ times the optimum where each degree constraint is violated by a multiplicative $O(\log n)$ factor and an additive $O(\log m)$ term. Subsequently, Bansal et al. [2] gave an algorithm that attains an optimal cost guarantee and an additive $\Delta - 1$ guarantee on degree; recall that Δ is the maximum number of degree constraints that an edge lies in. This algorithm used iterative relaxation as its main tool. Recently, Chekuri et al. [8] obtained an improved $\left(1, (1+\epsilon)b + O(\frac{1}{\epsilon}\log m)\right)$ approximation algorithm for MCST, for any $\epsilon > 0$; this algorithm is based on pipage rounding.

The minimum crossing matroid basis problem was introduced in [15], where the authors used iterative relaxation to obtain (1) $(1, b + \Delta - 1)$-approximation when there are only upper bounds on degree, and (2) $(1, b + 2\Delta - 1)$-approximation in the presence of both upper and lowed degree-bounds. The [8] result also holds in this matroid setting. [15] also considered a degree-bounded version of the *submodular flow* problem and gave a $(1, b+1)$ approximation guarantee.

The bounded-degree arborescence problem was considered in Lau et al. [20], where a $(2, 2b+2)$ approximation guarantee was obtained. Subsequently Bansal et al. [2] designed an algorithm that for any $0 < \epsilon \leq 1/2$, achieves a $(1/\epsilon, b_v/(1-\epsilon)+4)$ approximation guarantee. They also showed that this guarantee is the best one can hope for via the natural LP relaxation (for every $0 < \epsilon \leq 1/2$). In the absence of edge-costs, [2] gave an algorithm that violates degree bounds by at most an additive two. Recently Nutov [22] studied the arborescence problem under *weighted* degree constraints, and gave a $(2, 5b)$ approximation for it.

Lattice polyhedra were first investigated by Hoffman and Schwartz [13] and the natural LP relaxation was shown to be totally dual integral. Even though greedy-type algorithms are known for all examples mentioned earlier, so far no combinatorial algorithm has been found for lattice polyhedra in general. Two-phase greedy algorithms have been established only in cases where an underlying rank function satisfies a monotonicity property [10], [9].

2 Crossing Spanning Tree with Laminar Degree Bounds

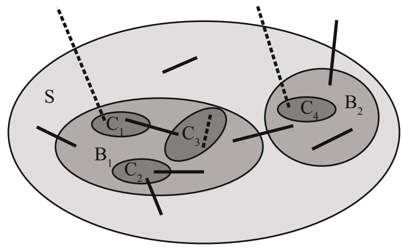
In this section we prove Theorem 1 by presenting an iterative relaxation-based algorithm with the stated performance guarantee. During its execution, the algorithm selects and deletes edges, and it modifies the given laminar family of degree bounds. A generic iteration starts with a subset F of edges already picked in the solution, a subset E of *undecided* edges, i.e., the edges not yet picked or dropped from the solution, a laminar family $\mathcal{L}$ on V, and residual degree bounds $b(S)$ for each $S \in \mathcal{L}$.

The laminar family $\mathcal{L}$ has a natural forest-like structure with *nodes* corresponding to each element of $\mathcal{L}$. A node $S \in \mathcal{L}$ is called the *parent* of node $C \in \mathcal{L}$ if S is the inclusion-wise minimal set in $\mathcal{L} \setminus \{C\}$ that contains C; and C is called a *child* of S. Node $D \in \mathcal{L}$ is called a *grandchild* of node $S \in \mathcal{L}$ if S is the parent of D's parent. Nodes $S, T \in \mathcal{L}$ are *siblings* if they have the same parent node. A node that has no parent is called *root*. The *level* of any node $S \in \mathcal{L}$ is the length of the path in this forest from S to the root of its tree. We also maintain a *linear ordering* of the children of each $\mathcal{L}$-node. A subset $\mathcal{B} \subseteq \mathcal{L}$ is called *consecutive* if all nodes in $\mathcal{B}$ are siblings (with parent S) and they appear consecutively in the ordering of S's children. In any iteration $(F, E, \mathcal{L}, b)$, the algorithm solves the following LP relaxation of the residual problem.

$$\begin{aligned}
\min \quad & \sum_{e \in E} c_e x_e && && (1)\\
\text{s.t.} \quad & x(E(V)) = |V| - |F| - 1 && \\
& x(E(U)) \leq |U| - |F(U)| - 1 && \forall U \subset V \\
& x(\delta_E(S)) \leq b(S) && \forall S \in \mathcal{L} \\
& x_e \geq 0 && \forall e \in E
\end{aligned}$$

For any vertex-subset $W \subseteq V$ and edge-set H, we let $H(W) := \{(u,v) \in H \mid u, v \in W\}$ denote the edges induced on W; and $\delta_H(W) := \{(u,v) \in H \mid u \in W,\ v \notin W\}$ the set of edges crossing W. The first two sets of constraints are spanning tree constraints while the third set corresponds to the degree bounds. Let x denote an optimal *extreme point solution* to this LP. By reducing degree bounds $b(S)$, if needed, we assume that x *satisfies all degree bounds at equality* (the degree bounds may therefore be fractional-valued). Let $\alpha := 24$.

Definition 2. *An edge $e \in E$ is said to be* local *for $S \in \mathcal{L}$ if e has at least one end-point in S but is neither in $E(C)$ nor in $\delta(C) \cap \delta(S)$ for any grandchild C of S. Let* $\text{local}(S)$ *denote the set of local edges for S. A node $S \in \mathcal{L}$ is said to be* good *if* $|\text{local}(S)| \leq \alpha$.



The figure on the left shows a set S, its children B_1 and B_2, and grand-children $C_1, \ldots, C_4$; edges in $\text{local}(S)$ are drawn solid, non-local ones are shown dashed.

Initially, E is the set of edges in the given graph, $F \leftarrow \emptyset$, $\mathcal{L}$ is the original laminar family of vertex sets for which there are degree bounds, and an arbitrary linear ordering is chosen on the children of each node in $\mathcal{L}$. In a generic iteration $(F, E, \mathcal{L}, b)$, the algorithm performs one of the following steps (see also Figure 1):

1. If $x_e = 1$ for some edge $e \in E$ then $F \leftarrow F \cup \{e\}$, $E \leftarrow E \setminus \{e\}$, and set $b(S) \leftarrow b(S) - 1$ for all $S \in \mathcal{L}$ with $e \in \delta(S)$.
2. If $x_e = 0$ for some edge $e \in E$ then $E \leftarrow E \setminus \{e\}$.
3. **DropN:** Suppose there at least $|\mathcal{L}|/4$ good non-leaf nodes in $\mathcal{L}$. Then either odd-levels or even-levels contain a set $\mathcal{M} \subseteq \mathcal{L}$ of $|\mathcal{L}|/8$ good non-leaf nodes. Drop the degree bounds of all *children* of $\mathcal{M}$ and modify $\mathcal{L}$ accordingly. The ordering of siblings also extends naturally.
4. **DropL:** Suppose there are more than $|\mathcal{L}|/4$ good leaf nodes in $\mathcal{L}$, denoted by $\mathcal{N}$. Then partition $\mathcal{N}$ into parts corresponding to siblings in $\mathcal{L}$. For any part $\{N_1, \cdots, N_k\} \subseteq \mathcal{N}$ consisting of ordered (not necessarily contiguous) children of some node S:
   (a) Define $M_i = N_{2i-1} \cup N_{2i}$ for all $1 \leq i \leq \lfloor k/2 \rfloor$ (if k is odd N_k is not used).
   (b) Modify $\mathcal{L}$ by removing leaves $\{N_1, \cdots, N_k\}$ and adding new leaf-nodes $\{M_1, \cdots, M_{\lfloor k/2 \rfloor}\}$ as children of S (if k is odd N_k is removed). The children of S in the new laminar family are ordered as follows: each node M_i takes the position of either N_{2i-1} or N_{2i}, and other children of S are unaffected.
   (c) Set the degree bound of each M_i to $b(M_i) = b(N_{2i-1}) + b(N_{2i})$.

Assuming that one of the above steps applies at each iteration, the algorithm terminates when $E = \emptyset$ and outputs the final set F as a solution. It is clear that the algorithm outputs a spanning tree of G. An inductive argument (see e.g. [20]) can be used to show that the LP (1) is feasible at each each iteration and $c(F) + z_{cur} \leq z_o$ where z_o is the original LP value, z_{cur} is the current LP value, and F is the chosen edge-set at the current iteration. Thus the cost of the final solution is at most the initial LP optimum z_o. Next we show that one of the four iterative steps always applies.

Lemma 1. *In each iteration, one of the four steps above applies.*

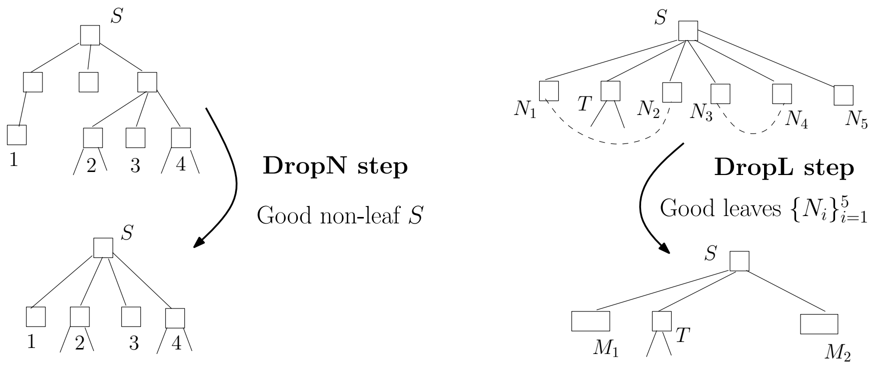


Fig. 1. Examples of the degree constraint modifications DropN and DropL

Proof. Let x^* be the optimal basic solution of (1), and suppose that the first two steps do not apply. Hence, we have $0 < x_e^* < 1$ for all $e \in E$. The fact that x^* is a basic solution together with a standard uncrossing argument (e.g., see [14]) implies that x^* is uniquely defined by

$$x(E(U)) = |U| - |F(U)| - 1 \quad \forall U \in \mathcal{S}, \qquad \text{and} \qquad x(\delta_E(S)) = b(S), \quad \forall S \in \mathcal{L}',$$

where $\mathcal{S}$ is a laminar subset of the tight spanning tree constraints, and $\mathcal{L}'$ is a subset of tight degree constraints, and where $|E| = |\mathcal{S}| + |\mathcal{L}'|$.

A simple counting argument (see, e.g., [27]) shows that there are at least 2 edges induced on each $S \in \mathcal{S}$ that are not induced on any of its children; so $2|\mathcal{S}| \le |E|$. Thus we obtain $|E| \le 2|\mathcal{L}'| \le 2|\mathcal{L}|$.

From the definition of local edges, we get that any edge $e = (u, v)$ is local to at most the following six sets: the smallest set $S_1 \in \mathcal{L}$ containing u, the smallest set $S_2 \in \mathcal{L}$ containing v, the parents P_1 and P_2 of S_1 and S_2 resp., the least-common-ancestor L of P_1 and P_2, and the parent of L. Thus $\sum_{S \in \mathcal{L}} |\text{local}(S)| \le 6|E|$. From the above, we conclude that $\sum_{S \in \mathcal{L}} |\text{local}(S)| \le 12|\mathcal{L}|$. Thus at least $|\mathcal{L}|/2$ sets $S \in \mathcal{L}$ must have $|\text{local}(S)| \le \alpha = 24$, i.e., must be good. Now either at least $|\mathcal{L}|/4$ of them must be non-leaves or at least $|\mathcal{L}|/4$ of them must be leaves. In the first case, step 3 holds and in the second case, step 4 holds. ■

It remains to bound the violation in the degree constraints, which turns out to be rather challenging. We note that this is unlike usual applications of iterative rounding/relaxation, where the harder part is in showing that one of the iterative steps applies.

It is clear that the algorithm reduces the size of $\mathcal{L}$ by at least $|\mathcal{L}|/8$ in each DropN or DropL iteration. Since the initial number of degree constraints is at most $2n - 1$, we get the following lemma.

Lemma 2. *The number of drop iterations (DropN and DropL) is* $T := O(\log n)$.

Performance guarantee for degree constraints. We begin with some notation. The iterations of the algorithm are broken into periods between successive drop iterations: there are exactly T drop-iterations (Lemma 2). In what follows, the t-th drop iteration is called *round* t. The *time* t refers to the instant just after round t; time 0 refers to the start of the algorithm. At any time t, consider the following parameters.

- $\mathcal{L}_t$ denotes the laminar family of degree constraints.
- E_t denotes the undecided edge set, i.e., support of the current LP optimal solution.
- For any set $\mathcal{B}$ of *consecutive siblings* in $\mathcal{L}_t$, $\mathsf{Bnd}(\mathcal{B}, t) = \sum_{N \in \mathcal{B}} b(N)$ equals the sum of the residual degree bounds on nodes of $\mathcal{B}$.
- For any set $\mathcal{B}$ of *consecutive siblings* in $\mathcal{L}_t$, $\mathsf{Inc}(\mathcal{B}, t)$ equals the number of edges from $\delta_{E_t}(\cup_{N \in \mathcal{B}} N)$ included in the final solution.

Recall that b denotes the *residual* degree bounds at any point in the algorithm. The following lemma is the main ingredient in bounding the degree violation.

Lemma 3. *For any set $\mathcal{B}$ of consecutive siblings in $\mathcal{L}_t$ (at any time t),* $\mathsf{Inc}(\mathcal{B}, t) \leq \mathsf{Bnd}(\mathcal{B}, t) + 4\alpha \cdot (T - t)$.

Observe that this implies the desired bound on each original degree constraint S: using $t = 0$ and $\mathcal{B} = \{S\}$, the violation is bounded by an additive $4\alpha \cdot T$ term.

Proof. The proof of this lemma is by induction on $T - t$. The base case $t = T$ is trivial since the only iterations after this correspond to including 1-edges: hence there is no violation in *any* degree bound, i.e. $\mathsf{Inc}(\{N\}, T) \leq b(N)$ for all $N \in \mathcal{L}_T$. Hence for *any* $\mathcal{B} \subseteq \mathcal{L}$, $\mathsf{Inc}(\mathcal{B}, T) \leq \sum_{N \in \mathcal{B}} \mathsf{Inc}(\{N\}, T) \leq \sum_{N \in \mathcal{B}} b(N) = \mathsf{Bnd}(\mathcal{B}, T)$.

Now suppose $t < T$, and assume the lemma for $t + 1$. Fix a consecutive $\mathcal{B} \subseteq \mathcal{L}_t$. We consider different cases depending on what kind of drop occurs in round $t + 1$.

DropN round. Here either all nodes in $\mathcal{B}$ get dropped or none gets dropped.

Case 1: *None of $\mathcal{B}$ is dropped.* Then observe that $\mathcal{B}$ is consecutive in $\mathcal{L}_{t+1}$ as well; so the inductive hypothesis implies $\mathsf{Inc}(\mathcal{B}, t + 1) \leq \mathsf{Bnd}(\mathcal{B}, t + 1) + 4\alpha \cdot (T - t - 1)$. Since the only iterations between round t and round $t + 1$ involve edge-fixing, we have $\mathsf{Inc}(\mathcal{B}, t) \leq \mathsf{Bnd}(\mathcal{B}, t) - \mathsf{Bnd}(\mathcal{B}, t+1) + \mathsf{Inc}(\mathcal{B}, t+1) \leq \mathsf{Bnd}(\mathcal{B}, t) + 4\alpha \cdot (T - t - 1) \leq \mathsf{Bnd}(\mathcal{B}, t) + 4\alpha \cdot (T - t)$.

Case 2: *All of $\mathcal{B}$ is dropped.* Let $\mathcal{C}$ denote the set of all children (in $\mathcal{L}_t$) of nodes in $\mathcal{B}$. Note that $\mathcal{C}$ consists of consecutive siblings in $\mathcal{L}_{t+1}$, and inductively $\mathsf{Inc}(\mathcal{C}, t + 1) \leq \mathsf{Bnd}(\mathcal{C}, t + 1) + 4\alpha \cdot (T - t - 1)$. Let $S \in \mathcal{L}_t$ denote the parent of the $\mathcal{B}$-nodes; so $\mathcal{C}$ are grand-children of S in $\mathcal{L}_t$. Let x denote the optimal LP solution *just before* round $t + 1$ (when the degree bounds are still given by $\mathcal{L}_t$), and $H = E_{t+1}$ the support edges of x. At that point, we have $b(N) = x(\delta(N))$ for all $N \in \mathcal{B} \cup \mathcal{C}$. Also let $\mathsf{Bnd}'(\mathcal{B}, t + 1) := \sum_{N \in \mathcal{B}} b(N)$ be the sum of bounds on $\mathcal{B}$-nodes just before round $t + 1$. Since S is a good node in round $t + 1$, $|\mathsf{Bnd}'(\mathcal{B}, t + 1) - \mathsf{Bnd}(\mathcal{C}, t + 1)| = |\sum_{N \in \mathcal{B}} b(N) - \sum_{M \in \mathcal{C}} b(M)| = |\sum_{N \in \mathcal{B}} x(\delta(N)) - \sum_{M \in \mathcal{C}} x(\delta(M))| \leq 2\alpha$. The last inequality follows since S is good; the factor of 2 appears since some edges, e.g., the edges between two children or two grandchildren of S, may get counted twice. Note also that the symmetric difference of $\delta_H(\cup_{N \in \mathcal{B}} N)$ and $\delta_H(\cup_{M \in \mathcal{C}} M)$ is contained in $\text{local}(S)$. Thus $\delta_H(\cup_{N \in \mathcal{B}} N)$ and $\delta_H(\cup_{M \in \mathcal{C}} M)$ differ in at most α edges.

Again since all iterations between time t and $t + 1$ are edge-fixing:

$$\begin{aligned}\mathsf{Inc}(\mathcal{B}, t) \leq\ & \mathsf{Bnd}(\mathcal{B}, t) - \mathsf{Bnd}'(\mathcal{B}, t + 1) + |\delta_H(\cup_{N \in \mathcal{B}} N) \setminus \delta_H(\cup_{M \in \mathcal{C}} M)| \\ & + \mathsf{Inc}(\mathcal{C}, t + 1)\end{aligned}$$

$$\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + \alpha + \mathsf{Inc}(\mathcal{C},t+1)$$
$$\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + \alpha + \mathsf{Bnd}(\mathcal{C},t+1) + 4\alpha\cdot(T-t-1)$$
$$\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + \alpha + \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha + 4\alpha\cdot(T-t-1)$$
$$\leq \mathsf{Bnd}(\mathcal{B},t) + 4\alpha\cdot(T-t)$$

The first inequality above follows from simple counting; the second follows since $\delta_H(\cup_{N\in\mathcal{B}}N)$ and $\delta_H(\cup_{M\in\mathcal{C}}M)$ differ in at most α edges; the third is the induction hypothesis, and the fourth is $\mathsf{Bnd}(\mathcal{C},t+1) \leq \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha$ (as shown above).

DropL round. In this case, let S be the parent of $\mathcal{B}$-nodes in $\mathcal{L}_t$, and $\mathcal{N} = \{N_1,\cdots,N_p\}$ be all the ordered children of S, of which $\mathcal{B}$ is a subsequence (since it is consecutive). Suppose indices $1 \leq \pi(1) < \pi(2) < \cdots < \pi(k) \leq p$ correspond to good leaf-nodes in $\mathcal{N}$. Then for each $1 \leq i \leq \lfloor k/2 \rfloor$, nodes $N_{\pi(2i-1)}$ and $N_{\pi(2i)}$ are merged in this round. Let $\{\pi(i) \mid e \leq i \leq f\}$ (possibly empty) denote the indices of good leaf-nodes in $\mathcal{B}$. Then it is clear that the only nodes of $\mathcal{B}$ that may be merged with nodes outside $\mathcal{B}$ are $N_{\pi(e)}$ and $N_{\pi(f)}$; all other $\mathcal{B}$-nodes are either not merged or merged with another $\mathcal{B}$-node. Let $\mathcal{C}$ be the inclusion-wise minimal set of *children of* S *in* $\mathcal{L}_{t+1}$ s.t.

- $\mathcal{C}$ is consecutive in $\mathcal{L}_{t+1}$,
- $\mathcal{C}$ contains all nodes of $\mathcal{B} \setminus \{N_{\pi(i)}\}_{i=1}^k$, and
- $\mathcal{C}$ contains all new leaf nodes resulting from merging *two good leaf nodes* of $\mathcal{B}$.

Note that $\cup_{M\in\mathcal{C}}M$ consists of some subset of $\mathcal{B}$ and at most two good leaf-nodes in $\mathcal{N} \setminus \mathcal{B}$. These two extra nodes (if any) are those merged with the good leaf-nodes $N_{\pi(e)}$ and $N_{\pi(f)}$ of $\mathcal{B}$. Again let $\mathsf{Bnd}'(\mathcal{B},t+1) := \sum_{N\in\mathcal{B}} b(N)$ denote the sum of bounds on $\mathcal{B}$ just before drop round $t+1$, when degree constraints are $\mathcal{L}_t$. Let $H = E_{t+1}$ be the undecided edges in round $t+1$. By the definition of bounds on merged leaves, we have $\mathsf{Bnd}(\mathcal{C},t+1) \leq \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha$. The term 2α is present due to the two extra good leaf-nodes described above.

Claim 6. *We have* $|\delta_H(\cup_{N\in\mathcal{B}}N) \setminus \delta_H(\cup_{M\in\mathcal{C}}M)| \leq 2\alpha$.

Proof. We say that $N \in \mathcal{N}$ is represented in $\mathcal{C}$ if either $N \in \mathcal{C}$ or N is contained in some node of $\mathcal{C}$. Let $\mathcal{D}$ be set of nodes of $\mathcal{B}$ that are *not* represented in $\mathcal{C}$ and the nodes of $\mathcal{N} \setminus \mathcal{B}$ that are represented in $\mathcal{C}$. Observe that by definition of $\mathcal{C}$, the set $\mathcal{D} \subseteq \{N_{\pi(e-1)}, N_{\pi(e)}, N_{\pi(f)}, N_{\pi(f+1)}\}$; in fact it can be easily seen that $|\mathcal{D}| \leq 2$. Moreover $\mathcal{D}$ consists of only good leaf nodes. Thus, we have $|\cup_{L\in\mathcal{D}}\,\delta_H(L)| \leq 2\alpha$. Now note that the edges in $\delta_H(\cup_{N\in\mathcal{B}}N) \setminus \delta_H(\cup_{M\in\mathcal{C}}M)$ must be in $\cup_{L\in\mathcal{D}}\delta_H(L)$. This completes the proof. ∎

As in the previous case, we have:

$$\begin{aligned}\mathsf{Inc}(\mathcal{B},t) &\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + |\delta_H(\cup_{N\in\mathcal{B}}N) \setminus \delta_H(\cup_{M\in\mathcal{C}}M)| \\ &\quad + \mathsf{Inc}(\mathcal{C},t+1) \\ &\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha + \mathsf{Inc}(\mathcal{C},t+1) \\ &\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha + \mathsf{Bnd}(\mathcal{C},t+1) + 4\alpha\cdot(T-t-1) \\ &\leq \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha + \mathsf{Bnd}'(\mathcal{B},t+1) + 2\alpha + 4\alpha\cdot(T-t-1) \\ &= \mathsf{Bnd}(\mathcal{B},t) + 4\alpha\cdot(T-t)\end{aligned}$$

The first inequality follows from simple counting; the second uses Claim 6, the third is the induction hypothesis (since $\mathcal{C}$ is consecutive), and the fourth is $\mathsf{Bnd}(\mathcal{C}, t+1) \leq \mathsf{Bnd}'(\mathcal{B}, t+1) + 2\alpha$ (from above).

This completes the proof of the inductive step and hence Lemma 3. ∎

3 Hardness Results

We now prove Theorem 2. The first step to proving this result is a hardness for the more general minimum crossing matroid basis problem: given a matroid $\mathcal{M}$ on a ground set V of elements, a cost function $c : V \to \mathbb{R}_+$, and degree bounds specified by pairs $\{(E_i, b_i)\}_{i=1}^m$ (where each $E_i \subseteq V$ and $b_i \in \mathbb{N}$), find a minimum cost basis I in $\mathcal{M}$ such that $|I \cap E_i| \leq b_i$ for all $i \in [m]$.

Theorem 7. *Unless $\mathcal{NP}$ has quasi-polynomial time algorithms, the unweighted minimum crossing matroid basis problem admits no polynomial time $O(\log^c m)$ additive approximation for the degree bounds for some fixed constant $c > 0$.*

Proof. We reduce from the label cover problem [1]. The input is a graph $G = (U, E)$ where the vertex set U is partitioned into pieces $U_1, \cdots, U_n$ each having size q, and all edges in E are between distinct pieces. We say that there is a *superedge* between U_i and U_j if there is an edge connecting some vertex in U_i to some vertex in U_j. Let t denote the total number of superedges; i.e.,

$$t = \left| \left\{ (i,j) \in \binom{[n]}{2} : \text{ there is an edge in } E \text{ between } U_i \text{ and } U_j \right\} \right|$$

The goal is to pick one vertex from each part $\{U_i\}_{i=1}^n$ so as to maximize the number of induced edges. This is called the value of the label cover instance and is at most t.

It is well known that there exists a universal constant $\gamma > 1$ such that for every $k \in \mathbb{N}$, there is a reduction from any instance of SAT (having size N) to a label cover instance $\langle G = (U, E), q, t \rangle$ such that:

- If the SAT instance is satisfiable, the label cover instance has optimal value t.
- If the SAT instance is not satisfiable, the label cover instance has optimal value $< t/\gamma^k$.
- $|G| = N^{O(k)}$, $q = 2^k$, $|E| \leq t^2$, and the reduction runs in time $N^{O(k)}$.

We consider a uniform matroid $\mathcal{M}$ with rank t on ground set E (recall that any subset of t edges is a basis in a uniform matroid). We now construct a crossing matroid basis instance $\mathcal{I}$ on $\mathcal{M}$. There is a set of degree bounds corresponding to each $i \in [n]$: for every collection C of edges incident to vertices in U_i such that no two edges in C are incident to the same vertex in U_i, there is a degree bound in $\mathcal{I}$ requiring *at most one* element to be chosen from C. Note that the number of degree bounds m is at most $|E|^q \leq N^{O(k\,2^k)}$. The following claim links the SAT and crossing matroid instances. Its proof is deferred to the full version of this paper.

Claim 8. [Yes instance] *If the SAT instance is satisfiable, there is a basis (i.e. subset $B \subseteq E$ with $|B| = t$) satisfying all degree bounds.*

[No instance] *If the SAT instance is unsatisfiable, every subset $B' \subseteq E$ with $|B'| \geq t/2$ violates some degree bound by an additive $\rho = \gamma^{k/2}/\sqrt{2}$.*

The steps described in the above reduction can be done in time polynomial in m and $|G|$. Also, instead of randomly choosing vertices from the sets W_i, we can use conditional expectations to derive a deterministic algorithm that recovers at least t/ρ^2 edges. Setting $k = \Theta(\log\log N)$ (recall that N is the size of the original SAT instance), we obtain an instance of bounded-degree matroid basis of size $\max\{m, |G|\} = N^{\log^a N}$ and $\rho = \log^b N$, where $a, b > 0$ are constants. Note that $\log m = \log^{a+1} N$, which implies $\rho = \log^c m$ for $c = \frac{b}{a+1} > 0$, a constant. Thus it follows that for this constant $c > 0$ the bounded-degree matroid basis problem has no polynomial time $O(\log^c m)$ *additive approximation* for the degree bounds, unless $\mathcal{NP}$ has quasi-polynomial time algorithms. ■

We now prove Theorem 2.

Proof. [Proof of Theorem 2] We show how the bases of a uniform matroid can be represented in a suitable instance of the crossing spanning tree problem. Let the uniform matroid from Theorem 7 consist of e elements and have rank $t \leq e$; recall that $t \geq \sqrt{e}$ and clearly $m \leq 2^e$. We construct a graph as in Figure 2, with vertices $v_1, \cdots, v_e$ corresponding to elements in the uniform matroid. Each vertex v_i is connected to the root r by two vertex-disjoint paths: $\langle v_i, u_i, r\rangle$ and $\langle v_i, w_i, r\rangle$. There are no costs in this instance. Corresponding to each degree bound (in the uniform matroid) of $b(C)$ on a subset $C \subseteq [e]$, there is a constraint to pick at most $|C| + b(C)$ edges from $\delta(\{u_i \mid i \in C\})$. Additionally, there is a *special degree bound* of $2e - t$ on the edge-set $E' = \bigcup_{i=1}^{e} \delta(w_i)$; this corresponds to picking a basis in the uniform matroid.

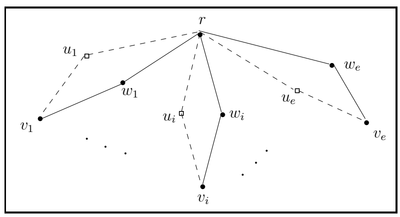


Fig. 2. The crossing spanning tree instance used in the reduction

Observe that for each $i \in [e]$, any spanning tree must choose exactly three edges amongst $\{(r, u_i), (u_i, v_i), (r, w_i), (w_i, v_i)\}$, in fact any three edges suffice. Hence every spanning tree T in this graph corresponds to a subset $X \subseteq [e]$ such that: (I) T contains both edges in $\delta(u_i)$ and one edge from $\delta(w_i)$, for each $i \in X$, and (II) T contains both edges in $\delta(w_i)$ and one edge from $\delta(u_i)$ for each $i \in [e] \setminus X$.

From Theorem 7, for the crossing matroid problem, we obtain the two cases:

Yes instance. There is a basis B^* (i.e. $B^* \subseteq [e]$, $|B^*| = t$) satisfying all degree bounds. Consider the spanning tree

$$T^* = \{(r, u_i), (u_i, v_i), (r, w_i) \mid i \in B^*\} \bigcup \{(r, w_i), (u_i, w_i), (r, u_i) \mid i \in [e] \setminus B^*\}.$$

Since B^* satisfies its degree-bounds, T^* satisfies all degree bounds derived from the crossing matroid instance. For the special degree bound on E', note that $|T^* \cap E'| = 2e - |B^*| = 2e - t$; so this is also satisfied. Thus there is a spanning tree satisfying all the degree bounds.

No instance. Every subset $B' \subseteq [e]$ with $|B'| \geq t/2$ (i.e. near basis) violates some degree bound by an additive $\rho = \Omega(\log^c m)$ term, where $c > 0$ is a fixed constant. Consider any spanning tree T that corresponds to subset $X \subseteq [e]$ as described above.

1. Suppose that $|X| \leq t/2$; then we have $|T \cap E'| = 2e - |X| \geq 2e - t + \frac{t}{2}$, i.e. the special degree bound is violated by $t/2 \geq \Omega(\sqrt{e}) = \Omega(\log^{1/2} m)$.
2. Now suppose that $|X| \geq t/2$. Then by the guarantee on the no-instance, T violates some degree-bound derived from the crossing matroid instance by additive ρ.

Thus in either case, every spanning tree violates some degree bound by additive $\rho = \Omega(\log^c m)$.

By Theorem 7, it is hard to distinguish the above cases and we obtain the corresponding hardness result for crossing spanning tree, as claimed in Theorem 2. ■

References

1. Arora, S., Babai, L., Stern, J., Sweedyk, Z.: The hardness of approximate optima in lattices, codes, and systems of linear equations. J. Comput. Syst. Sci. 54(2), 317–331 (1997)
2. Bansal, N., Khandekar, R., Nagarajan, V.: Additive guarantees for degree bounded network design. In: STOC, pp. 769–778 (2008)
3. Bansal, N., Khandekar, R., Könemann, J., Nagarajan, V., Peis, B.: On Generalizations of Network Design Problems with Degree Bounds (full version),Technical Report (2010)
4. Bilo, V., Goyal, V., Ravi, R., Singh, M.: On the crossing spanning tree problem. In: Jansen, K., Khanna, S., Rolim, J.D.P., Ron, D. (eds.) RANDOM 2004 and APPROX 2004. LNCS, vol. 3122, pp. 51–60. Springer, Heidelberg (2004)
5. Chaudhuri, K., Rao, S., Riesenfeld, S., Talwar, K.: What would Edmonds do? Augmenting paths and witnesses for degree-bounded MSTs. In: Chekuri, C., Jansen, K., Rolim, J.D.P., Trevisan, L. (eds.) APPROX 2005 and RANDOM 2005. LNCS, vol. 3624, pp. 26–39. Springer, Heidelberg (2005)
6. Chaudhuri, K., Rao, S., Riesenfeld, S., Talwar, K.: Push relabel and an improved approximation algorithm for the bounded-degree MST problem. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 191–201. Springer, Heidelberg (2006)
7. Chazelle, B.: The Discrepancy Method: Randomness and Complexity. Cambridge University Press, Cambridge (2000)
8. Chekuri, C., Vondrák, J., Zenklusen, R.: Dependent Randomized Rounding for Matroid Polytopes and Applications (2009), `http://arxiv.org/abs/0909.4348`
9. Faigle, U., Peis, B.: Two-phase greedy algorithms for some classes of combinatorial linear programs. In: SODA, pp. 161–166 (2008)
10. Frank, A.: Increasing the rooted connectivity of a digraph by one. Math. Programming 84, 565–576 (1999)
11. Goemans, M.X.: Minimum Bounded-Degree Spanning Trees. In: FOCS, pp. 273–282 (2006)
12. Grandoni, F., Ravi, R., Singh, M.: Iterative Rounding for Multiobjective Optimization Problems. In: Fiat, A., Sanders, P. (eds.) ESA 2009. LNCS, vol. 5757, pp. 95–106. Springer, Heidelberg (2009)
13. Hoffman, A., Schwartz, D.E.: On lattice polyhedra. In: Hajnal, A., Sos, V.T. (eds.) Proceedings of Fifth Hungarian Combinatorial Coll, pp. 593–598. North-Holland, Amsterdam (1978)
14. Jain, K.: A factor 2 approximation algorithm for the generalized Steiner network problem. In: Combinatorica, pp. 39–61 (2001)
15. Király, T., Lau, L.C., Singh, M.: Degree bounded matroids and submodular flows. In: Lodi, A., Panconesi, A., Rinaldi, G. (eds.) IPCO 2008. LNCS, vol. 5035, pp. 259–272. Springer, Heidelberg (2008)

16. Klein, P.N., Krishnan, R., Raghavachari, B., Ravi, R.: Approximation algorithms for finding low degree subgraphs. Networks 44(3), 203–215 (2004)
17. Könemann, J., Ravi, R.: A matter of degree: Improved approximation algorithms for degree bounded minimum spanning trees. SIAM J. on Computing 31, 1783–1793 (2002)
18. Könemann, J., Ravi, R.: Primal-Dual meets local search: approximating MSTs with nonuniform degree bounds. SIAM J. on Computing 34(3), 763–773 (2005)
19. Korte, B., Vygen, J.: Combinatorial Optimization, 4th edn. Springer, New York (2008)
20. Lau, L.C., Naor, J., Salavatipour, M.R., Singh, M.: Survivable network design with degree or order constraints (full version). In: STOC, pp. 651–660 (2007)
21. Lau, L.C., Singh, M.: Additive Approximation for Bounded Degree Survivable Network Design. In: STOC, pp. 759–768 (2008)
22. Nutov, Z.: Approximating Directed Weighted-Degree Constrained Networks. In: Goel, A., Jansen, K., Rolim, J.D.P., Rubinfeld, R. (eds.) APPROX 2008 and RANDOM 2008. LNCS, vol. 5171, pp. 219–232. Springer, Heidelberg (2008)
23. Ravi, R., Marathe, M.V., Ravi, S.S., Rosenkrantz, D.J., Hunt, H.B.: Many birds with one stone: Multi-objective approximation algorithms. In: STOC, pp. 438–447 (1993)
24. Ravi, R., Singh, M.: Delegate and Conquer: An LP-based approximation algorithm for Minimum Degree MSTs. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 169–180. Springer, Heidelberg (2006)
25. Schrijver, A.: Combinatorial Optimization. Springer, Heidelberg (2003)
26. Singh, M.: Personal Communication (2008)
27. Singh, M., Lau, L.C.: Approximating minimum bounded degree spanning trees to within one of optimal. In: STOC, pp. 661–670 (2007)