k-Edge-Connectivity: Approximation and LP Relaxation

David Pritchard*

July 27, 2010

Abstract

In the k-edge-connected spanning subgraph problem we are given a graph (V, E) and costs for each edge, and want to find a minimum-cost $F \subset E$ such that (V, F) is k-edge-connected. If $\mathsf{P} \neq \mathsf{NP}$, for unit costs, the best possible approximation ratio is known to be $1 + \Theta(1/k)$ for $k > 1$. We show there is a constant $\epsilon > 0$ so that for all $k > 1$, finding a $(1+\epsilon)$-approximation for k-ECSS is NP-hard, establishing a gap between the unit-cost and general-cost versions. Next, we consider the *multi-subgraph* cousin of k-ECSS, in which we purchase a *multi-subset* F of E, with unlimited parallel copies available at the same cost as the original edge. We conjecture that a $(1 + \Theta(1/k))$-approximation algorithm exists, and we describe an approach based on graph decompositions applied to its natural linear programming (LP) relaxation. The LP is essentially equivalent to the Held-Karp LP for TSP and the undirected LP for Steiner tree. We give a family of extreme points for the LP which are more complex than those previously known.

1 Introduction

In the k-edge-connected spanning subgraph problem (k-ECSS), we are given an input graph G with edge costs, and must select a minimum-cost subset of edges so that the resulting graph has edge-connectivity k between all vertices. This is a natural problem for applications, since it is the same as seeking resilience against $(k-1)$ edge failures, or the ability to route k units of flow between any pair of vertices. A natural variant of k-ECSS is to allow each edge to be purchased repeatedly, as many times as desired, with each copy at the same cost. We call this the *k-edge-connected spanning multi-subgraph problem* (k-ECSM).

When $k = 1$ the k-ECSS and k-ECSM problems are both equivalent to the *minimum spanning tree* problem, which is well-known to be solvable in polynomial time, but they are non-trivial for $k > 1$. We consider *approximation algorithms* for these problems: an algorithm that approximately solves k-ECSS or k-ECSM is said to be an *α-approximation*, or have *approximation ratio* α, if it always outputs a solution with cost at most α times optimal.

Here we survey the oldest and newest results for k-ECSM and k-ECSS. Frederickson & Jájá gave a 2-approximation algorithm for 2-ECSS [19], and a 3/2-approximation in the special case of metric costs [20]. A 3/2-approximation is possible for 2-ECSM [8]. For k-ECSS/k-ECSM in general, Khuller & Vishkin [27] gave a matroid-based 2-approximation, and Jain's iterated LP rounding framework [26] also gives a 2-approximation. Goemans & Bertsimas [25] give an approximation algorithm for k-ECSM with ratio $\frac{3}{2}$ when k is even, and $(\frac{3}{2} + \frac{1}{2k})$ when k is odd. Fernandes [18] showed 2-ECSS is APX-hard, even for unit costs.

*École Polytechnique Fédérale de Lausanne and partially supported by an NSERC post-doctoral fellowship.

An important special case is where all edges have unit cost. Then k-ECSS gets *easier* to approximate as k gets larger: Gabow et al. [22] gave an elegant $(1 + 2/k)$-approximation algorithm for k-ECSS/k-ECSM using iterated LP rounding, and they showed that for some fixed $\epsilon > 0$, for all $k > 1$, it is NP-hard to get a $(1+\epsilon/k)$-approximation algorithm for unit-cost k-ECSS. Together, these establish a $1 + \Theta(1/k)$ approximability threshold for unit-cost k-ECSS. Improvements to the constant, and improvements in the special case that the input graph is simple, appear in Cheriyan & Thurimella [13] and Gabow & Gallagher [21].

1.1 Contributions

1.1.1 Hardness Results (Section 2)

Our first main result is the following hardness for k-ECSS:

Theorem 1. *There is a constant $\epsilon > 0$ so that for all $k \geq 2$, it is* NP*-hard to approximate k-ECSS within ratio $1 + \epsilon$, even if the costs are* 0-1*.*

Although $\epsilon \approx \frac{1}{300}$ here is small, the qualitative difference is important: whereas the approximability of unit-cost k-ECSS tends to 1 as k tends to infinity, we see that the approximability of general-cost k-ECSS is bounded away from 1.

Next we establish a relatively straightforward hardness result for k-ECSM.

Proposition 2. *The 2-ECSM problem is* APX*-hard.*

The key step is to show that 2-ECSM and *metric 2-ECSS* are basically the same problem. First, we use the following well-known fact: in k-ECSM, the input is metric without loss of generality [25] (i.e. the graph is complete and its costs satisfy the triangle inequality).[1] Then, simple reduction techniques show that under metric costs, any 2-ECSM can be efficiently converted to a 2-ECSS without increasing the costs. We remark that this approach also yields a simpler 3/2-approximation for 2-ECSM (c.f. [8]), using the 3/2-approximation for metric 2-ECSS [20] as a black box.

What Proposition 2 leaves to be desired is hardness for k-ECSM, $k > 2$, and asymptotic dependence on k. Why is it hard to show these problems are hard? The hard instances for k-ECSS given by Theorem 1 and [22] contain certain *mandatory parts* that are "without loss of generality" included in the optimal feasible solution; the argument proceeds to show hardness of the residual problem once the mandatory parts are included. But coming up with suitable mandatory parts for k-ECSM, while keeping the residual problem hard, is tricky: e.g. the proof of Theorem 1 will use a spanning tree of zero-cost edges, but in k-ECSM this leads to a trivial instance (buy that spanning tree k times). The known hardness for k-VCSS (vertex connectivity) by Kortsarz et al. [29] is similar: we take hard instances of 2-VCSS and add $(k-2)$ new vertices, connected to all other vertices by 0-cost (mandatory) edges. A new trick seems to be needed to get a good hardness result for k-ECSM.

1.1.2 k-ECSM Conjecture (Section 3)

We conjecture that approximation ratio $1 + O(1/k)$ should be possible for k-ECSM, using LPs. Obtain the natural LP relaxation of k-ECSM by allowing edges to be purchased fractionally: intro-

[1] To see this, take the metric closure (i.e. shortest path costs), solve it, and replace each uv-edge in the solution with a shortest u-v path from the original graph; it is not hard to show this preserves k-edge-connectivity. In k-ECSS, note metricity is not WOLOG, since the replacement step here can introduce multiple edges.

$$\min\Big\{\sum_{e\in E} c_e x_e : \quad x\in\mathbb{R}^E \qquad (\mathcal{N}_k)$$
$$\sum_{e\in\delta(S)} x_e \ge k, \quad \forall\varnothing\ne S\subsetneq V \qquad (1)$$
$$x_e\ge 0, \quad \forall e\in E\Big\} \qquad (2)$$

$$\min\Big\{\sum_{e\in E} c_e x_e : \quad x\in\mathbb{R}^E \qquad (\mathcal{N}'_k)$$
$$\sum_{e\in\delta(v)} x_e = k, \quad \forall v\in V \qquad (3)$$
$$\sum_{e\in\delta(S)} x_e \ge k, \quad \forall\varnothing\ne S\subsetneq V \qquad (4)$$
$$x_e\ge 0, \quad \forall e\in E\Big\} \qquad (5)$$

Figure 1: The undirected relaxation for k-edge connected spanning multi-subgraph. The unbounded version ($\mathcal{N}_k$) is on the left, the bounded version ($\mathcal{N}'_k$) is on the right. They have the same value for metric costs, including all k-ECSM instances.

duce a variable x_e for each edge, and require that there is a fractional value of at least k spanning each cut (see Figure 1, where $\delta(S)$ denotes the set of edges with exactly one end in S).

Conjecture 3. *There is a polynomial-time approximation algorithm for k-ECSM which produces a solution of value at most* $(1 + C/k)\cdot \mathrm{OPT}(\mathcal{N}_k)$ *for some universal constant* C.

This conjecture implies a $(1+C/k)$-approximation algorithm, since $\mathrm{OPT}(\mathcal{N}_k)$ is a lower bound on the optimal k-ECSM cost. What makes us think Conjecture 3 is true? First, we know it holds for unit costs. Second, the same holds in related high-*width* problems; to explain, say an integer program has width W if in every constraint, the right-hand side is at least W times every coefficient. Multicommodity flow/covering problems in trees are closely related to ($\mathcal{N}_k$) via *uncrossing* (e.g. [26, 22, 21]) and they admit an LP-based $1+O(1/W)$-approximation algorithm [28, 33] (in that setting W is the minimum edge capacity). Similar phenomena are known for LP relaxations of other structured integer programs [12, 11, 32, 5]. In k-ECSM the width is k so one may view our conjecture as seeking integrality gap[2] and approximation ratio $1 + O(1/W)$.

Later, we show an open problem of [4] — can every k-edge connected graph be partitioned into two spanning $(\frac{k}{2} - O(1))$-edge-connected subgraphs? — would imply a nonconstructive version of Conjecture 3. Few partial results towards Conjecture 3 are known: the integrality gap of ($\mathcal{N}_1$) is $2(1-1/n)$ [25], and that of ($\mathcal{N}_2$) is at most $3/2$ [36]. For general k, the best integrality gap bounds known for ($\mathcal{N}_k$) come from the approximation algorithms [26, 25, 21, 22] mentioned earlier.

One further motivation to investigate the conjecture has to do with the *parsimonious property* of Goemans & Bertsimas [25]. Using metricity and *splitting-off*, they showed the constraint $\forall v \in V : x(\delta(v)) = k$ can be added to ($\mathcal{N}_k$) without affecting the value of the LP (the strengthened LP ($\mathcal{N}'_k$) is shown in Figure 1). As observed in [25], parsimony implies that Conjecture 3 would give a $(1 + \frac{C}{k})$-approximation algorithm for *subset k-ECSM*, where we require edge-connectivity k only amongst a pre-specified set of terminal nodes (generalizing the Steiner tree problem). Thus even if we don't care about LPs *a priori*, they have algorithmic dividends in Conjecture 3.

[2] The integrality gap is the worst-case ratio of the integral optimum to the LP optimum.

1.1.3 Complex Extreme Points (Section 4)

In both of the LPs ($\mathcal{N}_k$) and ($\mathcal{N}'_k$), note that k serves only as a scaling factor: x is feasible for ($\mathcal{N}_1$) iff kx is feasible for ($\mathcal{N}_k$). In fact, these LPs are well-studied: ($\mathcal{N}_1$) is equivalent (by the parsimonious property [25]) to the *undirected cut relaxation of the Steiner tree problem* and ($\mathcal{N}'_2$) is the *Held-Karp relaxation of the Traveling Salesman Problem*. We demonstrate a family of extreme point solutions to these ubiquitous LPs which are more complex than were previously known.

For a solution x, the *support* is the edge set $\{e \mid x_e > 0\}$, and the *support graph* is the graph with vertex set V and the support for its edge set. The *fractionality* of x is $\min\{x_e \mid e \in E, x_e > 0\}$.

Theorem 4. *There are extreme point solutions for the linear program* ($\mathcal{N}'_2$) *with fractionality exponentially small in* $|V|$*, and whose support graph has maximum degree linear in* $|V|$.

The members of the family are also extreme point solutions for ($\mathcal{N}_2$), since ($\mathcal{N}'_2$) is a face of ($\mathcal{N}_2$). The motivation for this theorem comes from a common design methodology in LP-based approximation algorithms [26, 22, 24, 35]: algorithmically exploit good properties of extreme point solutions. E.g., Jain's algorithm [26] uses the fact that when ($\mathcal{N}_k$) is generalized to skew-submodular connectivity requirements, every extreme solution x^* has an edge e with $x^*_e \geq \frac{1}{2}$. Hence, complex extreme points give some idea of what properties might or might not exist that can be exploited algorithmically.

Theorem 4 significantly improves previous results in the same vein. (A long-standing conjecture that the Held-Karp relaxation ($\mathcal{N}'_2$) has integrality gap at most 4/3 has motivated some of the work, e.g. [10, 6].) Boyd and Pulleyblank [9] showed that for any even $|V| \geq 10$, there is an extreme point of ($\mathcal{N}'_2$) with fractionality $2/(|V| - 4)$. Cheung [14] found extreme points of ($\mathcal{N}'_2$) whose support graph has maximum degree $\Theta(\sqrt{|V|})$. The construction in Theorem 4 was found with the assistance of computational methods, see the arXiv version [34] for details.

2 Hardness Results

In our hardness theorem for k-ECSS, we reduce from the following problem. (Here $\uplus$ denotes disjoint union.)

PATH-COVER-OF-TREE
Input: A tree $T = (V, E)$ and another set $X \subset \binom{V}{2}$ of edges/pairs.
Output: A subset of Y of X so that $(V, E \uplus Y)$ is 2-edge-connected.
Objective: Minimize $|Y|$.

PATH-COVER-OF-TREE is sometimes called the *tree augmentation problem* and a 1.8-approximation is published [17]; as an aside, it is basically equivalent to the special case of 2-ECSS where the input graph contains a connected subgraph of cost zero, plus some unit-cost edges. We give it the alternate name PATH-COVER-OF-TREE because it is more natural for us to interpret it as covering a tree's edges with a minimum-size subcollection of a given collection of paths. To make this explicit, for an edge $x = \{u, v\} \in X$ let P_x denote the edges of the unique u-v path in T. We rehash the proof of the following proposition since we will recycle its methodology.

Proposition 5 (folklore). *Y is feasible for* PATH-COVER-OF-TREE *if and only if* $\bigcup_{x \in Y} P_x = E$.

Proof. For every edge e of T, a *fundamental cut* of e and T means the vertex set of either connected component of $T \backslash e$.

Let $\delta_F(U)$ denote $\delta(U)$ in the graph (V, F). First, Y is feasible if $|\delta_{E \uplus Y}(U)| \geq 2$ for every set U with $\varnothing \neq U \subsetneq V$. But $|\delta_E(U)|$ is 1 when U is a fundamental cut and at least 2 otherwise; hence Y is feasible iff $|\delta_Y(U)| \geq 1$ for every fundamental cut U.

Second, when U is a fundamental cut, say for an edge $e \in E$, $|\delta_Y(U)| \geq 1$ iff $\bigcup_{x \in Y} P_x$ contains e. Taking this together with the previous paragraph, we are done. □

Path-Cover-of-Tree is shown NP-hard in [19] and a similar construction implies APX-hardness — see Appendix A. As an aside, it is even hard for trees of depth 2; compare this with the depth-1 instances which are in P since they can be shown isomorphic to *edge cover*. Now we prove the main hardness result:

Theorem 1. *Let it be* NP*-hard to approximate* Path-Cover-of-Tree *within ratio* $1 + \epsilon$*. Then for all integers* $k \geq 2$*, it is* NP*-hard to approximate* k*-ECSS within ratio* $1 + \epsilon$*, even for* 0-1 *costs.*

Proof. Let $(T = (V, E), X)$ denote an instance of Path-Cover-of-Tree. We construct a k-ECSS instance on the same vertex set, with edge set F. For each $e \in E$, we put $k - 1$ zero-cost copies of the edge e into F. For each $x \in X$, put one unit-cost copy of the edge x into F. These are all the edges of F; and although (V, F) is a multigraph, we later show that this can be avoided.

First we show the multigraph instance is hard. Clearly, there is an optimal solution for the k-ECSS instance which includes all copies of the 0-cost edges. Let $(k-1)E$ denote these 0-cost edges. The same logic as in the proof of Proposition 5 (analysis using fundamental cuts) shows that Y is a feasible solution for the Path-Cover-of-Tree instance if and only if $(k - 1)E \uplus Y$ is a feasible solution for the k-ECSS instance. Since costs are preserved between the two problems, it follows that an α-approximation algorithm for k-ECSS would also give an α-approximation algorithm for Path-Cover-of-Tree, and we are done.

Finally, here is how we make (V, F) a simple graph: replace every vertex $v \in V$ of the tree by a $(k + 1)$-clique of 0-cost edges; replace every edge $uv \in E$ of the tree by any $k - 1$ zero-cost edges between the two cliques for u and v; replace each edge $x \in X$ by any unit-cost edge between the cliques for u and v. We proceed similarly to before: when U is a vertex set of the newly constructed graph, we see $\delta(U)$ has at least k 0-cost edges unless U is a "blown-up" version of a fundamental cut (i.e., unless there is a fundamental cut U_0 of T so that U exactly equals the set of vertices in cliques corresponding to U_0). As before, the residual problem assuming these edges are bought is the same as the instance (T, X) (in a cost-preserving way), so we are done. □

2.1 Hardness of 2-ECSM (Proof of Proposition 2)

To show that 2-ECSM is APX-hard, we prove that it is "the same" as metric 2-ECSS, i.e. the special case of 2-ECSS on complete metric graphs. Metric 2-ECSS is APX-hard by a general result of [7][3] and so this gives us what we want. The key observation is the following.

[3] Here is a sketch for the reader, somewhat simpler than the more general results of [7]. Take a family of hard TSP instances with costs 1 and 2 [31]. Using a little case analysis, [7] shows that a 2-ECSS can be transformed to a Hamiltonian cycle (TSP tour) by repeatedly replacing two edges with one edge, which does not incerase the overall cost if edge costs are 1 and 2; so for these instances, TSP and 2-ECSS are the same. In particular on these (metric) instances, finding the min-cost 2-ECSS is APX-hard.

Proposition 6. *In a metric instance, given a 2-ECSM (V, F), we can obtain in polynomial time a 2-ECSS (V, F') with $c(F') \leq c(F)$, as long as $|V| \geq 3$.*

In other words, parallel edges can be eliminated without increasing the cost. (A similar observation in [20] turns a 2-ECSS into a 2-VCSS for metric instances.) Because the proof of Proposition 6 is relatively straightforward and not too long, we defer it to Appendix B.

Proof of Proposition 2. Since metric 2-ECSS is APX-hard [7], it is enough to show that any α-approximation algorithm for 2-ECSM gives an α-approximation for metric 2-ECSS. The metric 2-ECSS algorithm is: compute an α-approximately-optimal 2-ECSM F and apply Proposition 6 to get a 2-ECSS F' with $c(F') \leq c(F)$. Using Proposition 6 a second time, and using the fact that every 2-ECSS is trivially a 2-ECSM, we see the optimal 2-ECSS and 2-ECSM values are the same. Hence F' is an α-approximately-optimal 2-ECSS, as needed. ☐

3 k-ECSM Conjecture and Edge Connectivity Decomposition

Here is the conjecture made in the introduction. We will relate it to questions about graph decomposition.

Conjecture 3. *There is a polynomial-time approximation algorithm for k-ECSM which produces a solution of value at most $(1 + C/k) \cdot \mathrm{OPT}(\mathcal{N}_k)$ for some universal constant C.*

For positive integers A and B, define $f(A, B)$ to be the least integer f so that every f-edge-connected multigraph can be partitioned into two spanning subgraphs, one A-edge-connected and one B-edge-connected. The last question in a paper of Bang-Jensen and Yeo [4] is to answer the following question: *is there a constant C such that $f(k, k) \leq 2k+C$ for all integers k?* This question could provide a combinatorial approach to Conjecture 3; we give a nonconstructive consequence below.

Theorem 7. *If a constant C exists such that $f(k, k) \leq 2k + C$ for all k, then every k-ECSM instance has a solution with cost at most $(1 + C/k) \cdot \mathrm{OPT}(\mathcal{N}_k)$, i.e. the integrality gap of $(\mathcal{N}_k)$ is at most $1 + C/k$.*

Before proving Theorem 7 we make some other remarks on this function f. By applying an edge-counting argument to $(A + B)$-regular, $(A + B)$-edge-connected graphs, we have $f(A, B) \geq A + B$. The same sort of argument gives $f(A, 1) \geq A + 2$ and $f(1, 1) \geq 4$ since a 1-edge-connected graph has average degree at least $2 - \frac{2}{|V|}$. One consequence of the Nash-Williams/Tutte theorem is that every $2t$-edge-connected multigraph contains t edge-disjoint spanning trees, which implies $f(A, B) \leq 2(A + B)$. Thus $f(1, 1) = 4$. One can show $f(A, 2) \geq A + 3$ by considering a $(A + 2)$-regular, $(A + 2)$-edge-connected parallelization of the Petersen graph since it is non-Hamiltonian (see also [30]). Not much else seems to be known; e.g. we are not aware of any evidence against the conjecture $\forall A, B : f(A, B) \leq A + B + 2$ [4]. Resolving the gap $A + 2 \leq f(A, 1) \leq 2A + 2$ would be interesting; for example, given a 100-edge-connected graph, if we want to delete a spanning tree of our choice and keep high edge-connectivity, is 49 really the most we can guarantee?

Variants of f have received some attention. It is an open conjecture [4, 2] to prove a finite bound for the analogue of $f(1, 1)$ in *directed graphs*, i.e. it is conjectured that for some C, every strongly-C-arc-connected digraph contains two spanning arc-disjoint strongly-connected subdigraphs. For

[4] Matt DeVos independently conjectures this at http://garden.irmacs.sfu.ca/?q=category/connectivity.

edge-connectivity in hypergraphs, $f(1,1)$ is not finite, i.e. for every integer t there are t-edge-connected hypergraphs with no two disjoint connected spanning subhypergraphs [3].

Proof of Theorem 7. Let x^* be an optimal extreme point solution to ($\mathcal{N}_k$). Since x^* is rational, there is an integer t such that tx^* is integral. Then, it is easy to see that tx^* (or more precisely, the multigraph obtained by taking tx^*_e copies of each edge e) is a tk-edge-connected spanning multisubgraph. Likewise, for any positive integer α, αtx^* is a (αtk)-ECSM.

The following claim follows easily by induction on n, given the assumption $f(k,k) \leq 2k + C$ for all k.

Claim 8. *For all positive integers k and n, every $(2^n(k+C) - C)$-ECSM can be decomposed into 2^n disjoint k-ECSMs.*

Now, for any integer n, let us pick α just large enough that $\alpha tk \geq (2^n(k+C) - C)$. Therefore, αtx^* can be decomposed into 2^n disjoint k-ECSMs. The cheapest one has cost at most

$$\frac{c(\alpha tx^*)}{2^n} = \alpha t 2^{-n} c(x^*) = \left\lceil \frac{2^n(k+C) - C}{tk} \right\rceil t 2^{-n} \mathrm{OPT}(\mathcal{N}_k).$$

Then using $\lceil \frac{2^n(k+C)-C}{tk} \rceil \leq \lceil \frac{2^n(k+C)}{tk} \rceil \leq \frac{2^n(k+C)}{tk} + 1$, we see there is a k-ECSM with cost at most

$$\left(\frac{2^n(k+C)}{tk} + 1\right) t 2^{-n} \mathrm{OPT}(\mathcal{N}_k) = (1 + C/k + t/2^n) \mathrm{OPT}(\mathcal{N}_k).$$

This establishes that the integrality gap is no more than $1 + C/k + t/2^n$. Taking $n \to \infty$, we are done (since the integrality gap is some fixed real, and since t doesn't depend on n). ☐

4 Complex Extreme Points for ($\mathcal{N}'_2$)

Now we give our construction of a new family of extreme points for the TSP subtour relaxation ($\mathcal{N}'_2$); as mentioned earlier, it can be scaled by $k/2$ to give an extreme point for ($\mathcal{N}'_k$) or ($\mathcal{N}_k$), which is relevant to LP-based approaches for k-ECSM.

Let F_i denote the ith Fibonacci number, where $F_1 = F_2 = 1$. For a parameter $t \geq 3$, we denote the extreme point by x^*. The construction is given in the list below and pictured in Figure 2.

- For i from 1 to t, an edge $(2i-1, 2i)$ of x^*-value 1
- For i from 2 to $t-1$, an edge $(1, 2i)$ of x^*-value F_{t-i}/F_t
- An edge $(1, 2t)$ of x^*-value $1/F_t$
- For i from 3 to t, an edge $(2i-3, 2i-1)$ of x^*-value F_{t-i+1}/F_t
- For i from 3 to t, an edge $(2i-4, 2i-1)$ of x^*-value $1 - F_{t-i+2}/F_t$
- An edge $(2,3)$ of x^*-value F_{t-1}/F_t
- An edge $(2t-2, 2t)$ of x^*-value $1 - 1/F_t$

The support graph of x^* has $2t$ vertices and $4t-3$ edges with fractionality $1/F_t$ and maximum degree t. Therefore, in order to prove Theorem 4, it suffices to show that x^* is an extreme point solution.

Proposition 9. *The solution x^* described above is an extreme point solution for ($\mathcal{N}'_2$).*

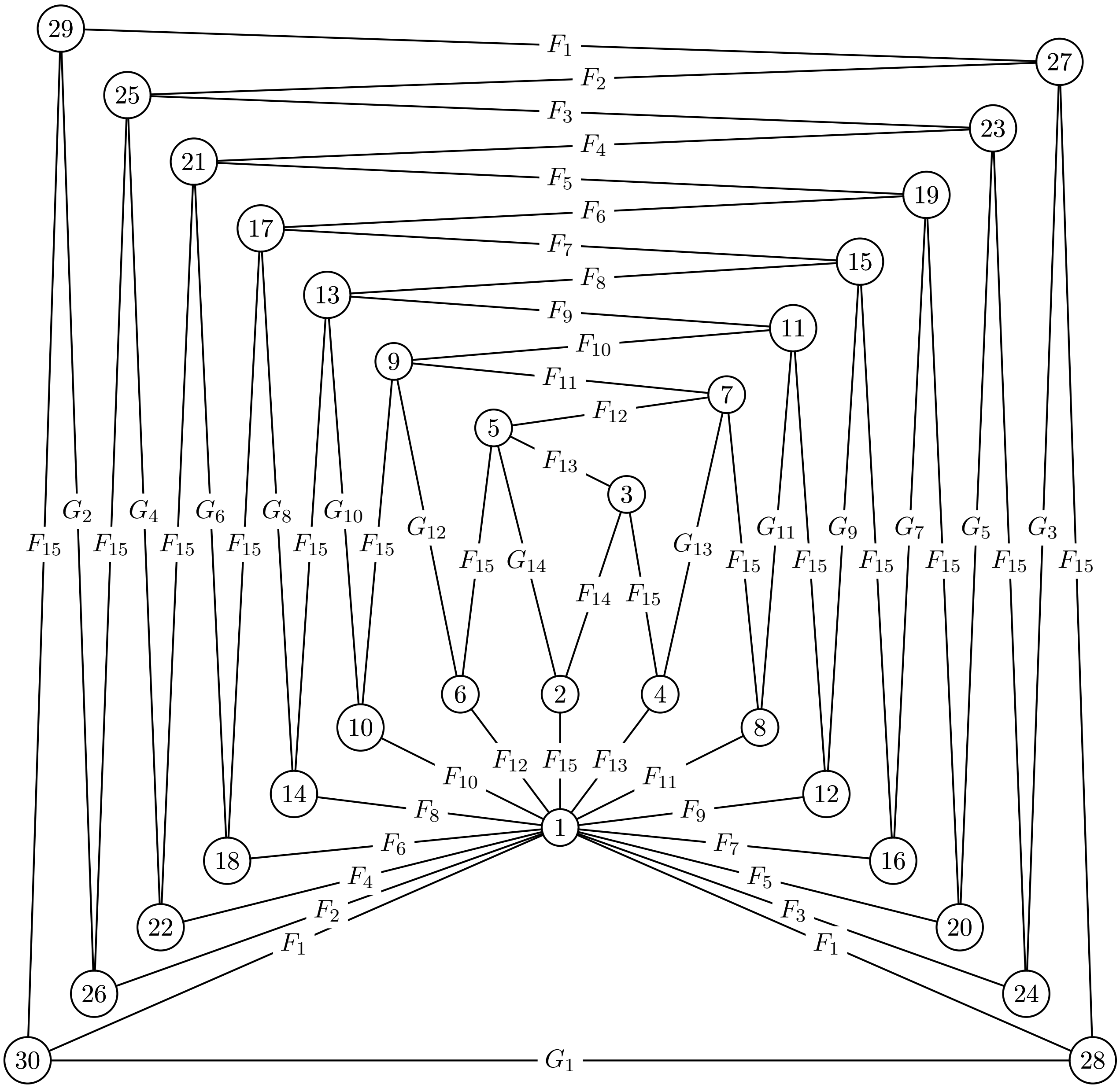

Figure 2: Our new construction of a complex extreme point x^* for the subtour TSP polytope ($\mathcal{N}_2'$), illustrated for $t = 15$. Scaled edge values are shown: the label F_i on an edge e indicates that $x_e^* = F_i/F_t$. The symbol G_i denotes $F_t - F_i$, i.e. an edge e with $x_e^* = 1 - (F_i/F_t)$.

Proof. With foresight, we write down the following family of $4t-3$ sets:

$$\mathcal{L} := \{\{i\}_{i=1}^{2t}, \{2i-1, 2i\}_{i=1}^{t}, \{1, \ldots, 2i\}_{i=2}^{t-2}\}.$$

The plan of our proof is to first show that x^* is the unique solution to $\{x(\delta(T)) = 2 \mid T \in \mathcal{L}\}$. It is easy to verify that x^* indeed satisfies all these conditions, so let us focus on the harder task of showing that x^* is the *only* solution. (Note, we are not assuming that x^* is feasible, so possibly $x^*(\delta(S)) < 2$ for some other sets, but we will deal with this later.)

A set S is *tight* for a solution x if $x(\delta(S)) = 2$. Consider any solution which is tight for all sets in $\mathcal{L}$. We first need a simple lemma. For disjoint sets S, T, let $\delta(S:T)$ denote the set of edges with one end in S and the other in T.

Lemma 10. *For some solution x, if S, T are disjoint tight sets and $S \cup T$ is also tight, then $x(\delta(S:T)) = 1$.*

Proof. We have $\delta(S) = \delta(S:T) \uplus \delta(S:V\backslash S\backslash T)$ and $\delta(T) = \delta(S:T) \uplus \delta(T:V\backslash S\backslash T)$. Also, $\delta(S \cup T) = \delta(S:V\backslash S\backslash T) \uplus \delta(T:V\backslash S\backslash T)$. Thus $2 = x(\delta(S)) + x(\delta(T)) - x(\delta(S\cup T)) = 2x(\delta(S:T))$. □

Consider a hypothetical solution x with $x(\delta(S)) = 2, \forall x \in \mathcal{L}$. The lemma shows all edges $\{2i-1, 2i\}_{i=1}^{t}$ have x-value 1 (take $S = \{2i-1\}, T = \{2i\}$). Define y_i equal to $x_{(2i+1,2i+3)}$ for i from 1 to $t-2$. The degree constraint at 3 (i.e., $x(\delta(3)) = 2$) forces $x_{(2,3)} = 1 - y_1$. The degree constraint at 2 forces $x_{(5,2)} = y_1$. Note $\{1, \ldots, 2t-2\}$ is tight since this set has the same constraint as $\{2t-1, 2t\}$. For i from 1 to $t-2$, note that the sets $\delta(\{1, \ldots, 2i\} : \{2i+1, 2i+2\})$ and $\delta(2i+1)$ differ only in that the former contains the edge $(2i+2, 1)$ and the latter contains the edges $\{(2i+1, 2i+2), (2i+1, 2i+3)\}$. Thus, using the lemma and degree constraint at $2i+1$, we see $x_{(2i+2,1)} + x_{(2i+1,2i+3)} = y_i$. The degree constraint at $2i+2$ then forces $x_{(2i+2,2i+5)} = 1 - y_i$ for $1 \le i \le t-3$. The degree constraint at $2t-2$ forces $x_{(1,2t-2)} = 1 - y_{t-2}$; the degree constraint at $2t$ forces $x_{(1,2t)} = y_{t-2}$. The degree constraint at $2t-1$ forces $y_{t-2} = y_{t-3}$, and the degree constraint at $2i+5$ forces $y_i = y_{i+1} + y_{i+2}$ for i from 1 to $t-4$; together this shows $y_i = F_{t-1-i} \cdot y_{t-2}$ for i from $t-4$ to 1 by induction. The degree constraint at 5 forces $2y_1 + y_2 = 1$, so $(2F_{t-2} + F_{t-3})y_{t-2} = 1$ and consequently $y_{t-2} = 1/F_t$. Thus we conclude that $x = x^*$, as desired.

Now, we show x^* is feasible using standard uncrossing arguments, plus the fact that $|\mathcal{L}| = 4t-3$. In $(\mathcal{N}_2')$, the constraints (2) for S and $V\backslash S$ are equivalent. Therefore, if we fix any root vertex $r \in V$, we may keep only the constraints for sets S not containing r without changing the LP. Correspondingly, we change $\mathcal{L}$ by complementing the sets that contain r, and it is easy to see $\mathcal{L}$ is a laminar family on $V\backslash\{r\}$. (This is along the lines of the standard argument by Cornuéjols et al. [16].) In fact $\mathcal{L}$ is a maximal laminar family, since any laminar family of nonempty subsets of X contains at most $2|X|-1$ elements, for any set X.

Finally, suppose for the sake of contradiction that x^* is not feasible, so there is a set S, with $r \notin S$, having $x^*(\delta(S)) < 2$. Clearly $S \notin \mathcal{L}$. Two sets S, T, neither containing r, *cross* if all three of $S\backslash T$, $T\backslash S$, and $T \cap S$ are non-empty. Take S with $x^*(\delta(S)) < 2$ such that S crosses a minimal number of sets in $\mathcal{L}$. If S crosses zero sets in $\mathcal{L}$, then $\mathcal{L} \cup \{S\}$ is laminar, but this is a contradiction since $S \notin \mathcal{L}$ and, crucially, $\mathcal{L}$ was maximal. Otherwise, set S crosses some tight set $T \in \mathcal{L}$, then since

$$2 + 2 > x^*(\delta(S)) + x^*(\delta(T)) \ge x^*(\delta(S \cup T)) + x^*(\delta(S \cap T)),$$

either $x^*(\delta(S \cup T)) < 2$ or $x^*(\delta(S \cap T)) < 2$. It is easy to verify that both $S \cup T$ and $S \cap T$ cross fewer sets of $\mathcal{L}$ than S, contradicting our choice of S. □

4.1 Relation to Asymmetric TSP

Asymmetric TSP is the analogue of TSP for directed graphs: we are given a metric directed cost function on the complete digraph (V, A), and seek a min-cost directed Hamiltonian cycle. Recently Asadpour et al. [1] obtained a breakthrough $O(\log n / \log \log n)$ approximation for this problem; its analysis uses the fact that extreme points of the natural LP relaxation

$$\{y \in \mathbb{R}_+^A : \forall \varnothing \neq U \subsetneq V, y(\delta^{\text{out}}(U)) \geq 1\} \tag{$\mathcal{A}$}$$

have denominator bounded by $2^{O(n \ln n)}$. Our undirected construction implies that for this directed variant, the extreme points attain denominator at least $2^{\Omega(n)}$; the proof is given in Appendix C.

Proposition 11. *For every even $n \geq 6$ there are extreme points for ($\mathcal{A}$) on n vertices with fractionality $1/F_{n/2}$ or smaller (and hence denominator at least $F_{n/2}$).*

References

[1] A. Asadpour, M. Goemans, A. Madry, S. O. Gharan, and A. Saberi. An $O(\log n / \log \log n)$-approximation algorithm for the asymmetric traveling salesman problem. In *Proc. 21st SODA*, 2010.

[2] J. Bang-Jensen. Problems and conjectures concerning connectivity, paths, trees and cycles in tournament-like digraphs. *Discrete Mathematics*, 309(18):5655–5667, 2009.

[3] J. Bang-Jensen and S. Thomassé. Highly connected hypergraphs containing no two edge-disjoint spanning connected subhypergraphs. *Discrete Appl. Math.*, 131:555–559, 2003.

[4] J. Bang-Jensen and A. Yeo. Decomposing k-arc-strong tournaments into strong spanning subdigraphs. *Combinatorica*, 24(3):331–349, 2004.

[5] N. Bansal, N. Korula, V. Nagarajan, and A. Srinivasan. On k-column sparse packing programs. In *Proc. 14th IPCO*, pages 369–382, 2010. Preliminary version at arXiv:0908.2256.

[6] G. Benoit and S. Boyd. Finding the exact integrality gap for small traveling salesman problems. *Math. Oper. Res.*, 33(4):921–931, 2008.

[7] H.-J. Böckenhauer, D. Bongartz, J. Hromkovič, R. Klasing, G. Proietti, S. Seibert, and W. Unger. On the hardness of constructing minimal 2-connected spanning subgraphs in complete graphs with sharpened triangle inequality. *Theor. Comput. Sci.*, 326(1-3):137–153, 2004.

[8] S. Boyd and A. Cameron. A 3/2-approximation algorithm for the multi-two-edge connected subgraph problem. Technical Report TR-2008-01, SITE, University of Ottawa, 2008.

[9] S. C. Boyd and W. R. Pulleyblank. Optimizing over the subtour polytope of the travelling salesman problem. *Math. Program.*, 49:163–187, 1991.

[10] R. D. Carr and S. Vempala. On the Held-Karp relaxation for the asymmetric and symmetric traveling salesman problems. *Math. Program.*, 100(3):569–587, 2004. Preliminary version appeared as "Towards a 4/3 approximation for the asymmetric traveling salesman problem" in *Proc. 11th SODA*, pages 116–125, 2000.

[11] C. Chekuri, A. Ene, and N. Korula. Unsplittable flow in paths and trees and column-restricted packing integer programs. In *Proc. 12th APPROX*, pages 42–55, 2009.

[12] C. Chekuri, M. Mydlarz, and F. B. Shepherd. Multicommodity demand flow in a tree and packing integer programs. *ACM Trans. Algorithms*, 3(3):27, 2007. Preliminary version appeared in *Proc. 30th ICALP*, pages 410–425, 2003.

[13] J. Cheriyan and R. Thurimella. Approximating minimum-size k-connected spanning subgraphs via matching. *SIAM J. Comput.*, 30(2):528–560, 2000. Preliminary version appeared in *Proc. 37th FOCS*, pages 292–301, 1996.

[14] K. Cheung. *Subtour Elimination Polytopes and Graphs of Inscribable Type.* PhD thesis, University of Waterloo, 2003.

[15] M. Chlebík and J. Chlebíková. Complexity of approximating bounded variants of optimization problems. *Theor. Comput. Sci.*, 354(3):320–338, 2006. Preliminary version appeared in *Proc. 14th FCT*, pages 27–38, 2003.

[16] G. Cornuéjols, D. Naddef, and J. Fonlupt. The traveling salesman problem on a graph and some related integer polyhedra. *Math. Programming*, 33:1–27, 1985.

[17] G. Even, J. Feldman, G. Kortsarz, and Z. Nutov. A 1.8 approximation algorithm for augmenting edge-connectivity of a graph from 1 to 2. *ACM Trans. Algorithms*, 5(2):1–17, 2009.

[18] C. G. Fernandes. A better approximation ratio for the minimum size k-edge-connected spanning subgraph problem. *J. Algorithms*, 28(1):105–124, 1998. Preliminary version appeared in *Proc. 8th SODA*, pages 629–638, 1997.

[19] G. N. Frederickson and J. JáJá. Approximation algorithms for several graph augmentation problems. *SIAM J. Comput.*, 10(2):270–283, 1981.

[20] G. N. Frederickson and J. JáJá. On the relationship between biconnectivity augmentation and the traveling salesman problem. *Theoretical Computer Science*, 19:189–201, 1982.

[21] H. N. Gabow and S. Gallagher. Iterated rounding algorithms for the smallest k-edge-connected spanning subgraph. In *Proc. 19th SODA*, pages 550–559, 2008.

[22] H. N. Gabow, M. X. Goemans, É. Tardos, and D. P. Williamson. Approximating the smallest k-edge connected spanning subgraph by LP-rounding. *Networks*, 53(4):345–357, 2009. Preliminary version appeared in *Proc. 16th SODA*, pages 562–571, 2005.

[23] N. Garg, V. V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18(1):3–20, May 1997. Preliminary version appeared in *Proc. 20th ICALP*, pages 64–75, 1993.

[24] M. X. Goemans. Minimum bounded degree spanning trees. In *Proc. 47th FOCS*, pages 273–282, 2006.

[25] M. X. Goemans and D. Bertsimas. Survivable networks, linear programming relaxations and the parsimonious property. *Math. Programming*, 60:145–166, 1993. Preliminary version appeared in *Proc. 1st SODA*, pages 388-396, 1990.

[26] K. Jain. A factor 2 approximation algorithm for the generalized Steiner network problem. *Combinatorica*, 21(1):39–60, 2001. Preliminary version appeared in *Proc. 39th FOCS*, pages 448–457, 1998.

[27] S. Khuller and U. Vishkin. Biconnectivity approximations and graph carvings. *J. ACM*, 41(2):214–235, 1994. Preliminary version appeared in *Proc. 24th STOC*, pages 759–770, 1992.

[28] J. Könemann, O. Parekh, and D. Pritchard. Max-weight integral multicommodity flow in spiders and high-capacity trees. In *Proc. 6th WAOA*, pages 1–14, 2008.

[29] G. Kortsarz, R. Krauthgamer, and J. R. Lee. Hardness of approximation for vertex-connectivity network design problems. *SIAM J. Comput.*, 33(3):704–720, 2004. Preliminary version appeared in *Proc. 5th APPROX*, pages 185-199, 2002.

[30] G. H. J. Meredith. Regular n-valent n-connected non-hamiltonian non-n-edge colorable graphs. *J. Combin. Theory Ser. B*, 14:55–60, 1973.

[31] C. H. Papadimitriou and M. Yannakakis. The traveling salesman problem with distances one and two. *Math. Oper. Res.*, 18:1–11, 1993.

[32] D. Pritchard. Approximability of sparse integer programs. In *Proc. 17th ESA*, pages 83–94, 2009. Preliminary version appears at arXiv:0904.0859.

[33] D. Pritchard. *Linear Programming Tools & Approximation Algorithms for Combinatorial Optimization*. PhD thesis, University of Waterloo, 2009.

[34] D. Pritchard. k-edge-connectivity: Approximation and LP relaxation. arXiv:1004.1917, 2010.

[35] M. Singh and L. C. Lau. Approximating minimum bounded degree spanning trees to within one of optimal. In *Proc. 39th STOC*, pages 661–670, 2007.

[36] L. Wolsey. Heuristic analysis, linear programming and branch and bound. *Math. Programming Study*, 13:121–134, 1980.

A Hardness of Path-Cover-of-Tree

Our arguments are based on those of [19], and also inspired by [23], who used the same approach to prove APX-hardness of a related packing problem. We reduce from minimum set cover in 3-uniform, 2-regular hypergraphs — i.e. set cover with sets of size 3, each set appearing in exactly 2 sets — which is equivalent to vertex cover in cubic graphs. The best known inapproximability ratio for this problem is about $\frac{100}{99}$, due to Chlebík and Chlebíková [15].

Here is the reduction. Let the instance of 3-uniform, 2-regular set cover be $(J, \mathcal{K})$ where J is the ground set and $\mathcal{K}$ is the family of triples from J. Let $k = |\mathcal{K}|$ (so $|J| = 3k/2$) and denote the sets by $K_i = \{a[i], b[i], c[i]\}$ for $1 \leq i \leq k$ (so $a[i], b[i], c[i]$ are elements of J). The tree T we construct for the Path-Cover-of-Tree instance has a root vertex r, a vertex v_j for each $j \in J$, and two vertices p_i, q_i for $1 \leq i \leq k$; T has an edge $\{r, v_j\}$ for every $j \in J$, and the two edges $\{v_{a[i]}, p_i\}, \{v_{a[i]}, q_i\}$ for $1 \leq i \leq k$. Finally, we define the set X to have the following $3k$ pairs: $\{p_i, q_i\}, \{p_i, v_{b[i]}\}, \{q_i, v_{c[i]}\}$ for $1 \leq i \leq k$.

Claim 12. $\mathrm{OPT}(T, X) = k + \mathrm{OPT}(J, \mathcal{K})$.

(We speak of PATH-COVER-OF-TREE in terms of covering $E(T)$ instead of as a 2-connectivity problem.)

Proof. Let $\{K_i \mid i \in I\}$ be an optimal set cover, i.e. a J-covering subfamily of $\mathcal{K}$ such that $|I| = \mathrm{OPT}(J, \mathcal{K})$. Define $Y \subset X$ as follows: if $i \in I$ we put $\{p_i, v_{b[i]}\}$ and $\{q_i, v_{c[i]}\}$ into Y, and if $i \notin I$ we put $\{p_i, q_i\}$ into Y. In either case, the corresponding paths in T cover the edges incident to p_i and q_i; and it is not hard to see that since I is a set cover, all edges incident to r are also covered. This proves $\mathrm{OPT}(T, X) \leq 2|I| + (k - |I|) = k + \mathrm{OPT}(J, \mathcal{K})$.

The reverse inequality is similar. The only step needing pause is to consider whether (T, X) always has an optimal solution Y of the form generated by the above mapping (since then it can be reversed). Indeed, if Y contains one or fewer of the 3 pairs $\{p_i, q_i\}, \{p_i, v_{b[i]}\}, \{q_i, v_{c[i]}\}$ then it must contain $\{p_i, q_i\}$ to cover the edges incident to p_i and q_i; and if Y contains two or more of the pairs, we can adjust such pairs to $\{p_i, v_{b[i]}\}$ and $\{q_i, v_{c[i]}\}$ without increasing $|Y|$ and without causing an edge of T to become uncovered. □

Here are the calculations that show the reduction works. We have $\mathrm{OPT}(J, \mathcal{K}) \geq k/2$ (since we need to cover $3k/2$ points by triples), and by the result of [15], no polynomial-time algorithm can determine $\mathrm{OPT}(J, \mathcal{K})$ within additive error $\frac{k}{2\cdot 99}$ on all instances, unless $\mathsf{P} = \mathsf{NP}$. Hence, no polynomial-time algorithm can determine $\mathrm{OPT}(T, X)$ within the same additive error. Finally, since $\mathrm{OPT}(T, X) \leq 2k$, we get an inapproximability ratio of $1 + \frac{k}{2\cdot 99}/2k = 1 + \frac{1}{396}$ for PATH-COVER-OF-TREE. However, if we actually look at the gap instances of [15], the same calculations give a slightly stronger ratio of $1 + \frac{1}{292.4}$.

B Proof of Proposition 6

We may assume F is minimal, i.e. that deleting any edge from (V, F) leaves a non-2-edge-connected graph. This implies there are no parallel triples. Next, suppose there is a parallel pair between some vertices u and v. If there is any u-v path not using a uv edge, it is easy to see that deleting one of the parallel uv edges contradicts minimality. Therefore we may assume uv is a cut edge (bridge) of the simplification of (V, F); call this the *bridge assumption*.

Since the graph is connected and $|V| \geq 3$, at least one of u or v (say u WOLOG) has another neighbour w. By the bridge assumption v is not adjacent to w. We will argue that the set F' obtained by deleting a uv edge, a uw edge, and adding a vw edge, is still 2-edge-connected. Iterating this operation we are done (since the cost does not increase and the number of parallel pairs decreases each time).

Since (V, F) is 2-edge-connected, it has a u-w path P not using the deleted uw edge. By the bridge assumption, P does not use any uv edge. Note that $|\delta_{F'}(S)| < |\delta_F(S)|$ only if S contains v and w but not u (or vice-versa). But then $\delta_{F'}(S)$ contains the remaining uv edge and at least one edge from P. So $|\delta_{F'}(S)| \geq 2$ for all $\varnothing \neq S \subsetneq V$ and we are done.

C Proof of Proposition 11

The key is to note that $(\mathcal{N}_2)$ equals the projection of $(\mathcal{A})$ to $\mathbb{R}^E_+$ obtained by setting $x_{\{u,v\}} = y_{(u,v)} + y_{(v,u)}$ for all $\{u, v\} \in \binom{V}{2}$ (call this map *dropping directions*). One direction is evident:

given y, it has value at least 1 both coming into and coming out of every nontrivial cut set U, hence its undirected image x has value at least 2 spanning the cut it defines, i.e. $x(\delta(U)) \geq 2$. Conversely, to show that for every $x \in (\mathcal{N}_2)$, there is a $y \in (\mathcal{A})$ of this type, just assign $y_{(u,v)} = y_{(v,u)} = x_{\{u,v\}}/2$ for all $\{u, v\} \in \binom{V}{2}$.

Now we prove Proposition 11. Consider x^* given by the construction, and consider the set of all y in $(\mathcal{A})$ such that y becomes x^* when dropping directions. The argument in the previous paragraph establishes that this set is nonempty, and it is not hard to see this set is a face of $(\mathcal{A})$ since x^* is an extreme point of $(\mathcal{N}_2)$. Finally, let y^* be any extreme point of this face. Our construction includes an edge e with $x^*_e = 1/F_{n/2}$, hence at least one of the two arcs corresponding to e has y^*-value in $(0, 1/F_{n/2}]$, giving the claimed result.

As a remark, the above proof leaves open the possibility that the extreme points y^* for $(\mathcal{A})$ could have *strictly worse* fractionality than $1/F_{n/2}$, but according to our computational experiments for $n = 6, 8$, the worst-case fractionality for such y^* is exactly $1/F_{n/2}$.