A primal-dual schema based approximation algorithm for the element connectivity problem

Kamal Jain* Ion Măndoiu† Vijay V. Vazirani‡ David P. Williamson§

Abstract

The element connectivity problem falls in the category of survivable network design problems – it is intermediate to the versions that ask for edge-disjoint and vertex-disjoint paths. The edge version is by now well understood from the view-point of approximation algorithms [17, 5, 8], but very little is known about the vertex version. In our problem, vertices are partitioned into two sets: terminals and non-terminals. Only edges and non-terminals can fail – we refer to them as *elements* – and only pairs of terminals have connectivity requirements, specifying the number of element-disjoint paths required. Our algorithm achieves an approximation guarantee of factor $2H_k$, where k is the largest requirement and $H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n}$. Besides providing possible insights for solving the vertex-disjoint paths version, the element connectivity problem is of independent interest, since it models a realistic situation.

1 Introduction

Given an undirected graph $G = (V, E)$ with non-negative costs c_e for edges $e \in E$, and a value r_{uv} for each pair of vertices $u, v \in V$, the *survivable network design problem* (SNDP) is that of finding a minimum-cost subgraph such that there are r_{uv} disjoint paths between each pair of vertices u and v. The paths can be required to be either edge-disjoint or vertex-disjoint; we refer to the former as the *edge-connectivity SNDP* (EC-SNDP) and the latter as the *vertex-connectivity SNDP* (VC-SNDP). The survivable network design problem is a natural generalization of the Steiner tree problem, and captures the problem of designing a minimum-cost network such that u and v are still connected in the network after up to $r_{uv} - 1$ links fail (for EC-SNDP) or up to $r_{uv} - 1$ links or nodes fail (VC-SNDP). The survivable network design problem arises from problems in the telecommunications industry (c.f. [7, 11]) and has been studied from many different approaches including polyhedral combinatorics [15, 7], interchange heuristics [12], min-max relations [1] (in the unweighted case), approximation algorithms [17, 5, 13], and implementations thereof [11]. In this paper, we consider approximation algorithms for the SNDP. A ρ-approximation algorithm for the SNDP runs in polynomial time and finds a solution of value no more than ρ times the value of an optimal solution.

There appears to be a qualitative difference in difficulty between EC-SNDP and VC-SNDP. For example, although an exact min-max formula is known for the number of edges needed to add to

*Microsoft Research. This research was performed while the author was a student at the Georgia Institute of Technology and supported by NSF Grant CCR 9627308. E-mail: kamalj@microsoft.com.

†University of California at San Diego. This research was performed while the author was a student at the Georgia Institute of Technology and supported by NSF Grant CCR 9627308. Email: mandoiu@cs.ucsd.edu.

‡College of Computing, Georgia Institute of Technology. Supported by NSF Grant CCR 9627308. Email: vazirani@cc.gatech.edu.

§IBM Almaden Research Center. E-mail: dpw@almaden.ibm.com.

a graph to have it satisfy the edge-disjoint paths constraint [1], no similar formula is known in the case of the vertex-disjoint paths.

The first approximation algorithm for EC-SNDP, achieving a guarantee of factor $2k$, where $k = \max_{u,v} r_{uv}$, was given by Williamson, Goemans, Mihail and Vazirani [17]. This paper also formalized a basic mechanism for using the primal-dual schema. The algorithm of [17] picks edges for the desired subgraph in k *phases*. The augmentation problem for each phase can be written as an integer program, the LP-relaxation of which is solved within factor 2 by the primal-dual process alluded to above. A clever reordering of the augmentations led to a $2H_k$-approximation algorithm due to Goemans, Goldberg, Plotkin, Shmoys, Tardos and Williamson [5], where $H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n} \approx \ln n$. Jain has given a 2-approximation algorithm for the problem [8], using a rounding-based algorithm. For a detailed overview of these algorithmic ideas and the primal-dual schema, we refer the reader to the survey article [6] or the book [16].

In the case of VC-SNDP, however, very little is known in terms of approximation algorithms. Using the basic algorithmic outline established in [17, 5], Ravi and Williamson [13, 14] give a 3-approximation algorithm in the case that $r_{uv} \in \{0, 1, 2\}$. For the problem in which $r_{uv} = k$ for all $u, v \in V$, also known as the minimum-cost k-vertex-connected subgraph problem, there is also a $(2 + \frac{2(k-1)}{n})$-approximation algorithm due to Khuller and Raghavachari [10] in the case that edge costs obey the triangle inequality. However, no non-trivial approximation algorithm is known for the vertex-connectivity survivable network design problem in its full generality.

In this paper we make progress on this important problem by considering a natural problem intermediate to EC-SNDP and VC-SNDP. We call it the *element connectivity survivable network design problem* (ELC-SNDP). In this version of the problem, the vertices are partitioned into *terminals* and *nonterminals*. Nonterminals and edges can fail; these are the *elements*. On the other hand, terminals cannot fail. Further, for each pair of terminals, u, v, we are given a connectivity requirement r_{uv}. The problem is to find a minimum-cost subgraph such that for each pair of terminals, u, v, despite the failure of any $r_{uv} - 1$ elements, there is still a path left connecting u and v; that is, there are r_{uv} *element*-disjoint paths between each pair of terminals u and v. Notice that nonterminals do not have any connectivity requirements. This model is realistic, since in many practical situations, the terminals are robust and do not fail, whereas intermediate nodes, which do not have connectivity requirements, do fail. In the VC-SNDP, all vertices and all edges are allowed to fail. The EC-SNDP is a special case of the ELC-SNDP with an empty set of nonterminals.

Our central result is a $2H_k$-approximation algorithm for this problem, where $k = \max_{u,v} r_{uv}$ is the largest connectivity requirement. Our algorithm also follows the basic algorithmic and proof outline established in [17, 5]. The main difficulty in solving VC-SNDP is that there is no known way of writing the augmentation problem for each phase as an integer program. (An example of the kinds of constraints one gets using the usual way of breaking the problem into phases is: pick either edge e_1 or both e_2 and e_3.) The problem ELC-SNDP is defined precisely in a way to get around this difficulty. However, even after the integer program is written, the ideas of [17, 5] do not apply in a straightforward manner.

The remainder of the paper is structured as follows. In Section 2 we give the integer programming formulation and its LP-relaxation. In Section 3 we show how the problem is decomposed into phases and we prove the approximation guarantee, assuming the correct implementation of a phase. In Section 4 we give the phase algorithm. In the last section we give a tight example, thereby showing that no better guarantee can be established for our algorithm.

Since the appearance of an extended abstract of this result [9], several related results have appeared. Cheriyan, Vempala, and Vetta [2] give a $6H_k$-approximation algorithm for VC-SNDP that contain at least $6k^2$ vertices. A $2H_k$-approximation algorithm for ELC-SNDP is also obtained as a special case of an algorithm by Zhao, Nagamochi, and Ibaraki [18]. Fleischer [3] has obtained a 2-approximation algorithm for VC-SNDP when $r_{ij} \in \{0, 1, 2\}$, and Fleischer, Jain, and Williamson [4] have developed a 2-approximation algorithm for ELC-SNDP. The latter two results draw upon the rounding algorithm and analysis developed by Jain [8] for EC-SNDP.

2 The Problem, its Integer Programming Formulation and LP-Relaxation

Let $G = (V, E)$ be an undirected graph with non-negative costs c_e on edges. The set V is partitioned into two disjoint sets R and S. R is the set of *terminals*; there is a non-negative connectivity requirement r_{uv} between each pair of terminals. We assume that these vertices are reliable. On the other hand vertices in S, also known as nonterminals, and all the edges are unreliable. We call the members of $S \cup E$ *elements*. We define the element connectivity problem as choosing a minimum-cost set of edges $E' \subseteq E$, so that in the subgraph $H = (V, E')$ for every pair u and v, there still remains a path between them in case $r_{uv} - 1$ elements fail. In other words there are r_{uv} element disjoint paths between u and v. For convenience we extend the definition of r_{uv} to any pair u and v of vertices by assuming that $r_{uv} = 0$ if at least one of u and v is a nonterminal.

Let H be a feasible solution to this problem. Suppose that a set of elements $A \subseteq S \cup E$ has failed. Now the surviving graph, $H - A$, should have at least $r_{uv} - |A|$ element-disjoint paths for every pair u and v. Note that as far as the integer program is concerned, for a pair u and v, it is sufficient to consider A's of cardinality $r_{uv} - 1$ only, and write cut constraints that ensure the existence of a path from u to v. However, it is easy to show that the LP-relaxation of this integer program has a bad integrality gap. Consider an unweighted complete graph on n vertices from which we want to select a low-cost k-edge-connected subgraph. Since the degree of every vertex in the optimal solution should be at least k, the cost of optimal solution is at least $\frac{nk}{2}$. If we form the linear program only for A's of cardinality $k - 1$ then picking every edge to the extent of $\frac{1}{n-k}$ will be a valid feasible solution. This gives us the integrality gap of at least $\frac{k(n-k)}{n-1}$. Choosing $k = \frac{n}{2}$ makes the integrality gap at least $\frac{n}{4}$.

Hence, we include in the integer program constraints corresponding to sets A of all cardinalities. As a corollary of our algorithm, we show that the LP-relaxation of this integer program has integrality gap bounded by $2H_k$.

Let $f(B) = \max_{u \in B, v \notin B} r_{uv}$. The integer program is:

$$\min \sum_{e \in E} c_e x_e \tag{1}$$

subject to

$$\forall A \subseteq S \cup E, B \subseteq V - A : \quad \sum_{e \in \delta_{G-A}(B)} x_e \geq f(B) - |A|$$

$$\forall e \in E : \quad x_e \in \{0, 1\},$$

where $\delta_{G-A}(B)$ denotes the set of edges with one endpoint in B after removing A from graph G.

To get the linear program we further relax the condition $x_e \in \{0, 1\}$ to $x_e \geq 0$. The following is the dual of the above linear program,

$$(2) \qquad \max \sum_{A\subseteq S\cup E, B\subseteq V-A} (f(B)-|A|)y_{BA}$$

subject to

$$\forall e\in E: \quad \sum_{B,A: A\subseteq S\cup E,\, B\subseteq V-A,\, e\in\delta_{G-A}(B)} y_{BA} \le c_e$$

$$\forall A\subseteq S\cup E, B\subseteq V-A: \quad y_{BA}\ge 0.$$

Let OPT be the optimal cost of IP (1). By weak duality theorem any solution to (2) will be a lower bound on OPT.

We will use the fact that f is *weakly supermodular* [5], i.e., $f(V)=0$ and, for every $A,B\subseteq V$, at least one of the following holds

- $f(A)+f(B)\le f(A-B)+f(B-A)$
- $f(A)+f(B)\le f(A\cap B)+f(A\cup B)$.

3 High level description of the algorithm

Given an infeasible solution H to IP (1), define the *deficiency of a constraint* as the difference between the right-hand side and the left-hand side of the constraint. Only unsatisfied constraints will have positive deficiency. The deficiency of a set $B\subseteq V$ is defined to be the maximum deficiency of a constraint in which B is involved.

As in [5] our algorithm has k phases numbered from k to 1. We design the algorithm so that at the start of p^{th} phase, the deficiency of a set can be at most p and at the end of the phase it can be at most $p-1$. So at the end of the 1^{st} phase we have a feasible solution and we output that.

Let H be the partial solution constructed at the beginning of the p^{th} phase. Let h be defined by

$$h(B)=\begin{cases}1, & \text{if deficiency of } B \text{ is } p\\ 0, & \text{otherwise.}\end{cases}$$

Let $\Gamma_H(B)$ be the set of nonterminals which are not in B but have a neighbor with respect to H in B. Let $\rho_H(B)$ be the set of those edges of H which have one endpoint in B and the other in $R-B$. Finally, we define the *element neighborhood of set B w.r.t. H* to be $\varepsilon_H(B)=\Gamma_H(B)\cup\rho_H(B)$.

Observation 3.1 *The deficiency of B is $f(B)-|\varepsilon_H(B)|$ and is the deficiency of the constraint corresponding to the set pair $\varepsilon_H(B)$ and B. Moreover inclusion of any edge of $E-H$ which decreases the deficiency of this constraint will also decrease the deficiency of B.*

So the integer program we want to solve for the p^{th} phase is:

$$(3) \qquad \min \sum_{e\in E-H} c_e x_e$$

subject to

$$\forall B\subseteq V: \quad \sum_{e\in\delta_{G-\varepsilon_H(B)}(B)} x_e \ge h(B)$$

$$\forall e\in E-H: \quad x_e\in\{0,1\}$$

Let I be a feasible solution of this integral program. By Observation 3.1 the deficiency of any set with respect to $I \cup H$ will be at most $p-1$.

By relaxing $x_e \in \{0,1\}$ to $x_e \geq 0$ we get a linear program, whose dual is:

$$\max \sum_{B \subseteq V} h(B) y_B \tag{4}$$

subject to

$$\forall e \in E - H : \quad \sum_{B: e \in \delta_{G-\varepsilon_H(B)}(B)} y_B \leq c_e$$

$$y_B \geq 0$$

In the next section we will prove:

Theorem 3.2 *We can find in polynomial time a feasible solution, F, for IP (3) and a feasible solution, y, for LP (4) such that* $\text{cost}(F) \leq 2 \sum_{B \subseteq V} h(B) y_B$.

Let y be as in the above theorem. Let

$$\overline{y}_{BA} = \begin{cases} y_B, & \text{if } A = \varepsilon_H(B) \\ 0, & \text{otherwise.} \end{cases}$$

It is easy to verify that $\overline{y}$ is a feasible solution for LP (2).

Lemma 3.3

$$\text{cost}(F) \leq \frac{2}{p} \cdot \text{OPT}.$$

Proof. By definition, $h(B) = 1$ iff $f(B) - |\varepsilon_H(B)| = p$. So,

$$\sum_{B \subseteq V} h(B) y_B = \sum_{B \subseteq V} h(B) \overline{y}_{B\varepsilon_H(B)} = \frac{1}{p} \sum_{B \subseteq V} (f(B) - |\varepsilon_H(B)|) \overline{y}_{B\varepsilon_H(B)} \leq \frac{1}{p} \cdot \text{OPT}.$$

The claim follows immediately from Theorem 3.2. ■

Corollary 3.4 *The cost of edges chosen by the algorithm in all k phases is at most*

$$2\left(\frac{1}{k} + \frac{1}{k-1} + \ldots + 1\right) \cdot \text{OPT} = 2H_k \cdot \text{OPT}.$$

4 The algorithm for a phase, and its analysis

4.1 The algorithm

We now present the algorithm for a phase, which will augment sets whose deficiency is p. Our augmentation algorithm follows those in [17] and [5]. We first give the algorithm, then turn to stating and proving Theorem 3.2.

Let I be an infeasible solution to IP (3). A set is *violated* with respect to I if the constraint corresponding to it in IP (3) is unsatisfied. A violated set is *active* if none of its proper subsets is violated. Our algorithm for a phase is as follows:

<u>*Algorithm for phase p*</u>

$I \longleftarrow \emptyset$, $y \leftarrow 0$, $i \leftarrow 0$
While there are violated sets w.r.t. I
  $i \leftarrow i+1$
  Find all active sets (an algorithm will be discussed below).
  Increase y_B *uniformly for all active sets* B *until some dual constraint becomes tight,*
    i.e., until $\sum_{B:e\in\delta_{G-\mathcal{E}_H(B)}(B)} y_B = c_{e_i}$ *for some edge* $e_i \notin I$*.*
  $I \leftarrow I \cup \{e_i\}$
For $l \leftarrow i$ *downto 1*
  If there are no violated sets w.r.t. $I - \{e_l\}$
    $I \leftarrow I - \{e_l\}$

4.2 Proof of Theorem 3.2

To prove that this algorithm satisfies Theorem 3.2, we follow the general proof framework of [17]. From now on H is fixed as the partial solution obtained at the end of $(p+1)^{\text{st}}$ phase and we fix an iteration i of the while loop of the algorithm above. We will call this the "current iteration". Let I be the partial solution at the beginning of this iteration. Let F be the final set of edges returned at the end of the phase. We claim that Theorem 3.2 follows from the theorem below; after stating the theorem, we will establish the claim.

Theorem 4.1 *If* $\mathcal{C}$ *is the set of active sets in this iteration, then*

$$\sum_{C\in\mathcal{C}} |F \cap \delta_{G-\mathcal{E}_H(C)}(C)| \leq 2|\mathcal{C}|.$$

Proof of Theorem 3.2. We wish to show that

$$\text{cost}(F) \leq 2 \sum_{B\subseteq V} h(B) y_B.$$

Note that by the properties of the algorithm $c_e = \sum_{B:e\in\delta_{G-\mathcal{E}_H(B)}(B)} y_B$ for each $e \in F$. Thus the theorem follows if

$$\sum_{e\in F} \sum_{B:e\in\delta_{G-\mathcal{E}_H(B)}(B)} y_B \leq 2 \sum_{B\subseteq V} h(B) y_B.$$

We can rewrite the lefthand side as $\sum_{B\subseteq V} |F \cap \delta_{G-\mathcal{E}_H(B)}(B)| y_B$, so that the theorem follows if

$$\sum_{B\subseteq V} |F \cap \delta_{G-\mathcal{E}_H(B)}(B)| y_B \leq 2 \sum_{B\subseteq V} h(B) y_B.$$

We prove this by induction on the value of y during the course of the algorithm. Initially $y = 0$, so the inequality holds. In any iteration, if $\mathcal{C}$ is the set of active sets in that iteration, and each y_C for $C \in \mathcal{C}$ is increased by γ, then the left-hand side increases by

$$\gamma \sum_{C\in\mathcal{C}} |F \cap \delta_{G-\mathcal{E}_H(C)}(C)|,$$

while the right-hand side increases by $2\gamma|\mathcal{C}|$. The theorem then follows by Theorem 4.1. ■

4.3 Proof of polynomial time

We now turn to proving that the algorithm can be implemented in polynomial time. To do this, we will need some definitions and a theorem, which we provide below.

Definition 4.2 *Two sets $A, B \subseteq V$* $\varepsilon_{H\cup I}$-cross *if $A \not\subseteq B$, $B \not\subseteq A$, $A \cap (B \cup \varepsilon_{H\cup I}(B)) \neq \emptyset$ and $B \cap (A \cup \varepsilon_{H\cup I}(A)) \neq \emptyset$.*

Note that since $A, B \subseteq V$ and $\varepsilon_{H\cup I} \subseteq E \cup V$, we could have replaced $\varepsilon_{H\cup I}$ with $\Gamma_{H\cup I}$ in the definition above.

Definition 4.3 *A family of subsets of V is* $\varepsilon_{H\cup I}$-laminar *if no two sets in the family $\varepsilon_{H\cup I}$-cross.*

Observation 4.4 *Let $X \subseteq Y \subseteq V \cup E$, and $A, B \subseteq V$. If A and B ε_X-cross, they also ε_Y-cross. Similarly, if a family of sets is ε_Y-laminar, then they are also ε_X-laminar.*

Note that this is a stronger notion than the usual notion of laminarity, which is reproduced below. A laminar family according to this notion is also laminar according to the usual notion.

Definition 4.5 *Two sets $A, B \subseteq V$* cross *if $A \not\subseteq B$, $B \not\subseteq A$, and $A \cap B \neq \emptyset$.*

Definition 4.6 *A family of subsets of V is* laminar *if no two sets in the family cross.*

Theorem 4.7 *If $A, B \subseteq V$ are violated sets with respect to I, then either $A \cap B$ and $A \cup B$ are also violated or $A - B - \varepsilon_{H\cup I}(B)$ and $B - A - \varepsilon_{H\cup I}(A)$ are also violated.*

We defer the proof of this theorem for the moment. The theorem implies the following corollary.

Corollary 4.8 *No violated set with respect to I $\varepsilon_{H\cup I}$-crosses any active set with respect to I.*

Proof. If a violated set A $\varepsilon_{H\cup I}$-crosses an active set B, then by the theorem, either $A \cap B$ or $B - A - \varepsilon_{H\cup I}(A)$ is also violated. This contradicts the minimality of B. ■

Theorem 4.7 thus implies that the algorithm can be implemented in polynomial time.

Theorem 4.9 *The algorithm for a phase can be implemented in polynomial time.*

Proof. It follows from Corollary 4.8 that the active sets are disjoint. Hence each vertex can be in at most one active set. Consider a network on $H \cup I$, where the capacity of each edge and each nonterminal is one and the capacity of each terminal is unbounded. Consider a vertex u: it will be in an active set if there exists a vertex v such that the minimum u-v cut with respect to edges $H \cup I$ is of capacity $r_{uv} - p$. Let v be one such vertex. The active set in which u lies is the minimal (inclusion-wise) u-v min-cut. This can be found in polynomial time using a max-flow subroutine. ■

We now prove Theorem 4.7. To prove this theorem we need the following definitions and lemmas.

Definition 4.10 *Let $\varphi : 2^V \to \mathbb{Z}^+$. We say that φ is* $\varepsilon_{H\cup I}$-submodular *if $\varphi(V) = 0$ and, for every $A, B \subseteq V$, the following two conditions hold:*

1. $\varphi(A) + \varphi(B) \geq \varphi(A \cap B) + \varphi(A \cup B)$
2. $\varphi(A) + \varphi(B) \geq \varphi(A - B - \varepsilon_{H\cup I}(B)) + \varphi(B - A - \varepsilon_{H\cup I}(A))$.

Lemma 4.11 *$|\varepsilon_{H\cup I}|$ is $\varepsilon_{H\cup I}$-submodular.*

Proof. We need to prove the two inequalities in the definition of $\varepsilon_{H\cup I}$-submodularity.

1. One can easily verify that the contribution of any element to the left-hand side of the inequality is at least the contribution of the element to the right-hand side of the inequality. This proves the first condition of $\varepsilon_{H\cup I}$-submodularity.
2. The proof of this inequality is similar to the proof of the first inequality except for the case when there is an edge rs, $r \in R$, $s \in S \cap A \cap B$ and either $r \in A - B - \varepsilon_{H\cup I}(B)$ or $r \in B - A - \varepsilon_{H\cup I}(A)$. In this case s contributes to the right-hand side of the inequality but does not contribute to the left-hand side. But to counteract the contribution of s, edge rs contributes only to the left-hand side of the inequality. ■

Definition 4.12 *Let $\varphi : 2^V \to \mathbb{Z}$. We say that φ is* weakly $\varepsilon_{H\cup I}$-supermodular *if $\varphi(V) = 0$ and, for every $A, B \subseteq V$, at least one of the following two conditions hold:*

1. $\varphi(A) + \varphi(B) \leq \varphi(A \cap B) + \varphi(A \cup B)$
2. $\varphi(A) + \varphi(B) \leq \varphi(A - B - \varepsilon_{H\cup I}(B)) + \varphi(B - A - \varepsilon_{H\cup I}(A))$.

Lemma 4.13 *$f(B) = \max_{u\in B, v\notin B} r_{uv}$ is weakly $\varepsilon_{H\cup I}$-supermodular.*

Proof. Since $\varepsilon_{H\cup I}(B)$ does not contain any terminals, $f(A - B - \varepsilon_{H\cup I}(B)) = f(A - B)$. Similarly, $f(B - A - \varepsilon_{H\cup I}(A)) = f(B - A)$. The lemma follows from the fact that f is weakly supermodular [5]. ■

Now we are ready to prove Theorem 4.7.

Proof of Theorem 4.7. Since A and B are violated sets, their deficiency is p, which is the maximum deficiency for this phase. Hence,

$$\begin{aligned} 2p &= (f(A) - |\varepsilon_{H\cup I}(A)|) + (f(B) - |\varepsilon_{H\cup I}(B)|) \\ &= (f(A) + f(B)) - (|\varepsilon_{H\cup I}(A)| + |\varepsilon_{H\cup I}(B)|). \end{aligned}$$

Since f is weakly $\varepsilon_{H\cup I}$-supermodular, either $f(A)+f(B) \leq f(A\cap B)+f(A\cup B)$ or $f(A)+f(B) \leq f(A - B - \varepsilon_{H\cup I}(B)) + f(B - A - \varepsilon_{H\cup I}(A))$. Suppose the former holds. By $\varepsilon_{H\cup I}$-submodularity, we also have $|\varepsilon_{H\cup I}(A)| + |\varepsilon_{H\cup I}(B)| \geq |\varepsilon_{H\cup I}(A \cap B)| + |\varepsilon_{H\cup I}(A \cup B)|$. Hence,

$$\begin{aligned} 2p &\leq (f(A \cap B) + f(A \cup B)) - (|\varepsilon_{H\cup I}(A \cap B)| + |\varepsilon_{H\cup I}(A \cup B)|) \\ &= (f(A \cap B) - |\varepsilon_{H\cup I}(A \cap B)|) + (f(A \cup B) - |\varepsilon_{H\cup I}(A \cup B)|) \\ &\leq 2p. \end{aligned}$$

The last inequality results from the fact that no set has deficiency more than p. Since $(f(A \cap B) - |\varepsilon_{H\cup I}(A \cap B)|) + (f(A \cup B) - |\varepsilon_{H\cup I}(A \cup B)|)$ is at most as well as at least $2p$, it is $2p$. This is possible only if $A \cap B$ and $A \cup B$ are violated.

Similarly, if $f(A) + f(B) \leq f(A - B - \varepsilon_{H\cup I}(B)) + f(B - A - \varepsilon_{H\cup I}(A))$, then $A - B - \varepsilon_{H\cup I}(B)$ and $B - A - \varepsilon_{H\cup I}(A)$ are violated. ■

4.4 Proof of Theorem 4.1

Finally, we turn to the proof of Theorem 4.1. Let $\mathcal{C}$ be the collection of active sets in the current iteration. Let Y be the set of edges $e \in F$ for which there exists $C \in \mathcal{C}$ such that $e \in \delta_{G-\varepsilon_H(C)}(C)$. For each edge $e \in Y$ we define a *witness set*, $W_e \subseteq V$, as a set that meets the following conditions:

1. $|\varepsilon_{H\cup I\cup F}(W_e)| = f(W_e) - p + 1$;
2. $|\varepsilon_{H\cup I\cup F-\{e\}}(W_e)| = f(W_e) - p$.

To see that a witness set W_e must exist for every $e \in Y$, observe that by construction of the algorithm $I \cup F - \{e\}$ is not a feasible solution; thus there must be some violated set W_e which is a witness set. A *witness family* for Y is a family of subsets of V, so that it exactly contains one witness for each edge in Y. Note that it cannot be the case that the same set W is a witness for two different edges e and f.

Theorem 4.14 *There is a ε_H-laminar witness family for Y.*

Proof. Given a witness family, suppose two sets W_e and W_f ε_H-cross. Then they will $\varepsilon_{H\cup F\cup I}$-cross also. Let X and Y be the two sets obtained from Theorem 4.7; that is, either $W_e \cap W_f$ and $W_e \cup W_f$ or $W_e - W_f - \varepsilon_{H\cup F\cup I}(W_f)$ and $W_f - W_e - \varepsilon_{H\cup F\cup I}(W_e)$. These sets do not $\varepsilon_{H\cup F\cup I}$-cross since neither sets $W_e \cap W_f$ and $W_e \cup W_f$ $\varepsilon_{H\cup F\cup I}$-cross nor sets $W_e - W_f - \varepsilon_{H\cup F\cup I}(W_f)$ and $W_f - W_e - \varepsilon_{H\cup F\cup I}(W_e)$ $\varepsilon_{H\cup F\cup I}$-cross. We will replace W_e and W_f by two other sets X_e and Y_f. We will show that these sets X_e and Y_f will be witnesses for edges e and f. When the first case of Theorem 4.7 is valid, we will have that these two sets are $W_e \cap W_f$ and $W_e \cup W_f$. When the second case of the theorem holds, X_e and Y_f will be subsets of $W_e - W_f - \varepsilon_{H\cup F\cup I}(W_f)$ and $W_f - W_e - \varepsilon_{H\cup F\cup I}(W_e)$. Note that in either case the two sets do not ε_H-cross. Also notice that this process cannot continue indefinitely without decreasing the minimum cardinality of a witness in the family, so it must end with a laminar witness family.

For the sake of argument, we assume that the first case of Theorem 4.7 holds; the other case is similar, and we will explain the minor differences at the end of the proof.

Since W_e and W_f are witnesses we get

$$|\varepsilon_{H\cup F\cup I}(W_e)| + |\varepsilon_{H\cup F\cup I}(W_f)| = f(W_e) + f(W_f) - 2p + 2.$$

By the arguments in the proof of Theorem 4.7 we can use the weak $\varepsilon_{H\cup F\cup I}$-supermodularity of $f(\cdot)$ and the $\varepsilon_{H\cup F\cup I}$-submodularity of $|\varepsilon_{H\cup F\cup I}(\cdot)|$ to get that

$$|\varepsilon_{H\cup F\cup I}(W_e \cap W_f)| + |\varepsilon_{H\cup F\cup I}(W_e \cup W_f)| \leq f(W_e \cap W_f) + f(W_e \cup W_f) - 2p + 2.$$

Note that this is possible, because the option which holds for $f(\cdot)$ does not depend upon the ε function. Since $F \cup I$ is a feasible solution to this phase, $|\varepsilon_{H\cup F\cup I}(W_e \cap W_f)| \geq f(W_e \cap W_f) - p + 1$

and $|\varepsilon_{H\cup F\cup I}(W_e \cup W_f)| \geq f(W_e \cup W_f) - p + 1$. Thus the above inequality implies that $|\varepsilon_{H\cup F\cup I}(W_e \cap W_f)| = f(W_e \cap W_f) - p + 1$ and $|\varepsilon_{H\cup F\cup I}(W_e \cup W_f)| = f(W_e \cup W_f) - p + 1$.

Now consider e. Applying the definition of witness for e we get

$$|\varepsilon_{H\cup F\cup I-\{e\}}(W_e)| + |\varepsilon_{H\cup F\cup I-\{e\}}(W_f)| \leq f(W_e) + f(W_f) - 2p + 1.$$

Using the weak $\varepsilon_{H\cup F\cup I-\{e\}}$-supermodularity of $f(\cdot)$ and $\varepsilon_{H\cup F\cup I-\{e\}}$-submodularity of $|\varepsilon_{H\cup F\cup I-\{e\}}(\cdot)|$, we get that

$$|\varepsilon_{H\cup F\cup I-\{e\}}(W_e \cap W_f)| + |\varepsilon_{H\cup F\cup I-\{e\}}(W_e \cup W_f)| \leq f(W_e \cap W_f) + f(W_e \cup W_f) - 2p + 1.$$

Now by the feasibility of H we know that $|\varepsilon_{H\cup F\cup I-\{e\}}(W_e \cap W_f)| \geq f(W_e \cap W_f) - p$ and $|\varepsilon_{H\cup F\cup I-\{e\}}(W_e \cup W_f)| \geq f(W_e \cup W_f) - p$. Thus for at least one of $W_e \cap W_f$ and $W_e \cup W_f$, say $W_e \cap W_f$, $|\varepsilon_{H\cup F\cup I-\{e\}}(W_e \cap W_f)| = f(W_e \cap W_f) - p$. Then $W_e \cap W_f$ is a witness set for e and we set $X_e = W_e \cap W_f$. A similar argument will show that the other set is a witness set for edge f.

We now return to the case that the result of Theorem 4.7 are the two sets $W_e - W_f - \varepsilon_{H\cup F\cup I}(W_f)$ and $W_f - W_e - \varepsilon_{H\cup F\cup I}(W_e)$. The argument proceeds much as above, except that it will show that one of $W_e - W_f - \varepsilon_{H\cup F\cup I-\{e\}}(W_f)$ and $W_f - W_e - \varepsilon_{H\cup F\cup I-\{e\}}(W_e)$ is a witness for e, and the corresponding opposite of the pair $W_e - W_f - \varepsilon_{H\cup F\cup I-\{f\}}(W_f)$ and $W_f - W_e - \varepsilon_{H\cup F\cup I-\{f\}}(W_e)$ is a witness for f. ■

Given the laminar witness family, we can construct a rooted tree T, as follows. Let $\mathcal{L}$ be the set of sets in the witness family, plus the set of vertices V. We create a node of the tree for each $L \in \mathcal{L}$; the node corresponding to V will be the root of T. The node corresponding to $A \in \mathcal{L}$ is a parent of the node corresponding to $B \in \mathcal{L}$ if A is the smallest set in $\mathcal{L}$ strictly containing B. We associate each active set $C \in \mathcal{C}$ with the node in the tree corresponding to the smallest set in $\mathcal{L}$ that contains C; note that this notion is well-defined by Corollary 4.8.

In a moment, we will prove the following lemma.

Lemma 4.15 *Let $\mathcal{B}$ be the set of active sets associated with a node v in the tree T. Then the degree of node v in the tree T is at least $\sum_{C\in\mathcal{B}} |F \cap \delta_{G-\varepsilon_H(C)}(C)|$.*

First, let us show how the lemma implies Theorem 4.1.

Proof of Theorem 4.1. Let us color the nodes of the tree T; we color a node blue if it has an active set associated with it, and red otherwise. Let *Red* and *Blue* be the sets of red and blue nodes respectively, and let $deg(v)$ be the degree of node v in tree T. Note that every leaf of the tree is blue: since each leaf corresponds to a violated set, it must contain some minimal violated set inside it. Then we have

$$\begin{aligned}
&(5) \quad & \sum_{C\in\mathcal{B}} |F \cap \delta_{G-\varepsilon_H(C)}(C)| &\leq \sum_{v\in Blue} deg(v) \\
&& &= \sum_{v\in Blue\cup Red} deg(v) - \sum_{v\in Red} deg(v) \\
&(6) && \leq 2(|Blue| + |Red| - 1) - 2(|Red| - 1) \\
&&& \leq 2|Blue| \\
&(7) && \leq 2|\mathcal{C}|,
\end{aligned}$$

where (5) follows by Lemma 4.15, (6) by the fact that T is a tree, and all red nodes (excepting perhaps the root) have degree at least two, and (7) follows since each blue node has at least one member of $\mathcal{C}$ associated with it. ■

We now complete the proof.

Proof of Lemma 4.15. By the definition of Y, $\sum_{C\in\mathcal{B}} |F \cap \delta_{G-\varepsilon_H(C)}(C)| = \sum_{C\in\mathcal{B}} |Y \cap \delta_{G-\varepsilon_H(C)}(C)|$. Given a node v of the tree T, and the set of active sets $\mathcal{B}$ corresponding to it, let Y_B be the set of edges $Y_B = \bigcup_{C\in\mathcal{B}}(Y \cap \delta_{G-\varepsilon_H(C)}(C))$. Note that there must be a witness set corresponding to each edge in Y_B, and at most one can be the set W corresponding to v itself. The proof follows if we can show that each remaining edge of Y_B can be mapped to a unique child of v; that is, a unique witness set X such that W is the smallest witness set containing X. Thus we simply need to show that for an edge $e \in Y_B$, if its witness set is not W, it must be the case that for its witness set X, $X \subset W$, and there is no witness set Z such that $X \subset Z \subset W$.

We prove these statements by contradiction. First assume that $X \not\subset W$. Since the witness sets are laminar, this implies that $X \cap W = \emptyset$. We know $e \in \delta_{G-\varepsilon_H(C)}(C)$ with $C \subset W$ for some active $C \in \mathcal{B}$, and $e \in \delta_{G-\varepsilon_H(X)}(X)$ by the definition of a witness set. If $e \in \delta_{G-\varepsilon_H(W)}(W)$, this contradicts the fact that X is the unique witness set for e. The only way this cannot occur is if the endpoint of e in X is in the element neighborhood of W, $\varepsilon_H(W)$, a possibility which is eliminated since X and W do not ε_H-cross. Next, suppose that there is a witness set Z such that $X \subset Z \subset W$. As above, we know that $e \in \delta_{G-\varepsilon_H(X)}(X)$ and $e \in \delta_{G-\varepsilon_H(C)}(C)$, where $C \subseteq W$, but $C \cap Z = \emptyset$. Thus there is one endpoint of e in C and the other in X. If $e \in \delta_{G-\varepsilon_H(Z)}(Z)$, this contradicts the fact that X is the unique witness set for e. The only way this cannot occur is if the endpoint of e in C is in the element neighborhood of Z, $\varepsilon_H(Z)$, a possiblity which is eliminated since by Corollary 4.8 the active set C does not ε_H-cross the violated set Z. ■

5 Tight example

For the special case when the set of nonterminals is empty, our algorithm reduces to the algorithm of [5] for edge connectivity. A $2H_k$-approximation tight example to this algorithm for edge-connectivity is given in the same paper.

Acknowledgements

We thank the two anonymous referees for useful comments.

References

[1] A. Frank, Augmenting graphs to meet edge-connectivity requirements, *SIAM Journal on Discrete Mathematics*, 5 (1992), pp. 25–53.

[2] J. Cheriyan, S. Vempala, and A. Vetta. An approximation algorithm for the minimum-cost k-vertex connected subgraph. To appear in STOC 2002.

[3] L. Fleischer. A 2-approximation for minimum cost $\{0, 1, 2\}$ vertex connectivity. In *Integer Programming and Combinatorial Optimization*, number 2081 in the Lecture Notes in Computer Science, pages 115–129. Springer, 2001.

[4] L. Fleischer, K. Jain, and D.P. Williamson. An iterative rounding 2-approximation algorithm for the element connectivity problem. FOCS 2001, pp. 339–347.

[5] M.X. Goemans, A. Goldberg, S. Plotkin, D. Shmoys, E. Tardos, and D.P. Williamson, Improved Approximation Algorithms for Network Design Problems, SODA '94, pp. 223–232.

[6] M.X. Goemans, D.P.Williamson, The Primal-Dual Method for Approximation Algorithms and its Application to Network Design Problems, in *Approximation Algorithms for NP-hard Problems*, D. Hochbaum, Ed., PWS, 1997.

[7] M. Grötschel and C.L. Monma and M. Stoer, Computational Results With a Cutting Plane Algorithm for Designing Communication Networks with Low-Connectivity Constraints, *Operations Research* 40 (1992), pp. 309–330.

[8] K. Jain, A Factor 2 Approximation Algorithm for the Generalized Steiner Network Problem, *Combinatorica* 21 (2001), pp. 39–60.

[9] K. Jain, I. Mandoiu, V.V. Vazirani, and D.P. Williamson. A primal-dual schema based approximation algorithm for the element connectivity problem, SODA '99, pp. 484–489.

[10] S. Khuller and B. Raghavachari, Improved Approximation Algorithms for Uniform Connectivity Problems, *Journal of Algorithms* 21 (1996), pp. 434–450.

[11] M. Mihail, N. Dean, M. Mostrel and D. Shallcross, A Commercial Application of Survivable Network Design: ITP/INPLANS CCS Network Topology Analyser, SODA '96, pp. 279–287.

[12] C.L. Monma and D.F. Shallcross, Methods for Designing Communication Networks With Certain Two-Connected Survivability Constraints, *Operations Research* 1989, pp. 531–541.

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

[14] R. Ravi and D.P. Williamson, Erratum: an approximation algorithm for minimum-cost vertex connectivity problems, to appear in *Algorithmica*.

[15] M. Stoer, *Design of Survivable Networks*, Lecture Notes in Mathematics 1531, Springer-Verlag, Berlin, 1992.

[16] V.V. Vazirani, *Approximation Algorithms*, Springer, 2001.

[17] D.P. Williamson, M.X. Goemans, M. Mihail, and V.V. Vazirani, A primal-dual approximation algorithm for generalized Steiner network problems, *Combinatorica* 15 (1995), pp. 435–454.

[18] L. Zhao, H. Nagamochi, and T. Ibaraki. A primal-dual approximation algorithm for the survivable network design problem in hypergraph. In *STACS 2001*, number 2010 in Lecture Notes for Computer Science, pages 478–489. Springer-Verlag, 2001.