On Generalizations of Network Design Problems with Degree Bounds

Nikhil Bansal[1], Rohit Khandekar[1], Jochen Könemann[2], Viswanath Nagarajan[3], and Britta Peis[4]

[1] IBM T.J. Watson Research Center. {nikhil,rohitk}@us.ibm.com
[2] University of Waterloo. jochen@uwaterloo.ca
[3] Tepper School of Business, CMU. viswa@cmu.edu
[4] Technische Universität Berlin. peis@math.tu-berlin.de

Abstract. The problem of designing efficient networks with degree-bound constraints has received a lot of attention recently. In this paper, we study several generalizations of this fundamental problem. Our generalizations are of the following two types:

- Generalize constraints on vertex-degree to *arbitrary subsets of edges*.
- Generalize the underlying network design problem to other combinatorial optimization problems like *polymatroid intersection* and *lattice polyhedra*.

We present several algorithmic results and lower bounds for these problems. At a high level, our algorithms are based on the iterative rounding/relaxation technique introduced in the context of degree bounded network design by Lau et al. [LNSS07] and Singh-Lau [SL07]. However many new ideas are required to apply this technique to the problems we consider. Our main results are:

- We consider the *minimum crossing spanning tree problem* [B+04] in the case that the 'degree constraints' have a *laminar structure* (this generalizes the well-known bounded degree MST [SL07]). We provide a $(1, b + O(\log n))$ bicriteria approximation for this problem, that improves over earlier results [B+04,BKN08].
- We introduce the *minimum crossing polymatroid intersection* problem, and give a $(2, 2b + \Delta - 1)$ bicriteria approximation (where Δ is the maximum number of degree-constraints that an element is part of). In the special case of bounded-degree arborescence (here $\Delta = 1$), this improves the previously best known $(2, 2b + 2)$ bound [LNSS07] to $(2, 2b)$.
- We also introduce the *minimum crossing lattice polyhedra* problem, and obtain a $(1, b + 2\Delta - 1)$ bicriteria approximation under certain condition. This result provides a unified framework and common generalization of various problems studied previously, such as degree bounded matroids [KLS08].

1 Introduction

Recently there has been a substantial progress in algorithms for network design problems with additional degree-bound constraints. These problems arise naturally in various contexts, the degree bound may correspond to limitations in outgoing bandwidth from a node, limitations in processing power, or even limitations of budget for the outgoing edges. The most widely studied problem in this line of research is the minimum cost degree bounded spanning tree problem. Here given a weighted undirected graph, and degree bounds for vertices, the goal is to find the minimum cost spanning tree subject to these degree bounds. Even in the absence of weights, this problem is NP-Hard, since finding a spanning tree with maximum degree two is equivalent to finding a Hamiltonian Path. A variety of techniques has been used for this problem [R+93,KR02,KR05,C+05,C+06,RS06,G06] which culminated in a recent breakthrough by Singh and Lau [SL07] who gave the best possible algorithm, that achieves the optimum cost, and an additive +1 violation in the degrees.

Subsequently, these results and techniques have been applied to different and more general settings such as matroids, arborescences, directed network design problems with intersecting and crossing super-modular connectivity requirements, and survivable network design [LNSS07,LS08,BKN08,KLS08]. Another direction has been to look at more general bounds than simply degree bounds, e.g., [B+04,BKN08,KLS08]. In this paper, we study further extensions and generalizations of these problems. In addition to several results, we also introduce and investigate the *degree bounded lattice polyhedron problem*. This problem forms a common generalization of several degree bounded optimization problems studied recently. We now formally describe various problems we consider, our results and techniques and how they connect to the previous work.

2 Our results, techniques and previous work

2.1 Minimum Crossing Spanning Tree

The algorithm of Singh and Lau [SL07] for the degree-bounded minimum spanning tree problem is based on an iterative rounding approach of Jain [J01] based on a natural linear programming relaxation. They show that in each iteration either some variable is set to 0 or 1, or else the degree constraint on some vertex can be dropped. A natural question is whether one can generalize this approach to bounds on arbitrary subsets of edges, instead of just degree bounds. This has been studied in the literature as the *Minimum Crossing Spanning Tree Problem (MCST)*. In this problem, we are given subsets of edges $E_1, \ldots, E_m \subseteq E$ and degree bounds $b_1, \ldots, b_m$, and the goal is to find a minimum cost spanning tree that contains at most b_i edges from E_i.

There are two previous results on the MCST problem. The first result, due to Bilo et al. [B+04], gives a multiplicative guarantee on both cost and degree violation: the algorithm finds a spanning tree where the degree is $O(\log n)b_i + O(\log m)$ and the cost violation is a multiplicative $O(\log n)$. This algorithm is based on randomized rounding. The second result [BKN08] gives an optimal cost guarantee and an additive guarantee on degree: if each edge lies in at most Δ sets $\{E_i\}_{i=1}^m$, then there is an algorithm that finds a spanning tree of optimum cost and degree at most $b_i + \Delta - 1$. This algorithm uses the iterative rounding/relaxation approach of Singh and Lau [SL07]. Note that these degree guarantees are incomparable: the first result is better for large Δ, whereas the second does better when Δ is small. For example, when $\Delta = \Theta(n)$ there is either an additive $O(n)$ or a multiplicative $O(\log n)$ bound for the degree.

We consider a special case of MCST when the degree-bounds have a 'laminar structure', and improve over the above bounds. Our motivation is to understand how far one can take the iterative rounding approach and provide small *additive* degree constraint violations with respect to the MCST problem. In the *laminar MCST problem*, we are given graph $G = (V, E)$ with edge-costs $c : E \to \mathbb{R}_+$, and degree bounds represented by a laminar family $\mathcal{C}$ on V along with a bound $b(S)$ for each $S \in \mathcal{C}$. The goal is to compute a minimum cost spanning tree in G that contains at most $b(S)$ edges from $\delta(S)$ for each $S \in \mathcal{C}$; here $\delta(S) := \{(u, v) \in E \mid u \in S,\ v \notin S\}$ is the set of edges *crossing* S. We refer to edge-subsets of the form $\delta(S)$ for some $S \subseteq V$ as *vertex-cuts*. We obtain the following result for laminar MCST in Section 3, that improves over both the previously known bounds.

Theorem 1 *There is a polynomial time algorithm for laminar MCST that computes a spanning tree of cost at most the optimum, and that contains at most* $b(S) + O(\log n)$ *edges from* $\delta(S)$ *for all* $S \in \mathcal{C}$.

This algorithm is again based on iterative rounding, and it has two new main ideas. First, we modify the iterative rounding procedure of Singh and Lau, to drop a constant fraction of constraints in each iteration. This is crucial as it can be shown that dropping one constraint at a time as in Singh and Lau can indeed lead to a degree violation of $\Omega(\Delta)$. Second, the algorithm does not just drop degree constraints, but in some iterations it also generates new degree constraints, by merging existing degree constraints.

Degree-bounded Matroids. A natural generalization of the MCST problem is the *minimum crossing matroid basis* problem [KLS08]. Here we are given a matroid on groundset E with rank function $r : 2^E \to \mathbb{N}$, cost function $c : E \to \mathbb{R}$ and degree bounds specified by subsets $\{E_i \subseteq E\}_{i=1}^m$ and respective bounds $\{b_i\}_{i=1}^m$. The goal is to find a min-cost basis B of the matroid satisfying $|B \cap E_i| \le b_i$ for all $i \in [m]$. Once again we denote by Δ, the maximum number of sets $\{E_i\}_{i=1}^m$ that any element of E lies in. A result of [BKN08,KLS08] shows that iterated rounding can be used to finds a basis of optimal cost that violates degree bounds by an at most and additive $\Delta - 1$ term. Our second result extends the guarantee of [B+04] to the crossing matroid problem.

Theorem 2 *There is a polynomial time algorithm for the minimum crossing matroid basis problem, that computes a basis of cost at most* $O(\log k)$ *times the optimum and with at most* $O(\log k) b_i + O(\log m)$ *elements from* E_i *for each* $i \in [m]$*. Here* m *is the number of degree constraints and* k *is the rank of the underlying matroid.*

This algorithm is based on randomly rounding an optimal LP solution. Although the algorithm in Theorem 2 is a natural extension of [B+04], the analysis is not completely straightforward. The algorithm of Bilo et al. [B+04] performs $O(\log n)$ rounds of the following: sample each edge independently according to the value produced by the LP solution. The key argument here is a result of Alon [A95] showing that w.h.p. the chosen edge-set (from the above procedure) contains a spanning tree: this proof relies on the graph structure and the notion of connected components. However it is not clear how to apply this argument to matroids, since there is no equivalent of a connected component in general matroids. Instead, we obtain the desired result by using a theorem of Polesskii [Pol90] (also proved in [Kar98]) that states: if a matroid of rank k contains $2L \cdot \ln k$ disjoint bases, then picking each element independently with probability $\frac{1}{L}$, results in a set containing a basis w.h.p. Details of Theorem 2 are given in Appendix B.

Hardness of approximation. Our next result shows that the crossing spanning tree problem is strictly harder than the bounded degree minimum spanning tree problem.

Theorem 3 *Unless $\mathcal{NP}$ has quasi-polynomial time algorithms, the minimum crossing matroid basis problem admits no $O(\log^\alpha m)$ additive approximation for some constant $\alpha > 0$. This holds even when there are no costs. Moreover, the minimum crossing spanning tree problem does not admit a $(1, b + O(\log^\alpha m))$-bicriteria approximation.*

To show the hardness result in Theorem 3, we give a reduction from the Label Cover Problem. The reduction proceeds in two steps. First, we show the hardness for a *uniform* matroid instance, without costs. Then, we show how to use this to reduce to an MCST problem with costs, such that any minimum spanning tree with optimum cost violates the degree bounds. The details are given in Section A.

We note that there is still a large gap between the positive and the negative results for MCST. There is also an additive $\Omega(\sqrt{\Delta})$ gap for the standard LP-based approaches, using discrepancy arguments[5]. While this integrality gap is substantially better than our hardness result, given the lack of any reasonable hardness results on discrepancy type problems, it is not clear how this could improve the hardness result for MCST.

2.2 Minimum Crossing Arborescence and Polymatroid Intersection

The degree bounded spanning tree problem has also been studied on directed graphs [KKRR04,LNSS07,BKN08]. Here we are given a weighted directed graph $G = (V, E)$ with root $r \in V$ and outdegree bounds b_v on the vertices $v \in V$. The *degree bounded min-cost arborescence problem* is to find a minimum cost arborescence rooted at r subject to the degree bounds. The results for arborescences are rather different from those for spanning trees. Bansal et al. [BKN08] designed an algorithm that for any $0 \le \epsilon \le 1/2$, produces a $(b_v/(1-\epsilon)+4, 1/\epsilon)$ bi-criteria guarantee. In fact this guarantee holds more generally for directed network design with 'intersecting supermodular requirements'. It turns out that this guarantee is best one can hope for via the natural LP relaxation, even for arborescences, since there is a similar integrality gap for every $0 \le \epsilon \le 1/2$. In particular, any approximation better than multiplicative factor 2 in the degree bounds causes a factor of at least 2 in the costs. If we do not care about costs, we can set $\epsilon = 0$ and obtain only an additive degree violation; in fact, [BKN08] improved this guarantee to plus 2.

Now, suppose we consider bounds on general edge sets. Given that additive guarantees exist for both crossing spanning trees and unweighted arborescences (with outdegree bounds), a natural question is whether results analogous to spanning trees or matroids also hold for unweighted arborescences. In particular, suppose we consider the unweighted arborescence problem with bounds $\{b_i\}_{i=1}^m$ on sets $\{E_i\}_{i=1}^m$, where the set system has $\Delta := \max_{e \in E} |\{i \in [m] : e \in E_i\}| = O(1)$, or even if $\Delta = 1$ (i.e., sets E_i are disjoint). Is there an additive degree violation guarantee in this case? Somewhat surprisingly, we show that for the natural LP relaxation, the answer is negative in a rather strong sense:

[5] This was pointed out to us by Mohit Singh.

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

On the positive side we show a tight upper bound matching the lower bound above, for the much more general polymatroid intersection problem.

Definition 1 (Minimum crossing polymatroid 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* minimum crossing polymatroid intersection problem *is:*

$$
\begin{aligned}
&\min c^T x \\
&\quad x(S) \geq \max\{r_1(S), r_2(S)\} && \forall S \subseteq E \\
&\quad x(E_i) \leq b_i && \forall i \in [m] \\
&\quad x_e \in \{0,1\} && \forall e \in E.
\end{aligned}
$$

Recall that the arborescence problem is an intersection of a partition matroid and a graphic matroid, and hence it is a special case of the matroid intersection problem. The following theorem captures our main result for this problem.

Theorem 5 *Any optimal basic solution x^* of the linear relaxation of the minimum crossing polymatroid 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*

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

We note that this result is the best one can hope given the lower bounds above. First, the integrality gap instance mentioned previously implies that the multiplicative factor in the degree cannot be improved beyond 2. Second, the [BKN08] lower bound for arborescences implies that one cannot hope to obtain a ratio better than 2 in costs (without violating factor strictly greater than 2 in degrees). For the special case of degree bounded arborescence, Theorem 5 improves the previously best known bicriteria bound of $(2, 2b+2)$ [LNSS07] to $(2, 2b)$.

The algorithm for this theorem uses iterative rounding, and its proof is based on a 'fractional token' counting argument similar to the one used in proving the $\Delta - 1$ additive guarantee for the MCST problem [BKN08]. Proofs of Theorems 4 and 5 are in Appendix C.

2.3 Minimum Crossing Lattice Polyhedron Problem

We generalize the minimum crossing polymatroid intersection problem even further to *minimum crossing lattice polyhedra*. Lattice polyhedra form a common framework for several discrete optimization problems such as polymatroids, intersection of two polymatroids, shortest paths, max flow/min cut in s,t-planar graphs, supermodular systems, etc. (see Appendix D). Lattice polyhedra were first investigated by Hoffman and

Schwartz [HS78] and the natural LP relaxation was shown to be totally dual integral. Even though greedy-type algorithms are known for all the examples mentioned above, 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 (see [Fra99],[FP08]).

Before formally defining the crossing lattice polyhedra problem, we need to introduce some terminology. Let $(\mathcal{F}, \leq)$ be a partially ordered set with $\mathcal{F} \neq \emptyset$. We consider a *lattice* $(\mathcal{F}, \leq)$, where there are two commutative binary operations, *meet* $\wedge$ and *join* $\vee$, that are defined on *all* pairs $A, B \in \mathcal{F}$, such that:

$$A \wedge B \ \leq\ A, B \ \leq\ A \vee B$$

Note that our definition is more general than the usual definition of a lattice, since the join $A \vee B$ is not required to be the least common upper bound of A and B. A function $r : \mathcal{F} \to \mathbb{Z}_+$ is said to be *supermodular* on $(\mathcal{F}, \leq, \wedge, \vee)$ iff:

$$r(A) + r(B) \leq r(A \wedge B) + r(A \vee B), \quad \text{for all } A, B \in \mathcal{F}$$

Given a supermodular function $r : \mathcal{F} \to \mathbb{Z}_+$, a ground set E, a cost function $c : E \to \mathbb{R}_+$, and a set-valued function $\rho : \mathcal{F} \to 2^E$ satisfying:

1. **Consecutive property:** If $A \leq B \leq C$ then $\rho(A) \cap \rho(C) \subseteq \rho(B)$,
2. **Submodularity:** For all $A, B \in \mathcal{F}$, $\rho(A \vee B) \cup \rho(A \wedge B) \subseteq \rho(A) \cup \rho(B)$,

the *lattice polyhedron problem* is defined as the following integer program:

$$\min\{c^T \cdot x \mid \sum_{e \in \rho(S)} x_e \geq r(S),\ \forall S \in \mathcal{F};\ \ x \in \{0,1\}^E\}$$

Definition 2 (Minimum crossing lattice polyhedron). *Given a lattice polyhedron specified by* $(\mathcal{F}, \leq, \wedge, \vee, E, \rho, r, c)$*, and a family* $\{E_i\}_{i=1}^m$ *of subsets of* E *with bounds* $\{b_i\}_{i=1}^m$*, the* minimum crossing lattice polyhedron *problem is:*

$$\begin{array}{rll} \min c^T \cdot x & & \\ x(\rho(S)) \geq r(S), & \forall S \in \mathcal{F} & \qquad \textit{Rank constraints} \\ x(E_i) \leq b_i, & \forall i \in I & \qquad \textit{Degree constraints} \\ x \in \{0,1\}^E & & \end{array}$$

We prove the following result for this problem.

Theorem 6 *Consider any instance of minimum crossing lattice polyhedra (Definition 2) that satisfies the following assumption:*

$$(*) \quad S < T \quad \Longrightarrow \quad |\rho(S)| < |\rho(T)|, \quad \textit{for all } S, T \in \mathcal{F}$$

Then there is an algorithm that computes a solution of cost at most the optimal, where all rank constraints are satisfied, and each degree bound is violated by at most an additive $2\Delta - 1$*. Here* $\Delta := \max_{e \in E} |\{i \in [m] : e \in E_i\}|$.

This theorem also holds in the presence of both lower and upper degree-bounds. We note that assumption $(*)$ is satisfied for matroids, so Theorem 6 matches the previously best-known bound [KLS08] for degree bounded matroids (with both upper/lower bounds). We also note that this theorem is only applicable when the rank constraints are separable in polynomial time; this corresponds to the problem of minimizing a submodular function on ground-set E over the subsets $\{\rho(S) \mid S \in \mathcal{F}\} \subseteq 2^E$. This is indeed possible in all aforementioned examples of lattice polyhedra.

Observe that property $(*)$ is valid in case of inclusion-wise ordering, i.e., if

$$S \leq T \quad \Longleftrightarrow \quad \rho(S) \subseteq \rho(T) \quad \forall S, T \in \mathcal{F}.$$

In this special case, we can improve the result of Theorem 6.

Theorem 7 *If the underlying lattice of the minimum crossing lattice polyhedron problem is ordered by inclusion, then there is an algorithm that computes a solution of cost at most the optimal, where all rank constraints are satisfied, and each degree bound is violated by at most an additive* $\Delta - 1$.

Theorems 6 and 7 are similar to the corresponding proofs for MCST [BKN08] and degree-bounded matroid [KLS08], however the arguments need to be carefully adapted in the more general setting of lattice polyhedra. Proofs appear in Appendix D.

3 Crossing Spanning Tree with Laminar degree bounds

We consider the *crossing spanning tree problem* with bounds on vertex-cuts that form a laminar family. In this problem, we are given an undirected graph $G = (V, E_o)$ on n vertices, non-negative edge-costs c_e for $e \in E_o$, and a family $\mathcal{D}$ of subsets of V with "degree-bounds" $b(S)$ for each $S \in \mathcal{D}$. We assume that $\mathcal{D}$ is a *laminar family* of vertex-sets: i.e. $S \subseteq T$ or $T \subseteq S$ or $S \cap T = \emptyset$ holds for any $S, T \in \mathcal{D}$. The problem involves computing a minimum-cost spanning tree T of G that contains at most $b(S)$ edges from $\delta(S)$ for each $S \in \mathcal{D}$. This problem reduces to the usual degree-bounded MST when $\mathcal{D} = \{\{v\} \mid v \in V\}$. In this section we prove Theorem 1.

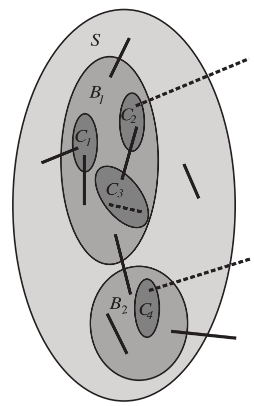
The algorithm uses iterative rounding based on an LP relaxation. The algorithm modifies the laminar family of degree bounds during its execution. A generic iteration starts with a subset F of edges already picked in the solution, a subset E of *undecided* edges, i.e., the edges not yet picked in or dropped 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.

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

For any vertex-subset $U \subseteq V$ and edge-set H, we let $H(U) := \{(u, v) \in H \mid u, v \in U\}$ denote the edges induced on U; and $\delta_H(S) := \{(u, v) \in H \mid u \in S,\ v \notin S\}$ the set of edges crossing S. 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 be fractional-valued). Let $\alpha := 24$.

Definition 3. *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 $\mathrm{local}(S)$ denote the set of local edges for S. A node $S \in \mathcal{L}$ is said to be* good *if $|\mathrm{local}(S)| \leq \alpha$.*

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



The algorithm is initialized with $F \leftarrow \emptyset$, $E \leftarrow E_o$, $\mathcal{L} \leftarrow \mathcal{D}$, the original degree bounds on $\mathcal{D}$, and an arbitrary linear ordering on the children of each node in $\mathcal{D}$. In a generic iteration $(F, E, \mathcal{L}, b)$, the algorithm performs one of the following steps:

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. 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 $b(M_i) = b(N_{2i-1}) + b(N_{2i})$.

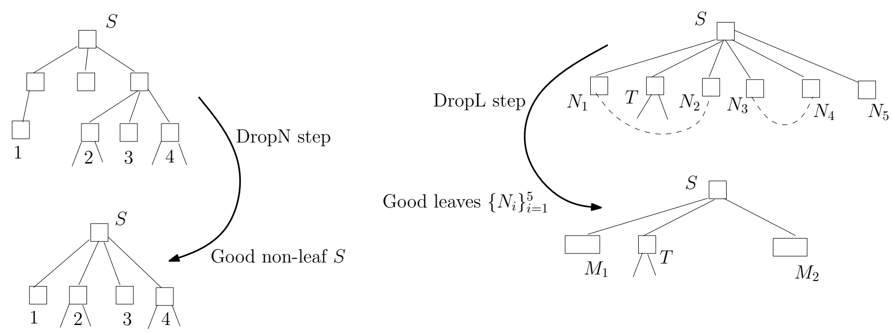


Fig. 1. Examples of the degree constraint modifications DropN and DropL.

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. [LNSS07]) can be used to show that the LP (1) is feasible at each each iteration and $c(F) + LP_{cur} \leq LP_o$ where LP_o is the original LP value, LP_{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 LP_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.*

Proof: We crucially use the fact that x is an extreme point solution of (1). This implies that x is uniquely defined by satisfying a linearly independent and laminar subset $\mathcal{S}$ of the spanning tree constraints at equality together with a sub-family $\mathcal{L}' \subseteq \mathcal{L}$ of degree-constraints, such that $|E| = |\mathcal{S}| + |\mathcal{L}'|$. If the first two steps do not apply, then $0 < x_e < 1$ for all $e \in E$. A counting argument (see, e.g., [SL07]) shows that there are at least 2 edges induced on each $S \in \mathcal{L}'$ that are not induced on any of its children; so $2|\mathcal{S}| \leq |E|$. 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 $\sum_{S \in \mathcal{L}} |\text{local}(S)| \leq 6|E|$. Combining these facts, we conclude that $\sum_{S \in \mathcal{L}} |\text{local}(S)| \leq 12|\mathcal{L}|$. Thus at least $|\mathcal{L}|/2$ sets $S \in \mathcal{L}$ must have $|\text{local}(S)| \leq \alpha = 24$, 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 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 $2n-1$,

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

3.1 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, define:

- $\mathcal{L}_t$ 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*), we have* $\mathsf{Inc}(\mathcal{B},t) \le \mathsf{Bnd}(\mathcal{B},t) + 4\alpha\cdot(T-t)$.

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. $\mathsf{Inc}(\{N\},T) \le b(N)$ for all $N\in\mathcal{L}_T$. Hence for any $\mathcal{B}\subseteq\mathcal{L}$, $\mathsf{Inc}(\mathcal{B},T) \le \sum_{N\in\mathcal{B}} \mathsf{Inc}(\{N\},T) \le \sum_{N\in\mathcal{B}} b(N) = \mathsf{Bnd}(\mathcal{B},T)$.

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.* The inductive hypothesis implies $\mathsf{Inc}(\mathcal{B},t+1) \le \mathsf{Bnd}(\mathcal{B},t+1)+4\alpha\cdot(T-t-1)$. Since the only iterations between round t and round $t+1$ involve edge-fixing, we have $\mathsf{Inc}(\mathcal{B},t) \le \mathsf{Bnd}(\mathcal{B},t) - \mathsf{Bnd}(\mathcal{B},t+1) + \mathsf{Inc}(\mathcal{B},t+1) \le \mathsf{Bnd}(\mathcal{B},t) + 4\alpha\cdot(T-t-1)$.

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 $\mathsf{Inc}(\mathcal{C},t+1) \le \mathsf{Bnd}(\mathcal{C},t+1) + 4\alpha\cdot(T-t-1)$. 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 $t+1$, $|\mathsf{Bnd}'(\mathcal{B}, t+1) - \mathsf{Bnd}(\mathcal{C}, t+1)| = |\sum_{N\in\mathcal{B}} b(N) - \sum_{M\in\mathcal{C}} b(M)| = |\sum_{N\in\mathcal{B}} x(\delta(N)) - \sum_{M\in\mathcal{C}} x(\delta(M))| \le 2\alpha$. 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 $\mathrm{local}(S)$. 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:

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

The first inequality 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 $\mathsf{Bnd}(\mathcal{C}, t+1) \le \mathsf{Bnd}'(\mathcal{B}, t+1) + 2\alpha$ (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 $1 \le \pi(1) < \pi(2) < \cdots < \pi(k) \le p$ correspond to good leaf-nodes in $\mathcal{N}$. Then for each $1 \le i \le \lfloor k/2 \rfloor$, nodes $N_{\pi(2i-1)}$ and $N_{\pi(2i)}$ are merged in this round. Let $\{\pi(i) \mid e \le i \le f\}$ (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 $\mathsf{Bnd}(\mathcal{C}, t+1) \le \mathsf{Bnd}'(\mathcal{B}, t+1) + 2\alpha$. The term 2α is present due to the two extra good leaf-nodes described above.

Claim 1 *We have $|\delta_H(\cup_{N\in\mathcal{B}} N) \setminus \delta_H(\cup_{M\in\mathcal{C}} M)| \le 2\alpha$.*

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

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

The first inequality follows from simple counting; the second uses Claim 1, 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. ■

References

[A95] N. Alon, *A note on network reliability*, Discrete Probability and Algorithms, D. Aldous et al. Eds., IMA volumes in mathematics and its applications, 72, 1995, 11-14.

[A+93] S. Arora, L. Babai, J. Stern, and Z. Sweedyk, *The hardness of approximate optima in lattices, codes, and systems of linear equations*, IEEE Symposium on Foundations of Computer Science, 1993.

[BKN08] N.Bansal, R. Khandekar and V. Nagarajan, *Additive guarantees for degree bounded network design*, Proceedings of the 40th annual ACM symposium on Theory of computing, Victoria, Canada, 2008, 769-778.

[B+04] V. Bilo, V. Goyal, R. Ravi and, M. Singh, *On the crossing spanning tree problem*, in APPROX 2004.

[C+05] K. Chaudhuri, S. Rao, S. Riesenfeld, and K. Talwar, *What would Edmonds do? Augmenting paths and witnesses for degree-bounded MSTs*, In Proceedings of 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, pp. 26-39, 2005.

[C+06] K. Chaudhuri, S. Rao, S. Riesenfeld, and K. Talwar. *Push relabel and an improved approximation algorithm for the bounded-degree mst problem*, In Proceedings of 33th International Colloquium on Automata, Languages and Programming, 2006.

[Edm70] J. Edmonds, *Submodular functions, matroids and certain polyhedra*, Proc. Int. Conf. on Combinatorics (Calgary), 1970, pp. 69–87.

[EJ85] P.H. Edelman and R.E. Jamison, *The theory of convex geometries*, Geometriae Dedicata **19** (1985), 247–270.

[FP08] U.Faigle and B. Peis, *Two-phase greedy algorithms for some classes of combinatorial linear programs*, Proc. of the 19th annual ACM-SIAM symposium on discrete algorithms, San Francisco, 2008, 161-166.

[FKP08] U. Faigle, W. Kern and B. Peis, *A ranking model for cooperative games, convexity and the greedy algorithm*, working paper.
[Fra99] A. Frank, *Increasing the rooted connectivity of a digraph by one*, Math. Programming **84** (1999), 565–576.
[Fuj05] S. Fujishige, *Submodular Functions and Optimization*, Annals of Discrete Mathematics, vol. 58, Elsevier Science Publishers, North-Holland, 2nd ed., 2005.
[G06] M.X. Goemans, *Minimum Bounded-Degree Spanning Trees*, Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, 273282, 2006.
[HS78] A. Hoffman and D.E. Schwartz, *On lattice polyhedra*, Proceedings of Fifth Hungarian Combinatorial Coll. (A. Hajnal and V.T. Sos, eds.), North-Holland, Amsterdam, 1978, pp. 593–598.
[J01] K. Jain, *A factor 2 approximation algorithm for the generalized Steiner network problem*, Combinatorica, 2001, 39-61.
[Kar98] D. R. Karger, *Random sampling and greedy sparsification for matroid optimization problems*, Mathematical Programming, 82, 1998, pp. 99–116.
[KLS08] T. Király, L.C. Lau and M. Singh, *Degree bounded matroids and submodular flows*, in IPCO 2008.
[KKRR04] P. N. Klein, R. Krishnan, B. Raghavachari and R. Ravi, *Approximation algorithms for finding low degree subgraphs*, Networks, 44(3), 2004, 203-215.
[KR02] J. Könemann and R. Ravi, *A matter of degree: Improved approximation algorithms for degree bounded minimum spanning trees,* SIAM J. on Computing, 31:1783-1793, 2002.
[KR05] J. Könemann and R. Ravi, *Primal-Dual meets local search: approximating MSTs with nonuniform degree bounds,* SIAM J. on Computing, 34(3):763-773, 2005.
[LNSS07] L.C. Lau, J. Naor, M. R. Salavatipour and M. Singh, *Survivable network design with degree or order constraints (full version),* in STOC 2007.
[LS08] L.C. Lau and M. Singh, *Additive Approximation for Bounded Degree Survivable Network Design*, in STOC 2008.
[Pol90] V. P. Polesskii, *Bounds on the connectedness probability of a random graph*, Problems of Information Transmission, 27, 1990, pp. 86–97.
[R+93] R. Ravi, M.V. Marathe, S.S. Ravi, D.J. Rosenkrants, and H.B. Hunt, *Many birds with one stone: Multi-objective approximation algorithms*, in Proceedings of the 25th ACM-Symposium on Theory of Computing, pp 438-447, 1993.
[RS06] R. Ravi and M. Singh, *Delegate and Conquer: An LP-based approximation algorithm for Minimum Degree MSTs.* In Proceedings of 33rd International Colloquium on Automata, Languages and Programming, 2006.
[Sch03] A. Schrijver, *Combinatorial Optimization*, Springer, 2003.
[SL07] M. Singh and L.C. Lau, *Approximating minimum bounded degree spanning trees to within one of optimal*, in STOC 2007.

A Hardness Result for Minimum Crossing Spanning Tree

In this section we will show that unless $\mathcal{NP}$ has quasi-polynomial time algorithms, any solution with optimum cost for minimum crossing spanning tree, must violate the degree by at least an additive term of $O(\log^c m)$ for some universal constant c. Before we prove this result, we show 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 $|I \cap E_i| \le b_i$ for all $i \in [m]$. Later we show how to adapt this hardness result to special case of the spanning tree matroid.

Theorem 8 *Unless $\mathcal{NP}$ has quasi-polynomial time algorithms, the minimum crossing matroid basis problem admits no $O(\log^c m)$ additive approximation for some fixed constant $c > 0$. This holds even if we do not care about the costs.*

Proof: We reduce from the label cover problem [A+93]. 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.

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

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. Note that the value can be 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 $< t/\gamma^k$.
- $|G| = N^{O(k)}$, $q = 2^k$, and the reduction runs in time $N^{O(k)}$.

We construct 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). 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 requiring *at most one* element to be chosen from C. Note that the number of degree bounds $m \leq n \cdot (nq)^q \leq n^{2q}$.

Observe that if the original SAT instance is satisfiable, then the matroid $\mathcal{M}$ contains a basis obeying all the degree bounds: namely the t edges covered in the optimal solution to the label cover instance. This is because if we consider any U_i, then all the edges having a vertex in U_i as their endpoint, have the same endpoint. Thus, by the way the collection C is defined, at most one such edge can lie in it.

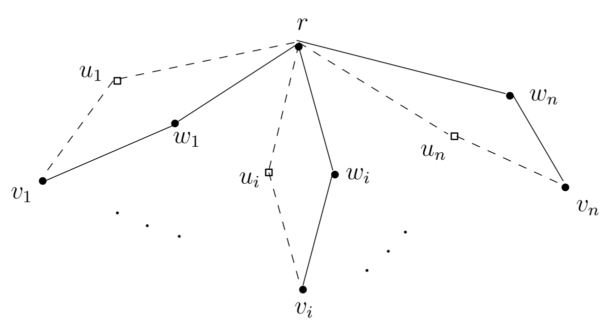
On the other hand, we will show that if the SAT instance is unsatisfiable, then every basis in $\mathcal{M}$ picks at least $\rho = \gamma^{k/2}$ edges from some degree-constrained set of edges. Suppose (for a contradiction) that there is a basis (i.e. set of t edges $B \subseteq E$) such that $|B \cap C| < \rho$ for each degree constraint C. This means that each part $\{U_i\}_{i=1}^n$ contains fewer than ρ vertices that are incident to edges B. For each part $i \in [n]$, let $W_i \subseteq U_i$ denote the vertices incident to edges of B; note that $|W_i| < \rho$. Consider the label cover solution obtained as follows. For each $i \in [n]$, choose one vertex from W_i uniformly at random. Clearly, the expected number of edges in the resulting induced subgraph is at least $t/\rho^2 = t/\gamma^k$, which contradicts that the value of label cover instance is strictly less than t/γ^k.

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 $\rho = \gamma^{k/2} = \log^b N$, 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 $O(\log^c m)$ *additive approximation*, unless $\mathcal{NP}$ has quasi-polynomial time algorithms. ■

We now consider the special case of minimum-cost crossing spanning tree: given an edge-weighted graph with degree-bounds on m edge-sets, find a minimum cost spanning tree satisfying all degree bounds. Using Theorem 3, we prove the following.

Corollary 1 *Unless $\mathcal{NP}$ has quasi-polynomial time algorithms, there is no $(b+\log^c m, 1)$ approximation for the minimum-cost crossing spanning tree problem, for some fixed constant $c > 0$.*

Proof: Recall that Theorem 3 actually shows the hardness of approximating the bounded-degree *uniform* matroid problem. We show how the bases of a uniform matroid can be represented in a suitable instance of the min-cost crossing spanning tree problem. Let the uniform matroid from Theorem 3 consist of e elements and have rank $t \le e$. 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 $\langle v_i, w_i, r\rangle$. The edges $\{(r, u_i) \mid i \in [e]\} \cup \{(v_i, u_i) \mid i \in [e]\}$ have cost zero, and edges $\{(r, w_i) \mid i \in [e]\} \cup \{(v_i, w_i) \mid i \in [e]\}$ have cost 1. 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 $\delta(\{u_i \mid i \in C\})$.



The dashed edges have cost 0, solid edges have cost 1.

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

Observe that for each $i \in [e]$, any spanning tree must choose at least three edges among $\{(r, u_i), (u_i, v_i), (r, w_i), (w_i, v_i)\}$, in fact any three edges suffice. Thus for any spanning tree of cost $2n - t$, there must be exactly t indices i for which both edges

(r, u_i) and (u_i, v_i) lie in the spanning tree. Thus we can associate a basis in the uniform matroid with every spanning tree of cost $2n - t$.

In Theorem 3, for the bounded-degree uniform matroid problem, it is hard to distinguish the following two cases: (yes-case) there is a basis b^* satisfying all degree bounds, and (NO-case) every basis violates some degree bound by an additive $\rho = \Omega(\log^c m)$ term. In the yes-case, whenever i lies in the basis b^*, we choose the edges $\{(r, u_i), (u_i, v_i), r(w_i)\}$. This solution has cost $2n - t$, and satisfies all the degree bounds. On the other hand, in the no-case any spanning tree with cost $2n - t$, must violate some degree bound by at least an additive ρ. This implies that there is no $(b + \rho, 1)$ approximation for minimum-cost crossing spanning tree: given some instance $\mathcal{I}$ (which is either a YES-instance or NO-instance) of the bounded degree matroid problem, we reduce $\mathcal{I}$ to a crossing spanning tree instance as above and apply the $(b + \rho, 1)$ algorithm. If we obtain a tree of cost at most $2n - t$ then $\mathcal{I}$ is a (YES-instance), otherwise it is a (NO-instance). ■

B An algorithm for minimum crossing matroid basis

In this section, we consider the minimum crossing matroid problem defined as follows. Given a matroid $\mathcal{M}$ having n elements, rank function $r : 2^{[n]} \to \mathbb{N}$ and cost $c : [n] \to \mathbb{R}$, and m *arbitrary* "degree" constraints $\{E_i, b_i\}_{i=1}^m$, find a minimum cost basis subject to the degree constraints. For the case of crossing spanning tree, Bilo et al. gave an $(O(\log n)b + O(\log m), O(\log n))$ approximation algorithm based on randomized rounding of the natural LP relaxation. We note that this result can be extended to the bounded-degree matroid problem. In particular, we show that

Theorem 9 *There is an $(O(\log k)b + O(\log m), O(\log k))$ bicriteria approximation algorithm for the bounded-degree matroid basis problem with m degree constraints on a matroid of rank k.*

The algorithm is very simple: We consider the following LP relaxation.

$$\min \sum_{e \in [n]} c_e \cdot x_e$$

$$x(S) \leq r(S) \qquad \forall S \subseteq [n]$$

$$x([n]) = r([n])$$

$$x(E_i) \leq b_i \qquad \forall i \in [m]$$

If x denotes an optimal solution to the above LP-relaxation, then the integer solution R consists of each element $e \in [n]$ chosen independently with probability $\min\{\rho \cdot x_e, 1\}$, where $\rho = 2\lceil \ln k \rceil$. We will show that w.h.p. R contains a basis and that all the degree violations are small. Using the Chernoff bound for each $i \in [m]$, $Pr[|R \cap E_i| > 2\rho b_i + 2\log m] \leq \frac{1}{m^2}$ (since the expected value of $|R \cap E_i|$ is at most $\rho \cdot b_i$). Thus with probability at least $1 - \frac{1}{m}$, for every degree bound $i \in [m]$, we have $|R \cap E_i| \leq 2\rho b_i + 2\log m$. Before showing that R contains a basis w.h.p., we state a relevant theorem of Polesskii [Pol90], which was also proved in Karger [Kar98].

Theorem 10 (Theorem 4.2, Karger [Kar98]) *Suppose a matroid $\mathcal{N}$ of rank k contains $2L \cdot \ln k$ disjoint bases. Then if each element of $\mathcal{N}$ is chosen independently with probability $\frac{1}{L}$, the resulting set contains a basis with probability at least $1 - O(\frac{1}{k})$.*

Claim 2 *The set R contains a basis of $\mathcal{M}$ with probability at least $1 - O(\frac{1}{k})$.*

Proof: This is a direct application of Theorem 10. Let L be some large integer so that $L \cdot x_e$ is integral for all $e \in [n]$ (recall that x is the optimal LP solution). Construct matroid $\mathcal{N}$ from $\mathcal{M}$ by keeping $2L\lceil \ln k \rceil \cdot x_e$ *copies* of each element $e \in [n]$; clearly the rank of $\mathcal{N}$ equals k (rank of $\mathcal{M}$). Since x is a fractional basis in $\mathcal{M}$, matroid $\mathcal{N}$ contains $P = 2L \cdot \lceil \ln k \rceil$ disjoint bases. This follows from the matroid base packing theorem, Corollary 42.1d, [Sch03], which states that a matroid on element set V with rank function r has ℓ disjoint bases if and only if $\ell(r(V) - r(S)) \leq |V \setminus S|$.

Consider picking each element in $\mathcal{N}$ independently with probability $\frac{1}{L}$, and let T' be the resulting set of elements. Let $T \subseteq [n]$ be the set of distinct element of $\mathcal{M}$ in T'; clearly T' contains a basis of $\mathcal{N}$ iff T contains a basis of $\mathcal{M}$. Theorem 10 implies that T contains a basis with probability at least $1 - O(\frac{1}{k})$. We now relate the random set T to the random set R. It is clear that each element $e \in [n]$ is chosen independently in T (as is the case in R). The probability $\overline{q}_e$ of *not* picking element $e \in [n]$ in T equals $(1 - \frac{1}{L})^{Px_e}$ which is at most $1 - \frac{P}{L}x_e$. Note that $\overline{q}_e \geq \max(0, 1 - \frac{P}{L}x_e)$. Now, $p_e = \min(1, \frac{P}{L}x_e)$ is the probability that e is picked in set R, and hence $\overline{p_e} = \max(0, 1 - \frac{P}{L}x_e)$ which is at most $\overline{q}_e$. Thus the random set R stochastically dominates T. Since basis containment is a monotone property, the probability that R contains a basis is larger than that for T, implying the claim. ■

Combining the high probability statements for degree-bound violation and Claim 2, we obtain Theorem 2.

C Minimum Crossing Arborescence and Polymatroid Intersection

Recall that there is an additive +2 approximation for the degree bounded arborescence problem without costs. In this section, we consider this problem when bounds on arbitrary edge sets are allowed. Surprisingly, we show that even if we add one extra ”non-degree” bound, the degree bounded arborescence problem without cost has a (multiplicative) integrality gap of 2. In particular, we prove Theorem 4:

Proof: [Theorem 4] We first define the graph. This graph is shown in Figure 3, and is similar to the one in [BKN08] (but has different parameters). Let k be an arbitrarily large integer, consider a k-ary arborescence rooted at root r, of depth $d > 2\ln(2/\epsilon)/\epsilon$. We call the edges of this arborescence solid edges. Consider the natural drawing of this tree, and label these leaves $1, \ldots, k^d$, from right to left. Next we define dashed edges as follows. There is one edge going from root r to leaf 1, and one edge from each leaf i to $i+1$ for $i = 1, \ldots, k^d - 1$. Finally, the dotted edges are defined as follows. For each internal node v (other than the root), there is an incoming dotted edge from the leftmost leaf root in the subtree rooted at v. This completes the description of the graph. The degree bounds are as follows. For each non-leaf vertex, there is an out-degree bound of

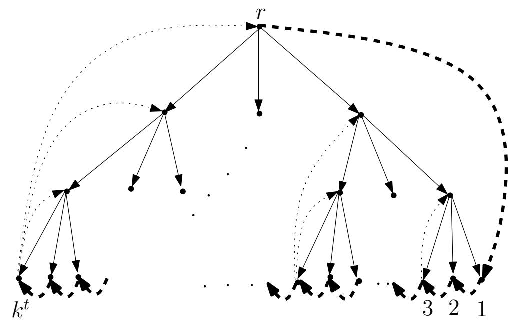


Fig. 3. The integrality gap instance. The set E_1 consists of all dashed edges.

$k/2$. In addition, we define the E_1 to be the set of all dashed edges and assign it a bound of $b_1 = k^d/2$. Note that $|E_1| = k^d$. It is easily verified that $\Delta = 1$.

Consider the LP solution which assigns $x_e = 0.5$ to every edge. It is easily verified that this is a valid arborescence solution (each vertex can be sent a unit of flow from the root by sending 0.5 unit of flow along the solid edges, and 0.5 unit along the dashed and dotted edges), and satisfies all the E_i bounds.

We now show that in any integral solution, the degree is violated by at factor of at least $2-\epsilon$. Let us assume that each internal vertex has an outdegree of at most $k(1-\epsilon/2)$, otherwise this is a violated vertex and we are done. It suffices to show that in this case, there must be at least $k^d(1-\epsilon/2)$ edges chosen from E_1 in a valid arborescence. This follows from the simple property (see [BKN08], Prop. 1, for a formal proof) that if a leaf i does not have path from root to itself using only solid edges, then the edge $(i-1, i)$ must be present in the arboresence. Now, if internal degree is at most $k(1-\epsilon/2)$, then the number of leaves with a path from root using only solid edges is at most $(1-\epsilon/2)^d k^d$ which, by our choice of d, is at most $\epsilon k^d/2$. Thus at least, $k^d(1-\epsilon/2)$ edges must be chosen from E_1 which proves the result. ■

Recall that several problems such as the minimum cost arboresence problem can be cast as a matroid intersection problem. While the degree bounded version of the minimum cost arborescence problem is well understood [BKN08], not much is known about its behavior with degree bounds on arbitrary subsets. We now consider the *minimum crossing polymatroid intersection problem* (see Defintion 1) and prove Theorem 5.

The algorithm 1 for minimum crossing polymatroid intersection is based on iteratively rounding the following natural LP relaxation.

$$
\begin{aligned}
&\min c^T x \\
&\quad x(S) \geq \max\{r_1(S), r_2(S)\} - |F \cap S| && \forall S \subseteq E \\
&\quad x(E_i) \leq b'_i && \forall i \in W \\
&\quad 0 \leq x_e \leq 1 && \forall e \in E.
\end{aligned}
$$

Above, E denotes the set of unfixed elements, F the set of chosen elements, $W \subseteq [m]$ the set of remaining degree bounds, and b'_i (for each $i \in W$) the residual degree-bound in the i^{th} constraint.

Algorithm 1 Algorithm for minimum crossing polymatroid intersection.

Intially, set $F = \emptyset$, $W = [m]$, $b'_i = b_i$, for all $i \in I$
while $E \neq \emptyset$ **do**
    Compute an optimal basic solution x^* of the LP;
    for all $e \in E$ with $x^*(e) = 0$ **do**
        $E \leftarrow E \setminus \{e\}$
    end for
    for all $e \in E$ with $x^*(e) \geq \frac{1}{2}$ **do**
        $F \leftarrow F \cup \{e\}$; $E \leftarrow E \setminus \{e\}$
        $b'_i \leftarrow b'_i - x^*(e)$, for all $i \in W$ with $e \in E_i$
    end for
    for all $i \in W$ with $|E_i| \leq \lceil 2b'_i \rceil + \Delta - 1$ **do**
        $W \leftarrow W \setminus \{i\}$
    end for
end while
Return the incidence vector x^F of F;

Note that this algorithm rounds variables of value $x^*(e) \geq \frac{1}{2}$ to 1, and hence we loose a factor of two in the cost and in the degree bounds. Theorem 5 follows as a consequence if we can show that in each iteration, either some variable can be rounded, or some constraint can be dropped. For this, we prove:

Lemma 1. *If $x^* \in \mathbb{R}^E$ is a basic optimal solution of $(LP2)$ with $0 < x^*(e) < \frac{1}{2}$ for all $e \in E$, then there exists at least one $i \in W$ such that*

$$|E_i| \leq \lceil 2b'_i \rceil + \Delta - 1$$

Proof: Since x^* is a basic feasible solution, there exist linearly independent tight sets $\mathcal{T}_1 \subseteq \{S \subseteq E \mid x^*(S) = r_1(S)\}$, $\mathcal{T}_2 \subseteq \{S \subseteq E \mid x^*(S) = r_2(S)\}$ and $\mathcal{B} \subseteq \{E_i \subseteq E \mid x^*(E_i) = b'_i\}$ such that

$$|E| = |\mathcal{T}_1| + |\mathcal{T}_2| + |\mathcal{B}|.$$

Since x^* is modular and r_1, r_2 are supermodular on the Boolean lattice $(2^E, \subseteq)$, it can be assumed (again, using uncrossing arguments) that each of $(\mathcal{T}_1, \subseteq)$ and $(\mathcal{T}_2, \subseteq)$ form a chain. We use the following claim from [BKN08] (which was originally stated for spanning trees, but immediately extends to any polymatroid).

Claim ([BKN08]). We have $|\mathcal{T}_1|, |\mathcal{T}_2| \leq \sum_{e \in E} x^*_e$. Additionally, $\mathcal{T}_j = x^*(E)$ (for $j \in \{1, 2\}$) only if $E \in \mathcal{T}_j$.

Suppose (for a contradiction) that for all $i \in W$, $|E_i| \geq \lceil 2b'_i \rceil + \Delta$. For each $i \in W$, define $\mathsf{Sp}_i := \sum_{e \in E_i} (1 - 2x^*_e) = |E_i| - 2x^*(E_i)$. Then we have $\mathsf{Sp}_i \geq |E_i| - 2b'_i \geq |E_i| - \lceil 2b'_i \rceil \geq \Delta$. Hence $\sum_{i \in W} \mathsf{Sp}_i \geq \Delta \cdot |W|$.

For each $e \in E$, let $r_e := |\{i \in W : e \in E_i\}| \leq \Delta$. Note also that $0 < 1 - 2x_e^* < 1$ for each $e \in E$. Now,

$$\sum_{i \in W} \mathsf{Sp}_i = \sum_{e \in E} r_e \cdot (1 - 2x_e^*) \leq \Delta \cdot \sum_{e \in E} (1 - 2x_e^*)$$
$$= \Delta \cdot (|E| - 2 \cdot x^*(E)) \leq \Delta \cdot (|E| - |\mathcal{T}_1| - |\mathcal{T}_2|)$$

Thus we have $\sum_{i \in W} \mathsf{Sp}_i \leq \Delta \cdot |\mathcal{B}| \leq \Delta \cdot |W|$ with equality only if $E \in \mathcal{T}_1 \cap \mathcal{T}_2$ (from Claim C), $r_e = \Delta$ for all $e \in E$, and $\mathcal{B} = W$.

We now claim that equality $\sum_{i \in W} \mathsf{Sp}_i = \Delta \cdot |W|$ is not possible. If this were the case, $\chi(E)$ is a constraint in each of $\mathcal{T}_1$ and $\mathcal{T}_2$; and $\sum_{i \in \mathcal{B}} \chi(E_i) = \sum_{i \in W} \chi(E_i) = \Delta \cdot \chi(E)$. However this contradicts the linear independence of constraints in $\mathcal{T}_1$ and $\mathcal{B}$. Thus it must be that $\sum_{i \in W} \mathsf{Sp}_i < \Delta \cdot |W|$, which contradicts the assumption that $|E_i| \geq \lceil 2b_i' \rceil + \Delta$ for all $i \in W$. ■

Proof: [Theorem 5] Lemma 1 implies that an improvement is possible in each iteration of Algorithm 1. Since we only round elements that the LP sets to value at least half, the cost guarantee is immediate. Consider any degree bound $i \in [m]$; let b_i' denote its residual bound when it is dropped, and F' the set of chosen elements at that iteration. Again, rounding elements of fractional value at least half implies $|E_i \cap F'| \leq \lfloor 2b_i - 2b_i' \rfloor = 2b_i - \lceil 2b_i' \rceil$. Furthermore, the number of E_i-elements in the support of the basic solution at the iteration when constraint i is dropped is at most $\lceil 2b_i' \rceil + \Delta - 1$. Thus the number of E_i-elements chosen in the final solution is at most $2b_i - \lceil 2b_i' \rceil + \lceil 2b_i' \rceil + \Delta - 1 = 2 \cdot b_i + \Delta - 1$ ■

D Minimum Crossing Lattice Polyhedra

Before we study *minimum crossing lattice polyhedra* (Definition 2), we give a few examples of well-known discrete optimization problems which can be formalized as the problem to find an optimal integral vector of a lattice polyhedron.

D.1 Examples of lattice polyhedra

The reductions given here can also be found in [Sch03] and [FKP08], for example.

Polymatroid intersection. Let $r_1, r_2 : E \to \mathbb{Z}_+$ be two supermodular rank functions on the same ground set E, $c : E \to \mathbb{R}$ and consider the polymatroid intersection problem

$$\min\{c^T x \mid x(T) \geq \max\{r_1(T), r_2(T)\} \forall T \subseteq E, x \in \{0,1\}^E\}.$$

We show that this problem might as well be formulated as a lattice polyhedron problem: Let E' and E'' be two disjoint copies of E and set $\tilde{E} = E' \cup E''$. We consider the lattice $(\mathcal{F}, \subseteq, \cup, \cap)$ defined on

$$\mathcal{F} = \{S \subseteq \tilde{E} \mid S \subseteq E' \text{ or } E' \subseteq S\}.$$

The set-valued function $\rho : \mathcal{F} \to 2^E$ and the rank function $r : \mathcal{F} \to \mathbb{Z}_+$ are now given by

$$r(T') := r_1(T) \quad \text{and} \quad r(\tilde{E} \setminus T'') := r_2(T) \quad \forall T \subseteq E,$$
$$\rho(T') := T \quad \text{and} \quad \rho(\tilde{E} \setminus T'') := T \quad \forall T \subseteq E,$$

where T' and T'' are the E' and E''-copies, respectively, of the set T. Since ρ satisfies the consecutivity and submodularity properties on $(\mathcal{F}, \subseteq, \cap, \cup)$, problem

$$\min\{c^T x \mid \sum_{e \in \rho(S)} x_e \geq r(S), \forall S \in \mathcal{F}; x \in \{0,1\}^E\}$$

is a lattice polyhedron problem and equivalent to the polymatroid intersection problem above.

Shortest paths. Let $D = (V, E)$ be a digraph with edge-costs $c : E \to \mathbb{R}_+$ and designated vertices $s, t \in V$. In the *shortest-path problem* one aims to find a directed s,t-path P in G of minimum cost $c(P) = \sum_{e \in P} c_e$. We formulate the shortest-path problem as a lattice polyhedron problem as follows: Consider the collection of all s,t-cuts

$$\mathcal{F} = \{U \subseteq V \mid s \in U, t \notin U\},$$

and map each cut $U \in \mathcal{F}$ to the set of its outgoing edges; i.e., $\rho(U) := \delta^+(U) \subseteq 2^E$. It is well-known that the function $\rho : \mathcal{F} \to 2^E$ satisfies the consecutivity and submodularity properties on $(\mathcal{F}, \subseteq, \cap, \cup)$. Since the constant function $r \equiv 1$ is certainly supermodular on $\mathcal{F}$, the shortest-path problem

$$\min\{c^T x \mid \sum_{e \in \delta^+(S)} x_e \geq 1, \forall S \in \mathcal{F}; x \in \{0,1\}^E\}$$

turns out to be a special instance of the lattice polyhedron problem.

Max flow/min cut in s,t-planar graphs. Let $G = (V, E)$ be a directed or undirected graph with $s, t \in V$ and denote by $\mathcal{P} \subseteq 2^E$ the collection of all cycle-free s,t-paths in G. Given edge-capacities $c : E \to \mathbb{R}$ the min cut problem can be formulated as

$$\min\{c^T x \mid \sum_{e \in P} x_e \geq 1, \forall P \in \mathcal{P}; x \in \{0,1\}^E\}.$$

Note that this problem is a lattice polyhedron (with constant rank function $r \equiv 1$ and identity function $\rho(P) = P$ for all $P \in \mathcal{P}$) as soon as we can find a lattice $(\mathcal{P}, \leq, \vee, \wedge)$ on the collection of paths satisfying the consecutivity and submodularity properties. Given a planar representation of G with s, t on the outer boundary of the representation (graphs for which such a representation exists are called *s,t-planar graphs*), we can define such a lattice in a natural manner: we simply set

$$P \leq Q \quad \Longleftrightarrow \quad Q \text{ is the uppermost path in } G[P \cup Q] \quad \forall P, Q \in \mathcal{P},$$

where $G[P \cup Q]$ is the subgraph of G induced by the edges in P and Q, and the *uppermost path* is constructed greedily as follows: start with the uppermost edge leaving s and always traverse the next outgoing edge in clockwise order (w.r.t. the planar representation). Consequently, the join $P \vee Q$ is the uppermost path, and the meet $P \wedge Q$ is the *lowestmost* path in $G(P \cup Q)$ (the latter is constructed analoguesly by starting with the lowest s-leaving edge and always traversing the next outgoing edge in counterclockwise order). It is not hard to see that the resulting lattice satisfies the desired consecutivity and submodularity properties.

We note that the two-phase greedy algorithms described in [Fra99], [FP08] find a min cut together with a max flow (i.e., the dual solution) in an s,t-planar graph even in the more general setting with supermodular monotone rank function $r : \mathcal{P} \to \mathbb{Z}_+$.

Supermodular systems. Following Fujishige [Fuj05], a *supermodular system* $(\mathcal{D}, r)$ consists of a family of subsets $\mathcal{D} \subseteq 2^E$ of a finite set E with $\emptyset, E \in \mathcal{D}$ such that $(\mathcal{D}, \subseteq, \cup, \cap)$ forms a distributive lattice, together with a supermodular function $r : \mathcal{D} \to \mathbb{R}$ which is normalized in the sense $r(\emptyset) = 0$. Fujishige described a greedy algorithm which optimizes a linear function over the base polyhedron of a supermodular system

$$\{x \in \mathbb{R}^E \mid x(E) = r(E); x(S) \geq r(S), \forall S \in \mathcal{D}\}.$$

Note that our iterative rounding algorithm for the minimum crossing lattice polyhedron problem also applies when we are interested in a basis solution, i.e., one satisfying $x(E) = r(E)$. Since any supermodular system defines a lattice polyhedron with inclusion-wise ordering, Theorem 7 applies in the special case where we are interested in an integral vector of a supermodular base polyhedron satisfying certain degree bounds.

D.2 Algorithm for minimum crossing lattice polyhedra

We consider the minimum crossing lattice polyhedron problem in a slightly more general form than Definition 2: we allow both *upper and lower bounds* on the family $\{E_i\}_{i=1}^m$. Let $\{a_i\}_{i=1}^m$ denote the respective lower-bounds, as in Definition 2, let $\{b_i\}_{i=1}^m$ denote the upper-bounds. We first give an algorithm for Theorem 6 and prove it.

The algorithm for minimum crossing lattice polyhedra is based on iterative rounding. At each iteration, we maintain the following:

- $F \subseteq E$ of elements that have been chosen into the solution.
- $E' \subseteq E \setminus F$ of undecided elements.
- $W \subseteq [m]$ of degree bounds.

Initially $E' = E$, $F = \emptyset$ and $W = [m]$. In a generic iteration with E', F, W, we solve the following LP relaxation on variables $\{x_e \mid e \in E'\}$, called $\mathcal{LP}_{\text{lat}}(E', F, W)$:

$$\begin{aligned} &\min c^T \cdot x && \\ x(\rho(S)) &\geq r(S) - |F \cap \rho(S)|, && \forall S \in \mathcal{F} \\ a_i - |F \cap E_i| \leq x(E_i) &\leq b_i - |F \cap E_i|, && \forall i \in W \\ 0 \leq x_e &\leq 1, && \forall e \in E' \end{aligned}$$

Consider an optimal basic feasible solution x to the above LP relaxation. The algorithm does one of the following in iteration (E', F, W), until $E' = W = \emptyset$.

1. If there is $e \in E'$ with $x_e = 0$, then $E' \leftarrow E' \setminus \{e\}$.
2. If there is $e \in E'$ with $x_e = 1$, then $F \leftarrow F \cup \{e\}$ and $E' \leftarrow E' \setminus \{e\}$.
3. If there is $i \in W$ with $|E_i \cap E'| \leq 2\Delta$, then $W \leftarrow W \setminus \{i\}$.

D.3 Proof of Theorem 6

Assuming that one of the steps (1)-(3) applies at each iteration, it is clear that we obtain a final solution F^* that has cost at most the optimal value, satisfies the rank constraints, and violates each degree constraint by at most an additive $2\Delta - 1$. We next show that one of (1)-(3) applies at each iteration (E', F, W).

Lemma 4 *Suppose $(\mathcal{F}, \leq)$ is a lattice satisfying the consecutive and submodular properties, and condition $(*)$, function r is supermodular, and x is a basic feasible solution to $\mathcal{LP}_{lat}$ with $0 < x_e < 1$ for all $e \in E'$. Then there exists some $i \in W$ with $|E_i \cap E'| \leq 2\Delta$.*

We first establish some standard uncrossing claims, before proving the lemma. We also need some more definitions. Two elements $A, B \in \mathcal{F}$ are said to be *comparable* if either $A \leq B$ or $B \leq A$; they are *non-comparable* otherwise. A subset $\mathcal{L} \subseteq \mathcal{F}$ is called a *chain* if $\mathcal{L}$ contains no pair of non-comparable elements.

Let $r'(S) := r(S) - |F \cap \rho(S)|$ for all $S \in \mathcal{F}$ denote the right hand side of the rank constraints in the LP solved in a generic iteration (E', F, W).

Claim. r' is supermodular.

Proof: This follows from the consecutive and submodular properties of lattice $(\mathcal{F}, \leq)$. Consider any $A, B \in \mathcal{F}$, and

$$\begin{aligned}
|F \cap \rho_A| + |F \cap \rho_B| &= |F \cap (\rho_A \cup \rho_B)| + |F \cap (\rho_A \cap \rho_B)| \\
&\geq |F \cap (\rho_{A \wedge B} \cup \rho_{A \vee B})| + |F \cap (\rho_A \cap \rho_B)| \\
&\geq |F \cap (\rho_{A \wedge B} \cup \rho_{A \vee B})| + |F \cap (\rho_{A \wedge B} \cap \rho_{A \vee B})| \\
&= |F \cap \rho_{A \wedge B}| + |F \cap \rho_{A \vee B}|
\end{aligned}$$

The second inequality follows from submodularity (i.e. $\rho_A \cup \rho_B \supseteq \rho_{A \wedge B} \cup \rho_{A \vee B}$), and the third inequality uses the consecutive property $\rho_{A \wedge B} \cap \rho_{A \vee B} \subseteq \rho_A, \ \rho_B$ (since $A \wedge B \leq A, B \leq A \vee B$). This combined with supermodularity of r implies $r'(A) + r'(B) \leq r'(A \wedge B) + r'(A \vee B)$ for all $A, B \in \mathcal{F}$. ■

For any element $A \in \mathcal{F}$, let $\chi(A) \in \{0,1\}^{E'}$ be the incidence vector of $\rho(A) \subseteq E'$. Let $\mathcal{T} := \{A \in \mathcal{F} \mid x(\rho_A) = r'(A)\}$ denote the elements in $\mathcal{F}$ that correspond to tight rank constraints in the LP solution x of this iteration. Using the fact that r' is supermodular (from above), and by standard uncrossing arguments, we obtain the following.

Lemma 5 *If $S, T \in \mathcal{F}$ satisfy $x(\rho_S) = r'(S)$ and $x(\rho_T) = r'(T)$, then:*

$$x(\rho(S \wedge T)) = r'(S \wedge T) \quad \textit{and} \quad x(\rho(S \vee T)) = r'(S \vee T)$$

Moreover, $\chi(S) + \chi(T) = \chi(S \wedge T) + \chi(S \vee T)$.

Proof: We have the following sequence of inequalities:

$$\begin{aligned}
r'(S\wedge T)+r'(S\vee T) &\le x(\rho_{S\wedge T})+x(\rho_{S\vee T})\\
&= x(\rho_{S\wedge T}\cap\rho_{S\vee T})+x(\rho_{S\wedge T}\cup\rho_{S\vee T})\\
&\le x(\rho_{S\wedge T}\cap\rho_{S\vee T})+x(\rho_S\cup\rho_T)\\
&\le x(\rho_S\cap\rho_T)+x(\rho_S\cup\rho_T)\\
&= x(\rho_S)+x(\rho_T)\\
&= r'(S)+r'(T)\\
&\le r'(S\wedge T)+r'(S\vee T)
\end{aligned}$$

The first inequality is by feasibility of x, the third inequality is the submodular lattice property, the fourth inequality is by consecutive property, and the last inequality is supermodularity of r'. Thus we have equality throughout, in particular $x(\rho(S\vee T)) = r'(S\vee T)$ and $x(\rho(S\wedge T)) = r'(S\wedge T)$. Finally since $x_e > 0$ for all $e\in E'$, we also have $\chi(S)+\chi(T)=\chi(S\wedge T)+\chi(S\vee T)$. ■

Lemma 6 ([Sch03]) *There exists a chain $\mathcal{L}\subseteq\mathcal{T}$ such that the vectors $\{\chi(A)\mid A\in\mathcal{L}\}$ are linearly independent and span $\{\chi(B)\mid B\in\mathcal{T}\}$.*

We are now ready for the proof of Lemma 4.
Proof: [Lemma 4] $|E'|$ is the number of non-zero variables in basic feasible x. Hence there exist tight linearly independent constraints: $\mathcal{L}\subseteq\mathcal{F}$ corresponding to rank-constraints and $\mathcal{B}\subseteq W$ degree-constraints, such that $|E'|=|\mathcal{L}|+|\mathcal{B}|$. Furthermore, by Lemma 6 $\mathcal{L}$ is a *chain* in $\mathcal{F}$, say consisting of the elements $S_1<S_2<\cdots<S_k$. We claim that,

$$|\rho(S_j)\setminus\left(\cup_{t=1}^{j-1}\rho(S_t)\right)|\ge 2,\qquad \text{for each } 1\le j\le k \tag{2}$$

The above condition is clearly true for $j=1$: since $x(\rho(S_1))=r'(S_1)\ge 1$ (it is positive and integer-valued), and $x_e<1$ for all $e\in E'$. Consider any $j\ge 2$. By the consecutive property on $S_t\le S_{j-1}<S_j$ (for any $1\le t\le j-1$), we have $\rho(S_j)\cap\rho(S_t)\subseteq\rho(S_{j-1})$. So, $\rho(S_j)\setminus\left(\cup_{t=1}^{j-1}\rho(S_t)\right)=\rho(S_j)\setminus\rho(S_{j-1})$. We now claim that $|\rho(S_j)\setminus\rho(S_{j-1})|\ge 2$, which would prove (2). Since $S_{j-1}<S_j$, assumption $(*)$ implies that there is at least one element $e\in\rho(S_j)\setminus\rho(S_{j-1})$. Moreover, if this is the only element, i.e., if $\rho(S_j)\setminus\rho(S_{j-1})=\{e\}$, then $\rho(S_{j-1})=\rho(S_j)\setminus\{e\}$ must be true (again by property $(*)$). But this causes a contradiction to the non-integrality of x_e:

$$x_e = x\left(\rho(S_j)\right)-x\left(\rho(S_{j-1})\right)=r'\left(\rho(S_j)\right)-r'\left(\rho(S_{j-1})\right)\in\mathbb{Z}.$$

Now, equation (2) implies that $k=|\mathcal{L}|\le\frac{|E'|}{2}$. Hence $|E'|\le 2|\mathcal{B}|$.

Suppose (for contradiction) that $|E_i\cap E'|\ge 2\Delta+1$ for all $i\in W$. Then $\sum_{i\in W}|E_i\cap E'|\ge(2\Delta+1)\cdot|W|$. Since each element in E' appears in at most Δ sets $\{E_i\}_{i\in W}$, we have $\Delta\cdot|E'|\ge\sum_{i\in W}|E_i\cap E'|\ge(2\Delta+1)\cdot|W|$. Thus $|E'|>2|W|\ge 2|\mathcal{B}|$, which contradicts $|E'|\le 2|\mathcal{B}|$ from above. ■

We are now able to prove the main result of this section:

Proof: [Theorem 6] Since the algorithm only picks 1-elements into the solution F, the guarantee on cost can be easily seen. As argued in Lemma 4, at each iteration (E', F, W) one of the Steps (1)-(3) apply. This implies that the quantity $|E'| + |W|$ decreases by 1 in each iteration; hence the algorithm terminates after at most $|E|+|I|$ iterations. To see the guarantee on degree violation, consider any $i \in I$ and let (E', F, W) denote the iteration in which it is dropped, i.e. Step (3) applies here with $|E_i \cap E'| \leq 2\Delta$ (note that there must be such an iteration, since finally $W = \emptyset$). Since a degree bound is dropped at this iteration, we have $0 < x_e < 1$ for all $e \in E'$ (otherwise one of the earlier steps (1) or (2) applies).

1. *Lower Bound:* $a_i - |F \cap E_i| \leq x(E_i \cap E') < |E' \cap E_i| \leq 2\Delta$, i.e. $a_i \leq |F \cap E_i| + 2\Delta - 1$. The final solution contains at least all elements in F, so the degree lower bound on E_i is violated by at most $2\Delta - 1$.
2. *Upper Bound:* The final solution contains at most $|F \cap E_i| + |E' \cap E_i|$ elements from E_i. If $E_i \cap E' = \emptyset$, the upper bound on E_i is not violated. Else, $0 < x(E_i \cap E') \leq b_i - |F \cap E_i|$, i.e. $b_i \geq 1 + |F \cap E_i|$, and $|F \cap E_i| + |E' \cap E_i| \leq b_i + 2\Delta - 1$. So in either case, the final solution violates the upper bound on E_i by at most $2\Delta - 1$.

Observing that all the steps (1)-(3) preserve the feasibility of the $\mathcal{LP}_{\text{lat}}$, it follows that the final solution satisfies all rank constraints (since $E' = \emptyset$ finally). ∎

D.4 Inlcusion-wise ordered lattice polyhedra

We now consider a special case of minimum crossing lattice polyhedra where the lattice $\mathcal{F}$is ordered by inclusion. This class of lattice polyhedra clearly satisfy assumption $(*)$, so Theorem 6 applies. However in this case, we prove the stronger guarantee in Theorem 7 for the setting with *only upper bounds* as in Definition 2. The algorithm remains the same as the one above for Theorem 6. In order to prove Theorem 7 it suffices to show the following strengthening of Lemma 4.

Lemma 7 *Suppose $(\mathcal{F}, \leq)$ is a lattice satisfying the consecutive and submodular properties, and condition*

$$S \leq T \quad \Longleftrightarrow \quad \rho_S \subseteq \rho_T \quad \forall S, T \in \mathcal{F},$$

function r is supermodular, and x is a basic feasible solution to $\mathcal{LP}_{lat}$ with $0 < x_e < 1$ for all $e \in E'$. Then there exists some $i \in W$ with $|E_i \cap E'| \leq b'_i + \Delta - 1$.

Proof: Since x is a basic feasible solution, there exist linearly independent tight rank function- and degree bound constraints $\mathcal{T}$ and $\mathcal{B} \subseteq W$ such that

$$|E'| = |\mathcal{T}| + |\mathcal{B}|.$$

Using uncrossing arguments, we can assume that $(\mathcal{T}, \leq)$ forms a chain

$$\mathcal{T} = \{T_1 < T_2 < \ldots < T_k\}.$$

Consider an arbitrary pair $T_i < T_{i+1}$ in $\mathcal{T}$. Since $x_e > 0$ for all $e \in E$ and $\rho(T_i) \subset \rho(T_{i+1})$, it follows that $0 < x(\rho(T_{i+1}) \setminus \rho(T_i))$ and therefore, by the integrality of r,

$$x(\rho(T_{i+1}) \setminus \rho(T_i)) = x(\rho(T_{i+1})) - x(\rho(T_i)) = r(T_{i+1}) - r(T_i) \geq 1.$$

Thus,

$$x(E) \geq x(\rho(T_k)) = \sum_{i=1}^{k-1} x(\rho(T_{i+1}) \setminus \rho(T_i)) \geq k = |\mathcal{T}|$$

with equality only if $E = \rho(T_k)$. This implies that

$$|E'| - x(E) = |\mathcal{T}| + |\mathcal{B}| - x(E) \leq |\mathcal{B}|. \tag{3}$$

Let $E'_i = E' \cap E_i$. To prove the statement of the Lemma, it suffices to show:

$$\sum_{i \in W} (|E'_i| - b'_i) = \sum_{i \in W} (|E'_i| - x(E_i)) < \Delta|W|.$$

In order to prove this, define $\Delta_e = |\{i \in W \mid e \in E_i\}|$ and consider the derivations

$$\begin{aligned}
\sum_{i \in W} (|E'_i| - x(E_i)) &= \sum_{i \in W} \sum_{e \in E'_i} (1 - x_e) = \sum_{e \in E} \Delta_e (1 - x_e) \\
&= \Delta \sum_{e \in E} (1 - x_e) - \sum_{e \in E} (\Delta - \Delta_e)(1 - x_e) \\
&\underbrace{\leq}_{\text{eq.}(3)} \Delta|\mathcal{B}| - \sum_{e \in E} (\Delta - \Delta_e)(1 - x_e) \\
&= \Delta|W| - \Delta|W \setminus \mathcal{B}| - \sum_{e \in E} (\Delta - \Delta_e)(1 - x_e) \leq \Delta|W|.
\end{aligned}$$

Note that equality can only hold if $E = \rho(T_k)$ and $\Delta|W \setminus \mathcal{B}| + \sum_{e \in E}(\Delta - \Delta_e)(1 - x_e) = 0$. The latter can only be true if $|\mathcal{B}| = |W|$ and $\Delta_e = \Delta$ for each $e \in E$. But this would imply that

$$\sum_{i \in \mathcal{B}} \chi^{E_i} = \Delta \chi^E = \Delta \chi^{T_k},$$

where $\chi^S \in \{0,1\}^{\mathcal{F} \times E}$ is the incidence vector of $S \in \mathcal{F}$ with $\chi^S_e = 1$ iff $e \in \rho(S)$. However, this contradicts the fact that the constraints $\mathcal{T}$ and $\mathcal{B}$ are linearly independent. ■