Hardness of approximation for vertex-connectivity network-design problems*

Guy Kortsarz† Robert Krauthgamer‡ James R. Lee§

July 9, 2002

Abstract

In the survivable network design problem (SNDP), the goal is to find a minimum-cost subgraph satisfying certain connectivity requirements. We study the vertex-connectivity variant of SNDP in which the input specifies, for each pair of vertices, a required number of vertex-disjoint paths connecting them.

We give the first lower bound on the approximability of SNDP, showing that the problem admits no efficient $2^{\log^{1-\epsilon} n}$ ratio approximation for any fixed $\epsilon > 0$ unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\mathrm{polylog}(n)})$. We also show hardness of approximation results for several important special cases of SNDP, including constant factor hardness for the k-vertex connected spanning subgraph problem (k-VCSS) and for the vertex-connectivity augmentation problem, even when the edge costs are severely restricted.

1 Introduction

A basic problem in network design is to find, in an input graph $G = (V, E)$ with nonnegative edge costs, a subgraph of minimum-cost that satisfies certain connectivity requirements, see e.g. the surveys [Fra94, Khu96]. A fundamental problem in this area is the vertex-connectivity variant of the *survivable network design problem* (SNDP). In this problem, the input also specifies a *connectivity requirement* $k_{u,v}$ for every pair of vertices $\{u, v\}$, and the goal is to find a subgraph of minimum-cost such that for every pair of vertices $\{u, v\}$ there are at least $k_{u,v}$ *vertex-disjoint* paths between u and v.

Many network design problems (including SNDP) are NP-hard, and a significant amount of research is concerned with *approximation algorithms* for these problems, i.e. polynomial-time algorithms that find a solution whose value is guaranteed to be within some factor (called the *approximation ratio*) of the optimum. A notable success is the 2-approximation of Jain [Jai01] for the edge-connectivity version of SNDP, in which the paths are only required to be *edge-disjoint*. (See also [FJW01] for an extension to a more general version of SNDP.) However, no algorithm is known to achieve a sublinear (in $|V|$) approximation ratio for the vertex-connectivity variant of SNDP.

---

*This is a preprint of a paper that will appear in APPROX 2002. (Copyright owned as specified in the conference proceedings.)

†Department of Computer Sciences, Rutgers University, Camden, NJ 08102, USA. Email: `guyk@crab.rutgers.edu`

‡International Computer Science Institute (ICSI) and Computer Science Division, University of California, Berkeley, CA 94720, USA. Supported in part by NSF grants CCR-9820951 and CCR-0121555 and DARPA cooperative agreement F30602-00-2-0601. Email: `robi@cs.berkeley.edu`

§Computer Science Division, University of California, Berkeley, CA 94720, USA. Email: `jrl@cs.berkeley.edu`

This disparity between approximating edge-connectivity and vertex-connectivity network design problems might suggest a lack in our understanding of vertex-connectivity or, perhaps, that vertex-connectivity problems are inherently more difficult to approximate. Resolving this question (see e.g. [Vaz01, Section 30.2]) is one of the important open problems in the field of approximation algorithms. We provide an answer by showing that there is a striking difference between the approximability of the edge and vertex-connectivity variants of this problem. Specifically, we show that it is hard to approximate the vertex-connectivity variant of SNDP within a factor of $2^{\log^{1-\epsilon}|V|}$ for any fixed $\epsilon > 0$.

In general, we address the *hardness of approximation* of vertex-connectivity problems by presenting relatively simple variants of SNDP that are nevertheless hard to approximate. Therefore, unless stated otherwise, *connectivity* means vertex-connectivity, *disjoint paths* means vertex-disjoint paths, and all graphs are assumed to be undirected. (For a more in-depth account of approximation algorithms for edge-connectivity problems, see [Fra94, Khu96].) Throughout, let n denote the number of vertices in the input graph G.

A classical and well-studied special case of SNDP is the problem of finding a minimum-cost k-vertex connected spanning subgraph, i.e., the special case where $k_{u,v} = k$ for all vertex pairs $\{u, v\}$. This is called the *k-vertex connected spanning subgraph problem* (k-VCSS). A related problem is the *vertex-connectivity augmentation problem* ($\mathrm{VCAP}_{\ell,k}$), where the goal is to find a minimum-cost set of edges that augments an ℓ-connected graph into a k-connected graph. We exhibit hardness of approximation results for both of these problems.

Other special cases of SNDP for which we show hardness of approximation are the *subset connectivity problem* and the *outconnectivity to a subset problem* (OSP). In the first problem, the input contains a subset S of the vertices, and the goal is to find a minimum-cost subgraph that contains at least k vertex-disjoint paths between every pair of vertices in S. This is SNDP with $k_{u,v} = k$ for all $u, v \in S$ and $k_{u,v} = 0$ otherwise. In the second problem, the input contains a special vertex r (called the *root*) and a subset S of the vertices, and the goal is to find a minimum-cost subgraph that contains at least k vertex-disjoint paths between r and any vertex in S. In other words, this is SNDP with $k_{r,v} = k$ for all $v \in S$ and $k_{u,v} = 0$ otherwise.

1.1 Previous work

k-VCSS is NP-hard even for $k = 2$ and *uniform costs* (i.e., all edges have the same cost), as this problem already generalizes the Hamiltonian cycle problem (note that a 2-connected subgraph of G has n edges if and only if it is a Hamiltonian cycle). By a similar argument, the outconnectivity to a subset problem is also NP-hard, even for $k = 2$ and $S = V \setminus \{r\}$ [CJN01]. It immediately follows that SNDP (which is a more general problem) is also NP-hard. $\mathrm{VCAP}_{0,2}$ is NP-hard by a similar argument [ET76], and $\mathrm{VCAP}_{1,2}$ is proved to be NP-hard in [FJ81].

Most previous work on approximating vertex-connectivity problems concentrated on upper bounds, i.e., on designing approximation algorithms. An approximation ratio of $2k$ for k-VCSS was obtained in [CJN01] by a straightforward application of [FT89], and the approximation ratio was later improved to k in [KN00]. Recently, Cheriyan, Vempala and Vetta [CVV02] devised improved approximation algorithms for the problem. Their first algorithm achieves approximation ratio $6H(k) = O(\log k)$, where $H(k)$ is the kth harmonic number, for the case where $k \le \sqrt{n/6}$. Their second algorithm achieves an approximation ratio $\sqrt{n/\epsilon}$ for the case where $k \le (1-\epsilon)n$. (An approximation ratio of $O(\log k)$ claimed in [RW97] was found to be erroneous, see [RW02].)

Better approximation ratios are known for several special cases of k-VCSS. For $k \le 7$ an approximation ratio of $\lceil (k+1)/2 \rceil$ is known (see [KR96] for $k = 2$, [ADNP99] for $k = 2, 3$, [DN99] for $k = 4, 5$, and [KN00] for $k = 6, 7$). For *metric costs* (i.e., when the costs satisfy the triangle inequality) an approximation ratio $2 + \frac{(k-1)}{n}$ is given in [KN00] (building on a ratio $2 + \frac{2(k-1)}{n}$ previously shown in [KR96]). For uniform costs, an approximation ratio of $1 + 1/k$ is obtained in [CT00]. For a complete Euclidean graph, a polynomial time approximation scheme (i.e., factor $1 + \epsilon$ for any fixed $\epsilon > 0$) is devised in [CL99].

The connectivity augmentation problem has also attracted a lot of attention. A 2-approximation for $\mathrm{VCAP}_{1,2}$ is shown in [FJ81, KT93]. In the case where every pair of vertices in the graph forms an augmenting edge of unit cost, $\mathrm{VCAP}_{k,k+1}$ is not known to be in P nor to be NP-hard. For the latter problem, a $k - 2$ additive approximation is presented in [Jor95], and optimal algorithms for small values of k are shown in [ET76, WN93, HR91, Hsu92].

The special case of OSP with $S = V \setminus \{r\}$ (called the k-outconnectivity problem), can be approximated within ratio 2, see for example [KR96]. Approximation algorithms for related problems are given in [CJN01].

In contrast, there are almost no lower bounds for approximating vertex-connectivity problems. It is shown in [CL99] that 2-VCSS is APX-hard even for bounded-degree graphs with uniform costs and for complete Euclidean graphs in $\mathbb{R}^{\log n}$. No stronger lower bound is known for the more general SNDP.

1.2 Our results

We give hardness of approximation results for several of these vertex-connectivity network design problems. In Section 2 we show that 2-VCSS is APX-*hard*, i.e., there exists some fixed $\epsilon > 0$ such that it is NP-hard to approximate 2-VCSS within ratio $1 + \epsilon$, even in the case of uniform costs. (This result is similar to, but was obtained independently from, [CL99].) We then conclude that k-VCSS is APX-hard for any $k \geq 2$, even in the case of 0-1 costs. In particular, the latter is strictly harder to approximate then k-VCSS with uniform costs (which can be approximated within ratio $1+1/k$), unless $\mathsf{P} = \mathsf{NP}$. In addition, we exhibit APX-hardness for $\mathrm{VCAP}_{1,2}$, even in the case where all edges have cost 1 or 2. From this, it follows that $\mathrm{VCAP}_{k,k+1}$ with uniform costs is APX-hard for every $k \geq 2$. For fixed k, these hardness results match, up to constant factors, the approximation algorithms mentioned in Section 1.1.

In Section 3, we show that the outconnectivity to a subset problem (OSP) cannot be approximated within a ratio of $(\frac{1}{3} - \epsilon) \ln n$ for any fixed $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{O(\log \log n)})$. It follows that OSP with a general subset S is much harder to approximate than the special case $S = V \setminus \{r\}$ (which can be approximated within ratio 2). This problem is a special case of SNDP that is already harder to approximate than the edge-connectivity variant of SNDP (which can be approximated within ratio 2), unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{O(\log \log n)})$.

In Section 4, we show that SNDP cannot be approximated within a ratio of $2^{\log^{1-\epsilon} n}$, for any fixed $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\mathrm{polylog}(n)})$. This hardness of approximation result extends also to the subset k-connectivity problem which is a special case of SNDP. The lower bound holds for $k = n^{\rho}$ where $0 < \rho < 1$ is any fixed constant. It then follows that SNDP is provably harder to approximate than k-VCSS (which can be approximated within a ratio of $O(\log k)$ for $k \leq \sqrt{n/6}$), unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\mathrm{polylog}(n)})$.

1.3 Preliminaries

For an arbitrary graph G, let $V(G)$ denote the vertex set of G and let $E(G)$ denote the edge set of G. For a nonnegative cost function c on the edges of G and a subgraph $G' = (V', E')$ of G we use the notation $c(G') = c(E') = \sum_{e \in E'} c(e)$. We denote the set of *neighbors* of a vertex v in $W \subset V$

(namely, the vertices $w \in W$ such that $(v, w) \in E$) by $N(v, W, G)$. When $W = V(G)$ we omit W and write $N(v, G)$, and when G is clear from the context, we use simply $N(v)$.

A set W of k vertices in a graph $G = (V, E)$ is called a *k-vertex-cut* (or just a *vertex-cut*) if the subgraph of G induced on $V \setminus W$ has at least two connected components. A vertex $w \in V$ is called a *cut-vertex* if $W = \{w\}$ is a vertex-cut. We will use the following classical result.

Theorem 1 (Menger's Theorem, see e.g. [Die00]).

a. A graph G contains at least k vertex-disjoint paths between two of its vertices u, v if and only if every vertex-cut that separates u from v must be of size at least k.

b. A graph G is k-vertex connected if and only if it has no $(k-1)$-vertex cut.

2 Vertex-connectivity spanning subgraph and augmentation

2.1 The k-vertex connected spanning subgraph problem

We consider the minimum-cost k-vertex connected spanning subgraph problem (k-VCSS), both in the uniform cost model, and in models with restricted edge costs. The main result of this section is that for some fixed $\epsilon > 0$ and every $k \geq 2$, it is NP-hard to approximate k-VCSS within a factor of $(1+\epsilon)$, even in the special case where edges have only costs 0 and 1.

Throughout, let k-VCSS(a, b) denote the k-VCSS problem on the *complete* graph where edges have cost a or b. Note that k-VCSS with uniform costs is just k-VCSS$(1, \infty)$.

Theorem 2. *For some fixed $\epsilon > 0$, it is NP-hard to approximate the 2-VCSS$(1, 2)$ problem within a factor of $(1+\epsilon)$.*

We prove Theorem 2 by a reduction from the MinTSP$(1, 2)$ problem: Given a complete graph $G = (V, E)$, and a weight function on edges $w : E \to \{1, 2\}$, find a minimum weight tour of G. As noted in the introduction, a similar reduction was discovered independently by Czumaj and Lingas [CL99]. We include a proof here for completeness. The following hardness of approximation result for MinTSP$(1, 2)$ follows from [PY93] (and the results of [ALM$^+$98]).

Theorem 3 ([PY93] and [ALM$^+$98]). *For some fixed $\epsilon > 0$, it is NP-hard to distinguish whether an instance of MinTSP$(1, 2)$ on n vertices has an optimal tour of cost n or every tour has cost at least $n(1+\epsilon)$.*

Theorem 2. We show a gap-preserving reduction from MinTSP$(1, 2)$. Let $G = (V, E)$ be an instance of this problem. Let MinTSP(G) be the weight of an optimal tour of G and let 2-VCSS(G) be the cost of an optimal 2-connected spanning subgraph of G, where the cost of an edge is equal to its weight. We will show that for some suitable $\epsilon' > 0$,

$$\begin{aligned} \textsc{MinTSP}(G) = n &\implies \text{2-VCSS}(G) = n \\ \textsc{MinTSP}(G) \geq n(1+\epsilon) &\implies \text{2-VCSS}(G) \geq n(1+\epsilon') \end{aligned}$$

Suppose that MinTSP$(G) = n$. Then an optimal tour of G is a cycle with n edges of weight 1. Since a cycle is the smallest possible 2-connected graph on n vertices, it follows that 2-VCSS$(G) = n$.

Now consider a graph G for which MinTSP$(G) \geq n(1+\epsilon)$ and suppose, for the sake of contradiction, that there exists a 2-connected spanning subgraph $H = (V, E_H)$ such that cost$(H) < n(1+\epsilon')$. Let $\deg_H(v)$ be the degree of a vertex v in H. Since H is 2-connected, $\deg_H(v) \geq 2$ for all v. We will repeatedly remove from H edges that are adjacent to vertices whose degree exceeds 2 until no such vertices remain, and then paste the connected components of the resulting graph H' together to obtain a tour of G. Let R be the number of edges removed in this process. Note that for any vertex v, we remove at most $\deg_H(v) - 2$ edges adjacent to that vertex. It follows that

$$R \leq \sum_{v \in V} (\deg_H(v) - 2) = 2 \cdot |E_H| - 2n \leq 2 \cdot \text{cost}(H) - 2n < 2n\epsilon'.$$

Since we remove R edges from a connected graph H, the resulting graph H' has at most $R + 1$ connected components. Each of these components consists of vertices whose degree is at most 2, and hence it is either an isolated vertex, a path, or a cycle. We arbitrarily remove one edge from every cycle without increasing the cost of H'. Finally, we add at most $R + 1$ edges to connect the paths (and isolated vertices) from end to end to obtain a tour T of G. The cost of each of these additional edges is at most 2, and thus

$$\text{weight}(T) \leq \text{cost}(H) - R + 2(R + 1) < n(1 + \epsilon') + 2n\epsilon' + 2 = n(1 + 3\epsilon') + 2.$$

Choosing $\epsilon' < \epsilon/3$ yields the desired contradiction for sufficiently large n. The theorem follows. □

It is straightforward to extend Theorem 2 to 2-VCSS with uniform costs. Indeed, restrict the above instance G of 2-VCSS(1, 2) to just the edges of cost 1. Let $\bar{G}$ denote the resulting instance of 2-VCSS with uniform costs. It is easy to see that 2-VCSS($\bar{G}$) $\geq$ 2-VCSS(G), and that if 2-VCSS(G) $= n$ then also 2-VCSS($\bar{G}$) $= n$. We thus obtain the following.

Corollary 4. *For some fixed $\epsilon > 0$, it is* NP*-hard to approximate* 2-VCSS *with uniform costs within a factor of* $(1 + \epsilon)$.

The main result of this section now follows easily from Corollary 4.

Theorem 5. *For every $k \geq 2$ and some fixed $\epsilon > 0$ (which is independent of k), it is* NP*-hard to approximate k-*VCSS *with edge costs 0 and 1 within a factor of* $(1 + \epsilon)$.

Proof. Given a $(0, 1)$-weighted graph $G = (V, E)$ which is k-connected, we construct a graph G' by adding a new vertex v' that is connected to every vertex in V by an edge of cost 0. Call this set of edges $E_{new} = \{(v', v) : v \in V\}$. We claim (below) that k-VCSS(G) $=$ $(k + 1)$-VCSS(G') (the gap is trivially preserved). The theorem then follows by induction with hardness of the base case, $k = 2$, following from Corollary 4.

To prove the claim, suppose first that $H = (V, E_H)$ is a k-connected subgraph of G. Then $H' = (V \cup \{v'\}, E_H \cup E_{new})$ is a $(k + 1)$-connected subgraph of G' and has the same cost as H. Hence, $(k + 1)$-VCSS(G') $\leq$ k-VCSS(G). Conversely, let H' be a $(k + 1)$-connected subgraph of G'. Obtain a graph H by removing v' and all incident edges from H'. H is a k-connected subgraph of G and has the same cost as H'. It follows that k-VCSS(G) $\leq$ $(k + 1)$-VCSS(G'). □

It is shown in [CT00] that k-VCSS with uniform costs is approximable within a factor of $1 + 1/k$, for any fixed k. Notice that this ratio approaches 1 as $k \to \infty$. Together with Theorem 5, this yields the following.

Corollary 6. *For a sufficiently large fixed $k > 0$, k-*VCSS *with uniform costs is strictly easier to approximate than k-*VCSS *with edge costs 0 and 1, unless* P = NP.

Finally, the analysis in Theorem 2 and the preceding corollary extend also to the corresponding edge-connectivity problem. Such a hardness of approximation result was previously shown in [Fer98] but our proof is considerably simpler.

Theorem 7. *For some fixed $\epsilon > 0$, it is* NP*-hard to approximate the minimum-cost 2-edge-connected spanning subgraph* (2-ECSS) *problem with uniform costs within a factor of* $(1+\epsilon)$.

2.2 The vertex-connectivity augmentation problem

The vertex-connectivity augmentation problem ($\mathrm{VCAP}_{k,\ell}$) is defined as follows. Given a k-connected graph $G_0 = (V, E_0)$ and a cost function $c : V \times V \to \mathbb{N}$, find a set $E_1 \subseteq V \times V$ of minimum cost so that $G_1 = (V, E_0 \cup E_1)$ is ℓ-connected. Since all graphs considered here are simple, we will not allow G_1 to contain self-loops. $\mathrm{VCAP}_{k,\ell}(a, b)$ will represent a version of the problem where edges have only cost a or b (so $c : V \times V \to \{a, b\}$). The main result of this section is that for some fixed $\epsilon > 0$ and for every $k \geq 1$, it is NP-hard to approximate $\mathrm{VCAP}_{k,k+1}(1, 2)$ within a factor of $(1+\epsilon)$.

It is possible to convert any instance of $\mathrm{VCAP}_{k_0,k_0+\alpha}$ to an "equivalent" instance of $\mathrm{VCAP}_{k_0+1,k_0+1+\alpha}$ by adding to G_0 a new vertex that is connected to every old vertex (similar to the proof of Theorem 5). This yields the following easy lemma.

Lemma 1. *If it is* NP*-hard to approximate* $\mathrm{VCAP}_{k_0,k_0+\alpha}$ *with edge costs from a set S within a factor of* $(1+\epsilon)$ *for some $\alpha \geq 1$, then the same is true of* $\mathrm{VCAP}_{k,k+\alpha}$ *with edge costs from S for any $k \geq k_0$.*

It is straightforward that α-VCSS with edge costs 0 and 1 can be thought of as $\mathrm{VCAP}_{0,\alpha}(0, 1, \infty)$, by setting the cost of each $(u, v) \notin E$ to ∞. (Note that we need not use ∞; instead we can make each edge not in E have cost n^2 so that no optimal solution uses such an edge). Therefore, combining Theorem 5 and Lemma 1 yields the following result.

Corollary 8. *For any $k \geq 2$, for any $\alpha \geq 2$, and for some fixed $\epsilon > 0$ (independent of α and k), it is* NP*-hard to approximate* $\mathrm{VCAP}_{k,k+\alpha}(0, 1, \infty)$ *within a factor of* $(1+\epsilon)$.

Notice that the above corollary is not applicable to the case $\alpha = 1$. Furthermore, one may wonder about the difficulty of the problem when the edge weights are severely restricted, say to only 1 and 2, or in a model with uniform edge costs. We resolve both of these concerns in the following theorem.

Theorem 9. *For any $k \geq 1$ and some fixed $\epsilon > 0$ (independent of k), it is* NP*-hard to approximate* $\mathrm{VCAP}_{1,2}(1, 2)$ *within a factor of* $(1+\epsilon)$.

The proof of theorem 9 employs a reduction from 3-dimensional matching (3DM) that was used in [FJ81] to prove that solving $\mathrm{VCAP}_{1,2}(1, 2)$ (optimally) is NP-hard. We obtain a stronger result (i.e., hardness of approximation) by a more involved analysis of the reduction (and by relying on the hardness of approximating a bounded version of the 3-dimensional matching problem, shown in [Pet94]). Due to space considerations, the proof is deferred to the full version of this paper. The main result of this section follows by applying Lemma 1 to the above theorem.

Corollary 10. *For any $k \geq 1$ and some fixed $\epsilon > 0$ (independent of k), it is* NP*-hard to approximate* $\mathrm{VCAP}_{k,k+1}(1, 2)$ *within a factor of* $(1+\epsilon)$. *The same result holds for* $\mathrm{VCAP}_{k,k+1}(1, \infty)$, *i.e.* VCAP *with uniform costs.*

Remark. The techniques used to prove Theorem 9 can be used to show that a number of other connectivity augmentation problems are APX-hard (including the edge-connectivity variant of VCAP). Also, both Theorem 5 and Corollary 10 remain true even when $k = k(n)$ grows with n and $k(n)$ is computable in polynomial time.

3 The outconnectivity to a subset problem

We prove the following theorem in the full version of this paper. The proof is based on a hardness result shown by Feige et al. [FHKS02] for the problem of packing set-covers (in a set-system).

Theorem 11. *The outconnectivity to a subset problem cannot be approximated within a ratio of* $(\frac{1}{3} - \epsilon) \ln n$ *for any fixed* $\epsilon > 0$*, unless* $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{O(\log \log n)})$.

4 Survivable network design and the subset connectivity problem

In this section we show hardness results for approximating the subset connectivity problem, and thus also SNDP (which is a more general problem), within a ratio of $2^{\log^{1-\epsilon} n}$ for any fixed $\epsilon > 0$. These lower bounds are proven by a reduction from a graph-theoretic problem, called MinRep that is defined in [Kor01]. This problem is closely related to the LabelCover$_{\max}$ problem of [AL96] and to the parallel repetition theorem of [Raz98]. We first describe the MinRep problem and the hardness results known for it in Section 4.1. Then, in Section 4.2, we give a reduction from MinRep to SNDP. Finally, we adapt this reduction to the subset connectivity problem in Section 4.3.

4.1 The MinRep problem

Arora and Lund [AL96] introduced the LabelCover$_{\max}$ problem as a graph-theoretic description of one-round two-prover proof systems, for which the parallel repetition theorem of Raz [Raz98] applies. The MinRep problem described below is closely related to LabelCover$_{\max}$ and was defined in [Kor01] for the same purpose.

The input to the MinRep problem consists of a bipartite graph $G(A, B, E)$, with an explicit partitioning of each of A and B into equal-sized subsets, namely, $A = \bigcup_{i=1}^{q_A} A_i$, $B = \bigcup_{j=1}^{q_B} B_j$ where all the sets A_i have the same size m_A and all the sets B_j have the same size m_B. The bipartite graph G induces a *super-graph* H, as follows. The *super-vertices* (i.e., the vertices of H) are the $q_A + q_B$ sets A_i and B_j. A *super-edge* (i.e., an edge in H) connects two super-vertices A_i and B_j if there exist $a \in A_i$ and $b \in B_j$ that are adjacent in G (i.e., $(a, b) \in E$).

A pair (a, b) *covers* a super-edge (A_i, B_j) if $a \in A_i$ and $b \in B_j$ are adjacent in G (i.e., $(a, b) \in E$). Let $A'_i \subseteq A_i$ and $B'_j \subseteq B_j$. (The vertices of A'_i and B'_j can be thought of as *representatives* of A_i and B_j, respectively.) We say that $A'_i \cup B'_j$ *covers* the super-edge (A_i, B_j) if there exists a pair (a, b) that covers the super-edge (A_i, B_j), namely, there exists a vertex $a_i \in A'_i$ and a vertex $b_j \in B'_j$ such that $(a_i, b_j) \in E$.

The goal in the MinRep problem is to select representatives from the sets A_i and B_j such that all the super-edges are covered and the total number of representatives selected is minimal. That is, we wish to find subsets $A' \subseteq A$ and $B' \subseteq B$ with minimal total size $|A| + |B|$, such that for every super-edge (A_i, B_j) there exist representatives $a \in A' \cap A_i$ and $b \in B' \cap B_j$ that are adjacent in G (i.e., $(a, b) \in E$).

For our purposes, it is convenient (and possible) to restrict the MinRep problem so that for every super-edge (A_i, B_j), each vertex in B_j is adjacent to at most one vertex in A_i. We call this additional property of G the *star property* because it is equivalent to saying that for every super-edge (A_i, B_j) the subgraph of G induced on $A_i \cup B_j$ is a collection of vertex-disjoint stars whose centers are in A_i. [1] (See Figure 1 for an illustration.)

[1]A *star* is a graph all of whose vertices have degree 1, except for one vertex that has degree larger than 1. The vertex with degree larger than 1 is called the *center* of the star. The vertices of degree 1 are called *leaves* of the star.

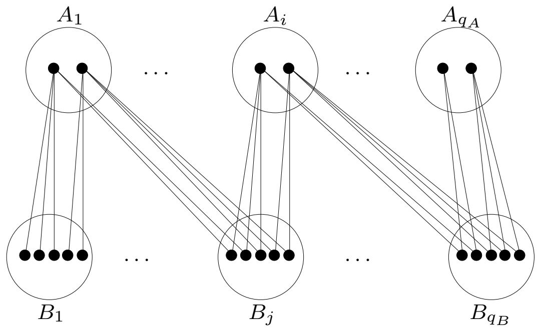

Figure 1: An instance of MinRep with the star property

The next theorem follows by a straightforward application of the parallel repetition theorem of Raz [Raz98], since the MinRep problem is graph-theoretic description of two-prover one-round proof systems. The additional property of the MinRep instance (namely, the star property) is achieved by using a specific proof system for Max3SAT(5). A description can be found in [Fei98, Section 2.2].

Theorem 12. *Let L be any NP-complete language and fix $\epsilon > 0$. Then there exists an algorithm (i.e., a reduction), whose running time is quasi-polynomial, namely $n^{\text{polylog}(n)}$, and that given an instance x of L produces an instance $G(A, B, E)$ of the* MinRep *problem with the star property, such that the following holds.*

- *If $x \in L$ then the* MinRep *instance G has a solution of value $q_A + q_B$ (namely, with one representative from each A_i and one from each B_j).*
- *If $x \notin L$ then the value of any solution of the* MinRep *instance G is at least $(q_A+q_B) \cdot 2^{\log^{1-\epsilon} n}$, where n is the number of vertices in G.*

Hence, the MinRep *problem cannot be approximated within ratio $2^{\log^{1-\epsilon} n}$, for any fixed $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\text{polylog}(n)})$.*

4.2 The survivable network design problem

We prove the following theorem by a reduction whose starting point is Theorem 12.

Theorem 13. *The* SNDP *problem cannot be approximated within ratio $2^{\log^{1-\epsilon} n}$, for any fixed $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\text{polylog}(n)})$.*

The reduction. Given the instance $G(A, B, E)$ of the MinRep problem as described in Section 4.1, create an instance $\bar{G}(\bar{V}, \bar{E})$ of SNDP as follows. (See Figure 2 for illustration.)

1. Take G and let all its edges have cost 0.
2. For each $i = 1, \ldots, q_A$ create a new vertex u_i that is connected to every vertex in A_i by an edge of cost 1. Similarly, for each $j = 1, \ldots, q_B$ create a new vertex w_j that is connected to every vertex in B_j by an edge of cost 1. Let $U = \{u_1, \ldots, u_{q_A}\}$ and $W = \{w_1, \ldots, w_{q_B}\}$.

3. For each super-edge (A_i, B_j) create two new vertices x_i^j and y_j^i. For every i, let $X_i = \{x_i^j\}$ consist of the d vertices x_i^j (note that the values of j need not be consecutive), and connect every vertex of X_i to u_i by edges of cost 0. Similarly, for every j let $Y_j = \{{y_j}^i\}$ and connect every vertex in Y_j to w_j by an edge of cost 0.

4. For every super-edge (A_i, B_j) connect every vertex in $\{x_i^j, y_j^i\}$ to every vertex in $(A \setminus A_i) \cup (B \setminus B_j)$ by an edge of cost 0.

5. Let $X = \cup_{i=1}^{q_A} X_i$ and $Y = \cup_{j=1}^{q_B} Y_j$. Connect every two vertices in $X \cup Y$ by an edge of cost 0.

6. Finally, require $k = |X| + |Y| + (q_A - 1)m_A + (q_B - 1)m_B$ vertex-disjoint paths from x_i^j to y_j^i for every super-edge (A_i, B_j). (Recall that $|X| = |Y| = q_A \cdot d_A = q_B \cdot d_B$.)

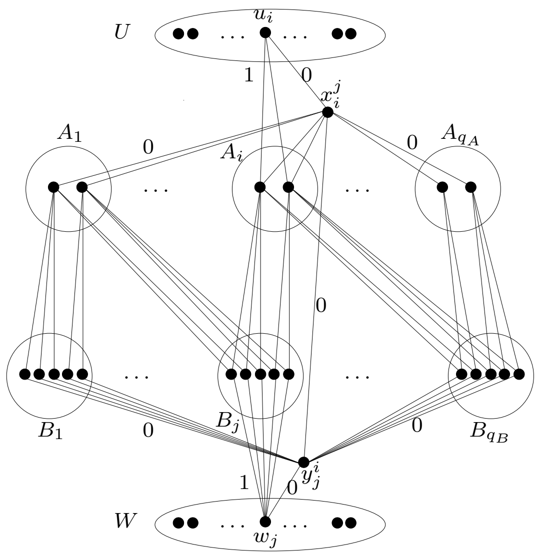

Figure 2: The vertices x_i^j, y_j^i in the SNDP instance $\bar{G}$

The analysis. Suppose $x \in L$ and then by Theorem 12 there exists a choice of $q_A + q_B$ representatives (one representative from each A_i and one from each B_j) that cover all the super-edges. Let G' be the subgraph of $\bar{G}$ that contains an edge between each u_i and the representative chosen in A_i, an edge between each w_j and the representative chosen in B_j, and all the edges of cost 0 in $\bar{G}$. Let us now show that G', whose cost is clearly $2q$, is a solution to the instance $\bar{G}$ of the SNDP problem. Consider a pair of vertices x_i^j, y_j^i such that (A_i, B_j) is a super-edge in G. Each vertex in $\mathcal{F}_{i,j} = (X \setminus \{x_i^j\}) \cup (Y \setminus \{y_j^i\}) \cup (A \setminus A_i) \cup (B \setminus B_j)$ defines a path of length 2 in G' between x_i^j and y_j^i, and the edge (x_i^j, y_j^i) defines a path of length 1, so we get a total of $|X| - 1 + |Y| - 1 + (q_A - 1)m_A + (q_B - 1)m_B + 1 = k - 1$ vertex-disjoint paths between x_i^j and y_j^i. There is an additional path that goes through $V \setminus \mathcal{F}_{i,j} = U \cup W \cup A_i \cup B_j \cup \{x_i^j, y_j^i\}$, namely, $x_i^j - u_i - a_i - b_j - w_j - y_j^i$ where a_i and b_j are the representatives chosen from A_i and B_j, respectively. This additional path is clearly vertex-disjoint from the others, yielding k vertex-disjoint paths between x_i^j and y_j^i.

The next lemma will be used to complete the proof of Theorem 13. Let G' be a feasible solution to this instance of the SNDP problem, i.e. a subgraph of $\bar{G}$ in which x_i^j is k-vertex connected to y_j^i for every super-edge (A_i, B_j).

Lemma 2. *For every super-edge (A_i, B_j), the subgraph G' contains an edge connecting u_i to some $a_i \in A_i$, and an edge connecting w_j to some $b_j \in B_j$, such that $(a_i, b_j) \in E$ (i.e., the pair (a_i, b_j) covers the super-edge).*

Proof. Since G' is a feasible solution, it contains k vertex-disjoint paths between x_i^j and y_j^i. Let $\mathcal{F}_{i,j} = (X \setminus \{x_i^j\}) \cup (Y \setminus \{y_j^i\}) \cup (A \setminus A_i) \cup (B \setminus B_j)$. At most $|\mathcal{F}_{i,j}| = |X| - 1 + |Y| - 1 + (q_A - 1)m_A + (q_B - 1)m_B = k - 2$ of these k paths can visit vertices of $\mathcal{F}_{i,j}$, and at most one of these paths can use the edge (x_i^j, y_j^i). (This is essentially a "mixed" cut that contains both vertices and edges; see Figure 3.) Hence, G' contains a path between x_i^j and y_j^i, that visits only vertices of $\bar{V} \setminus \mathcal{F}_{i,j} = U \cup W \cup A_i \cup B_j \cup \{x_i^j, y_j^i\}$ and whose length is at least two.

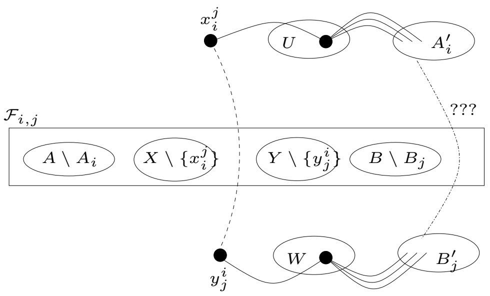

Figure 3: A mixed cut with $k - 2$ vertices and one edge.

Observe that in the subgraph of G' induced on $\bar{V} \setminus \mathcal{F}_{i,j}$ the following holds. (Assume without loss of generality that G' contains all the edges of cost 0 in $\bar{G}$.) The only neighbor of x_i^j is u_i, so the vertices at distance 2 from x_i^j (i.e., the neighbors of u_i except for x_i^j) form a subset A'_i of A_i. Thus, the vertices at distance 3 from x_i^j (i.e. all the neighbors of A'_i except for u_i) are all from B_j. Similarly, the only neighbor of y_j^i is w_j, so vertices at distance 2 from y_j^i form a subset B'_j of B_j, and all vertices at distance 3 from y_j^i are from A_i. Note that the subgraph of $\bar{G}$ induced on $A_i \cup B_j$ is a collection of vertex-disjoint stars, whose centers are in A_i and whose leaves are in B_j. The aforementioned path in G' between x_i^j and y_j^i (that visits only vertices of $\bar{V} \setminus \mathcal{F}_{i,j}$) then must be have the form $x_i^j - u_i - a_i - b_j - w_j - y_j^i$ with $a_i \in A'_i$ and $b_j \in B'_j$ (note that the other vertices of $U \cup W$ are unreachable), and the lemma follows. $\square$

We now complete the proof of Theorem 13. Suppose that $x \notin L$ and let G' be a feasible solution to the instance $\bar{G}$ of the SNDP problem. Let A'_i be the set of neighbors of u_i among A_i (in G'), and let B'_j be the set of neighbors of w_j among B_j (in G'). By Lemma 2 the representatives $A' = \cup_i A'_i$ and $B' = \cup_j B'_j$ cover all the super-edges (A_i, B_j), thus forming a feasible solution to the MinRep instance G. By Theorem 12 the value of this MinRep solution is $|A'| + |B'| \geq (q_A + q_B) \cdot 2^{\log^{1-\epsilon} n}$, where n denotes the number of vertices in G. Observe that the cost of G' is also $|A'| + |B'|$. Since the number of vertices in $\bar{G}$ is $n^{O(1)}$, the cost of G' is at least $(q_A + q_B) \cdot 2^{\log^{1-\epsilon} |V(\bar{G})|}$, and the theorem follows.

4.3 The subset k-connectivity problem

We adapt the reduction of Theorem 13 to the subset k-connectivity problem, as follows. We require that the subset $S = X \cup Y$ is k-vertex connected. For this S to be k-vertex connected in the case $x \in L$, we add, for every $z, z' \in S$ that are not a pair x_i^j, y_j^i, a set $Q_{z,z'}$ of k new vertices that are all connected to z and to z' by edges of cost 0. The analysis of the case $x \notin L$ (including the proof of Lemma 2) remains valid. Specifically, in the mixed cut defined by the vertices $\mathcal{F}_{i,j}$ and the edge (x_i^j, y_j^i) (see Figure 3), the sets $Q_{x_i^j,z}$ are on the side of x_i^j and similarly for y_j^i, and all other sets $Q_{z,z'}$ are isolated cliques. We thus obtain the following hardness of approximation result for the subset k-connectivity problem.

Theorem 14. *The subset k-connectivity problem cannot be approximated within ratio $2^{\log^{1-\epsilon} n}$, for any fixed $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\mathrm{polylog}(n)})$.*

Note that in our reduction above k is $\Theta(|E(\bar{G})|)$ while the number of vertices is $\Theta(k|E(\bar{G})|^2)$. Therefore, our hardness result for the subset k-connectivity applies for $k = \Theta(n^{1/3})$, where n denotes the number of vertices in the input graph. In this problem, it is straightforward to achieve $k = n^\alpha$ for any fixed $0 < \alpha < 1$ by adding sufficiently many vertices that are either isolated or connected to all other vertices by edges of cost 0.

It follows that the min-cost subset k-connectivity problem is provably harder to approximate than the min-cost k-connectivity problem (for values of k as above). Indeed, it is shown in [CVV02] that the latter problem can be approximated within ratio $O(\log k)$ for $k \le \sqrt{n/6}$.

References

[ADNP99] V. Auletta, Y. Dinitz, Z. Nutov, and D. Parente. A 2-approximation algorithm for finding an optimum 3-vertex-connected spanning subgraph. *J. Algorithms*, 32(1):21–30, 1999.

[AL96] S. Arora and C. Lund. Hardness of approximations. In D. Hochbaum, editor, *Approximation Algorithms for NP-Hard Problems*. PWS Publishing Company, 1996.

[ALM+98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998.

[CJN01] J. Cheriyan, T. Jordán, and Z. Nutov. On rooted node-connectivity problems. *Algorithmica*, 30(3):353–375, 2001.

[CL99] A. Czumaj and A. Lingas. On approximability of the minimum-cost k-connected spanning subgraph problem. In *Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 281–290. ACM, 1999.

[CT00] J. Cheriyan and R. Thurimella. Approximating minimum-size k-connected spanning subgraphs via matching. *SIAM J. Comput.*, 30(2):528–560, 2000.

[CVV02] J. Cheriyan, S. Vempala, and A. Vetta. Approximation algorithms for minimum-cost k-vertex connected subgraphs. In *34th Annual ACM Symposium on the Theory of Computing*, 2002. To appear.

[Die00] R. Diestel. *Graph theory*. Springer-Verlag, New York, second edition, 2000.

[DN99] Y. Dinitz and Z. Nutov. A 3-approximation algorithm for finding optimum 4, 5-vertex-connected spanning subgraphs. *J. Algorithms*, 32(1):31–40, 1999.

[ET76] K. P. Eswaran and R. E. Tarjan. Augmentation problems. *SIAM J. Comput.*, 5(4):653–665, 1976.

[Fei98] U. Feige. A threshold of $\ln n$ for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

[Fer98] C. G. Fernandes. A better approximation ratio for the minimum size k-edge-connected spanning subgraph problem. *J. Algorithms*, 28(1):105–124, 1998.

[FHKS02] U. Feige, M. M. Halldórsson, G. Kortsarz, and A. Srinivasan. Approximating the domatic number. *SIAM J. Comput.*, 2002. To appear.

[FJ81] G. N. Frederickson and J. JáJá. Approximation algorithms for several graph augmentation problems. *SIAM J. Comput.*, 10(2):270–283, 1981.

[FJW01] L. Fleischer, K. Jain, and D. P. Williamson. An iterative rounding 2-approximation algorithm for the element connectivity problem. In *42nd Annual IEEE Symposium on Foundations of Computer Science*, pages 339–347, 2001.

[Fra94] A. Frank. Connectivity augmentation problems in network design. In J. R. Birge and K. G. Murty, editors, *Mathematical Programming: State of the Art 1994*, pages 34–63. The University of Michigan, 1994.

[FT89] A. Frank and É. Tardos. An application of submodular flows. *Linear Algebra Appl.*, 114/115:329–348, 1989.

[HR91] T. Hsu and V. Ramachandran. A linear time algorithm for triconnectivity augmentation. In *32nd Annual IEEE Symposium on Foundations of Computer Science*, pages 548–559, 1991.

[Hsu92] T. Hsu. On four-connecting a triconnected graph. In *33nd Annual IEEE Symposium on Foundations of Computer Science*, pages 70–79, 1992.

[Jai01] K. Jain. A factor 2 approximation algorithm for the generalized Steiner network problem. *Combinatorica*, 21(1):39–60, 2001.

[Jor95] T. Jordán. On the optimal vertex-connectivity augmentation. *J. Combin. Theory Ser. B*, 63(1):8–20, 1995.

[Khu96] S. Khuller. Approximation algorithms for finding highly connected subgraphs. In D. Hochbaum, editor, *Approximation Algorithms for NP-Hard Problems*. PWS Publishing Company, 1996.

[KN00] G. Kortsarz and Z. Nutov. Approximating node connectivity problems via set covers. In *3rd International workshop on Approximation algorithms for combinatorial optimization (APPROX)*, pages 194–205. Springer, 2000.

[Kor01] G. Kortsarz. On the hardness of approximating spanners. *Algorithmica*, 30(3):432–450, 2001.

[KR96] S. Khuller and B. Raghavachari. Improved approximation algorithms for uniform connectivity problems. *J. Algorithms*, 21(2):434–450, 1996.

[KT93] S. Khuller and R. Thurimella. Approximation algorithms for graph augmentation. *J. Algorithms*, 14(2):214–225, 1993.

[Pet94] E. Petrank. The hardness of approximation: gap location. *Comput. Complexity*, 4(2):133–157, 1994.

[PY93] C. H. Papadimitriou and M. Yannakakis. The traveling salesman problem with distances one and two. *Math. Oper. Res.*, 18(1):1–11, 1993.

[Raz98] R. Raz. A parallel repetition theorem. *SIAM J. Comput.*, 27(3):763–803, 1998.

[RW97] R. Ravi and D. P. Williamson. An approximation algorithm for minimum-cost vertex-connectivity problems. *Algorithmica*, 18(1):21–43, 1997.

[RW02] R. Ravi and D. P. Williamson. Erratum: An approximation algorithm for minimum-cost vertex-connectivity problems. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1000–1001, 2002.

[Vaz01] V. V. Vazirani. *Approximation algorithms*. Springer-Verlag, Berlin, 2001.

[WN93] T. Watanabe and A. Nakamura. A minimum 3-connectivity augmentation of a graph. *J. Comput. System Sci.*, 46(1):91–128, 1993.