Approximating the Smallest k-Edge Connected Spanning Subgraph by LP-Rounding

Harold N. Gabow [*] Michel X. Goemans [†] Éva Tardos [‡] David P. Williamson [§]

September 1, 2005

Abstract

The *smallest k-ECSS problem* is, given a graph along with an integer k, find a spanning subgraph that is k-edge connected and contains the fewest possible number of edges. We examine a natural approximation algorithm based on rounding an LP solution. A tight bound on the approximation ratio is $1 + 3/k$ for undirected graphs with $k > 1$ odd, $1 + 2/k$ for undirected graphs with k even, and $1 + 2/k$ for directed graphs with k arbitrary. Using iterated rounding improves the first upper bound to $1+2/k$. On the hardness side we show that for some absolute constant $c > 0$, for any integer $k \geq 2$ ($k \geq 1$), a polynomial-time algorithm approximating the smallest k-ECSS on undirected (directed) multigraphs to within ratio $1 + c/k$ would imply $P = NP$.

1 Introduction

A *k-ECSS* of a k-edge connected graph is a k-edge connected spanning subgraph. We are interested in finding a smallest k-ECSS, i.e., one that has the fewest possible number of edges. This is a basic problem in network design. When the graph is undirected and $k = 1$ the solution is simply a spanning tree. For the other cases, it is intuitively plausible that the problem becomes easier to approximate as k increases, and several previous results support this belief. We prove this conjecture, presenting polynomial-time algorithms with approximation ratio $1 + 2/k$. We start by reviewing the previous results. Throughout this paper n and m denote the number of vertices and edges of the given graph, respectively.

First consider simple graphs. Cheriyan and Thurimella [1] gave a $1 + 2/(k + 1)$ approximation algorithm for the smallest k-ECSS of a simple undirected graph. Their approach is based on an analog of a theorem of Mader on k-node connected graphs. The approach also extends to simple digraphs, achieving approximation ratio $1 + 4/\sqrt{k}$. This work gave striking evidence that the problem gets easier for larger k. However the approximation bounds depend crucially on the assumption of simplicity. (The bounds become 2 when parallel edges are allowed [5].) [6] improves the bound for the simple digraph algorithm to $1 + \sqrt{2/k} + O(1/k)$, an essentially tight bound.

[*]Department of Computer Science, University of Colorado at Boulder, Boulder, CO 80309. E-mail: hal@cs.colorado.edu

[†]MIT, 77 Mass. Ave., Room 2-351, Cambridge, MA 02139. Research supported by NSF grant CCR-0098018. E-mail: goemans@math.mit.edu

[‡]Department of Computer Science, Cornell University, Ithaca, NY 14853. Research supported by ONR grant N00014-98-1-0589. E-mail: eva@cs.cornell.edu

[§]School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY 14853. E-mail: dpw@cs.cornell.edu

Regarding hardness, Khuller, Raghavachari and Young [13] show the smallest directed 1-ECSS problem is MAXSNP-hard, and Fernandes [3] shows the same for smallest undirected 2-ECSS.

Next consider multigraphs. Khuller and Raghavachari [12] were the first to go beyond the naive performance bound of 2 (on any type of graph, simple or multi-). Their algorithm works on undirected multigraphs and is based on depth-first search. They prove a performance bound < 1.85. Gabow [5] improved the analysis to $< 1 + \sqrt{1/e} < 1.61$, the best currently known combinatorial bound. [6] achieves the performance bound $2 - 1/3k$ for any multidigraph. Karger [10] gave a randomized algorithm for smallest k-ECSS, based on sampling and randomized rounding. It works on undirected multigraphs and has performance ratio $1 + O(\sqrt{(\log n)/k})$. This is the only known bound that improves with k when parallel edges are allowed.

We analyze an algorithm for the smallest k-ECSS problem based on linear programming. The algorithm simply rounds up an extreme point of the LP corresponding to the natural integer linear program. This algorithm achieves approximation ratio $1 + 2/k$ for digraphs and arbitrary k, as well as for undirected graphs and even k. For undirected graphs and odd $k > 1$ the approximation ratio is $1 + 3/k$. This shows the smallest k-ECSS problem gets easier to approximate for larger k. These upper bounds are tight to within lower order terms. We also show Jain's iterated rounding algorithm [9] improves the performance bound for the odd undirected case to $1 + 2/k$. Taken together these are the best-known bounds for the smallest k-ECSS problem. They also show the natural linear program for smallest k-ECSS has integrality gap at most $1 + 2/k$ for both directed and undirected graphs. On the hardness side we show that for some absolute constant $c > 0$, for any integer $k \geq 2$ ($k \geq 1$), a polynomial-time algorithm approximating the smallest k-ECSS on undirected (directed) multigraphs to within ratio $1 + c/k$ would imply $P = NP$. This is achieved by extending the reductions of [13, 3] to show that for any integer $k \geq 2$ ($k \geq 1$), the smallest k-ECSS on undirected (directed) multigraphs is MAXSNP-hard.

Section 2 states the LP-rounding algorithm and sketches short proofs of a $1+4/k$ bound on the approximation ratio. Section 3 derives our tight upper bounds. Section 4 gives the lower bound examples. Section 5 gives the hardness results. We conclude this section with some terminology: A fraction written a/bc is an abbreviation for $a/(bc)$. The expression $a \bmod b$ where both arguments are positive real values has the obvious interpretation $(a - \lfloor a/b \rfloor b)$. We consider real vectors whose components are indexed by a set of edges, e.g., the edge set of the given graph. Boldface variables denote vectors, e.g., $\boldsymbol{x}$ with components x_e. If F is a set of edges then $\boldsymbol{x}(F)$ denotes $\sum_{e \in F} x_e$. Similarly for a function $f : X \to \mathbf{R}$ and $S \subseteq X$, $f(S)$ denotes $\sum_{s \in S} f(s)$. All graphs that we consider allow parallel edges.

2 Rounding Algorithm

The problem of finding a smallest k-ECSS of graph $G = (V, E)$ is modelled by the integer linear program

$$\begin{array}{lll} \text{minimize} & \boldsymbol{x}(E) & \\ \text{subject to} & d_x(S) \geq k & \emptyset \subset S \subset V \\ & x_e \in \{0, 1, \ldots, u_e\} & e \in E. \end{array}$$

If G is undirected then $d_x(S)$ denotes the degree of S when edges are weighted by $\boldsymbol{x}$, i.e., $d_x(S) = \sum\{x_e : \text{edge } e \text{ is incident to } S\}$. If G is directed "degree" becomes "in-degree", i.e., $d_x(S) = \sum\{x_e : \text{edge } e \text{ enters } S\}$. Finally u_e is an integer representing the multiplicity of edge e in G. Without loss of generality assume G is k-edge connected.

To approximate the smallest k-ECSS relax this system to the linear program

$$\text{(1)} \qquad \begin{array}{lll} \text{minimize} & \boldsymbol{x}(E) & \\ \text{subject to} & d_x(S) \geq k & \emptyset \subset S \subset V \\ & 0 \leq x_e \leq u_e & e \in E. \end{array}$$

Find an optimum extreme point. From now on $\boldsymbol{x}$ denotes this extreme point. Convert $\boldsymbol{x}$ to an integral solution by rounding each fractional part up to 1.

This algorithm clearly produces a k-ECSS. It can be implemented in polynomial or strongly polynomial time, for both directed and undirected graphs. (This is straightforward if we use an LP equivalent to (1) but of polynomial size. Such an LP can be based on interpreting x_e as the capacity of edge e in a flow network, see [9, 15].) So the only issue is determining the approximation ratio. We will sketch short derivations of an upper bound $1 + 4/k$ on this ratio.

Suppose first that the graph is undirected. Let B be the set of variables x_e that the extreme point sets to its lower or upper bound, i.e., $x_e \in \{0, u_e\}$. Since $\boldsymbol{x}$ is extreme some subset of the remaining constraints uniquely determines the remaining values of $\boldsymbol{x}$. That is, for some family $\mathcal{L}$ of subsets of V, if we fix the values of variables $x_e \in B$ then the remaining values form the unique solution to the system of equations $d_x(S) = k$, $S \in \mathcal{L}$. Jain [9] showed that $\mathcal{L}$ can be chosen to be laminar. ([2] gives a similar argument for the case $k = 2$.) A laminar family on a ground set of n elements contains $< 2n$ sets. Hence there are $< 2n$ constraints in our system of equations. Since the solution is unique, the number of variables equals the number of constraints (assuming $\mathcal{L}$ is minimal). Thus $< 2n$ variables are not in B. Since every variable of B is integral, $< 2n$ variables have a fractional part. So rounding up all fractional parts increases the objective value $\boldsymbol{x}(E)$ by $< 2n$. In any k-ECSS any vertex has degree $\geq k$, so there are $\geq kn/2$ edges. Thus the approximation ratio is $< 1 + 2n/(kn/2) = 1 + 4/k$.

A similar analysis holds for the directed case. Choose an arbitrary vertex r and define B as before. The degree constraints of the integer program amount to requiring each set S, $\emptyset \subset S \subseteq V - r$, to have in-degree and out-degree both $\geq k$. Since $\boldsymbol{x}$ is extreme there are families of subsets of V, say $\mathcal{L}_1$ and $\mathcal{L}_2$, such that if we fix the values of all variables $x_e \in B$ then the remaining values of $\boldsymbol{x}$ correspond to the unique solution to the system of equations that state each $S \in \mathcal{L}_1$ ($\mathcal{L}_2$) has in-degree (out-degree) exactly k, respectively. (These in- and out-degrees are computed using the weighting $\boldsymbol{x}$.) The same argument as [9] shows that $\mathcal{L}_1$ and $\mathcal{L}_2$ can both be chosen to be laminar. There are $< 2n + 2n = 4n$ variables in the system. Hence $< 4n$ variables have a fractional part. Rounding increases $\boldsymbol{x}(E)$ by $< 4n$. Any k-ECSS contains $\geq kn$ edges so the approximation ratio is $< 1 + 4n/kn = 1 + 4/k$.

3 Upper Bound Analysis

This section improves the analysis of the simple rounding algorithm and eventually shows the integrality gap of (1) is $\leq 1 + 2/k$.

It is convenient to transform the problem (in a fashion similar to [9]). Enlarge G by adding a parallel copy of each edge e that has both an integral and a fractional part, i.e., both $\lfloor x_e \rfloor$ and (x_e mod 1) are positive. Assign the value $\lfloor x_e \rfloor$ to one copy of e and (x_e mod 1) to the other. We continue to use the vector $\boldsymbol{x}$ to refer to the values in this new graph. In this new graph we view the linear system $d_x(S) = k$, $S \in \mathcal{L}$ as having the fractional values x_e as the unknowns. The integral values x_e are interpreted as known constants. Clearly this linear system still has a unique solution. Furthermore we can assume our linear system is minimal. (Note that when we switch to the new graph, some equations of the system can become redundant because of edges with no fractional part.) We continue to use $\mathcal{L}$ to refer to this new minimal laminar family.

We start by analyzing the rounding algorithm for undirected graphs. Consider a set $S \in \mathcal{L}$. Say that S *owns vertex* v if S is the smallest set of $\mathcal{L}$ containing v (as in [15]). Furthermore say that S *owns edge* e if S is the smallest set of $\mathcal{L}$ containing both ends of e. Thus a vertex or edge is owned at most once. Let $\phi(S)$ denote the total weight in $\boldsymbol{x}$ of all fractional edges owned by S,

$$\phi(S) = \sum \{x_e : 0 < x_e < 1 \text{ and } S \text{ owns } e\}.$$

Lemma 3.1 *Any set $S \in \mathcal{L}$ that does not own a vertex has $\phi(S)$ a positive half-integer (integer) when k is odd (even) respectively.*

Proof: Assume S does not own any vertex. Hence S has at least two children in $\mathcal{L}$, say children $S_i, 1 \leq i \leq r$, $r > 1$. (Throughout the paper we apply tree terminology to laminar families in the obvious way.) The sets S_i form a partition of S.

Since an edge entering S_i either enters S or leaves another set S_j we get

$$\sum_{i=1}^{r} d_x(S_i) = d_x(S) + 2 \sum_{i<j} d_x(S_i, S_j).$$

Since $d_x(S_i) = d(S) = k$ this says

$$rk = k + 2 \sum \{x_e : S \text{ owns } e\}.$$

Since the above summation equals $\phi(S)$ plus various integral values x_e, rearranging gives

$$(r-1)k \equiv 2\,\phi(S) \pmod{2}.$$

This shows $\phi(S)$ is a half integer if k is odd and an integer if k is even.

It remains to show $\phi(S)$ is positive, i.e., S owns a fractional edge. In the opposite case the fractional edges leaving the sets S_i, $1 \leq i \leq r$ form a partition of the fractional edges leaving S. This makes the constraint $d_x(S) = k$ the sum of the constraints of the children S_i. But this contradicts the assumed minimality of $\mathcal{L}$. □

Theorem 3.2 *LP-rounding achieves an approximation guarantee of $1+3/k$ ($1+2/k$) on undirected graphs with k odd (even) respectively.*

Proof: Rounding up a fractional weight x_e increases the objective function by $1 - x_e$. There are exactly $|\mathcal{L}|$ fractional edges, since the linear system $d_x(S) = k$, $S \in \mathcal{L}$ uniquely determines the fractional weights. Thus rounding increases the objective function by a total amount $\leq |\mathcal{L}| - \phi(\mathcal{L})$. (Recall that our notational convention makes $\phi(\mathcal{L}) = \sum\{\phi(S) : S \in \mathcal{L}\}$. Also, we write inequality above since a fractional edge need not be owned by a set.) We will show

$$|\mathcal{L}| - \phi(\mathcal{L}) \leq \Delta n$$

where Δ equals $3/2$ (1) for k odd (even) respectively. This implies the lemma since it shows the approximation ratio is $\leq 1 + \Delta n/(kn/2) = 1 + 2\Delta/k$.

Suppose that k is even. Every set of $\mathcal{L}$ owns either a vertex or fractional edges of weight $\phi(S) \geq 1$ (Lemma 3.1). Thus

$$|\mathcal{L}| \leq n + \phi(\mathcal{L})$$

as desired.

Suppose k is odd. Using $2\phi(S) \geq 1$ in the argument for k even gives $|\mathcal{L}| \leq n + 2\,\phi(\mathcal{L})$. Together with $|\mathcal{L}| \leq 2n$ this implies

$$|\mathcal{L}| - \phi(\mathcal{L}) \leq \min\{n + \phi(\mathcal{L}),\ 2n - \phi(\mathcal{L})\}.$$

The right-hand side achieves its maximum value $3n/2$ when $\phi(\mathcal{L}) = n/2$. This gives the desired bound. □

We turn to digraphs. All of our digraphs are weighted, and we allow the weight of an edge to have arbitrary sign, positive, negative or 0. We use ρ and δ denote the in-degree and out-degree functions respectively. These functions count edges according to their weight, e.g., $\rho(S)$ is the total weight of all edges entering S.

The uncrossing argument of Jain [9] can be combined with an argument of Frank [4] to get a single laminar family $\mathcal{L}$ for digraphs. This is proved in [14] so we only state the result. Recall that we have switched to the graph where each value x_e is either integral or fractional (strictly between 0 and 1). Let H be the digraph containing the integral edges, weighted according to $\boldsymbol{x}$. $\mathcal{L}$ will consist of two families of vertex sets $\mathcal{I}$, $\mathcal{O}$ having a corresponding system (2) of linear equations in unknowns y_e where e ranges over the fractional edges of $\boldsymbol{x}$:

$$\begin{array}{rcll} \rho_y(S) & = & k - \rho_H(S) & S \in \mathcal{I}, \\ \delta_y(S) & = & k - \delta_H(S) & S \in \mathcal{O}. \end{array} \tag{2}$$

It is possible for a set to belong to both families $\mathcal{I}$ and $\mathcal{O}$. Let $\hat{\boldsymbol{x}}$ be the vector $\boldsymbol{x}$ restricted to its fractional edges. The result of [14] states:

Lemma 3.3 *There is a laminar family of sets $\mathcal{L} = \mathcal{I} \cup \mathcal{O}$ such that the linear system (2) in unknowns $\boldsymbol{y}$ has $\hat{\boldsymbol{x}}$ as its unique solution.*

The above family is not entirely adequate for our purposes. Before refining it recall two simple facts about weighted digraphs.

Lemma 3.4 *In an arbitrary weighted digraph G consider a vertex set S partitioned into subsets S_i, $i = 1, \ldots, r$.*

(i) $\delta(S) + \sum_{i=1}^{r} \rho(S_i) = \rho(S) + \sum_{i=1}^{r} \delta(S_i)$.

(ii) *If G is k-edge connected then $\rho(S_1) = \ldots = \rho(S_r) = \delta(S) = k$ implies $\delta(S_1) = \ldots = \delta(S_r) = \rho(S) = k$.*

Proof: (i) Both sides of the equation count the total weight of all edges incident to some set S_i. (Arbitrarily signed weights are allowed.)

(ii) Using the hypothesis, (i) and k-edge connectedness gives

$$(r+1)k = \delta(S) + \sum_{i=1}^{r} \rho(S_i) = \rho(S) + \sum_{i=1}^{r} \delta(S_i) \geq (r+1)k.$$

This implies equality holds throughout, giving the conclusion of (ii). □

Choose $\mathcal{I}$ and $\mathcal{O}$ so together they give a set of constraints satisfying Lemma 3.3 and in addition they are *light*, i.e., they minimize the quantity $\sum\{|S| : S \in \mathcal{I} \cup \mathcal{O}\}$. This sum counts a set $S \in \mathcal{I} \cap \mathcal{O}$ twice. Note that lightness implies $\mathcal{I}$ and $\mathcal{O}$ form a minimal family.

Define ownership of edges exactly as before: A set $S \in \mathcal{L}$ *owns edge* e if S is the smallest set of $\mathcal{L}$ containing both ends of e. In contrast we allow vertices to be owned twice, once in $\mathcal{I}$ and once in $\mathcal{O}$: A set $S \in \mathcal{I}$ *owns vertex* v *in* $\mathcal{I}$ if S is the smallest set of $\mathcal{I}$ containing v. A similar definition applies to sets of $\mathcal{O}$.

The next two lemmas are stated for a set $S \in \mathcal{I}$; symmetric versions hold for a set of $\mathcal{O}$.

Lemma 3.5 *Suppose a set $S \in \mathcal{I}$ does not own any vertex in $\mathcal{I}$.*
(i) *S owns a fractional edge.*
(ii) *If $S \in \mathcal{I} \cap \mathcal{O}$ then S owns a vertex in $\mathcal{O}$.*

Proof: Throughout this argument $\boldsymbol{y}$ stands for an arbitrary vector over the fractional edges.

Let the children of S in $\mathcal{L}$ be $S_i, 1 \le i \le r$. Since S does not own any vertex in $\mathcal{I}$ it is partitioned into proper subsets belonging to $\mathcal{I}$. So the children S_i also form a partition of S. Furthermore each child S_i is partitioned into sets of $\mathcal{I}$.

Claim: *For each $i = 1, \ldots, r$, $\rho_x(S_i) = k$ and so $\hat{\boldsymbol{x}}$ satisfies*

$$\rho_y(S_i) = k - \rho_H(S_i). \tag{3}$$

To prove the claim we can clearly assume child S_i belongs to $\mathcal{O} - \mathcal{I}$. $S_i \in \mathcal{O}$ implies $\delta_x(S_i) = k$. We have already observed that S_i is partitioned into sets T_j all belonging to $\mathcal{I}$, so $\rho_x(T_j) = k$. Applying Lemma 3.4(ii) shows $\rho_x(S_i) = k$ as claimed. (The lemma is applicable since the weighting $\boldsymbol{x}$ makes the graph k-edge connected.)

(i) Suppose S does not own a fractional edge. Hence the fractional edges entering the sets S_i, $1 \le i \le r$ form a partition of the fractional edges entering S. Equivalently we have the identity $\rho_y(S) = \sum_{i=1}^{r} \rho_y(S_i)$. In other words the constraint of (2) for S,

$$\rho_y(S) = k - \rho_H(S),$$

is the sum of equations (3) for $i = 1, \ldots, r$. (The right-hand sides add up correctly since $\hat{\boldsymbol{x}}$ satsfies all $r + 1$ equations.)

Since we have chosen $\mathcal{I}$ and $\mathcal{O}$ minimal, the constraints of (2) are linearly independent. Hence equation (3) for some child S_i is linearly independent of the constraints of (2) with S deleted from $\mathcal{I}$. For definiteness suppose this child is S_1. Replacing $\mathcal{I}$ by $\mathcal{I} - S + S_1$ in (2) gives another linearly independent system with a unique solution. The Claim shows this unique solution is $\hat{\boldsymbol{x}}$. However this system is lighter than the original. This is the desired contradiction.

(ii) We again argue by contradiction. Suppose $S \in \mathcal{I} \cap \mathcal{O}$ and S does not own a vertex in either $\mathcal{I}$ or $\mathcal{O}$. The Claim applied twice shows that for $i = 1, \ldots, r$, $\hat{\boldsymbol{x}}$ satisfies

$$\rho_y(S_i) = k - \rho_H(S_i), \quad \delta_y(S_i) = k - \delta_H(S_i). \tag{4}$$

Lemma 3.4(i) gives a linear relation for the terms $\rho_y(S_i), \delta_y(S_i), \rho_y(S)$ and $\delta_y(S)$. In other words the two constraints of (2) for S,

$$\rho_y(S) = k - \rho_H(S), \quad \delta_y(S) = k - \delta_H(S),$$

are linearly dependent with equations (4), $i = 1, \ldots, r$.

The rest of the argument is similar to part (i): At least one of the equations in (4), $i = 1, \ldots, r$ is linearly independent of the constraints of (2) with S deleted from $\mathcal{I}$. Since $\hat{\boldsymbol{x}}$ satisfies this equation we get a contradiction to the lightness of the original system just as in part (i). $\square$

We can now proceed analogously to the undirected case. Define $\phi(S)$ as before.

Lemma 3.6 *Any set $S \in \mathcal{I}$ that does not own a vertex in $\mathcal{I}$ has $\phi(S)$ a positive integer.*

Proof: As before let the children of S in $\mathcal{L}$ be $S_i, 1 \leq i \leq r$, with these sets forming a partition of S. $\phi(S)$ is positive by Lemma 3.5(i). Recalling from the Claim that $\rho_x(S_i) = k$, we have the following equations analogous to Lemma 3.1:

$$\sum_{i=1}^{r} \rho_x(S_i) = \rho_x(S) + \sum_{i \neq j} \rho_x(S_i, S_j),$$

$$rk = k + \sum \{x_e : S \text{ owns } e\},$$
$$\phi(S) \in \mathbf{Z}.$$
□

Theorem 3.7 *LP-rounding achieves an approximation guarantee of $1 + 2/k$ on directed graphs.*

Proof: The number of fractional edges equals the number of equations $|\mathcal{I}| + |\mathcal{O}|$ of (2), since (2) has a unique solution. Hence rounding increases the objective function by $\leq |\mathcal{I}| + |\mathcal{O}| - \phi(\mathcal{L})$. Each set $S \in \mathcal{I}$ either owns a vertex in $\mathcal{I}$ or has $\phi(S) \geq 1$ (Lemma 3.6). In the latter case either $S \notin \mathcal{O}$ or S owns a vertex in $\mathcal{O}$ (Lemma 3.5(ii)). Thus we can conclude $|\mathcal{I}| + |\mathcal{O}| \leq 2n + \phi(\mathcal{L})$. Hence rounding increases the objective by $\leq 2n$. This gives approximation ratio $\leq 1 + 2n/kn = 1 + 2/k$. □

Finally we return to the undirected case to achieve performance ratio $1 + 2/k$ for k odd. Recall Jain's iterated rounding algorithm [9, 15] achieves approximation ratio 2 for the Steiner network problem. Call such a problem a *smallest Steiner network problem* if every edge costs one, i.e., we wish to minimize the number of edges. (The input is a multigraph as usual.)

Theorem 3.8 *Jain's iterated rounding algorithm always finds a solution to a smallest Steiner network problem that contains $\leq n$ edges more than optimum.*

Proof: We sketch the proof, relying on some details from the analysis of Jain's algorithm (see [9] or [15]). The algorithm begins by solving (1) in the case of smallest k-ECSS, and its analog for more general smallest Steiner network problems. Let F be the set of fractional edges in the optimum solution $\boldsymbol{x}$ (after we have broken each edge into integral and fractional parts as usual). As before let $\mathcal{L}$ be the laminar family that uniquely determines the fractional values of $\boldsymbol{x}$. We will show that Jain's algorithm gives an integral solution whose objective value exceeds the LP optimum by at most

$$\min\{|\mathcal{L}| - \boldsymbol{x}(F),\ \boldsymbol{x}(F)\}.$$

This quantity is clearly $\leq |\mathcal{L}|/2$. Since $|\mathcal{L}| \leq 2n - 3$ the theorem follows.

To prove the first term of the upper bound recall that $\boldsymbol{x}$ has $|\mathcal{L}|$ fractional values. Jain's algorithm works by rounding up a subset of the variables of F. So it increases the objective function by $\leq |\mathcal{L}| - \boldsymbol{x}(F)$ as desired.

For the second term let B be the subset of F defined by $B = \{e : x_e \geq 1/2\}$. The algorithm starts by rounding up the variables of B. This increases the objective by $\leq \boldsymbol{x}(B)$. The algorithm then proceeds recursively on a problem whose optimum objective value is $\leq \boldsymbol{x}(F - B)$. Since Jain's algorithm is a 2 approximation, the recursive invocation increases the objective by $\leq \boldsymbol{x}(F - B)$. So the final solution amounts to the original optimum value increased by $\leq \boldsymbol{x}(B) + \boldsymbol{x}(F - B) = \boldsymbol{x}(F)$. □

For general Steiner problems the theorem sometimes gives a better bound: Let s be the number of vertices x where the singleton set $\{x\}$ has a positive connectivity requirement (i.e., the given cut requirement function f has $f(\{x\}) > 0$). Then $|\mathcal{L}| < s+n$. Hence the proof shows Jain's algorithm finds a solution of $\le OPT + (s+n)/2$ edges.

The theorem implies iterated rounding achieves approximation ratio $1 + 2/k$ for the smallest k-ECSS of an undirected graph. (Analyses of iterated rounding on directed graphs [14, 8] do not seem to imply a similar bound, although of course we have proved it.) The results of this section then establish the following bound.

Corollary 3.9 *The integrality gap of LP (1) is $\le 1 + 2/k$ for both directed and undirected graphs.*

Note that for $k > 1$ odd, the cycle C_n has integrality gap $1 + 1/k - O(1/kn)$.

4 Lower Bound Examples

This section proves the bounds of Theorems 3.2–3.7 for simple LP-rounding are tight:

Theorem 4.1 *There is an infinite family of undirected graphs where LP-rounding can find a solution with approximation ratio*

$$1 + \Delta/k - O(1/n),$$

where $\Delta = 3$ for $k > 1$ odd and $\Delta = 2$ for k even. There is a similar family of directed graphs where $\Delta = 2$ for any $k \ge 1$.

The Theorem is proved first for undirected graphs and then for digraphs. Throughout this section w denotes the given edge weight function.

We begin with two convenient examples of regular degree k k-edge connected graphs. If G is a graph the notation $c \cdot G$ denotes the graph obtained from G by making c copies of each edge. C_n denotes the cycle on n vertices. So for example $2 \cdot C_n$ is regular of degree 4.

A *cylinder* on $n = 2h$ vertices consists of two copies of the cycle C_h, plus h edges forming a matching that pairs the ith vertex of one cycle with the ith vertex of the other. (So for instance $h = 4$ gives the cube.) A cylinder is *odd* or *even* depending on the parity of h. We often use these edge weights:

$$w(e) = \begin{cases} a & e \text{ is a matching edge} \\ (k-a)/2 & e \text{ is a cycle edge} \end{cases} \tag{5}$$

Proposition 4.2 (i) *For even k, $\frac{k}{2} \cdot C_n$ is a regular degree k k-edge connected graph.*

(ii) *A cylinder with weights (5) is k-edge connected when $k/h \le a \le k/2$. For odd $k > 1$ taking $a = 1$ (and $h \ge k$) gives a regular degree k k-edge connected graph.*

Proof: (ii) To see the weighting is k-edge connected observe that $a \le k/2$ makes any two vertices in the same cycle k-edge connected, and $k/h \le a$ makes the two cycles k-edge connected. □

Here is a simple illustration of the lower bound argument. Suppose $k > 1$ is odd. Let Y denote an unweighted cylinder on n vertices. Consider an input graph $k \cdot Y$. An optimum extreme point for LP (1) is $\frac{k}{2} \cdot C_n$. (Y contains a Hamiltonian cycle C_n. Assigning each edge of this cycle weight $k/2$ gives an extreme point, since this is the only feasible solution with every vertex and edge tight.) Rounding up each edge weight increases $\boldsymbol{x}(E)$ by $n/2$. On the other hand Proposition

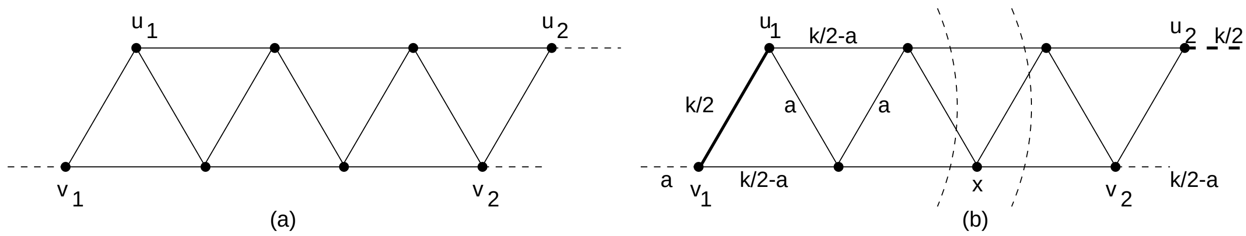

Figure 1: Gadget for undirected graphs. (a) The gadget consists of the solid edges. Edges incident to the gadget, drawn dashed, go to v_1, u_2 or v_2. (b) Tight sets and edge weights for an extreme point: The gadget has one edge of weight $k/2$; the remaining gadget edges weigh a or $k/2-a$. The two dashed arcs are the right boundaries of the tight sets $[v_1, x)$ and $[v_1, x]$ respectively.

4.2(ii) shows a smallest k-ECSS has $kn/2$ edges. Hence the algorithm's approximation ratio is $1 + (n/2)/(kn/2) = 1 + 1/k$. We need to increase this by $2/k$.

We will use the following obvious principle. The principle allows nonsimple paths.

Path Principle *Let P be a path where the first edge weighs a and the weights of every two consecutive edges sum to s. Then the weights of P are alternately a and $s-a$.*

The lower bound is based on the gadget illustrated in Fig.1. A gadget consists of kites (a *kite* being $K_4 - e$). A gadget can be expanded arbitrarily by adding on new kites, each new kite introducing four new edges. Edges incident to the gadget go to one of the vertices v_1, u_2, or v_2. More than one edge may be go to each of these vertices.

The vertices of a gadget have a natural left-to-right order. So we will use interval notation to designate sets of gadget vertices. Fig.1(b) illustrates the intervals $[v_1, x]$ and $[v_1, x) = [v_1, x^-]$ where x^- is the vertex preceding x.

Call the edge weights shown in Fig.1(b) (including the edges incident to the gadget) the *generic weights for a*. Here a is an arbitrary value, $0 \le a \le k/2$. Each horizontal gadget edge weighs $k/2 - a$ and each diagonal gadget edge except the first weighs a.

Lemma 4.3 *Consider a gadget occurring as an induced subgraph H of a graph G. Let a be an arbitrary value, $0 \le a \le k/2$.*

(i) *Suppose the edges incident to H at v_1 have total weight a. Then the generic weights for a are the unique values that make every set $\{x\}$ and $[v_1, x]$, $x \in V(H)$ tight (i.e., degree k).*

(ii) *Suppose the edges containing a vertex of H have the generic weights for a. Let G' be the result of contracting the top (bottom) row of H to a vertex u (v) respectively and assigning edge uv the weight $k/2$. Then G is k-edge connected if G' is.*

Proof: (i) It is easy to check that the generic weights make all singletons $\{x\}$ and all intervals $[v_1, x]$ tight. The rest of the argument is devoted to showing the weights are unique.

Claim: *At any vertex $x \in V(H) - v_1$, the edges going leftward (rightward) from x must have total weight exactly $k/2$. (For $x = u_2, v_2$ this includes edges incident to the gadget.)*

To show the Claim compare the tight sets $[v_1, x)$ and $[v_1, x]$, as illustrated in Fig.1(b). The two sets are incident to the same edges except for the edges going leftward and rightward from x. (This holds at the boundaries $x = u_1, v_2, u_2$ too.) Since $d(x) = k$, the edges going leftward (rightward) have total weight $k/2$ as claimed.

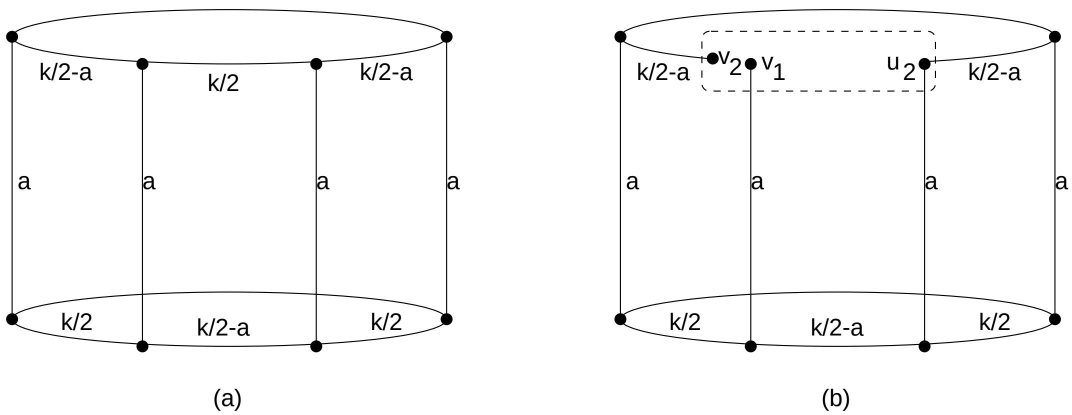

Figure 2: (a) Even cylinder Y. (b) Y with an edge replaced by a gadget.

The Claim at vertex u_1 shows $w(u_1v_1) = k/2$. Now consider the following path P through the remaining edges of the gadget: P starts at v_1 and traverses the first kite minus u_1v_1, by zigzagging up three edges and going down the fourth; then P traverses the remaining kites in the same pattern, ending by going down u_2v_2. The first edge of P weighs $k/2 - a$ by the tightness of v_1. The weights of every two consecutive edges of P sum to $k/2$ by the Claim. The Path Principle then gives the weights of all the gadget edges. The weights of the edges incident to the gadget at v_2 and u_2 follow from tightness of these vertices.

(ii) We will construct a family of edge-disjoint cycles in G of total weight $k/2$, each cycle passing through every vertex of H. This implies the vertices of H are k-edge connected (since each cycle provides two paths joining any two vertices of H). Now a cut of weight $< k$ in G gives a similar cut in G', which is impossible. Thus we will deduce that G is k-edge connected.

Split the gadget edge u_1v_1 into edges of weight a and $k/2 - a$. Now every gadget edge has weight a or $k/2 - a$; we use the terms *a-edge* and *$(k/2 - a)$-edge* accordingly.

Graph $G' - uv$ contains a collection of edge-disjoint uv-paths of total weight $k/2$. In G the collection corresponds to u_2v_1-paths of total weight a and u_2v_2-paths of total weight $k/2 - a$. Form a cycles by starting with the u_2v_1-paths and extending them back to u_2 via the a-edges. Form $k/2 - a$ cycles by starting with the u_2v_2-paths of total weight $k/2 - a$ and extending them back to u_2 via the $(k/2 - a)$-edges. Each cycle traverses all of $V(H)$ as desired. □

Fig.2 illustrates how the gadget will be used. We first discuss the cylinder Y of Fig.2(a). Y is constructed from two cycles C_h, h even, and has the weights shown, for a an arbitrary real value, $k/h \le a \le k/2$. Y has the following properties (a)–(d):

(a) *Y is k-edge connected.* In proof, split each weight $k/2$ edge into edges of weight a and $k/2 - a$. Then the edges can be partitioned into a top cycle of weight $k/2 - a$, a bottom cycle of weight $k/2 - a$, and a Hamiltonian cycle of weight a. This implies the vertices of the top cycle are k-edge connected, as are the vertices of the bottom cycle. The assumption $ha \ge k$ then implies all vertices are k-edge connected.

(b) *Every vertex is tight. So is every weight $k/2$ edge.*

(c) *Fixing the weight of one vertical edge to be a, the remaining weights shown are the unique values satisfying property (b).* To prove this first recall that in general if a graph has disjoint vertex sets A, B with $d(A) = d(B) = d(A \cup B) = k$ then $d(A, B) = k/2$. For instance an edge xy with x, y and $\{x, y\}$ tight necessarily has $w(xy) = k/2$. Thus property (b) implies all the weight $k/2$

edges are forced. The remaining edges form a Hamiltonian cycle C. We can deduce their weights by applying the Path Principle to C, using the tightness of each vertex.

(d) *If $ha = k$ (i.e., our assumption holds with equality) then Y is an optimum extreme point of LP (1) (where each upper bound $u_e = \infty$).* In fact the weights shown are the unique values making tight all the sets of (b) plus one more, the upper cycle. (Note that although the number of tight sets we have specified is one more than the number of edges, the graph is bipartite, so the vertex degrees are linearly dependent.) Property (b) guarantees optimum weight $kn/2$.

The extreme point for our lower bound is constructed by gluing a gadget into Y, as illustrated by Figs.2(b) and 1(a). Call this graph G. G satisfies analogs of properties (a)–(d):

(a$'$) *G is k-edge connected.* Prove this by combining (a) and Lemma 4.3(ii) (notice $G' = Y$).

(b$'$) *Every vertex, every weight $k/2$ edge and every interval of Lemma 4.3*(i) *is tight.*

(c$'$) *Fixing the weight of the vertical cylinder edge at v_1 to be a, the remaining weights shown are the unique values satisfying (b$'$).* The argument is similar to (c): The weights of gadget edges follow from Lemma 4.3(i). For the cylinder edges, the weight $k/2$ edges are forced as before. The remaining cylinder edges form a v_1v_2-path P starting with the vertical edge at v_1 of weight a. The weights of every two consecutive edges sum to $k/2$. (For the two edges at any cylinder vertex $w \neq v_1, v_2, u_2$ this follows as before – w is incident to a weight $k/2$ cylinder edge. For the two edges at u_2 this follows from Lemma 4.3(i).) The Path Principle for P then gives all the weights.

(d$'$) *If $ha = k$ then G is an optimum extreme point of LP (1) (where each $u_e = \infty$).*

Now imagine extending the gadget with a new kite. This adds four new edges, two of weight a and two of weight $k/2 - a$. Let Δ be the amount that rounding up these four edges increases the value of the objective function. Let $\alpha = a \bmod 1$.

Case 1: k is even. Assume $0 < \alpha < 1$. When we round up the extreme point the four edges increase the objective function by $\Delta = 2(1-\alpha) + 2(\alpha) = 2$.

Case 2: k is odd. Assume $0 < \alpha < 1/2$. When we round up the four edges increase the objective function by $\Delta = 2(1-\alpha) + 2(1/2+\alpha) = 3$. (Note that for $1/2 \leq \alpha < 1$ the objective increases by only 1.)

In summary, define the graph G by taking $h = 4k$ and $a = k/h = 1/4$. G is an optimum extreme point. Let the gadget contain $2 + g$ vertices, where g is an even integer that will approach infinity. So the gadget is built by starting with the edge u_1v_1 and extending it by $g/2$ kites. Since $\alpha = a = 1/4 < 1/2$ rounding up this extreme point increases the objective function by $\geq \Delta g/2$. Since G has $n = g + O(k)$ vertices the increase is $\geq \Delta n/2 - O(k)$.

Observe that n is even (both the gadget and the cylinder Y have an even number of vertices, and G discards exactly two cylinder vertices). So by adding extra edges, we can assume that the input graph contains a k-ECSS of $kn/2$ edges as in Proposition 4.2. (In fact, for even k G itself has a Hamiltonian cycle, consisting of the cylinder edges of weight $\neq k/2 - a$ and the diagonal gadget edges. For odd k we can introduce the cylinder Z of Proposition 4.2 by adding just three edges. The vertical edges of Z are the current cylinder's vertical edges plus the gadget diagonals not parallel to u_1v_1.) Hence the approximation ratio for the rounded solution is $\geq 1 + (\Delta n/2 - O(k))/(kn/2) = 1 + \Delta/k - O(1/n)$. As g tends to infinity so does n. Thus we have the ratio claimed in Theorem 4.1.

Digraphs

The example for digraphs follows the same outline. We first discuss the gadget of Fig.3(a). Call the weights shown in the figure the *generic weights for a*. Note that every diagonal edge except the last is directed both ways, and the first diagonal u_1v_1 has different weights in the two directions.

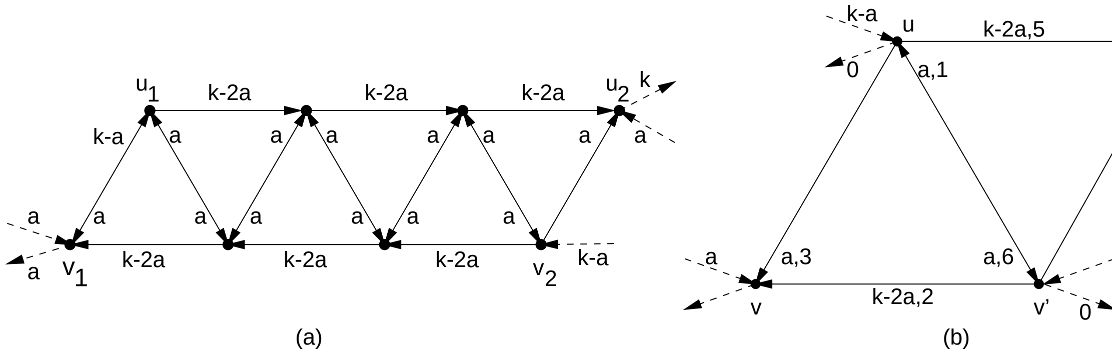

Figure 3: (a) Gadget for digraphs with its generic edge weights. As before edges incident to the gadget, drawn dashed, go to v_1, u_2 or v_2. (b) Inductive step deducing the edge weights on a kite.

Use the same interval notation for gadgets as before. Call two vertices *k-edge connected* if they are not separated by any cut of in-degree k (equivalently, the maximum flow between the vertices in both directions is at least k).

Lemma 4.4 *Consider a gadget occurring as an induced subgraph H of a graph G. Let a be an arbitrary value, $0 \le a \le k/2$.*

(i) *Suppose the edges entering H at v_1 have total weight a, as do the edges leaving H at v_1. Then the generic weights for a are the unique values that make every vertex of H and every interval $[v_1, x]$, $x \in V(H) - u_2$ have in- and out-degree both equal to k.*

(ii) *Suppose the edges containing a vertex of H have the generic weights for a. Suppose the subgraph $G - E(H)$ contains a collection of edge-disjoint paths consisting of v_1u_2-paths of total weight a, u_2v_1-paths of total weight a and u_2v_2-paths of total weight $k - a$. Then every pair of vertices of H is k-edge connected in G.*

Proof: (i) It is easy to check that the generic weights make the in- and out-degree of every set $\{x\}$, $x \in V(H)$ and $[v_1, x]$, $x \in V(H) - u_2$ both equal to k. The rest of the argument is devoted to showing the weights are unique.

We first deduce the weights of the gadget edges, using induction. For the base case observe that $w(v_1u_1) = k - a$. This follows from the constraint $\delta(v_1) = k$.

For the inductive step consider a typical kite of the gadget, as in Fig.3(b). Edge vu is not drawn – it is considered part of the previous kite (for the first kite it is treated in the base case v_1u_1). Similarly $u'v'$ is not drawn – it is part of the next kite (for the last kite it does not exist).

Assume the edges incident to the kite have the weights shown. For instance the total weight of all edges entering (leaving) the kite at v is exactly a (arbitrary) respectively. The label $k - a$ at vertex u means the total weight of edges entering u from the left is exactly $k-a$. (This is consistent with the convention that vu is not in the current kite; rather it contributes to this label $k-a$. Note this label is correct for the first kite by the base case.) The edge entering the kite at v' represents a horizontal edge and, if the kite is not the last one, a diagonal edge.

We assert that the above assumptions imply that the edges of the kite have the weights shown. More precisely, the label on each edge gives its weight followed by an integer (1-6) indicating the order in which we will deduce that weight. For example edge $v'u$ labelled $a, 1$ has weight a and it will be the first weight deduced.

Observe that once we deduce all the weights as shown, then by induction the weights of all gadget edges are determined: The weight of vu is deduced as that of $v'u'$ in the previous kite. The

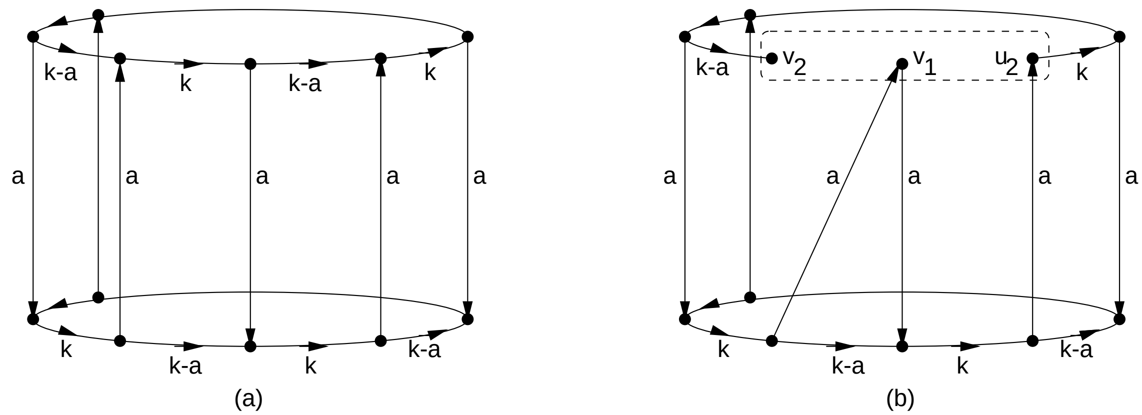

Figure 4: (a) Even cylinder Y for the digraph lower bound. (b) Y with a gadget replacing two edges.

weight of $u'v'$ is deduced as that of uv in the next kite, unless $u'v'$ does not exist, i.e., the last kite. Also the values shown for one kite imply the weights assumed for the next kite: For instance for u, $k-a=(k-2a)+a$. For the two labels of 0, the gadget has no corresponding edges.

Deduce that $w(v'u)=a$ using the constraint $\rho(u)=k$. Use this, plus the constraint $\rho([v_1,u])=k$, to deduce $w(v'v)=k-2a$. The four remaining deductions are based on these constraints: $\rho(v)$ for uv, $\delta(v')$ for $v'u'$, $\delta([v_1,v'])$ for uu', $\delta(u)$ for uv'.

Now by induction the weights of all the gadget edges have been deduced. It remains to deduce the weights of the edges incident to the gadget at v_2 and u_2. For this use the constraints on $\rho(v_2)$, $\rho(u_2)$ and $\delta(u_2)$.

(ii) We will construct a family of edge-disjoint cycles of total weight k, each cycle passing through every vertex of H. Clearly this implies every two vertices of H are k-edge connected.

Split the gadget edge v_1u_1 into two edges of weight a and $k-2a$. Now every gadget edge has weight a or $k-2a$; we will use the terms *a-edge* and *$(k-2a)$-edge* accordingly.

Form $k-2a$ cycles by starting with the v_2u_2-path of $(k-2a)$-edges and adding on $k-2a$ of the $k-a$ hypothesized u_2v_2-paths. Form a cycles by starting with the v_1u_2-path of a-edges and adding on the a hypothesized u_2v_1-paths. Finally form a more cycles by starting with the v_2v_1-path of a-edges and adding on the a hypothesized v_1u_2-paths followed by the remaining a hypothesized u_2v_2-paths. It is easy to see that no gadget edge occurs twice in this family of cycles. $\square$

The Path Principle is valid in a digraph G even when P is a path in the undirected version of G, i.e., directions of edges are ignored. We will apply the principle to *alternating paths* of the digraph, i.e., each internal vertex has its two edges both entering or both leaving. This is natural since we can have constraints on the total weight of edges entering or leaving a given vertex.

Consider the cylinder Y illustrated in Fig.4(a). Y is constructed from two cycles C_h, h even, and has the weights shown, for a an arbitrary real value, $2k/h \le a \le k$. Y has the following properties (a)–(d):

(a) *Y is k-edge connected.* In proof, split each weight k edge into edges of weight a and $k-a$. Then the edges can be partitioned into a top cycle of weight $k-a$, a bottom cycle of weight $k-a$, and a Hamiltonian cycle of weight a. This implies the vertices of the top cycle are k-edge connected, as are the vertices of the bottom cycle. The assumption $ah/2 \ge k$ then implies all vertices are k-edge connected.

(b) *Every vertex has in-degree k and out-degree k.*

(c) *Fixing the weight of one weight a edge, the remaining weights shown are the unique values satisfying property (b).* In proof it is clear that (b) forces the weight k edges. The remaining weights follow from the Path Principle applied to the edges of weight $\neq k$ (they form an alternating Hamiltonian cycle).

(d) *If $ha = 2k$ (i.e., our assumption holds with equality) then Y is an optimum extreme point of LP (1) (where each $u_e = \infty$).* In fact the weights shown are the unique values satisfying (b) and making both cycles have in-degree k. Property (b) implies total weight kn which is optimum.

The lower bound extreme point for digraphs is illustrated by Figs.4(b) and 3(a). Call this graph G. In brief construct G by starting with Y, deleting two consecutive edges of the upper cycle and replacing them with the gadget of Fig.3(a), and rotating an upward directed cylinder edge to the right into vertex v_1 as shown. Assume $2k/h \le a \le k/2$. For now we leave unspecified both the parameter h for Y and the number of vertices in the gadget. G satisfies analogs of properties (a)–(d):

(a$'$) *G is k-edge connected.* G contains the collection of paths hypothesized in Lemma 4.4(ii). (The paths contain every cylinder edge except for weight $k-a$ of every edge in the lower cycle.) Hence the vertices of H are all k-edge connected. Contracting H in G gives a graph that is a contraction of Y, and hence is k-edge connected. We conclude G is k-edge connected.

(b$'$) *Every vertex of G and every interval of Lemma 4.4*(i) *has in- and out-degree k.*

(c$'$) *Fixing the weight of the cylinder edge entering v_1 to be a, the remaining weights shown are the unique values satisfying (b$'$).* The Path Principle applied to the length three alternating path starting and ending at v_1 determines the weights of these three edges. Now the hypothesis of Lemma 4.4(i) is satisfied. Hence the weights of all the gadget edges and the cylinder edge entering u_2 are determined. The remaining cylinder edges are determined as in (c) (the remaining edges of weight $\neq k$ form an alternating u_2v_2-path).

(d$'$) *If $ha = 2k$ then G is an optimum extreme point of LP (1) (where each $u_e = \infty$).*

Imagine extending the gadget with a new kite. This adds six new edges, four of weight a and two of weight $k-2a$. Let $\alpha = a \bmod 1$ and assume $0 < \alpha < 1/2$. When we round up the extreme point these edges increase the objective function by $4(1-\alpha) + 2(2\alpha) = 4$. (Note that for $1/2 \le \alpha < 1$ the objective increases by only 2.)

In summary, define the graph G by taking $h = 6k$ and $a = 2k/h = 1/3$. G is an optimum extreme point. Let the gadget contain $4+g$ vertices, where g is an even integer that will approach infinity. So the gadget is built by starting with the initial kite (on vertices v_1, u_1, u_2, v_2) and extending it by $g/2$ kites. Since $\alpha = a = 1/3 < 1/2$ rounding up the extreme point increases the objective function by $\ge 4(g/2) = 2g$. Since G has $n = g + O(k+1)$ vertices the increase is $\ge 2n - O(k)$.

We can assume that the input graph also contains a directed Hamiltonian cycle of weight k edges. (In fact adding just the edge from v_2 to the cylinder vertex below v_1 completes a Hamiltonian cycle. The cycle traverses all gadget vertices except v_2 on a v_1u_2-path, continues along the top cylinder cycle to v_2, and uses the new edge to traverse the bottom cylinder cycle and return to v_1.) Thus the smallest k-ECSS has the same size as the LP optimum, kn. Hence the approximation ratio for the rounded solution is $\ge 1 + 2/k - O(1/n)$. As g tends to infinity so does n. This gives the ratio claimed in Theorem 4.1.

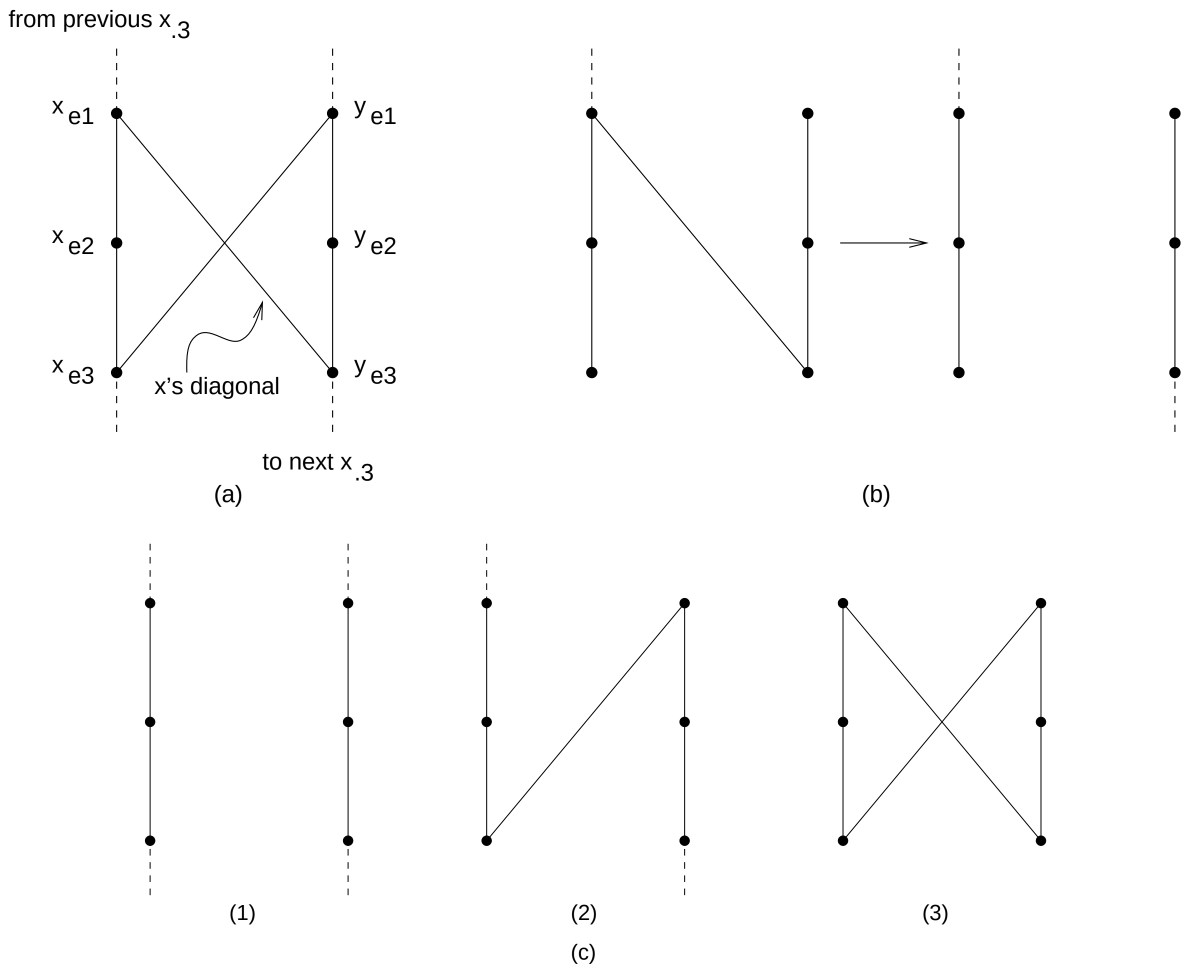

Figure 5: Hardness gadget for undirected graphs. (a) Gadget and x's chain. (b) Degree reduction on vertex x_{e1}. (c) 3 states of a gadget.

5 Hardness Results

We begin by sketching the proof of [3] that the smallest 2-ECSS problem is MAXSNP-hard for undirected graphs.[1] The proof is an L-reduction from *VC3*, the problem of finding a smallest vertex cover of a graph G having maximum degree at most 3. The graph H for smallest 2-ECSS is constructed as follows. Each edge $e = xy$ of G gives rise to the gadget illustrated by the 6 solid edges in Fig.5(a). A gadget consists of 4 *vertical edges* and 2 *diagonal edges*. The diagonal incident to x_{e1} (y_{e1}) is *x's (y)'s diagonal.*

In addition to the gadget vertices, H contains 1 other vertex r. In addition to the gadget edges, H contains for each vertex x of G a *chain* of edges stringing together the gadgets for the edges incident to x. This chain is illustrated by the 2 dashed edges marked in Fig.5(a), and is constructed as follows: Order the edges of G arbitrarily. For e the first edge incident to x, H contains the edge rx_{e1}. For $e = xy$ and $f = xz$ 2 consecutive edges incident to x, H contains edge $y_{e3}x_{e1}$. For $e = xy$ the last edge incident to x, H contains the edge $y_{e3}r$.

[1]The argument is an adaptation of the proof that the smallest 1-ECSS problem is MAXSNP-hard by Khuller et.al. [13]. They in turn attribute the fundamentals of their argument to Karp [11].

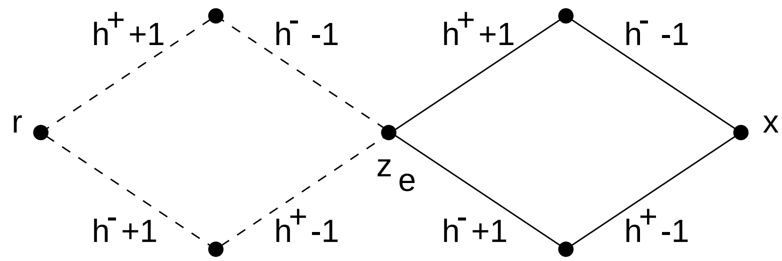

Figure 6: Padding edges for vertex $x(= x_{ei})$ in gadget for edge e. h^+ (h^-) denotes $\lceil k/2 \rceil$ ($\lfloor k/2 \rfloor$).

Let m be the number of edges of G. We shall also use the notation C for a vertex cover of G and K for a 2-ECSS of H. We shall think of K as a set of edges (rather than a subgraph), so $|K|$ denotes the number of edges in K. We will show that a smallest vertex cover C^* of G and a smallest 2-ECSS K^* of H satisfy

$$|K^*| = 6m + |C^*|, \tag{6}$$

by showing how to transform a vertex cover into a 2-ECSS of appropriate size and vice versa.

A vertex cover C of G gives a 2-ECSS K of H containing $6m+|C|$ edges, as follows: K contains the chain edges of all vertices of C, the diagonal edges of all vertices of $G-C$, plus the vertical edges of all gadgets. Since C is a vertex cover, for any gadget the edges chosen by K are as illustrated in Fig.5(c.1) or (c.2). Fig.5(c.2) corresponds to the case of $x \in C$, $y \notin C$; the symmetric edges are chosen in the opposite case. It is easy to see that K is an edge-disjoint union of cycles through r (by our ordering of the edges of G) and hence is 2-edge connected. Furthermore K has exactly $6m + |C|$ edges: Associate each gadget with 6 edges – those shown in Fig.5(c.1–2) except for the edges on the bottom leaving the gadget. This gives $6m$ edges in K. The remaining edges are each the last edge of the chain of a vertex of C, and hence they number $|C|$.

Next we show that a 2-ECSS K gives a vertex cover of size $\le |K| - 6m$. We start by executing a procedure called *degree reduction* which ensures that every vertex of $K - r$ has degree exactly 2: As illustrated in Fig.5(b), if a gadget vertex has degree 3 in K we discard its incident diagonal edge, and add the corresponding edge leaving the gadget (if it is not already in our subgraph). Repeating this process as long as some gadget vertex has degree 3, we eventually get a new subgraph K' with no such vertices.

Fig.5(c.3) shows a gadget with no incident (chain) edges. K' does not contain such a gadget, since it cannot be created by degree reduction. It is easy to see that Fig.5(c) gives all the possibilities for a gadget in K', since each gadget vertex has degree 2. Hence each gadget is as shown in Fig.5(c.1) or (c.2). This implies that K' is an edge-disjoint union of cycles through r (by the ordering of the edges of G). Furthermore the chain edges in K' are the union of the chain edges for some set of vertices C. C is a vertex cover of G (since a chain edge is incident to each gadget).

Our two transformations establish the relation (6). This implies the MAXSNP-hardness of the smallest 2-ECSS problem because our construction satisfies the 2 defining properties of an L-reduction: Starting with vertex cover C, the corresponding 2-ECSS has size $6m + |C| \le 19|C|$ ($m/3 \le |C|$ since G has maximum degree 3). And starting with 2-ECSS K, containing Δ more edges than a smallest 2-ECSS, the corresponding vertex cover has $\le \Delta$ more edges than a minimum vertex cover.

Now we extend this construction to reduce VC3 to smallest undirected k-ECSS, for any fixed $k > 2$. The reduction starts by forming the graph H defined above. Then for every gadget, say for edge e, we introduce a new vertex z_e along with 2 others, plus 4 edges forming length 2 paths from z_e to r – these are the dashed edges in Fig.6, having the multiplicities shown. Note that $h^- - 1 \geq 0$. Also for each of the 6 vertices in the gadget, say vertex x, we add 2 new vertices plus 4 edges forming length 2 paths from x to z_e – these are the solid edges in Fig.6, having the multiplicities shown (these are the same multiplicites as the dashed edges). To illustrate the construction, z_e has degree $7k + 10$ while the other 4 vertices of Fig.6 have degree exactly k.

Let H denote the new k-ECSS graph. The number of gadget or padding edges in H is

$$M = (6[1 + 2k] + 2k)m = (6 + 14k)m.$$

Similar to the previous reduction we will show that a smallest vertex cover C^* of G and a smallest k-ECSS K^* of H satisfy

$$|K^*| = M + |C^*|, \tag{7}$$

by transforming a vertex cover into a k-ECSS of appropriate size and vice versa.

We construct a k-ECSS K from a vertex cover C as follows. Just as before K contains the chain edges of all vertices of C, the diagonal edges of all vertices of $G - C$, and the vertical edges of all gadgets; in addition K contains all padding edges. Counting edges as before it is easy to see $|K| = M + |C|$.

We show K is k-edge connected in the following 3 steps. As before the nonpadding edges of K partition into edge disjoint cycles that span all gadget vertices and r.

(a) Any gadget vertex x is k-edge connected to r: The cycle through x gives 2 xr-paths. The padding edges for x give $k - 2$ xr-paths (of length 4), since $(h^- - 1) + (h^+ - 1) = k - 2$ and $h^+ + 1 \geq h^- - 1$, $h^- + 1 \geq h^+ - 1$.

(b) Each vertex z_e is k-edge connected to r: The dashed padding edges give $k - 2$ $z_e r$-paths (of length 2). The solid padding edges give $k - 2 \geq 1$ $z_e x_{ei}$-paths (of length 2), for $i = 1, 2$. This gives k paths to r's k-edge connected component.

(c) Now each of the 4 other padding vertices in Fig.6 have their k edges going to r's k-edge connected component.

We construct a vertex cover C from a k-ECSS K as follows. Every gadget vertex x_{e2}, y_{e2} has degree exactly k in H, as do all the padding vertices except the z_e vertices. So K contains all their incident edges, i.e., all padding edges and all vertical edges. These edges number $(4 + 14k)m$.

Now execute the degree reduction procedure on the nonpadding edges of K, obtaining a new graph K'. Observe that no gadget is in state Fig.5(c.3) in K': Such a gadget has no nonpadding edges incident to it in K', which implies it had no nopadding edges incident to it in K. But then the 19 vertex set, formed from the 6 gadget vertices plus each of their 2 neighboring padding vertices plus z_e, has exactly $k - 2$ incident edges (coming from 2 dashed edges in Fig.6). This contradicts the k-edge connectedness of K.

As before the nonpadding edges of K' partition into cycles containing r, and the chain edges in those cycles belong to the vertices of a vertex cover C. Each gadget can be associated with 2 chain or diagonal edges of K', giving $2m + |C|$ edges total. So $|K| \geq |K'| = (4+14k)m + 2m + |C| = M + |C|$.

We conclude that (7) holds. We get the following two results.

Theorem 5.1 *The smallest k-ECSS problem on undirected multigraphs is MAXSNP-hard for any fixed integer $k \geq 2$.*

Proof: As before the construction satisfies the 2 defining properties of an L-reduction: Starting with vertex cover C, the corresponding k-ECSS has size

$$M + |C| \leq (3[6 + 14k] + 1)|C| = (19 + 42k)|C|.$$

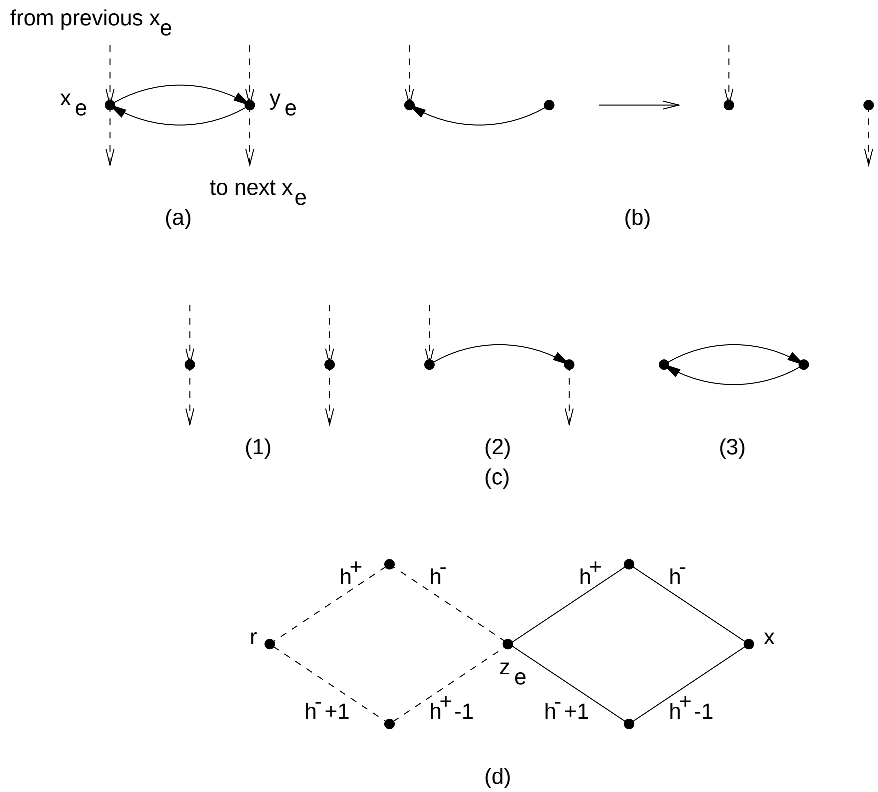

Figure 7: Reduction for multidigraphs.

And starting with k-ECSS K, containing Δ more edges than a smallest k-ECSS, the corresponding vertex cover has $\leq \Delta$ more edges than a minimum vertex cover. $\square$

Theorem 5.2 *There exists a constant $c > 0$ such that for any fixed integer $k \geq 2$, no polynomial-time algorithm approximates the smallest k-ECSS on undirected multigraphs to within ratio $1 + c/k$ unless $P = NP$.*

Proof: Suppose an algorithm approximates smallest k-ECSS to within ratio $1 + c/k$, where k is some fixed integer ≥ 2 and for the moment the constant c is arbitrary. We will show how to approximate VC3 to within ratio $1 + 52c$. Since VC3 is MAXSNP hard there is a constant $\epsilon > 0$ such that an approximating VC3 to within factor $1 + \epsilon$ imples $P = NP$. Hence the same can be said for approximating smallest k-ECSS to within factor $1 + \epsilon/(52k)$.

Given an instance of VC3, transform it as above to an instance of smallest k-ECSS. We have noted above the relation $|K^*| = M + |C^*| \leq (19 + 42k)|C^*|$. Hence the approximation algorithm finds a k-ECSS of size $\leq (1 + c/k)|K^*| \leq (M + |C^*|) + c(19/k + 42)|C^*| \leq M + |C^*|(1 + 52c)$. As above this gives a vertex cover of size $\leq |C^*|(1+52c)$. Clearly the entire procedure takes polynomial time. $\square$

Digraphs

The same two results can be deduced for multidigraph k-ECSS. We could use a directed version of the previous reduction but it is simpler to work with Khuller et.al.'s reduction for directed 1-ECSS [13]. Fig.7(a)–(c) is the analog of Fig.5, for an L-reduction of VC3 to smallest directed 1-ECSS. Equation (6) becomes $|K^*| = 2m + |C^*|$. Degree reduction (Fig.7(b)) is done on any gadget vertex of in- or out-degree 2. Fig.7(d) gives the new padding graph, the analog of Fig.6. Here each edge, drawn undirected, represents a pair of antiparallel directed edges, both directed edges having the multiplicity shown. Now the number of gadget or padding edges in H is $M = (2 + 12k)m$.

Corollary 5.3 *Theorems 5.1–5.2 hold for multidigraphs, with the condition on k loosened to $k \geq 1$.*

Similar reductions for k-VCSS (vertex connected spanning subgraph) are presented in [7].

Acknowledgments

The first author learned of work on this topic by the other authors in an anonymous communication. We are very grateful for this assistance. We also thank Guy Kortsarz for suggesting the possibility of a hardness result.

References

[1] J. Cheriyan and R. Thurimella, *Approximating minimum-size k-connected spanning subgraphs via matching*, SIAM J. Comput., 30, 2 (2000), pp. 528–560.

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

[3] C. G. Fernandes, *A better approximation ratio for the minimum size k-edge-connected spanning subgraph problem*, J. Algorithms, 28, 1 (1998), pp. 105–124.

[4] A. Frank, *Submodular functions in graph theory*, Discrete Math., 111 (1993), pp. 231–243.

[5] H. N. Gabow, *An improved analysis for approximating the smallest k-edge connected spanning subgraph of a multigraph*, SIAM J. Disc. Math. 19, 1 (2005), pp. 1–18.

[6] H.N. Gabow, *Special edges, and approximating the smallest directed k-edge connected spanning subgraph,* Proc. 15th Annual ACM-SIAM Symp. on Discrete Algorithms, 2004, pp. 227–236.

[7] H.N. Gabow, *On the difficulty of k-connected spanning subgraph problems*, unpublished notes.

[8] H.N. Gabow, *On the L_∞-norm of extreme points for crossing supermodular directed network LPs*, Eleventh MPS Conf. on Integer Programming and Combinatorial Optimization, 2005, M. Jünger and V. Kaibel eds., Springer-Verlag LNCS 3509, pp. 392–406.

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

[10] D. R. Karger, *Random sampling in cut, flow, and network design problems*, Math. of OR 24, 2 (1999), pp. 383–413.

[11] R.M. Karp, *Reducibility among combinatorial problems*, in Complexity of Computer Computations, R.E. Miller and J.W. Tatcher, eds., Plenum Press, NY (1972), pp. 85–103.

[12] S. Khuller and B. Raghavachari, *Improved approximation algorithms for uniform connectivity problems*, J. Algorithms 21, 2 (1996), pp. 434–450.

[13] S. Khuller, B. Raghavachari and N. Young, *Approximating the minimum equivalent digraph,* SIAM J. Comput., 24, 4 (1995), pp. 859–872.

[14] V. Melkonian and E. Tardos, *Algorithms for a network design problem with crossing supermodular demands*, Networks 43, 4 (2004), pp. 256–265.

[15] V. V. Vazirani, *Approximation Algorithms*, Springer-Verlag, NY, 2001.