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 [11, 2, 5], 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. [4, 7]) and has been studied from many different approaches including polyhedral combinatorics [10, 4], interchange heuristics [8], min-max relations [1] (in the unweighted case), approximation algorithms [11, 2, 9], and implementations thereof [7]. 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 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. Similarly, in the case of approximation algorithms, a $2H_k$-approximation algorithm has been known for a few years for EC-SNDP [11, 2], where $H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n} \approx \ln n$ and $k = \max_{u,v} r_{uv}$. Very recently, Jain gave a 2-approximation algorithm for the problem [5]. However, in the case of VC-SNDP very little is known in terms of approximation algorithms. Ravi and Williamson [9] show a 3-approximation algorithm in the case that $r_{uv} \in \{0, 1, 2\}$, and a $2H_k$-approximation algorithm in the case that $r_{uv} = k$ for all $u, v \in V$. For the latter problem, 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 [6] 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 *Steiner vertices*. Steiner vertices 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 Steiner vertices do not have any

*College of Computing, Georgia Institute of Technology. Supported by NSF Grant CCR 9627308.

†IBM T.J. Watson Research Center.

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 Steiner vertices.

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 follows the basic algorithmic approach used in the $2H_k$-approximation algorithms for EC-SNDP in Williamson et al. [11] and Goemans et al. [2] and the cases of VC-SNDP considered in Ravi and Williamson [9]. The algorithms in these papers break down the problem into a number of *phases*. In each phase, we solve a certain augmentation problem by specifying vertex sets S which must be augmented; that is, we must choose an edge from $\delta(S)$, the set of edges with one endpoint each in S and $V-S$. This augmentation problem is formulated as an integer programming problem, and the problem is solved by using the primal-dual method for approximation algorithms. We follow the paper of Ravi and Williamson [9], which shows that if the augmentation problem meets certain conditions, then the primal-dual method gives a good approximation algorithm for the augmentation problem. For a more detailed presentation of these algorithms and an overview of the primal-dual method, we refer the reader to the survey of Goemans and Williamson [3].

The central technical difficulty in modifying these algorithms is defining the appropriate notion of an augmentation for this problem and then showing that the conditions of [9] are met for this notion. The adaption is non-trivial. For example, previous work on the EC-SNDP needed to augment sets S by adding an edge from $\delta(S)$, and proofs relied on the well-known properties of the function $\delta(S)$. Here we must not only choose carefully the sets S to augment and the set of edges $e(S)$ from which we must choose, but also prove the corresponding properties about $e(S)$.

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.

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 Steiner vertices, 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 Steiner.

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 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 selected 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:

$$(2.1) \qquad \min \sum_{e\in E} c_e x_e$$

subject to

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

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

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,

$$\text{(2.2)} \qquad \max \sum_{A \subseteq S \cup E, B \subseteq V-A} (f(B) - |A|) y_{BA}$$

subject to

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

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

Let OPT be the optimal cost of IP (2.1). By weak duality theorem any solution to (2.2) will be a lower bound on OPT.

We will use the fact that f is *weakly supermodular* [2], i.e., $f(V) = 0$ and, for every $A, B \subseteq V$, at least one of the following holds

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

3 High level description of the algorithm

Given a partially feasible solution H to IP (2.1), 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 [2] 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 Steiner vertices 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 $\varepsilon_H(B) = \Gamma_H(B) \cup \rho_H(B)$.

Lemma 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:

$$\text{(3.3)} \qquad \min \sum_{e \in E-H} c_e x_e$$

subject to

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

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

Let I be a feasible solution of this integral program. By Lemma 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:

$$\text{(3.4)} \qquad \max \sum_{B \subseteq V} h(B) y_B$$

subject to

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

$$y_B \geq 0$$

In the next section we will prove:

Theorem 3.1. *We can find in polynomial time a feasible solution, I, for IP (3.3) and a feasible solution, y, for LP (3.4) such that* $\mathrm{cost}(I) \leq 2 \sum_{B \subseteq V} h(B) y_B$.

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 (2.2).

Lemma 3.2.

$$\mathrm{cost}(I) \leq \frac{2}{p} \cdot \mathrm{OPT}.$$

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

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

The claim follows immediately from Theorem 3.1. ■

Corollary 3.1. *The cost of edges chosen by the algorithm in all k phases is at most*

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

4 The algorithm for a phase, and its analysis

We now present the algorithm for a phase, which will augment sets whose deficiency is p. Our augmentation algorithm follows those in [11] and [9]. To prove Theorem 3.1, we use results of Ravi and Williamson [9] (see Corollary 4.8), which imply Theorem 3.1 if certain conditions on the augmentation problem are met. We first give the algorithm, then turn to stating and proving the conditions.

Let I be a partial solution to IP (3.3). A set is *violated* with respect to I if the constraint corresponding to it in IP (3.3) is unsatisfied. A violated set is *active* if none of its proper subsets is violated. Our algorithm for a phase is as follows:

Algorithm for phase p

$I \longleftarrow \emptyset$
$y \leftarrow 0$
$i \leftarrow 0$
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 a dual constraint becomes tight (i.e. $\sum_{B: e \in \delta_{G-\varepsilon_H(B)}(B)} y_B = c_e$) for some $e_i \in E - (\varepsilon_H(B) \cup I)$ for some active set B.
  $I \leftarrow I \cup \{e_i\}$
For $l \leftarrow i$ downto 1
  If there are no violated sets w.r.t. $I - \{e_l\}$
    $I \leftarrow I - \{e_l\}$

To prove that this algorithm satisfies Theorem 3.1 we need to prove the three conditions given in [9]. To state those three conditions in our setting we need the following definitions.

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.

Definition 4.1. *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 $B \cap (A \cup \varepsilon_{H\cup I}(A)) \neq \emptyset$.*

Definition 4.2. *A family of subsets of V is* $\varepsilon_{H\cup I}$-laminar *if no two sets in the family $\varepsilon_{H\cup I}$-cross.*

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 4.3. *Two sets $A, B \subseteq V$* cross *if $A \not\subseteq B$, $B \not\subseteq A$, and $A \cap B = \emptyset$.*

Definition 4.4. *A family of subsets of V is* laminar *if no two sets in the family cross.*

Let F be the final set of edges returned by the algorithm for phase p. Let $\mathcal{B}$ be the collection of active sets in the current iteration. Let Y be the set of edges $e \in F$ for which there exists $B \in \mathcal{B}$ such that $e \in \delta_{G-\varepsilon_H(B)}(B)$. 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.

Now we can state the needed three conditions in our setting with respect to the current iteration.

1. No violated set with respect to I $\varepsilon_{H\cup I}$-crosses any active set with respect to I.
2. The active sets with respect to I can be computed in polynomial time.
3. There exists a laminar witness family for Y.

Though we replaced the properties of crossing and laminarity with $\varepsilon_{H\cup I}$-crossing and $\varepsilon_{H\cup I}$-laminarity, given these conditions the proof of Theorem 3.1 follows from the techniques of [11, 9]. We do not repeat the part of these papers showing how Theorem 3.1 follows from the above conditions.

To prove the conditions, we will need the following theorem.

Theorem 4.1. *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.*

The theorem implies the following corollary, which gives the first condition.

Corollary 4.1. *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. ∎

It is also not difficult to see that Theorem 4.1 implies the second condition. It follows from the corollary 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 capacity of each edge and each Steiner vertex is one and 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 4.1, and then see how it implies the last condition. To prove this theorem we need the following definitions and lemmas.

DEFINITION 4.5. *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 4.1. *$|\varepsilon_{H\cup I}|$ 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 4.6. *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 4.2. *$f(B) = \max_{u \in B, v \notin B} r_{uv}$ is weakly $\varepsilon_{H\cup I}$-supermodular.*

Proof. Since $\varepsilon_{H\cup I}(B)$ does not contain a required vertex $f(A - B - \varepsilon_{H\cup I}(B)) = f(A - B)$. Similarly, $f(B - A - \varepsilon_{H\cup I}(A)) = f(B - A)$. The lemma follows from the fact that f is weakly supermodular [2]. ∎

Now we are ready to prove Theorem 4.1.

Proof of Theorem 4.1. 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,

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

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. ∎

We can now see how the theorem implies the last condition.

THEOREM 4.2. *There is a laminar witness family for Y.*

Proof. Given a witness family, suppose two sets W_e and W_f cross. Then they will $\varepsilon_{H\cup F\cup I}$-cross also. Let X and Y be the two sets obtained from Theorem 4.1 (observing that neither sets $W_e \cup W_f$ and $W_e \cap W_f$ $\varepsilon_{H\cup F\cup I}$-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 these sets, X_e and Y_f will be witnesses for e and f. When the first case of Theorem 4.1 is valid, we will have $X_e = X$ and $Y_f = Y$. When the second case of the theorem is valid X_e and Y_f will be the subsets of $W_e - W_f$ and $W_f - W_e$. 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. So only claim remains to show is that X_e and Y_f are witnesses for e and f.

Since W_e and W_f are witnesses we get

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

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

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

Since $F \cup I$ is a feasible solution to this phase, the above equality implies that $|\varepsilon_{H \cup F \cup I}(X)| = f(X) - p + 1$ and $|\varepsilon_{H \cup F \cup I}(Y)| = f(Y) - p + 1$.

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

$$|\varepsilon_{H \cup F \cup I - \{e\}}(X')| + |\varepsilon_{H \cup F \cup I - \{e\}}(Y')| \\ \leq f(X') + f(Y') - 2p + 1,$$

where X' and Y' are obtained by applying the same option of weak $\varepsilon_{H \cup F \cup I - \{e\}}$-supermodularity which was applied to obtain the sets X and Y from the options of weak $\varepsilon_{H \cup F \cup I}$-supermodularity (e.g., if $X = W_e \cup W_f$ and $Y = W_e \cap W_f$, then $X' = W_e \cup W_f$ and $Y' = W_e \cap W_f$). Note that this is possible, because the option which holds does not depend upon the ε function.

Now by the feasibility of H we know that $|\varepsilon_{H \cup F \cup I - \{e\}}(X')| \geq f(X') - p$ and $|\varepsilon_{H \cup F \cup I - \{e\}}(Y')| \geq f(Y') - p$. Thus for at least one of X' and Y', say X', $|\varepsilon_{H \cup F \cup I - \{e\}}(X')| = f(X') - p$. In case X and X' are same then X is a witness for e, so we set $X_e = X = X'$.

In case X and X' are different then, it must be the case that the definition of X and X' involved $\varepsilon_{H \cup F \cup I}$ and $\varepsilon_{H \cup F \cup I - \{e\}}$ respectively. There are now two possibilities,

1. $X = W_e - W_f - \varepsilon_{H \cup F \cup I}(W_f)$ and $X' = W_e - W_f - \varepsilon_{H \cup F \cup I - \{e\}}(W_f)$
2. $X = W_f - W_e - \varepsilon_{H \cup F \cup I}(W_e)$ and $X' = W_f - W_e - \varepsilon_{H \cup F \cup I - \{e\}}(W_e)$

Whichever be the case clearly X' is a witness for e. So we set $X_e = X'$. We apply the same procedure with f to obtain a witness Y_f. ■

5 Tight example

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

References

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

[2] 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.

[3] 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.

[4] 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.

[5] K. Jain, A Factor 2 Approximation Algorithm for the Generalized Steiner Network Problem, submitted to *Combinatorica*. To appear in FOCS '98.

[6] S. Khuller and B. Raghavachari, Improved Approximation Algorithms for Uniform Connectivity Problems, STOC '95, pp. 1–10.

[7] 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.

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

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

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

[11] 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.