

Thin trees for laminar families

Nathan Klein

University of Washington
Seattle, USA
nwklein@cs.washington.edu

Neil Olver

London School of Economics and Political Science
London, UK
n.olver@lse.ac.uk

Abstract—In the laminar-constrained spanning tree problem, the goal is to find a minimum-cost spanning tree which respects upper bounds on the number of times each cut in a given laminar family is crossed. This generalizes the well-studied degree-bounded spanning tree problem, as well as a previously studied setting where a chain of cuts is given. We give the first constant-factor approximation algorithm; in particular we show how to obtain a multiplicative violation of the crossing bounds of less than 22 while losing less than a factor of 5 in terms of cost.

Our result compares to the natural LP relaxation. As a consequence, our results show that given a k -edge-connected graph and a laminar family $\mathcal{L} \subseteq 2^V$ of cuts, there exists a spanning tree which contains only an $O(1/k)$ fraction of the edges across every cut in \mathcal{L} . This can be viewed as progress towards the *Thin Tree Conjecture*, which (in a strong form) states that this guarantee can be obtained for all cuts simultaneously.

I. INTRODUCTION

Let $G = (V, E)$ be a connected undirected graph. Given any proper $S \subset V$, we use $\delta(S)$ to denote the cut with shores S and $V \setminus S$. A spanning tree T of G is called α -thin if the number of edges of T crossing any given cut of G is at most an α fraction of the total number of edges: $|T \cap \delta(S)| \leq \alpha |\delta(S)|$ for each $S \subseteq V$.

In 2004, Goddyn [10] made the following conjecture: there exists a function $f : \mathbb{Z}_+ \rightarrow [0, 1]$ with $\lim_{k \rightarrow \infty} f(k) = 0$ such that every k -edge-connected graph G has an $f(k)$ -thin spanning tree. This has become known as the *thin tree conjecture*, and it remains open despite substantial efforts.

A natural strengthening of the conjecture, which we will refer to as the *strong thin tree conjecture* makes the same claim, but for $f(k) = C/k$ for some constant C . This conjecture is found explicitly in [3] and is the best that one could hope for up to constant factors; clearly no k -edge-connected graph has an α -thin tree for any $\alpha < 1/k$. In a different direction, there is also an algorithmic question one can ask: if a thin tree always exists, can we *find* one in polynomial time?

The thin tree conjecture has some nice implications. It implies the *weak 3-flow conjecture* of Jaeger [14]. This has since been resolved, by Thomassen [30], however this would provide an alternate proof. Another application lies in the asymmetric traveling salesman problem (ATSP). As shown by

Asadpour, Goemans, Madry, Oveis Gharan and Saberi [3, 25], if the constructive form of the strong thin tree conjecture is true, it would yield an $O(1)$ -approximation algorithm to ATSP. This has since been resolved by Svensson, Tarnawski and Végé [29] using completely different methods. Nonetheless, a new algorithm stemming from thin trees would be of significant interest. Furthermore, a constant factor approximation algorithm to the *bottleneck* version of the asymmetric traveling salesman problem, where the goal is to minimize the longest edge in the tour rather than the sum, is not known. This would follow from the constructive form of the thin tree conjecture [1].

Although the (strong) thin tree conjecture would no longer imply breakthroughs to these other problems, it remains a natural question in its own right. Turning things around, the positive resolution of these implications can perhaps be viewed as some weak evidence for the conjecture.

For the following discussion, it is useful to observe that the strong thin tree conjecture has the following equivalent formulation. Suppose we are given a graph G as well as a point x in the spanning tree polytope (that is, a convex combination of characteristic vectors of edge sets of spanning trees of G). We say that a spanning tree T is α -thin with respect to x if $|T \cap \delta(S)| \leq \alpha x(\delta(S))$ for every $S \subseteq V$. The conjecture is that there is a universal constant α such that an α -thin tree with respect to x always exists, for any instance and point in the spanning tree polytope. The equivalence follows from the observation that the point x' defined by $x'_e = 2/k$ for all $e \in E$ is in the dominant of the spanning tree polytope for every k -edge-connected graph G , and so there is a point x in the spanning tree polytope with $x_e \leq 2/k$ for all e . An α -thin tree with respect to x is then a $(2\alpha/k)$ -thin tree for the graph.

a) *Progress on the thin tree conjecture.*: The conjecture is known to hold for some graph classes, most notably planar and bounded genus graphs [25]. For general graphs, the best known result is that there always exists an $O(\frac{\text{poly} \log \log n}{k})$ -thin tree in any k -connected graph [2]. This is non-constructive; constructively, the best known is only $O(\frac{\log n}{\log \log n \cdot k})$ -thinness.

One difficulty with the constructive form of the conjecture is that it's not even clear how to *check* if a given tree T is α -thin, or even $O(\alpha)$ -thin. Nor do we know of a polynomially checkable certificate that can certify thinness. The problem, of course, is that there are an exponential number of cuts to be concerned with. An easier question presents itself: what if we consider an explicitly given family of cuts, and require the

N.K. was supported in part by Air Force Office of Scientific Research grant FA9550-20-1-0212 and NSF grants DGE-1762114 and CCF-1813135. N.O. was supported by NWO Vidi grant 016.Vidi.189.087.

thinness condition $|T \cap \delta(S)| \leq \alpha |\delta(S)|$ only for these specific cuts? And one step further: what if we consider a family of cuts with some specific structure?

b) Explicitly given cut collections.: Related questions have been considered from an algorithmic perspective already, independently from the thin tree conjecture. The first class considered was that of *singleton cuts*. Suppose we are given an integer-valued degree bound b_v for each node v of the graph G . The degree bounded spanning tree problem asks for a spanning tree satisfying these bounds, if such a spanning tree exists. This problem is easily seen to be NP-hard, since it captures the question of finding a Hamiltonian path with a specified start and end node. So it is necessary to allow for some relaxation of the degree bounds. Fürer and Raghavachari [9] showed that relaxing the degree bounds by 1 additively suffices. That is, they showed how to efficiently find a spanning tree T satisfying $|T \cap \delta(v)| \leq b_v + 1$ for all $v \in V$, if there exists a spanning tree T^* that satisfies the degree bounds exactly.

One can also consider a minimum cost version of the question. Now each edge $e \in E$ has a nonnegative cost $c(e)$, and the goal is to find a *cheapest* spanning tree satisfying the degree bounds (again, assuming one exists). Goemans [12] showed how to efficiently find a spanning tree T which violates the degree bounds by at most an additive 2, and satisfies $c(T) \leq c(T^*)$, where T^* is a minimum cost spanning tree that satisfies all the degree bounds exactly. Singh and Lau [28] then showed how to improve the degree violation to just 1, while maintaining the same bound on the cost. They use the method of *iterative relaxation*; we use iterative relaxation as well, so we will discuss this further in the sequel.

That ends the story for degree bounds; what about other families of constraints? So we have a given family \mathcal{F} of subsets, and a “degree bound” b_S for each $S \in \mathcal{F}$. Olver and Zenklusen [24] showed how to obtain, constructively, a constant multiplicative violation of all cut constraints if \mathcal{F} is a *chain*; that is, $\mathcal{F} = \{S_1, S_2, \dots, S_t\}$ with $S_1 \subsetneq S_2 \subsetneq \dots \subsetneq S_t$. Linhares and Swamy [20] showed that a minimum cost version of this result also holds, if one allows a constant factor approximation in the cost as well as in the cut constraints.

All of these results compare to the natural fractional relaxation. That is, they do not require that there is an actual tree satisfying the degree bounds, merely that there is a point in the spanning tree polytope which does. As such, we can view them in the context of thin trees. They show that weaker versions of the strong thin tree conjecture hold, where the cut bounds are enforced only on singleton cuts, or only on a chain of cuts. We will say that the strong thin tree conjecture holds for a given family \mathcal{F} if given any x in the spanning tree polytope, there is a spanning tree T satisfying $|T \cap \delta(S)| \leq O(1)x(\delta(S))$ for all $S \in \mathcal{F}$.

c) Our results.: Given that the strong thin tree conjecture holds for the family of singletons, and for a chain family, a very natural question presents itself. Suppose that \mathcal{L} is an arbitrary *laminar family* of subsets of V ; that is, for every $S, T \in \mathcal{L}$, $S \cap T$ is either equal to \emptyset , S , or T . Does the strong thin tree conjecture hold for \mathcal{L} ?

We show that this is indeed true. Further, our proof is constructive, and allows for costs. More precisely, given arbitrary nonnegative edge costs, our returned tree has cost within a constant factor of the cost of the starting fractional solution x .

We briefly sketch our high-level approach, leaving a full overview until Section II-D. As already mentioned, iterative relaxation has been applied very successfully to the degree-bounded spanning tree problem, so it is a natural candidate approach. However, there is an immediate obstruction. Iterative relaxation for degree-bounded spanning tree is fairly insensitive to the use of the graphic matroid; it works just as well (essentially without changes) if the graphic matroid is replaced by any other matroid.¹ However, the matroid generalization of the laminar-constrained spanning tree problem does *not* have a constant integrality gap, and even obtaining a constant factor multiplicative violation is hard. This was shown by Olver and Zenklusen [24] already for the chain case. So any successful approach will need to exploit the graphic matroid specifically; it is not clear how to do this directly with iterative relaxation.

We manage to bypass this obstruction and make use of iterative relaxation. We do this by first reducing to a special class of instances that we call \mathcal{L} -aligned, where the fractional solution x has the property that for every set S in the laminar family of constraints \mathcal{L} , the restriction of x to S is a point in the base polytope of the graphic matroid for the graph restricted to S . Our reduction crucially exploits properties of spanning trees, and does not apply to general matroids. We then give an iterative relaxation proof of this \mathcal{L} -aligned case. This part *does* generalize to arbitrary matroids.

d) Other related work.: For laminar families, the most directly comparable work is from 2013 by Bansal, Khandekar, Knemann, Nagarajan, and Peis. They give an *additive* $O(\log n)$ approximation for the laminar constrained spanning tree problem [4], improving upon an earlier more general result which given a family of m constraints obtains a violation of $(1 + \epsilon)b + O(\frac{1}{\epsilon} \log m)$ for each bound b [6]. As previously mentioned, Olver and Zenklusen [24] demonstrated a constant factor multiplicative violation for a family of cuts given by a chain. These three results are with respect to the fractional relaxation, and thus also solve the related thin tree problems. Nägele and Zenklusen [22] demonstrated that in quasi-polynomial time the violation for the chain-constrained spanning tree problem can be improved to a $(1 + \epsilon)$ multiplicative factor, for any $\epsilon > 0$. They further generalize this slightly towards laminar families, by allowing for a family of cuts that form a laminar family of constant *width*, meaning that the maximum number of disjoint sets in the laminar family is bounded by a constant. (Put differently, the number of leaves in the tree representing the laminar family is constant). However, this result is not based on the LP relaxation, and so does not imply anything for the strong thin tree conjecture for chains or constant-width laminar families.

¹With the notable exception of [28] which solves the bounded degree spanning problem with an additive error of 1 on both lower *and* upper bounds. When translated to the general matroid setting, the additive error is only known to be 2 [16].

This problem has also been studied for general matroids. Király, Lau and Singh [16] showed that given a matroid \mathcal{M} and a collection of upper bound constraints, one can achieve an additive violation of $\Delta - 1$ for all constraints, so long as every element of the matroid is in at most Δ constraints. They achieve a similar guarantee if lower bounds (or both lower and upper bounds) are present. Similar results and further generalizations can be found in [6, 4].

Pritchard [26] conjectured that every k -edge-connected graph contains a spanning tree after whose deletion the graph remains $k - f(k)$ connected, where $f(k)$ is any function for which $\lim_{k \rightarrow \infty} f(k)/k = 0$. This can easily be seen as a weakening of the thin tree conjecture. The strong version of this conjecture (which is a consequence of the strong thin tree conjecture) is that $f(k)$ is an absolute constant. Currently the best known bound for this problem (to the best of our knowledge) is $f(k) = \lfloor \frac{k}{2} \rfloor - 1$ by the Nash-Williams theorem [23].

There is a natural *spectral* strengthening of thin trees. Let L_H denote the Laplacian of a graph H , and let \preceq denote the Löwner ordering on symmetric matrices². We say T is α -*spectrally-thin* if $L_T \preceq \alpha L_G$; that is, if $z^T L_T z \leq \alpha z^T L_G z$ for any vector $z \in \mathbb{R}^V$. This is a stronger condition than α -thinness, as can be seen by choosing z to be the characteristic vector of a set $S \subseteq V$. A big advantage of spectral thinness is that it can be efficiently checked. A natural analogue of the strong thin tree conjecture, where connectivity is replaced by the minimum effective conductance, can be derived [13] as a consequence of results on the Kadison-Singer problem [21]. This demonstrates that the strong thin tree conjecture holds for edge transitive graphs (or any graph where the minimum edge conductance is within a constant factor of the connectivity). Unfortunately, spectral thinness is too strong a property to directly aid in proving the (strong or weak) thin tree conjecture in general; there are instances where no $o(\sqrt{n}/k)$ -spectrally thin tree exists [13, 11]. Nonetheless, spectral approaches have been fruitful. The current best result by Anari and Oveis Gharan [2] mentioned previously, that $O(\log \log n/k)$ -thin trees exist, makes use of spectral methods in a sophisticated way. Our approach on the other hand is completely combinatorial; we will not make use of any spectral techniques.

II. PRELIMINARIES AND RESULTS

A. Notation

Given a graph $G = (V, E)$ and a subset $S \subseteq V$, let $\delta(S) = \{\{u, v\} : |\{u, v\} \cap S| = 1\}$ denote the set of edges with exactly one endpoint in S . Let $G[S]$ denote the induced graph of G whose vertex set is S , and let $E(S) \subseteq E$ denote the set of edges in $G[S]$. For $\mathcal{P} = \{P_1, \dots, P_k\}$ a partition of a subset of the vertices of G , we let $\delta(\mathcal{P})$ denote the set of edges with endpoints in two different sets P_i . If the choice of G is not clear, we may write, e.g., $\delta_G(S)$ or $\delta_G(\mathcal{P})$.

²That is, $A \preceq B$ if $B - A$ is positive semidefinite.

For any edge weight function $x : E \rightarrow \mathbb{R}$, we write $x(F) := \sum_{e \in F} x(e)$. For $F \subseteq E$, we write $x|_F$ to denote x restricted to F .

B. Polyhedral Background

Edmonds [7] gave the following description for the convex hull of the spanning trees of any graph $G = (V, E)$, known as the *spanning tree polytope*.

$$P_{\text{st}}(G) = \left\{ x \in \mathbb{R}_{\geq 0}^E : x(E) = |V| - 1, \right. \\ \left. x(E(S)) \leq |S| - 1 \ \forall S \subseteq V \right\}. \quad (1)$$

The following is the natural LP relaxation for the problem given in Definition II.2.

$$\min \sum_{e \in E} x_e c_e \\ \text{s.t., } x(\delta(S)) \leq b_S \quad \forall S \in \mathcal{L}, \\ x \in P_{\text{st}}(G) \quad (2)$$

For two $x, x' \in \mathbb{R}^E$, we say x **dominates** x' if $x - x' \geq 0$. Let $P_{\text{st}}^\uparrow(G)$ denote the dominant of the spanning tree polytope of G , that is, the set of points in $x \in \mathbb{R}^E$ which dominate some point in $P_{\text{st}}(G)$. $P_{\text{st}}^\uparrow(G)$ has the following characterization:

$$P_{\text{st}}^\uparrow(G) = \left\{ x \in \mathbb{R}_{\geq 0}^E : x(\delta(\mathcal{P})) \geq |\mathcal{P}| - 1 \ \forall \text{ partitions } \mathcal{P} \text{ of } V \right\} \quad (3)$$

It is well-known that $P_{\text{st}}^\uparrow(G)$ can be separated efficiently.

Theorem II.1 ([5]). *Given a graph $G = (V, E)$ and a point $x \in \mathbb{R}_{\geq 0}^E$, A partition \mathcal{P} of G minimizing $x(\delta(\mathcal{P})) - (|\mathcal{P}| - 1)$ can be found in polynomial time.*

Suppose $\mathcal{M} = (E, \mathcal{I})$ is a matroid with groundset E and independent sets \mathcal{I} . The *matroid base polytope* of \mathcal{M} , which we will denote $P_{\mathcal{M}}$, is the convex hull of the incidence vectors of all bases of \mathcal{M} . The *rank* of \mathcal{M} , denoted $\text{rank}_{\mathcal{M}}$, is the cardinality of the largest independent set of \mathcal{M} . Given $F \subseteq E$:

- 1) The *deletion* of F from \mathcal{M} is the matroid on the groundset $\mathcal{M} \setminus F$ with independent sets $\{I \setminus F : I \in \mathcal{I}\}$. If $F = \{e\}$, i.e. it is a singleton, we will use the shorthand $\mathcal{M} - e$.
- 2) The *restriction* of \mathcal{M} to F , denoted $\mathcal{M}|_F$, is the matroid on groundset F with independent sets $\{I \cap F : I \in \mathcal{I}\}$. This is equivalent to the deletion of $E \setminus F$.
- 3) The *contraction* of \mathcal{M} by F , denoted \mathcal{M}/F , is the matroid on the groundset $E \setminus F$ with independent sets $\{I \subseteq E \setminus F : I \cup B \in \mathcal{I}\}$, where B is an arbitrary basis of $\mathcal{M}|_F$ (equivalently, an independent set of \mathcal{M} contained in F of largest cardinality). If $F = \{e\}$, i.e. it is a singleton, we will use the shorthand \mathcal{M}/e .

C. Our Results

We recall that a family of sets $\mathcal{L} \subseteq 2^V$ is **laminar** if for all $S, T \in \mathcal{L}$, $S \cap T$ is either equal to \emptyset , S , or T .

Definition II.2 (Laminar constrained spanning tree problem). Let $G = (V, E)$ be a connected graph, and \mathcal{L} a laminar family on V , with an associated degree bound $b_S \in \mathbb{Z}_{\geq 0}$ for each

$S \in \mathcal{L}$. The goal is to find efficiently a spanning tree T for which $|T \cap \delta(S)| \leq \alpha b_S$ for each $S \in \mathcal{L}$, assuming that there does exist a spanning tree T^* satisfying $|T^* \cap \delta(S)| \leq b_S$. In such a case, we say T is an α -approximate solution to the laminar constraints. We assume for convenience that $V \in \mathcal{L}$, though the associated constraint is of course vacuous.

To solve the above problem, we first determine if LP (2) is feasible, which can be done in polynomial time. If it is not, we may return “no” to the above problem since this would certify that such a tree does not exist. Thus to obtain an α approximation for the problem above, it is enough to obtain an α -thin tree with respect to a solution x of (2).

Definition II.3 (Laminar α -thin tree for (G, \mathcal{L}, x)). As input we get a graph $G = (V, E)$, a laminar family \mathcal{L} over V , and a feasible LP solution x to LP (2). Our goal is to find a spanning tree T such that $|T \cap \delta(S)| \leq \alpha x(\delta(S))$ for all $S \in \mathcal{L}$, i.e. a tree that is α thin with respect to x .

The main result of this paper is the following. We remark it also gives an $O(1)$ approximation in terms of the cost of the tree.

Theorem II.4. *Given an instance (G, \mathcal{L}, x) , we can in polynomial time find a spanning tree T such that:*

- i) $c(T) \leq (2 + \sqrt{7})c(x) < 5c(x)$, and
- ii) $|T \cap \delta(S)| \leq (2 + \sqrt{7})^2 x(\delta(S)) < 22x(\delta(S))$, i.e., it is a 22-thin tree for (G, \mathcal{L}, x) .

Our theorem can be generalized as follows, which can be used to reduce the cost of the tree arbitrarily close to 2 (at the expense of incurring a larger multiplicative loss).

Theorem II.5 (Main). *Given an instance (G, \mathcal{L}, x) and any $\eta > 2$, we can in polynomial time find a spanning tree T such that:*

- i) $c(T) \leq \eta c(x)$, and
- ii) $|T \cap \delta(S)| \leq \frac{1}{1-\frac{2}{\eta}}(2\eta + 3)x(\delta(S))$.

We will prove the latter theorem, since the previous theorem follows by setting $\eta = 2 + \sqrt{7}$.

D. Proof Overview

A key observation of this paper is the usefulness of the following definition.

Definition II.6 (\mathcal{L} -aligned). Given a graph $G = (V, E)$ and a laminar family $\mathcal{L} \subseteq 2^V$ we say a point $x \in P_{\text{st}}(G)$ is \mathcal{L} -aligned if $x_{|E(S)} \in P_{\text{st}}(G[S])$ for all $S \in \mathcal{L}$.

Note that $G[S]$ should be connected for each set $S \in \mathcal{L}$, otherwise no point can be \mathcal{L} -aligned.

In Section III and Section IV we show the following two theorems which when combined immediately give Theorem II.5.

Theorem II.7 (Laminar thin trees for \mathcal{L} -aligned points). *Given an instance (G, \mathcal{L}, x) for which x is \mathcal{L} -aligned, we can find a tree T of cost at most $c(x)$ in polynomial time for which*

$$|T \cap \delta(S)| \leq 2\lceil x(\delta(S)) \rceil + 1 \leq 2x(\delta(S)) + 3$$

for all $S \in \mathcal{L}$.

Theorem II.8 (Reduction to \mathcal{L} -aligned points). *For any instance (G, \mathcal{L}, x) and any $\eta > 2$, we can find an instance (G, \mathcal{L}', x') in polynomial time such that:*

- i) x' is \mathcal{L}' -aligned,
- ii) x' is dominated by ηx ,
- iii) *If for a spanning tree T there are $\alpha, \beta \geq 0$ such that $|T \cap \delta(S)| \leq \alpha x'(\delta(S)) + \beta$ for all $S \in \mathcal{L}'$, then we have $|T \cap \delta(S)| \leq \frac{1}{1-\frac{2}{\eta}}(\eta\alpha + \beta)x(\delta(S))$ for all $S \in \mathcal{L}$.*

To obtain our main theorem, given an instance (G, \mathcal{L}) we solve LP (2) to obtain an instance (G, \mathcal{L}, x) . We then apply Theorem II.8 to obtain a \mathcal{L} -aligned instance (G, \mathcal{L}', x') . Finally, we apply Theorem II.7 to obtain our tree with the desired properties.

We remark that while Theorem II.7 can be generalized to hold for any matroid over the edges of a graph and any laminar family (see Section IV), Theorem II.8 cannot be. Olver and Zenklusen [24] showed that there is a matroid and a laminar family of constraints (in fact, their family is a chain, and their matroid simply a partition matroid) with no constant-thin basis, in particular giving a lower bound of $O(\frac{\log n}{\log \log n})$ on the multiplicative violation. Thus it is necessary that one of these two pieces cannot be generalized to all matroids.

Theorem II.8 is proved via a natural combinatorial procedure which iteratively replaces sets in \mathcal{L} that are far from meeting the criteria $x_{|E(S)} \in P_{\text{st}}(G[S])$ with some partition of them. We first consider the scaling ηx , and show that if $\eta \cdot x_{|E(S)} \in P_{\text{st}}^\uparrow(G[S])$ for all $S \in \mathcal{L}$, then there is a point x' dominated by ηx which is \mathcal{L} -aligned. If not, we iteratively find a minimal cut S for which $\eta \cdot x_{|E(S)} \notin P_{\text{st}}^\uparrow(G[S])$, and then find the partition $\mathcal{P} = \{P_1, \dots, P_k\}$ of S which maximally violates an inequality in $P_{\text{st}}^\uparrow(G[S])$. We then delete S from the laminar family and add P_1, \dots, P_k . We show that $\eta \cdot x_{|E(P_i)} \in P_{\text{st}}^\uparrow(G[P_i])$ for all i . Therefore, by applying this procedure we get closer to obtaining an \mathcal{L} -aligned point. To finish the proof, we show that this process allows us to still effectively maintain (iii) of Theorem II.8.

Theorem II.7 (and its generalization to arbitrary matroids) is proved via an iterative relaxation procedure. The criteria that x is \mathcal{L} -aligned is, in some sense, exactly what is needed to make the iterative relaxation procedure work.

III. REDUCTION TO \mathcal{L} -ALIGNED POINTS

The following definition is key to our reduction to \mathcal{L} -aligned points.

Definition III.1 (Well-connected). Call a set $S \subseteq V$ η -well-connected if $\eta \cdot x_{|E(S)} \in P_{\text{st}}^\uparrow(G[S])$, i.e., if $\eta x(\delta_{G[S]}(\mathcal{P})) - (|\mathcal{P}| - 1) \geq 0$ for all partitions \mathcal{P} of S .

We will make us of the following simple fact, that allows us to contract η -well-connected subsets of a given set when evaluating the well-connectedness of a given set S .

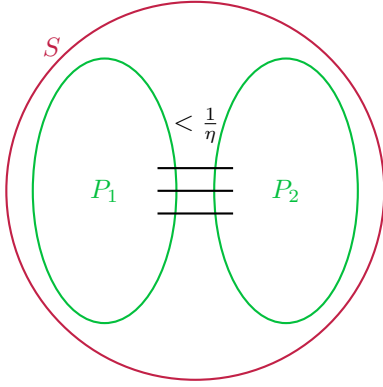


Fig. 1. An example of a set which is not η -well-connected (see Definition III.1). In this case, Algorithm 1 may replace S by P_1 and P_2 in \mathcal{L} .

Lemma III.2. Consider a set $S \subseteq V$, and suppose that S_1, \dots, S_r are disjoint subsets of S that are all η -well-connected. Let $G_S = (V_S, E_S)$ be the graph obtained from $G[S]$ after contracting each of S_1, \dots, S_r . Then S is η -well-connected, i.e., $\eta \cdot x_{|E_S} \in P_{\text{st}}^\uparrow(G[S])$, if and only if $\eta \cdot x_{|E_S} \in P_{\text{st}}^\uparrow(G_S)$.

Proof. Let $y = \eta x_{|E_S}$.

First, if $y \in P_{\text{st}}^\uparrow(G[S])$, then certainly $y_{|E_S} \in P_{\text{st}}^\uparrow(G_S)$, since given any convex combination of spanning trees of $G[S]$ that dominates y , the same convex combination of the images of these spanning trees upon contracting S_1, \dots, S_r is a convex combination of connected spanning subgraphs of G_S with marginals $y_{|E_S}$.

Conversely, suppose that $y_{|E_S} \in P_{\text{st}}^\uparrow(G_S)$. If $y_{|E_S} = \chi(T)$ for some spanning tree T of G_S , and each $y_{|E(S_i)} = \chi(T_i)$ for some spanning tree T_i of S_i , then the claim is clear; $T \cup T_1 \cup \dots \cup T_r$ is a spanning tree of $G[S]$. But the claim clearly remains true upon taking convex combinations, and moreover taking the dominant of any convex combination. \square

Algorithm 1 Reduction to a new laminar family

- 1: $\mathcal{L}' \leftarrow \emptyset$.
 - 2: **while** \mathcal{L} is nonempty **do**
 - 3: Choose a minimal set $S \in \mathcal{L}$.
 - 4: Let G_S be obtained from $G[S]$ by contracting all the maximal sets in \mathcal{L}' contained in S .
 - 5: Compute a partition \mathcal{P}' of G_S minimizing $\eta x(\delta_{G_S}(\mathcal{P}')) - (|\mathcal{P}'| - 1)$. Let \mathcal{P} be the corresponding partition of S obtained by uncontracting.
 - 6: Delete S from \mathcal{L} and add all parts of \mathcal{P} to \mathcal{L}' .
 - 7: **end while**
 - 8: Return \mathcal{L}' .
-

In this section we prove Theorem II.8, which heavily relies on Algorithm 1. This algorithm will be used to output the new family \mathcal{L}' in the theorem statement. As such, we first prove some properties of this algorithm.

Lemma III.3. Algorithm 1 can be implemented in polynomial time.

Proof. In each iteration, $|\mathcal{L}|$ decreases, so there are at most $2|V| - 1$ iterations. Each iteration can be implemented in polynomial time using Theorem II.1. \square

Lemma III.4. Consider any graph $G = (V, E)$ and $\eta > 0$. Let \mathcal{P} be a partition of G that minimizes $\eta x(\delta(\mathcal{P})) - (|\mathcal{P}| - 1)$. Then each part of \mathcal{P} is η -well-connected.

Proof. Fix any part $P \in \mathcal{P}$, and consider any partition $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_r\}$ of P . Let \mathcal{P}' be the partition of V' obtained by replacing P with the parts of \mathcal{Q} .

Write $\delta_{\mathcal{P}}(\mathcal{Q})$ for $\delta_{G[P]}(\mathcal{Q})$. Since $|\mathcal{P}'| = |\mathcal{P}| + |\mathcal{Q}| - 1$ and $\eta x(\delta(\mathcal{P}')) = \eta x(\delta(\mathcal{P})) + \eta x(\delta_{\mathcal{P}}(\mathcal{Q}))$, we have $\eta x(\delta_{\mathcal{P}}(\mathcal{Q})) - (|\mathcal{Q}| - 1)$ being equal to

$$\eta x(\delta(\mathcal{P}')) - (|\mathcal{P}'| - 1) - (\eta x(\delta(\mathcal{P})) - (|\mathcal{P}| - 1)).$$

This is nonnegative, by our choice of \mathcal{P} , and so $\eta x \in P_{\text{st}}^\uparrow(G[P])$. \square

Lemma III.5. The output \mathcal{L}' of Algorithm 1 is a laminar family, and each $S \in \mathcal{L}'$ is η -well-connected.

Proof. We claim that throughout the algorithm, we maintain the invariant that $\mathcal{L} \cup \mathcal{L}'$ is a laminar family, and that each $S \in \mathcal{L}'$ is η -well-connected. Certainly this holds at the start of the algorithm. Consider a partition \mathcal{P}' of G_S generated in step 5. By Lemma III.4, each part of \mathcal{P}' is η -well-connected. Then since the sets that were contracted in forming G_S are η -well-connected, by Lemma III.2 all parts of \mathcal{P}' are η -well-connected in G . Further, no part of \mathcal{P}' crosses a set in \mathcal{L}' , by construction. So the invariant is maintained. \square

The following is the main relevant quality of our reduction.

Lemma III.6. Let $S \in \mathcal{L}$ and let \mathcal{L}' be the output of Algorithm 1. Let S_1, \dots, S_ℓ be the unique maximal sets in \mathcal{L}' whose union is S . Then, $\sum_{i=1}^{\ell} x(\delta(S_i)) \leq \frac{1}{1-\frac{2}{\eta}} x(\delta(S)) - \frac{2}{\eta-2}$.

Proof. Consider the iteration of the algorithm where S is deleted from \mathcal{L} , and a partition \mathcal{P} of S (corresponding to a partition \mathcal{P}' of G_S) is added to \mathcal{L}' . Then $\mathcal{P} = \{S_1, \dots, S_\ell\}$. Note that by the choice of \mathcal{P}' , $\eta x(\delta_{G_S}(\mathcal{P}')) - (|\mathcal{P}'| - 1) \leq 0$ (either \mathcal{P}' is a violated constraint for $P_{\text{st}}^\uparrow(G_S)$, or if G_S is η -well-connected, \mathcal{P}' can be chosen to be the trivial partition of size 1, and equality is attained). Converting this to a statement about \mathcal{P} , we have $\eta x(\delta_{G[S]}(\mathcal{P})) - (|\mathcal{P}| - 1) \leq 0$. Thus

$$\begin{aligned} x(\delta(S)) &= \sum_{i=1}^{\ell} x(\delta(S_i)) - 2x(\delta_{G[S]}(\mathcal{P})) \\ &\geq \sum_{i=1}^{\ell} x(\delta(S_i)) - \frac{2}{\eta} (|\mathcal{P}| - 1) \\ &\geq \left(1 - \frac{2}{\eta}\right) \sum_{i=1}^{\ell} x(\delta(S_i)) + \frac{2}{\eta} \end{aligned}$$

Where the last line is a consequence of $x(\delta(S_i)) \geq 1$ for each S_i . The claim follows. \square

We now prove [Theorem II.8](#).

Theorem II.8 (Reduction to \mathcal{L} -aligned points). *For any instance (G, \mathcal{L}, x) and any $\eta > 2$, we can find an instance (G, \mathcal{L}', x') in polynomial time such that:*

- i) x' is \mathcal{L}' -aligned,
- ii) x' is dominated by ηx ,
- iii) If for a spanning tree T there are $\alpha, \beta \geq 0$ such that $|T \cap \delta(S)| \leq \alpha x'(\delta(S)) + \beta$ for all $S \in \mathcal{L}'$, then we have $|T \cap \delta(S)| \leq \frac{1}{1-\frac{2}{\eta}}(\eta\alpha + \beta)x(\delta(S))$ for all $S \in \mathcal{L}$.

Proof. First, apply [Algorithm 1](#) to \mathcal{L} to obtain a new family \mathcal{L}' (which requires only polynomial time by [Lemma III.3](#)). By [Lemma III.5](#), \mathcal{L}' is a laminar family of η -well-connected sets.

We now show that ηx dominates a point x' which is \mathcal{L}' -aligned, giving i) and ii). Let $G_S = (V_S, E_S)$ denote the graph obtained by restricting to $S \in \mathcal{L}'$ and contracting all children in \mathcal{L}' . By definition of η -well-connected, for any $S \in \mathcal{L}'$, $\eta x|_{E_S} \in P_{\text{st}}^{\uparrow}(G_S)$. It follows that for every $S \in \mathcal{L}'$ we can find $y_S \in P_{\text{st}}(G_S)$ with $y_S \leq \eta x|_{E_S}$. Combining y_S for each S , we obtain $x' \in P_{\text{st}}(G)$ with $x' \leq \eta x$, and where x' is \mathcal{L} -aligned.

It remains to show (iii). Fix some $S \in \mathcal{L}$. The algorithm replaces S by some partition S_1, S_2, \dots, S_ℓ of S in \mathcal{L}' . Then we have

$$\begin{aligned} |T \cap \delta(S)| &\leq \sum_{i=1}^{\ell} |T \cap \delta(S_i)| && \text{(as } \bigcup_{i=1}^{\ell} \delta(S_i) \subseteq \delta(S)) \\ &\leq \sum_{i=1}^{\ell} (\alpha \cdot x'(\delta(S_i)) + \beta) && \text{(by assumption)} \\ &\leq \sum_{i=1}^{\ell} (\eta\alpha \cdot x(\delta(S_i)) + \beta) && (x' \leq \eta x) \\ &\leq (\eta\alpha + \beta) \sum_{i=1}^{\ell} x(\delta(S_i)) && (x(\delta(S_i)) \geq 1, \forall S_i). \end{aligned}$$

By [Lemma III.6](#), $\sum_{i=1}^{\ell} x(\delta(S_i)) \leq \frac{1}{1-\frac{2}{\eta}}x(\delta(S))$. The claim follows. \square

IV. LAMINAR THIN TREES FOR \mathcal{L} -ALIGNED POINTS VIA ITERATIVE RELAXATION

We will now prove [Theorem II.7](#), or rather a generalization of it where the graphic matroid is replaced by an arbitrary matroid. First, we define the obvious generalization of \mathcal{L} -aligned for a point in the base polytope of a matroid \mathcal{M} .

Definition IV.1. Given a graph $G = (V, E)$, a matroid \mathcal{M} with groundset E , and a laminar family \mathcal{L} of G , we say that a point $x \in P_{\mathcal{M}}$ is \mathcal{L} -aligned if $x(E(S)) = \text{rank}_{\mathcal{M}}(E(S))$ for all $S \in \mathcal{L}$.

(In the case where \mathcal{M} is a graphic matroid, this is just slightly different from the previous definition, if some sets in \mathcal{L} are not connected. The previous definition did not allow for

any \mathcal{L} -aligned points in this case, but here it is possible. This relaxation of the definition is irrelevant; there is no real reason to consider disconnected sets in \mathcal{L} , since they could simply be split into their connected components.)

The following is the primary reason it is useful for a point x to be \mathcal{L} -aligned in the iterative relaxation process.

Lemma IV.2. *Let x be \mathcal{L} -aligned. Let $S \in \mathcal{L}$ and let $S_1, \dots, S_k \in \mathcal{L}$ such that $S_i \cap S_j = \emptyset$ for all $1 \leq i, j \leq k$, $i \neq j$. Let $G_S = (V_S, E_S)$ be the graph arising from contacting S_1, \dots, S_k in the graph $G[S]$.*

Then, $x(E_S)$ is an integer.

Proof. Since $x(E(S)) = \text{rank}_{\mathcal{M}}(E(S))$, it is an integer. Similarly, $x(E(S_i))$ is an integer for all $1 \leq i \leq k$. However $E_S = E(S) \setminus (\cup_{i=1}^k E(S_i))$, from which the claim follows. \square

Next, we define the notion of a *matroid* (rather than a point) being \mathcal{L} -aligned.

Definition IV.3. Given a graph $G = (V, E)$, a matroid \mathcal{M} with groundset E , and a laminar family \mathcal{L} of G , we say that \mathcal{M} is \mathcal{L} -aligned if for any basis B of \mathcal{M} , and every $S \in \mathcal{L}$, $B \cap E(S)$ is a basis of $\mathcal{M}|_{E(S)}$.

The relationship between the notion of a matroid being \mathcal{L} -aligned, and a point $x \in P_{\mathcal{M}}$ being \mathcal{L} -aligned, is captured by the following lemma.

Lemma IV.4. *A matroid \mathcal{M} is \mathcal{L} -aligned if and only if for every point $x \in P_{\mathcal{M}}$, x is \mathcal{L} -aligned.*

Proof. First suppose \mathcal{M} is \mathcal{L} -aligned and let $x \in P_{\mathcal{M}}$. Then, we can write x as a convex combination of some bases B_1, \dots, B_k of \mathcal{M} . Since \mathcal{M} is \mathcal{L} -aligned, $B_i \cap E(S)$ is a basis of $\mathcal{M}|_{E(S)}$ for all $S \in \mathcal{L}$. Thus $|B_i \cap E(S)| = \text{rank}(\mathcal{M}|_{E(S)})$ for all i . It follows that $x(E(S)) = \text{rank}(\mathcal{M}|_{E(S)}) = \text{rank}_{\mathcal{M}}(E(S))$ for all $S \in \mathcal{L}$ as desired, demonstrating that x is \mathcal{L} -aligned.

For the other direction, suppose every point $x \in P_{\mathcal{M}}$ is \mathcal{L} -aligned. Then for any basis B of \mathcal{M} , by taking x to be the characteristic vector of B , we have $|B \cap E(S)| = x(E(S)) = \text{rank}(\mathcal{M}|_{E(S)})$. Thus \mathcal{M} is \mathcal{L} -aligned. \square

In the previous section, we saw how to reduce to the case where x is a point in the base polytope of the graphic matroid that is \mathcal{L} -aligned. It will be more convenient for our purposes to work with a matroid that is \mathcal{L} -aligned; this is a stronger property that will ensure that all fractional points we consider later in the iterative relaxation algorithm are all \mathcal{L} -aligned as well. We can ensure this by *refining* the matroid, in the sense defined in [19].

Definition IV.5. Given a matroid \mathcal{M} and a nonempty proper subset R of the groundset, the *refinement* of \mathcal{M} with respect to R is the matroid \mathcal{M}' obtained as the direct sum of $\mathcal{M}|_R$ and \mathcal{M}/R .

Note that if \mathcal{M}' is a refinement of \mathcal{M} , then every base of \mathcal{M}' is a base of \mathcal{M} . It is easy to show that for $R \subseteq E$ with $x(R) = \text{rank}_{\mathcal{M}}(R)$, x remains in the base polytope of the

matroid obtained by refining \mathcal{M} with respect to R (see [19] for details). As such, given a point $x \in P_{\mathcal{M}}$ that is \mathcal{L} -aligned, we can repeatedly refine \mathcal{M} by each set of \mathcal{L} in turn, to obtain a new matroid \mathcal{M}' such that $x \in P_{\mathcal{M}'}$ and \mathcal{M}' is \mathcal{L} -aligned. For \mathcal{M} the graphic matroid, this refinement procedure corresponds to taking \mathcal{M}' to be the direct sum of graphic matroids on G_S for each $S \in \mathcal{L}$.

So we consider the generalization of the laminar thin tree problem to matroids, under the restriction that the matroid is aligned with the laminar family. An instance of the problem is defined by a graph $G = (V, E)$, a matroid \mathcal{M} with groundset E , and a laminar family \mathcal{L} with degree bounds b_S for $S \in \mathcal{L}$, such that \mathcal{M} is \mathcal{L} -aligned. Edge costs c_e may also be given. The goal is to find a minimum cost basis of \mathcal{M} satisfying the cut constraints, if a solution exists.

The following LP is the natural relaxation that we will use. Note that since \mathcal{M} is \mathcal{L} -aligned, no explicit additional constraints on x are required; *any* feasible solution must satisfy $x_{|E(S)} \in P_{\mathcal{M}_{|E(S)}}$, and thus must be \mathcal{L} -aligned.

$$\begin{aligned} \min \quad & \sum_{e \in E} x_e c_e \\ \text{s.t.} \quad & x(\delta(S)) \leq b_S \quad \forall S \in \mathcal{L}, \\ & x \in P_{\mathcal{M}}. \end{aligned} \quad (4)$$

Theorem IV.6 (Laminar-constrained matroid basis). *Given an instance $(G, \mathcal{M}, \mathcal{L}, b)$ in which \mathcal{M} is \mathcal{L} -aligned, and where the LP relaxation (4) has a feasible solution x , we can find a basis T of \mathcal{M} in polynomial time for which $c(T) \leq c(x)$ and $|T \cap \delta(S)| \leq 2b_S + 1$ for all $S \in \mathcal{L}$.*

Theorem II.7 is an immediate consequence, by first refining the graphic matroid as described above.

The algorithm we will use to prove this theorem is shown in **Algorithm 2**. Our algorithm follows the usual iterative relaxation recipe: it ignores edges set to 0 and 1 and then drops constraints which are close to being satisfied. We have one non-standard step which drops a set in \mathcal{L} if it is *approximately implied* by its immediate parent or child in the family of tight constraints. This non-standard step is what leads to a multiplicative violation instead of an additive one.

If a recursive call to LAMCONSTRAINEDBASIS returns “Fail”, then we consider that the result of the procedure as a whole is also “Fail”. We also note that if LAMCONSTRAINEDBASIS is recursively called in any of steps 3, 4 or 6, the required properties of the input to the recursive call are satisfied. In particular, (4) is feasible. For steps 3 and 4, $x_{|E-e}$ is feasible for the smaller instance; for step 6, simply x is. With this in mind, LAMCONSTRAINEDBASIS is well-defined.

We first show that as long as the algorithm does succeed, the returned basis obeys the theorem statement.

Lemma IV.7. *If **Algorithm 2** does not return “Fail”, the returned set B is a basis and obeys $c(B) \leq c(x)$ and $|B \cap \delta(S)| \leq 2b_S + 1$ for all $S \in \mathcal{L}$.*

Proof. We prove the claim by induction on $|E| + |\mathcal{L}|$. The claim is trivially true if $E = \emptyset$.

Algorithm 2 Procedure LAMCONSTRAINEDBASIS, used to demonstrate **Theorem IV.6**.

Input: Instance $(G = (V, E, c), \mathcal{M}, \mathcal{L}, b)$ where \mathcal{M} is tight for \mathcal{L} and (4) is feasible.

Output: Basis B of \mathcal{M} .

- 1: If $E = \emptyset$, **return** \emptyset .
 - 2: Let x be a basic optimal solution to (4).
 - 3: If there is an edge e with $x_e = 0$, **return** LAMCONSTRAINEDBASIS($G - e, \mathcal{M} - e, \mathcal{L}, b$).
 - 4: If there is an edge e with $x_e = 1$, **return** $\{e\} \cup$ LAMCONSTRAINEDBASIS($G - e, \mathcal{M}/e, \mathcal{L}, b'$), where $b'_S = b_S$ if $e \notin \delta(S)$, and $b'_S = b_S - 1$ if $e \in \delta(S)$.
 - 5: Let $\mathcal{L}_{\text{tight}}$ be the set of cuts $S \in \mathcal{L}$ with $x(\delta(S)) = b_S$.
 - 6: If there is a set $S \in \mathcal{L}_{\text{tight}}$ for which either $\sum_{e \in \delta(S)} (1 - x_e) < 3$, or there is an $S' \neq S \in \mathcal{L}_{\text{tight}}$ with $\delta(S') \subseteq \delta(S)$ and $\sum_{e \in \delta(S) \setminus \delta(S')} (1 - x_e) < 2$, then **return** LAMCONSTRAINEDBASIS($G, \mathcal{M}, \mathcal{L} \setminus \{S\}, b$).
 - 7: **return** “Fail”. ▷ Should not reach this line
-

So suppose the claim holds for all smaller values of $|E| + |\mathcal{L}|$. If $x_e = 0$ for some e in step 3, then the claim is immediate; as long as the recursive call succeeds, returning a basis B' of $\mathcal{M} - e$ approximately satisfying the constraints, then $B = B'$ is of course a basis of \mathcal{M} still approximately satisfying the constraints. Furthermore, since $c(B') \leq c(x')$ where x' is a basic optimal solution to the problem on $\mathcal{M} - e$, and $x_{|E \setminus \{e\}}$ is feasible for the problem on $\mathcal{M} - e$, $c(B) = c(B') \leq c(x') \leq c(x)$. If $x_e = 1$ for some e in step 4, and the recursive call succeeds and returns a basis B' of \mathcal{M}/e , then $B := B' \cup \{e\}$ is a basis of \mathcal{M} . Further, for any set $S \in \mathcal{L}$ with $e \notin \delta(S)$, we have

$$|B \cap \delta(S)| = |B' \cap \delta(S)| \leq 2b'_S + 1 = 2b_S + 1.$$

On the other hand if $e \in \delta(S)$, we have

$$|B \cap \delta(S)| = |B' \cap \delta(S)| + 1 \leq 2b'_S + 2 < 2b_S + 1.$$

Finally, since $c(B') \leq c(x')$ where x' was a basic optimal solution to the problem on \mathcal{M}/e , and $x_{|E \setminus \{e\}}$ is feasible for the problem on \mathcal{M}/e , $c(B) = c(B') + c(e) \leq c(x') + c(e) \leq c(x)$.

It remains to consider the situation where we drop a constraint in step 6. Suppose a set $S \in \mathcal{L}_{\text{tight}}$ is dropped because $|\delta(S)| - x(\delta(S)) = \sum_{e \in \delta(S)} (1 - x_e) < 3$. Since the constraint is tight, we deduce that $|\delta(S)| - b_S < 3$, and so $|\delta(S)| \leq b_S + 2 \leq 2b_S + 1$ as desired.

Now suppose $S \in \mathcal{L}_{\text{tight}}$ is dropped because there is an $S' \neq S \in \mathcal{L}_{\text{tight}}$ with $\delta(S') \subseteq \delta(S)$ and $\sum_{e \in \delta(S) \setminus \delta(S')} (1 - x_e) < 2$. By tightness, $x(\delta(S) \setminus \delta(S')) = x(\delta(S)) - x(\delta(S')) = b_S - b_{S'}$ is an integer. Note that either $\delta(S') = \delta(S)$, in which case clearly we can drop the duplicate constraint, or $b_S > b_{S'}$; assume the latter. We have $|\delta(S) \setminus \delta(S')| \leq 1 + b_S - b_{S'}$.

Suppose B is any basis satisfying $|B \cap \delta(S')| \leq 2b_{S'} + 1$. Then

$$\begin{aligned} |B \cap \delta(S)| &\leq |B \cap \delta(S')| + |\delta(S) \setminus \delta(S')| \\ &\leq (2b_{S'} + 1) + 1 + b_S - b_{S'} \\ &= b_{S'} + 2 + b_S \\ &\leq (b_S - 1) + 2 + b_S \leq 2b_S + 1. \end{aligned}$$

Of course, dropping a constraint can only decrease the cost of a basic optimal solution to (4), so $c(B) \leq c(x)$ is immediate by induction in this case. \square

Now we are ready to prove the theorem.

Proof of Theorem IV.6. By the above lemma, it is enough to prove that the algorithm succeeds. For this, it suffices to show that whenever the preconditions of LAMCONSTRAINEDBASIS are satisfied, the procedure never reaches step 7.

Suppose for a contradiction that we do reach step 7. By assumption, none of the constraints defining the extreme point x are of the form $x_e = 0$ or $x_e = 1$, so they all come from tight cut constraints and tight matroid constraints. Let $\mathcal{C}_{\text{basis}} = \{C_1, C_2, \dots, C_r\}$, with $C_1 \subsetneq C_2 \subsetneq \dots \subsetneq C_r \subseteq E$ and $\mathcal{L}_{\text{basis}} \subseteq \mathcal{L}_{\text{tight}}$ be such that the constraints $x(\delta(S)) = b_S$ for $S \in \mathcal{L}_{\text{basis}}$ and $x(C) = \text{rank}_{\mathcal{M}}(C)$ for $C \in \mathcal{C}_{\text{basis}}$ are a collection of linearly independent tight constraints defining x . Moreover, choose this basis of tight constraints in such a way that $|\mathcal{C}_{\text{basis}}|$ is as large as possible. The fact that the tight matroid constraints form a chain follows from standard uncrossing arguments (see [27] Chapter 41 or [16]). Since there are precisely $|E|$ defining constraints, we have

$$|E| = |\mathcal{L}_{\text{basis}}| + |\mathcal{C}_{\text{basis}}|.$$

We note that since \mathcal{M} is \mathcal{L} -aligned, the maximality of $\mathcal{C}_{\text{basis}}$ ensures that $E(S) \in \text{span}(\mathcal{C}_{\text{basis}})$ for each $S \in \mathcal{L}$.

Assign 1 splittable token to each $e \in E$; our goal will be to assign these tokens to the constraints of $\mathcal{L}_{\text{basis}}$ and $\mathcal{C}_{\text{basis}}$ so that each tight constraint gets 1 token, and there is something left over. This will be our desired contradiction.

We will assign x_e tokens to C_i for each $e \in C_i \setminus C_{i-1}$. Since $0 < x_e < 1$ for each e , and $x(C_i)$ and $x(C_{i-1})$ are both integers with $x(C_{i-1}) < x(C_i)$, we can deduce that $x(C_i \setminus C_{i-1}) \geq 1$.

Now each edge has $1 - x_e$ tokens remaining. Our token assignment scheme will be as follows. We start with an assignment that is very reminiscent of the scheme for degree bounded spanning trees [28]. For each $e = \{u, v\}$, we assign $(1 - x_e)/2$ tokens to the smallest set in $\mathcal{L}_{\text{basis}}$ containing u , and $(1 - x_e)/2$ tokens to the smallest set in $\mathcal{L}_{\text{basis}}$ containing v . After this, we work bottom up on $\mathcal{L}_{\text{basis}}$, and if $S \in \mathcal{L}_{\text{basis}}$ has strictly more than the 1 token needed, we assign the excess to its parent in $\mathcal{L}_{\text{basis}}$.

First, any minimal set $S \in \mathcal{L}_{\text{basis}}$ satisfies $\sum_{e \in \delta(S)} (1 - x_e) \geq 3$, meaning that at least $\frac{3}{2}$ tokens are initially assigned to S . So S receives enough tokens to give a half token as excess to its parent. Inductively, we claim that every set gets 1 token, and moreover, has an excess of at least $\frac{1}{2}$ that it

can give to its parent. For any non-minimal $S \in \mathcal{L}_{\text{basis}}$, we have three cases depending on the number of disjoint maximal children of S in $\mathcal{L}_{\text{basis}}$. In each case we will consider the graph $G_S = (V_S, E_S)$ resulting from contracting the maximal children of S in $\mathcal{L}_{\text{basis}}$ in the graph $G[S]$. In Cases 2 and 3 we crucially use that $x(E_S)$ is an integer by Lemma IV.2.

- **Case 1: S has at least three maximal children in $\mathcal{L}_{\text{basis}}$.** Then inductively, each of these children has an excess of at least $\frac{1}{2}$. This gives us at least $\frac{3}{2}$ tokens for S , as desired.
- **Case 2: S has exactly two maximal children $A, B \in \mathcal{L}_{\text{basis}}$.**

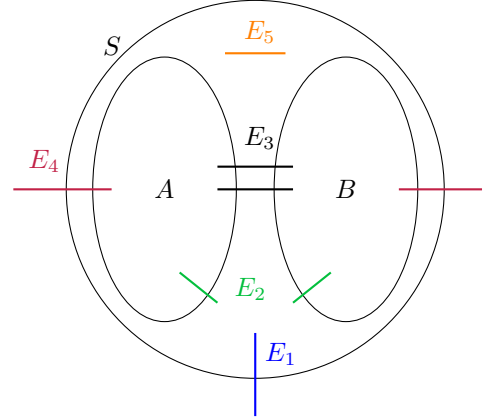


Fig. 2. Setting for Case 2. Note some edge sets may be empty.

Inductively, each child has an excess of at least $\frac{1}{2}$, giving us at least one token. Thus we need to collect at least $\frac{1}{2}$ additional tokens.

Consider the edge sets as defined in Fig. 2. In particular,

$$\begin{aligned} E_1 &= \delta(S) \setminus (\delta(A) \cup \delta(B)) \\ E_2 &= (\delta(A) \triangle \delta(B)) \setminus \delta(S) \\ E_3 &= \delta(A) \cap \delta(B) \\ E_4 &= (\delta(A) \cup \delta(B)) \cap \delta(S) \\ E_5 &= E_S \setminus (E_2 \cup E_3) \end{aligned}$$

First we observe that $E_1 \cup E_2 \cup E_5$ is nonempty. For suppose not; then, with χ denoting the incidence vector of a set, we can write $\chi(\delta(S)) + 2\chi(E(S))$ as

$$\chi(\delta(A)) + \chi(\delta(B)) + 2\chi(E(A)) + 2\chi(E(B)).$$

However, by the maximality of our choice of $\mathcal{C}_{\text{basis}}$, $E(A)$, $E(B)$ and $E(S)$ are all in the span of $\mathcal{C}_{\text{basis}}$, whereas $\delta(S)$, $\delta(A)$ and $\delta(B)$ are all in $\mathcal{L}_{\text{basis}}$. Thus we have a linear dependence among the constraints defined by $\mathcal{C}_{\text{basis}}$ and $\mathcal{L}_{\text{basis}}$, a contradiction.

So $x(E_1) + x(E_2) + x(E_5) > 0$. Therefore, we get

$$\begin{aligned} &\frac{|E_1| + |E_2| - x(E_1) - x(E_2)}{2} + |E_5| - x(E_5) \\ &= z - \left(\frac{x(E_1) + x(E_2)}{2} + x(E_5) \right) > 0 \end{aligned}$$

fractional tokens for some $z \in \mathbb{Z}_{\geq 0}$. We will prove that $\frac{x(E_1)+x(E_2)}{2} + x(E_5)$ is half integral, from which the claim follows. By the integrality of $x(E_S)$ (using \mathcal{L} -alignment) and the tightness of the constraints on A, B and S , we have that

$$\begin{aligned} a &:= x(E_2) + x(E_3) + x(E_5) \\ b &:= x(E_2) + 2x(E_3) + x(E_4) \\ c &:= x(E_1) + x(E_4) \end{aligned}$$

are all integers. Since $a - b/2 + c/2 = \frac{x(E_1)+x(E_2)}{2} + x(E_5)$, the claim follows.

- **Case 3:** S has precisely one maximal child S' in $\mathcal{L}_{\text{basis}}$.

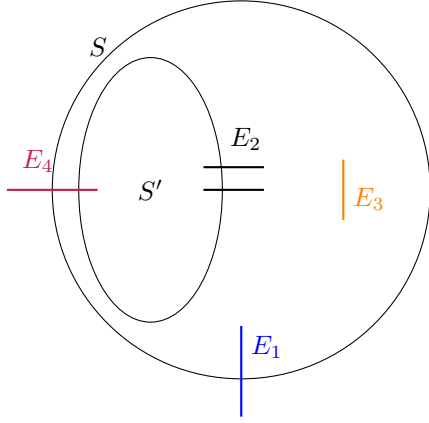


Fig. 3. Setting for Case 3.

We need to find 1 token that has been given by edges directly to S , so that the $\frac{1}{2}$ excess token from S' can be carried over as the excess of S .

If $\delta(S') \subseteq \delta(S)$, then because no relaxation step was possible in line 6, $\sum_{e \in \delta(S) \setminus \delta(S')} (1 - x_e) \geq 2$. Since each edge in $\delta(S) \setminus \delta(S')$ contributes $(1 - x_e)/2$ tokens, this gives us our token as necessary. Similarly, if $\delta(S') \supseteq \delta(S)$ we get the desired one token.

So assume that $\delta(S) \setminus \delta(S')$ and $\delta(S') \setminus \delta(S)$ are both nonempty. Let

$$\begin{aligned} E_1 &:= \delta(S) \setminus \delta(S'), \\ E_2 &:= \delta(S') \setminus \delta(S), \\ E_3 &:= E_S \setminus \delta(S'), \text{ and} \\ E_4 &:= \delta(S) \cap \delta(S'). \end{aligned}$$

(See Figure 3.)

Let $\delta \in [0, 1)$ be the fractional part of $x(E_4)$. Note that the number of tokens assigned to S is

$$|E_3| - x(E_3) + \frac{1}{2}(|E_1| + |E_2| - x(E_1) - x(E_2)). \quad (5)$$

Also observe that

$$x(E_2) + x(E_3), \quad x(E_1) + x(E_4), \quad \text{and} \quad x(E_2) + x(E_4) \quad (6)$$

are all integer-valued, by tightness of the cut constraints and \mathcal{L} -alignment. We distinguish two subcases.

- $\delta = 0$. Then $x(E_1)$ and $x(E_2)$ are both integers, and moreover since E_1 and E_2 are nonempty, $|E_1| - x(E_1)$ and $|E_2| - x(E_2)$ are both positive integers. This already gives us the desired 1 token by (5).
- $\delta > 0$. Then by (6) the fractional parts of $x(E_1)$ and $x(E_2)$ are both $1 - \delta$, and the fractional part of $x(E_3)$ is then δ . Thus $|E_1| - x(E_1) \geq \delta$ (being positive, with fractional part δ); similarly, $|E_2| - x(E_2) \geq \delta$ and $|E_3| - x(E_3) \geq 1 - \delta$. Substituting into (5), we have at least $1 - \delta + (2\delta)/2 = 1$ tokens assigned to S , as required.

We have demonstrated that all sets in $\mathcal{L}_{\text{basis}}$ receive a full tokens; moreover, any maximal set in $\mathcal{L}_{\text{basis}}$ will have an extra token that is not needed, since it has no parent to give it to. So we have our desired contradiction: $|E| > |\mathcal{C}_{\text{basis}}| + |\mathcal{L}_{\text{basis}}|$. \square

V. CONCLUSION

Besides the (strong) thin tree conjecture, our work leaves open several directions. One fascinating question is whether it is possible to leverage or strengthen our results to give a novel constant factor approximation algorithm for ATSP. While an algorithmic version of the strong thin tree conjecture is sufficient to give a constant factor approximation algorithm for ATSP, it is unclear if it is necessary: indeed, the current constant factor approximation algorithm for ATSP is not known to imply anything about thin trees. We ask if perhaps it is sufficient to focus on thinness for a laminar (or near laminar) family of cuts.

A second open question is to generalize our framework for other network design problems. One can ask, for example, if laminar constrained survivable network design and laminar constrained Steiner network admit bi-criteria constant factor approximations. Many results are known in the degree bounded case, see e.g. [17, 18, 8].

A third open question is whether it is possible to achieve a *minimum cost* tree which violates the degree bounds in a laminar family by any constant factor. One would need to avoid the scaling currently present in our reduction. A natural relaxation of this question is to ask for a $1 + \epsilon$ approximation for arbitrarily small ϵ as has been done for the chain case [20].

Finally, we note that our results immediately give a thin tree with respect to the set of minimum cuts of any graph, and we believe it may be possible to extend it to the set of all $(1 + \epsilon)$ near minimum cuts for some small $\epsilon > 0$ using results from [15]. We ask whether it is possible to extend our result to more general families of cuts such as the union of a constant number of laminar families or the set of cuts with at most αk edges in the graph for some constant α significantly larger than 1.

ACKNOWLEDGMENTS

We thank Shayan Oveis Gharan for asking about thin trees for minimum cuts, discussions of which were crucial in the early stages of this project, and we thank Deeparnab

Chakrabarty for several helpful conversations about that problem. We thank David Shmoys and Mohit Singh for asking the question about ATSP now stated in the conclusion, and an anonymous reviewer for their question on laminar constrained network design problems. N.O. thanks Michel Goemans and Rico Zenklusen for varied discussions on the topic of thin trees. This work began during the Trimester Program on Combinatorial Optimization at the Hausdorff Research Institute for Mathematics; we gratefully acknowledge the institute for its generous hosting and support.

REFERENCES

- [1] H.-C. An, R. Kleinberg, and D. B. Shmoys. “Approximation Algorithms for the Bottleneck Asymmetric Traveling Salesman Problem”. *ACM Transactions on Algorithms* 17(4). 2021. DOI: [10.1145/3478537](https://doi.org/10.1145/3478537) (cit. on p. 1).
- [2] N. Anari and S. Oveis Gharan. “Effective-Resistance-Reducing Flows, Spectrally Thin Trees, and Asymmetric TSP”. In: *Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 2015, pp. 20–39. DOI: [10.1109/FOCS.2015.11](https://doi.org/10.1109/FOCS.2015.11) (cit. on pp. 1, 3).
- [3] A. Asadpour, M. X. Goemans, A. Madry, S. Oveis Gharan, and A. Saberi. “An $O(\log n / \log \log n)$ -Approximation Algorithm for the Asymmetric Traveling Salesman Problem”. *Operations Research* 65(4). 2017, pp. 1043–1061 (cit. on p. 1).
- [4] N. Bansal, R. Khandekar, J. Könemann, V. Nagarajan, and B. Peis. “On generalizations of network design problems with degree bounds”. *Mathematical Programming* 141(1). 2013, pp. 479–506. DOI: [10.1007/s10107-012-0537-8](https://doi.org/10.1007/s10107-012-0537-8) (cit. on pp. 2, 3).
- [5] F. Barahona. “Separating from the dominant of the spanning tree polytope”. *Operations Research Letters* 12(4). 1992, pp. 201–203. DOI: [10.1016/0167-6377\(92\)90045-5](https://doi.org/10.1016/0167-6377(92)90045-5) (cit. on p. 3).
- [6] C. Chekuri, J. Vondrak, and R. Zenklusen. “Dependent Randomized Rounding via Exchange Properties of Combinatorial Structures”. In: *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 2010, pp. 575–584. DOI: [10.1109/FOCS.2010.60](https://doi.org/10.1109/FOCS.2010.60) (cit. on pp. 2, 3).
- [7] J. Edmonds. “Submodular functions, matroids and certain polyhedra”. In: *Combinatorial Structures and Their Applications*. Gordon and Breach, 1970, pp. 69–87 (cit. on p. 3).
- [8] A. Ene and A. Vakilian. “Improved Approximation Algorithms for Degree-Bounded Network Design Problems with Node Connectivity Requirements”. In: *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*. STOC ’14. Association for Computing Machinery, 2014, 754763. ISBN: 9781450327107. DOI: [10.1145/2591796.2591837](https://doi.org/10.1145/2591796.2591837) (cit. on p. 9).
- [9] M. Frer and B. Raghavachari. “Approximating the Minimum Degree Spanning Tree to Within One from the Optimal Degree”. In: *Proceedings of the Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1992, pp. 317–324. DOI: [10.1145/139404.139469](https://doi.org/10.1145/139404.139469) (cit. on p. 2).
- [10] L. A. Goddyn. “Some Open Problems I Like”. Available at <https://www.sfu.ca/~goddyn/Problems/problems.html>. 2004 (cit. on p. 1).
- [11] M. Goemans. *Thin spanning trees*. Lecture at “Graph Theory @ Georgia Tech: Conference Honoring the 50th Birthday of Robin Thomas”. 2012 (cit. on p. 3).
- [12] M. X. Goemans. “Minimum Bounded Degree Spanning Trees”. In: *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 2006, pp. 273–282. DOI: [10.1109/FOCS.2006.48](https://doi.org/10.1109/FOCS.2006.48) (cit. on p. 2).
- [13] N. J. A. Harvey and N. Olver. “Pipage Rounding, Pessimistic Estimators and Matrix Concentration”. In: *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2014, pp. 926–945 (cit. on p. 3).
- [14] F. Jaeger. “On circular flows in graphs, Finite and Infinite Sets”. *Colloquia Mathematica Societatis Janos Bolyai* 37. 1984, pp. 391–402 (cit. on p. 1).
- [15] A. Karlin, N. Klein, and S. Oveis Gharan. “A (Slightly) Improved Bound on the Integrality Gap of the Subtour LP for TSP”. In: *Proceedings of the 63rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 2022, pp. 844–855. DOI: [10.1109/FOCS54457.2022.00084](https://doi.org/10.1109/FOCS54457.2022.00084) (cit. on p. 9).
- [16] T. Király, L. C. Lau, and M. Singh. “Degree Bounded Matroids and Submodular Flows”. *Combinatorica* 32. 2012, pp. 703–720. DOI: [10.1007/s00493-012-2760-6](https://doi.org/10.1007/s00493-012-2760-6) (cit. on pp. 2, 3, 8).
- [17] L. C. Lau, J. S. Naor, M. R. Salavatipour, and M. Singh. “Survivable Network Design with Degree or Order Constraints”. *SIAM Journal on Computing* 39(3). 2009, pp. 1062–1087. DOI: [10.1137/070700620](https://doi.org/10.1137/070700620). eprint: <https://doi.org/10.1137/070700620> (cit. on p. 9).
- [18] L. C. Lau and M. Singh. “Additive Approximation for Bounded Degree Survivable Network Design”. *SIAM Journal on Computing* 42(6). 2013, pp. 2217–2242. DOI: [10.1137/110854461](https://doi.org/10.1137/110854461). eprint: <https://doi.org/10.1137/110854461> (cit. on p. 9).
- [19] A. Linhares, N. Olver, C. Swamy, and R. Zenklusen. “Approximate multi-matroid intersection via iterative refinement”. *Mathematical Programming* 183(1). 2020, pp. 397–418. DOI: [10.1007/s10107-020-01524-y](https://doi.org/10.1007/s10107-020-01524-y) (cit. on pp. 6, 7).
- [20] A. Linhares and C. Swamy. “Approximating Min-Cost Chain-Constrained Spanning Trees: A Reduction from Weighted to Unweighted Problems”. *Mathematical Programming* 172. 2018, pp. 17–34. DOI: [10.1007/s10107-017-1150-7](https://doi.org/10.1007/s10107-017-1150-7) (cit. on pp. 2, 9).

- [21] A. Marcus, D. A. Spielman, and N. Srivastava. “Interlacing Families II: Mixed Characteristic Polynomials and the Kadison-Singer Problem”. *Annals of Mathematics* 182(1). 2015, pp. 327–350 (cit. on p. 3).
- [22] M. Nägele and R. Zenklusen. “A New Dynamic Programming Approach for Spanning Trees with Chain Constraints and Beyond”. In: *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2019, pp. 1550–1569 (cit. on p. 2).
- [23] C. S.J. A. Nash-Williams. “Edge disjoint spanning trees of finite graphs”. *Journal of the London Mathematical Society* 36. 1961, pp. 445–45 (cit. on p. 3).
- [24] N. Olver and R. Zenklusen. “Chain-Constrained Spanning Trees”. *Mathematical Programming* 167. 2018, pp. 293–314. DOI: [10.1007/s10107-017-1126-7](https://doi.org/10.1007/s10107-017-1126-7) (cit. on pp. 2, 4).
- [25] S. Oveis Gharan and A. Saberi. “The Asymmetric Traveling Salesman Problem on Graphs with Bounded Genus”. In: *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2011, pp. 967–975 (cit. on p. 1).
- [26] D. Pritchard. “k-Edge-Connectivity: Approximation and LP Relaxation”. In: *Proceedings of the Ninth International Workshop on Approximation and Online Algorithms (WAOA)*. Springer Berlin Heidelberg, 2011, pp. 225–236. DOI: [10.1007/978-3-642-18318-8_20](https://doi.org/10.1007/978-3-642-18318-8_20) (cit. on p. 3).
- [27] A. Schrijver. *Combinatorial Optimization*. English. Vol. 2. Algorithms and Combinatorics. Springer, 2003. ISBN: 978-3-540-44389-6 (cit. on p. 8).
- [28] M. Singh and L. C. Lau. “Approximating minimum bounded degree spanning trees to within one of optimal”. *Journal of the ACM* 62(1). 2015. DOI: [10.1145/2629366](https://doi.org/10.1145/2629366) (cit. on pp. 2, 8).
- [29] O. Svensson, J. Tarnawski, and L. A. Végh. “A Constant-Factor Approximation Algorithm for the Asymmetric Traveling Salesman Problem”. *Journal of the ACM* 67(6). 2020. DOI: [10.1145/3424306](https://doi.org/10.1145/3424306) (cit. on p. 1).
- [30] C. Thomassen. “The weak 3-flow conjecture and the weak circular flow conjecture”. *Journal of Combinatorial Theory, Series B* 102(2). 2012, pp. 521–529. DOI: [10.1016/j.jctb.2011.09.003](https://doi.org/10.1016/j.jctb.2011.09.003) (cit. on p. 1).