SURVIVABLE NETWORK DESIGN PROBLEMS WITH VERTEX CONNECTIVITY REQUIREMENTS

PHILIP HODGES
UNDER SUPERVISION OF: JOSEPH CHERIYAN

Abstract. This document is a summary of the work done by the author and supervisor in the Spring 2018 term for the Combinatorics and Optimization department's Undergraduate Research Assistantship Program at the University of Waterloo. In this project, we explored a problem in network design, on survivable networks with small requirements.

Given a complete graph $G = (V, E)$ with edge costs c_e, and requirements $r_{uv} \in \mathbb{Z}$ for $u, v \in V$, we desire a set of edges $F \subseteq E$ of minimum cost such that each pair of vertices (u, v) is connected by r_{uv} vertex-disjoint paths. This is the Vertex-Connectivity Survivable Network Design Problem (VC-SNDP). In this project, we focused on the case where the requirements are less than or equal to 3, where no constant factor approximation is known. We document here many approaches that we took to try to gain traction on the problem, and several new lemmas that may prove useful in future work.

1. Introduction

Network design is an interesting area with many applications, mainly in the area of the design of telecommunications networks. In particular, survivable network problems model the idea of "failure points", that is, how many junctions in a network can fail before the network is no longer connected?

The Vertex-Connectivity Survivable Network Design Problem (VC-SNDP) is a very general problem in survivable network design. The problem is formulated as follows: given a complete graph $G = (V, E)$ with edge costs c_e for $e \in E$, and requirements $r_{uv} \in \mathbb{Z}$ for $u, v \in V$, we desire a set of edges $F \subseteq E$ of minimum cost such that each pair of vertices (u, v) is connected by r_{uv} vertex-disjoint paths.

This problem generalizes the Steiner Tree Problem, which was show by [Karp, 1972] to be NP-complete. Thus, unless $P = NP$, we cannot find a polynomial-time algorithm that finds the optimal solution to VC-SNDP. Instead, we try to find algorithms that run in polynomial time and provide an approximation guarantee; that is, the edge-set F output by the algorithm is guaranteed to have cost at most f times the optimal solution, where f is some function of the problem instance.

Approximation algorithms with approximation factor 2 exist for the analogous problem of edge-connectivity ([Jain, 2001]), requiring only edge-disjoint paths, as well as for element-connectivity ([Fleischer et al., 2006]), a concept that is intermediate between edge- and vertex-connectivity. However, when vertex-disjoint paths are required, the problem appears to be more difficult. Let $k = \max_{u,v \in V} r_{uv}$ and let T be the set of vertices which are an end of a pair with nonzero requirement, called terminals. The best known approximation for VC-SNDP is $O(k^3 \ln |T|)$ due to an algorithm from [Chuzhoy and Khanna, 2009].

Certain special cases on the requirements admit algorithms with better approximations. In the case of requirements $r_{uv} \in \{0, 1, 2\}$, a 2-approximation algorithm is given in [Fleischer et al., 2006]. If the all pairs with positive requirements share one vertex (we call these *rooted requirements*), then an $O(k^2)$-approximation algorithm is given in [Nutov, 2012a]. If $r_{uv} = k$ for all $u, v \in T$, then an $O(k \ln k)$-approximation algorithm is given in [Nutov, 2012b].

Date: August 17, 2018.

These results motivate the major open problems related to VC-SNDP; in the general case, does there exist an algorithm with approximation factor independent of $|T|$? If not, then in the case of $r_{uv} \in \{0, 1, 2, 3\}$, is there an algorithm with a constant factor approximation?

In this project, we start to build some foundation and present some ideas that may prove useful in addressing the latter question. In Sections 2 and 3, we summarize the ideas of [Nutov, 2009]. In Section 4, we prove a new lemma for the general case that is similar to some structural results from [Nutov, 2009]. In Section 5 we provide a counterexample showing that this lemma only holds in instances of small requirements. In Section 6, we present a new framework that attempts to use the new lemma along with the ideas and results from [Nutov, 2009] to attempt to generalize those results from the rooted case to the general case. Section 7 contains a proof of a separate structural lemma that, while not explicitly used anywhere, may prove important in the future. Section 8 is a brief look into an alternate approach, where we explore what a generalization of the 2-approximation algorithm for the Steiner Forest Problem ([Williamson and Shmoys, 2011], page 170) would require.

2. Summary of [Nutov, 2009]

Let $G = (V, E)$ be a complete graph with edge costs c_e and connectivity requirements $\{r_{uv} : u, v \in V\}$. Let $\kappa_H(u, v)$ denote the maximum number of internally-vertex-disjoint paths from u to v in the graph H.

Let $J = (V, E_J)$ be a subgraph of G. Define $\Gamma_J(X) = \Gamma(X) = \{v \in V - X : uv \in E_J \text{ for some } u \in X\}$, the set of neighbours of X in J, and let $X^* = V - (X \cup \Gamma(X))$.

Assume that there is a set of terminals $T \subseteq V$ and a vertex $s \in V$ such that $r_{st} = \ell + 1$ for all $t \in T$ and $r_{st} = 0$ otherwise, so the instance of VC-SNDP is rooted at s. Assume that the subgraph J has cost 0 and $\kappa_J(s, t) = \ell$ for all $t \in T$. Thus we are considering an instance of the Rooted SND Augmentation problem, where we wish to find an augmenting set of edges I of minimum cost such that the connectivity between s and nodes in T is at least $\ell + 1$ in $G + I$.

We require the following definitions.

Definition 1 (t-tight, min-core, max-core)**.** *A node subset $X \subseteq V$ is t-tight for $t \in T$ if $t \in X, s \in X^*$, and $|\Gamma_J(X)| = l$. A tight set is a core if it does not contain two inclusion minimal tight sets. An inclusion minimal core is a min-core, and an inclusion maximal core is a max-core.*

Definition 2 ($C_i, M_i, \mathcal{C}_J, \mathcal{M}_J, T_i, \Gamma_i$)**.** *Let $\mathcal{C}_J = \{C_1, ..., C_\nu\}$ be the set of min-cores of J with $\nu = |\mathcal{C}_J|$, and $\mathcal{M}_J = \{M_1, ..., M_\nu\}$ be a set of max-cores such that M_i contains C_i. Let $T_i = T \cap C_i$ and $\Gamma_i = \Gamma(M_i)$.*

Definition 3 (independence of max-cores)**.** *$M_i, M_j \in \mathcal{M}_J$ are independent if the sets $T_i \cap M_j^*, T_j \cap M_i^*$ are both nonempty. If M_i, M_j are not independent, then $T_i \subseteq \Gamma_j$ or $T_j \subseteq \Gamma_i$ and we say that M_i, M_j are dependent.*

The following two propositions concern basic properties of the sets $\mathcal{C}_J$ and $\mathcal{M}_J$, and the proofs follow directly from Lemma 2.2 in the paper. The first proposition states that min-cores do not T-intersect, so terminals belong to only one min-core.

Proposition 1 (Lemma 2.3 from [Nutov, 2009])**.** *For any tight set X and any $C_i \in \mathcal{C}_J$, either $C_i \cap X \cap T = \emptyset$ or $C_i \subseteq X$. Thus $C_i \cap C_j \cap T = \emptyset$ for any $i \neq j$.*

The statement of the lemma has been modified slightly, as the proof in the paper actually implies that $C_i \subseteq X$ and not just that $C_i \cap T \subseteq X$.

Proposition 2 (Corollary 2.4 from [Nutov, 2009])**.** *For any i the set M_i is unique. For any $i \neq j$, if M_i, M_j are independent, then $M_i \cap M_j^*, M_j \cap M_i^*$ are tight.*

Given a subfamily $\mathcal{M} \subseteq \mathcal{M}_J$, denote by $\mathcal{F}(\mathcal{M})$ the set of tight subsets of max-cores in $\mathcal{M}$. That is,

$$\mathcal{F}(\mathcal{M}) = \{X : X \subseteq M \in \mathcal{M}, X \text{ is tight}\}.$$

Then we have

Lemma 3 (Lemma 2.7 from [Nutov, 2009]). *If $I \subseteq E$ covers $\mathcal{F}(\mathcal{M}_J)$ then the number of min-cores in $G+I$ is at most $\nu/2$.*

This lemma informs our augmentation strategy. We can iteratively find $O(\log \nu) = O(\log |T|)$ covering sets, decreasing the number of min-cores by a factor of at least $1/2$ at each step until there are no min-cores remaining. Thus we increase the connectivity of J.

The remainder of Section 2 of the paper consists of finding a cover for $\mathcal{F}(\mathcal{M}_J)$. We do this by partitioning $\mathcal{M}_J$ into parts $\mathcal{M}_i$ for which the members of each part are pairwise independent. Then the families $\mathcal{F}(\mathcal{M}_i)$ will have a special structure (bi-uncrossability) allowing us to find a low cost cover set I for each of them.

Lemma 4 (Lemma 2.6 from [Nutov, 2009]). *The family $\mathcal{M}_J$ can be partitioned, in polynomial time, into at most $2\ell + 1$ parts so that the members of each part are pairwise independent.*

Definition 4 (bi-uncrossable). *A subfamily $\mathcal{F}$ of tight sets is bi-uncrossable if for any $X, Y \in \mathcal{F}$ at least one of the following holds: $X \cap Y, X \cup Y \in \mathcal{F}$ and equality holds in the first part of Proposition 2.2, or $X \cap Y^*, Y \cap X^* \in \mathcal{F}$ and equality holds in the second part of Proposition 2.2.*

Lemma 5 (Lemma 2.5 from [Nutov, 2009]). *If the members of $\mathcal{M}$ are pairwise independent, then the family $\mathcal{F}(\mathcal{M})$ is bi-uncrossable.*

Thus we can decompose the family $\mathcal{F}(\mathcal{M}_J)$ into bi-uncrossable subfamilies. Our final lemma allows us to find covers for each of these subfamilies.

Lemma 6 (Follows from section 3 of [Nutov, 2009]). *There exists a 2-approximation algorithm for the problem of finding a minimum cost edge-cover of a bi-uncrossable family $\mathcal{F}$.*

Then we can combine the $2\ell + 1$ 2-approximate edge-covers of Lemma 6 to get a $2(2\ell + 1)$-approximate edge cover of $\mathcal{F}(\mathcal{M}_J)$. This gives an $O(\ell \log |T|)$-approximation algorithm for the Rooted SND Augmentation problem, and thus an $O(k^2 \log |T|)$-approximation algorithm for the Rooted SND problem, where $k = \max(r_{ij} : i, j \in V)$.

A final improvement to this method that is detailed in [Nutov, 2012a] involves bounding the number of iterations that must be done for the augmentation. Once each core contains enough terminals, the whole family $\mathcal{F}(\mathcal{M}_J)$ becomes bi-uncrossable, and thus one final covering will cover all required tight sets. This number of terminals required in each core is on the order of ℓ, and so the best known approximation factor for the rooted problem is no longer dependent on T, and becomes $O(k^2)$. See [Nutov, 2012a] for details.

3. Proof of Proposition 2.1 from [Nutov, 2009]

The proof of Proposition 2.1 from [Nutov, 2009] is omitted from that paper, and the reference provided for it is somewhat unclear, as it is written in a different context. As such, we provide a proof here that will hopefully clearly illustrate the details of the properties described. The definitions and notation used here are the same as in the previous section.

Proposition 7 (Proposition 2.1 from [Nutov, 2009]). *For any $X, Y \subseteq V$ the following hold:*

(i) $|\Gamma(X)| + |\Gamma(Y)| \geq |\Gamma(X \cap Y)| + |\Gamma(X \cup Y)|$*, and if equality holds, then* $(X \cap Y)^* = X^* \cup Y^*$ *and* $(X \cup Y)^* = X^* \cap Y^*$

(ii) $|\Gamma(X)| + |\Gamma(Y)| \geq |\Gamma(X \cap Y^*)| + |\Gamma(Y \cap X^*)|$, *and if equality holds, then* $(X \cap Y^*)^* = X^* \cup Y$ *and* $(Y \cup X^*)^* = X \cup Y^*$

Proof. (i) First note that

$$|\Gamma(X)| + |\Gamma(Y)| = |\Gamma(X \cup Y)| + |\Gamma(X) \cap \Gamma(Y)| + |\Gamma(X) \cap Y| + |\Gamma(Y) \cap X|$$

and

$$K \cup \Gamma(X \cap Y) = [\Gamma(X) \cap \Gamma(Y)] \cup [\Gamma(X) \cap Y] \cup [\Gamma(Y) \cap X],$$

where

$$K = [\Gamma(X - Y) \cap Y] \cup [\Gamma(Y - X) \cap X] \cup [\Gamma(X - Y) \cap \Gamma(Y - X) - (X \cap Y)].$$

These can be seen from the diagram in Figure 1.

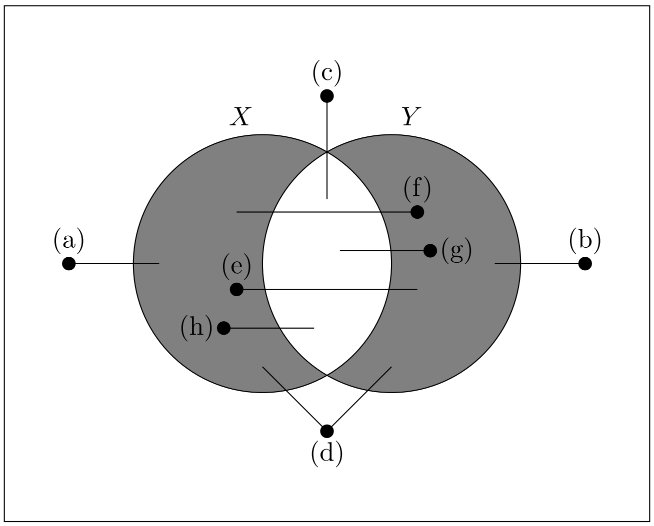

FIGURE 1

For example, $\Gamma(X \cap Y) = (c) \cup (g) \cup (h)$ and $\Gamma(X \cup Y) = (a) \cup (b) \cup (c) \cup (d)$. The equalities can be verified in this way. In particular, the set K has been constructed as $K = (d) \cup (e) \cup (f)$, and we have

$$\Gamma(X \cap Y) \subseteq [\Gamma(X) \cap \Gamma(Y)] \cup [\Gamma(X) \cap Y] \cup [\Gamma(Y) \cap X],$$

with equality if and only if $K = \emptyset$. Thus,

$$\begin{aligned} |\Gamma(X)| + |\Gamma(Y)| &= |\Gamma(X \cup Y)| + |\Gamma(X) \cap \Gamma(Y)| + |\Gamma(X) \cap Y| + |\Gamma(Y) \cap X| \\ &\geq |\Gamma(X \cap Y)| + |\Gamma(X \cup Y)|. \end{aligned}$$

If equality holds, then

$$\begin{aligned} (X \cap Y)^* &= V - [(X \cap Y) \cup \Gamma(X \cap Y)] \\ &= V - [(X \cap Y) \cup (\Gamma(X) \cap \Gamma(Y)) \cup (\Gamma(X) \cap Y) \cup (\Gamma(Y) \cap X)] \\ &= V - [(X \cup \Gamma(X)) \cap (Y \cup \Gamma(Y))] \\ &= [V - (X \cup \Gamma(X))] \cup [V - (Y \cup \Gamma(Y))] \\ &= X^* \cup Y^*. \end{aligned}$$

Note that

$$X \cup Y \cup \Gamma(X) \cup \Gamma(Y) = X \cup Y \cup \Gamma(X \cup Y)$$

since $\Gamma(X) \cup \Gamma(Y) - \Gamma(X \cup Y) \subseteq X \cup Y$. Thus

$$\begin{aligned} X^* \cap Y^* &= [V - (X \cup \Gamma(X))] \cap [V - (Y \cup \Gamma(Y))] \\ &= V - (X \cup Y \cup \Gamma(X) \cup \Gamma(Y)) \\ &= V - (X \cup Y \cup \Gamma(X \cup Y)) \\ &= (X \cup Y)^*. \end{aligned}$$

Thus $X^* \cap Y^* = (X \cup Y)^*$ actually holds in general, and does not require equality of the inequality in the proposition.

(ii) First note that for any $X, Y \subseteq V$, if $v \in \Gamma(X \cap Y)$, then $v \in \Gamma(X) \cap Y$, $v \in \Gamma(Y) \cap X$, or $\Gamma(X) \cap \Gamma(Y)$. This can also be seen in Figure 1. Combining this with $\Gamma(X^*) \subseteq \Gamma(X)$ for any $X \subseteq V$, we have

$$\begin{aligned} \Gamma(X \cap Y^*) &\subseteq [\Gamma(X) \cap Y^*] \cup [\Gamma(Y) \cap X] \cup [\Gamma(X) \cap \Gamma(Y)] \\ \Gamma(Y \cap X^*) &\subseteq [\Gamma(Y) \cap X^*] \cup [\Gamma(X) \cap Y] \cup [\Gamma(X) \cap \Gamma(Y)] \end{aligned} \tag{1}$$

Now we write $\Gamma(X)$ and $\Gamma(Y)$ as unions of disjoint sets to obtain

$$\begin{aligned} |\Gamma(X)| + |\Gamma(Y)| &= |[\Gamma(X) \cap Y] \cup [\Gamma(X) \cap Y^*] \cup [\Gamma(X) \cap \Gamma(Y)]| \\ &\quad + |[\Gamma(Y) \cap X] \cup [\Gamma(Y) \cap X^*] \cup [\Gamma(X) \cap \Gamma(Y)| \\ &= |\Gamma(X) \cap Y| + |\Gamma(X) \cap Y^*| + |\Gamma(X) \cap \Gamma(Y)| \\ &\quad + |\Gamma(Y) \cap X| + |\Gamma(Y) \cap X^*| + |\Gamma(X) \cap \Gamma(Y)| \\ &= |[\Gamma(X) \cap Y^*] \cup [\Gamma(Y) \cap X] \cup [\Gamma(X) \cap \Gamma(Y)]| \\ &\quad + |[\Gamma(Y) \cap X^*] \cup [\Gamma(X) \cap Y] \cup [\Gamma(X) \cap \Gamma(Y)]| \\ &\geq |\Gamma(X \cap Y^*)| + |\Gamma(Y \cap X^*)|, \end{aligned}$$

where the last inequality is due to the subsets in 1. In particular, equality holds when the subsets in 1 give equality, and then in this case

$$\begin{aligned} (X \cap Y^*)^* &= V - (X \cap Y^* \cup \Gamma(X \cap Y^*)) \\ &= V - ([X \cap Y^*] \cup [\Gamma(X) \cap Y^*] \cup [\Gamma(Y) \cap X] \cup [\Gamma(X) \cap \Gamma(Y)]) \\ &= V - ([X \cup \Gamma(X)] \cap [Y^* \cup \Gamma(Y)]) \\ &= [V - (X \cup \Gamma(X))] \cup [V - (Y^* \cup \Gamma(Y))] \\ &= X^* \cup Y \end{aligned}$$

and $(Y \cap X^*)^* = X \cup Y^*$ similarly.

□

4. A New Structural Lemma for 2-to-3 Augmentation in VC-SNDP

We now turn our attention to the case of general requirements $r_{st} \in \{0, 1, 2, 3\}$, and attempt to generalize the ideas used for the rooted case in [Nutov, 2009]. In this section, we describe basic properties similar to those in the rooted case that still hold, and prove a lemma analogous to Lemma 2.3 from [Nutov, 2009] that only holds when augmenting from $\ell = 2$ to $\ell = 3$ (we call this the 2-to-3 augmentation). Together, these can give a basis for an algorithm similar to that of [Nutov, 2009].

As in Section 2, we are given a complete graph $G = (V, E)$ with edge costs c_e for $e \in E$, and requirements r_{st} for $s, t \in V$. We desire a set of edges $F \subseteq E$ of minimum cost such that each pair of vertices s, t is connected by r_{ij} vertex-disjoint paths. As before, we wish to consider an augmentation framework.

Let $J = (V, E_J)$ be a subgraph of G. Let $\kappa_J(s,t)$ denote the maximum number of internally-vertex-disjoint paths from s to t in the subgraph J. Define $\Gamma_J(X) = \Gamma(X) = \{t \in V - X : st \in E_J \text{ for some } s \in X\}$, the set of neighbours of X in J, and let $X^* = V - (X \cup \Gamma(X))$.

Let $R \subseteq E$ be the set of edges such that $r(s,t) \neq 0$ and let $T \subseteq V$ be the set of nodes that are endpoints of edges in R, called terminals. Assume that the subgraph J has cost 0 and $\kappa_J(s,t) = 2$ for all $(s,t) \in R$. Thus we are considering an instance of the 2-to-3 SND Augmentation problem, where we wish to find an augmenting set of edges I of minimum cost such that the connectivity between requirement pairs $(s,t) \in R$ is at least 3 in $G + I$.

Definition 5 (s,t-tight, core)**.** *Given $(s,t) \in R$, a node subset $X \subseteq V$ is (s,t)-tight, or tight for (s,t), if $|\{s,t\} \cap X| = |\{s,t\} \cap X^*| = 1$ (we say X separates s and t), and $|\Gamma(X)| = 2$. We call X tight if it is tight for some $(s,t) \in R$. If X is tight for every pair of endpoints of edges in $R' \subseteq R$, we say X is tight for R'. If every $t \in T' \subseteq T$ is the endpoint of some edge for which X is tight, we will also say that X is tight for T'. Inclusion-minimal tight sets are called cores. Let $\mathcal{C} = \mathcal{C}_J = \{C_1, ..., C_\nu\}$ denote the set of cores in J. Let $T_i \subseteq T$ denote the set of nodes such that each $t \in T_i$ is an endpoint of an edge for which C_i is tight, and let T_X denote the same for X. Similarly let $R_i \subseteq R$ denote the set of edges for which C_i is tight, and R_X the same for X.*

The ideas we require can be alternatively formulated in terms of bisets. A biset is a pair of sets of vertices $\hat{X} = (X, X^+)$ such that $X \subseteq X^+$. Let $\Gamma(\hat{X}) = X^+ \setminus X$ and $X^* = V \setminus X^+$. Let $\delta_J(\hat{X})$ be the set of edges with one end in X and the other in X^*. We can then define a tight biset for $(s,t) \in R$ to be one where $|X \cap \{s,t\}| = |X^* \cap \{s,t\}| = 1$ and $|\Gamma(\hat{X})| + |\delta_J(\hat{X})| = 2$. We can verify that if $\hat{X}$ is a tight biset, then X is a tight set using our definition above and $\Gamma(\hat{X}) \subseteq \Gamma_J(X)$, the set of neighbours of X in the subgraph J. The biset formulation is useful for results about families of sets; in these results, we have few results about families of sets, and so we continue to use notation regarding vertex sets and their neighbourhoods.

Proposition 7 regarding vertex sets and their neighbourhoods still holds in this case, and we use it to prove the following lemma.

Lemma 8. *(1) If each of $X \cap Y$ and $X \cup Y$ separate some pair of terminals, then $X \cap Y$ and $X \cup Y$ are tight.*
(2) If each of $X \cap Y^*$ and $X^* \cap Y$ separate some pair of terminals, then $X \cap Y^*$ and $X^* \cap Y$ are tight.

Proof. (1) We have

$$\begin{aligned} 2 + 2 &= |\Gamma(X)| + |\Gamma(Y)| \\ &\geq |\Gamma(X \cap Y)| + |\Gamma(X \cup Y)| \\ &\geq 2 + 2 \end{aligned}$$

where the last inequality follows from the fact that each of $X \cap Y$ and $X \cup Y$ separate some pair of terminals. Thus equality holds throughout, and $X \cap Y$ and $X \cup Y$ are tight.

(2) We have

$$\begin{aligned} 2 + 2 &= |\Gamma(X)| + |\Gamma(Y)| \\ &\geq |\Gamma(X \cap Y^*)| + |\Gamma(X^* \cap Y)| \\ &\geq 2 + 2 \end{aligned}$$

where the last inequality follows from the fact that each of $X \cap Y^*$ and $X^* \cap Y$ separate some pair of terminals. Thus equality holds throughout, and $X \cap Y^*$ and $X^* \cap Y$ are tight.

□

Lemma 9 provides some structure to the tight sets of G that is analogous to the case of rooted requirements, as covered in [Nutov, 2009]. This lemma does not hold for arbitrarily high requirements in the augmentation framework for the case of general requirements.

Lemma 9. *For any tight sets X and Y such that $X \cap Y \cap (T_X \cup T_Y)$, either $X \cap Y$ is tight, or $X \cap Y^*$ and $X^* \cap Y$ are tight.*

Proof. Let X and Y be tight sets that $X \cap T_X \cap Y \neq \emptyset$. If $X \subseteq Y$ or $Y \subseteq X$ then $X \cap Y$ is trivially tight, so suppose otherwise. Let $(s_1, t_1) \in R_X$ such that $s_1 \in X \cap Y$.

If $t_1 \in Y^*$, then $X \cap Y$, $X \cup Y$ both separate s_1 and t_1, and so are tight. So we can assume that $t_1 \in Y$ or $t_1 \in \Gamma(Y)$.

Since Y does not separate s_1 and t_1, it must be tight for some other pair of terminals, say (s_2, t_2), with $s_2 \in Y$. If $t_2 \in X^*$, then $X \cap Y$ separates s_1 and t_1 and $X \cup Y$ separates s_2 and t_2, and thus both are tight. So we can assume that $t_2 \in X$ or $t_2 \in \Gamma(X)$. Note that in the arguments so far, it does not matter where s_2 is with regard to X. This is true for the rest of the proof as well.

There are 4 more cases to consider, corresponding to whether $t_1 \in Y$ or $t_1 \in \Gamma(Y)$ and whether $t_2 \in X$ or $t_2 \in \Gamma(X)$.

(1) If $t_1 \in Y$ and $t_2 \in X$, then $Y \cap X^*$ separates s_1 and t_1 and $X \cap Y^*$ separates s_2 and t_2. Thus they are both tight.

(2) If $t_1 \in \Gamma(Y)$ and $t_2 \in \Gamma(X)$, then since $t_1 \neq t_2$ (as otherwise, e.g., X is not tight for (s_1, t_1)), we have $|\Gamma(X \cup Y)| \geq 2$. Since $X \cap Y$ separates s_1 and t_1, $|\Gamma(X \cap Y)| \geq 2$, and so both are equal to 2 and hence $X \cap Y$ is tight.

(3) If $t_1 \in \Gamma(Y)$ and $t_2 \in X$, then assume for a contradiction that $|\Gamma(X \cup Y)| \leq 1$. Then $\Gamma(X) \subseteq Y$. Since $\Gamma(X \cap Y^*) \subseteq (\Gamma(Y) \cap \Gamma(X)) \cup (Y^* \cap \Gamma(X)) \cup (\Gamma(Y) \cap X)$, we then conclude that $\Gamma(X \cap Y^*) \subseteq \Gamma(Y) \cap X$. Then every path from $t_2 \in X \cap Y^*$ to $s_2 \in Y$ must use at least one vertex from $\Gamma(Y) \cap X$. But $|\Gamma(Y) \cap X| \leq 1$, since $|\Gamma(Y)| = 2$ and one neighbour of Y is t_1, which is in X^*. Therefore $\kappa_J(s_2, t_2) \leq 1$, a contradiction. Hence $|\Gamma(X \cup Y)| \geq 2$. But since $X \cap Y$ separates s_1 and t_1, $|\Gamma(X \cap Y)| \geq 2$, and so both are equal to 2 and therefore $X \cap Y$ is tight.

(4) This case is very similar to case 3, but we write it out for completeness. If $t_1 \in Y$ and $t_2 \in \Gamma(X)$, then assume for a contradiction that $|\Gamma(X \cup Y)| \leq 1$. Then $\Gamma(Y) \subseteq X$. Since $\Gamma(X^* \cap Y) \subseteq (\Gamma(X) \cap \Gamma(Y)) \cup (X^* \cap \Gamma(Y)) \cup (\Gamma(X) \cap Y)$, we then conclude that $\Gamma(X^* \cap Y) \subseteq \Gamma(X) \cap Y$. Then every path from $t_1 \in X^* \cap Y$ to $s_1 \in X$ must use at least one vertex from $\Gamma(X) \cap Y$. But $|\Gamma(X) \cap Y| \leq 1$, since $|\Gamma(X)| = 2$ and one neighbour of X is t_2, which is in Y^*. Therefore $\kappa_J(s_2, t_2) \leq 1$, a contradiction. Hence $|\Gamma(X \cup Y)| \geq 2$. But since $X \cap Y$ separates s_1 and t_1, $|\Gamma(X \cap Y)| \geq 2$, and so both are equal to 2 and therefore $X \cap Y$ is tight.

□

Taking one of the sets in the above lemma to be some core $C_i \in \mathcal{C}$, we get the following lemma as a corollary.

Lemma 10. *For any tight set X and any $C_i \in \mathcal{C}$, either $C_i \cap X \cap (T_X \cup T_i) = \emptyset$ or $C_i \subseteq X$.*

Proof. Suppose $C_i \cap X \cap (T_X \cup T_i) \neq \emptyset$. Then by Lemma 3, we have that either $X \cap C_i$ or $X^* \cap C_i$ is tight. This contradicts the minimality of C_i unless $C_i \subseteq X$. □

5. Counterexamples to Lemma 9 with Higher Requirements

Here we provide counterexamples to Lemma 9 in augmentation instances with higher requirements. We use the same notation and definitions as Section 4.

We consider augmentations from 4 to 5 and from 3 to 4, and give counterexamples to Lemma 9 in each case. These counterexamples must consist of tight sets X and Y that intersect on some terminal for which X is tight, but the desired sets are not tight. We can verify that the graphs in Figure 2 and 3 satisfy these conditions for 4-to-5 and 3-to-4 augmentation respectively.

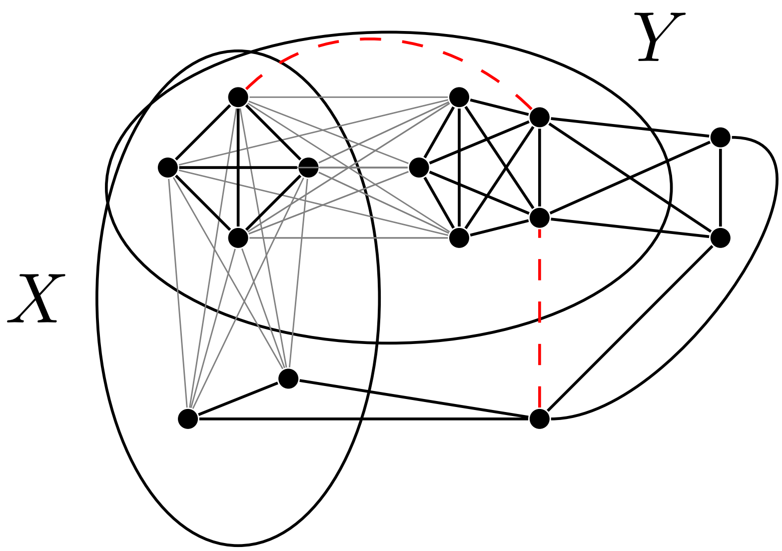

FIGURE 2

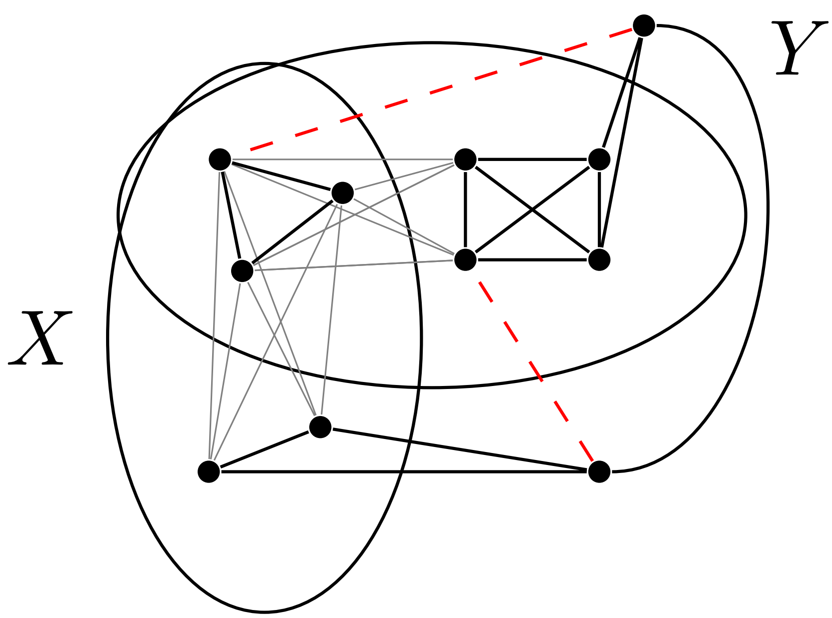

FIGURE 3

There is no significance to the gray edges; the colour is simply for visibility. Red dashed lines indicate terminal pairs.

The graph in Figure 3 requires some stronger properties to be an appropriate counterexample. In particular, we must have that the terminal in Y for which Y is tight must be in $\Gamma(X)$ or X, as well as the other conditions in the relevant case of the proof of Lemma 9 for 2-to-3 augmentation. The proof that this is the only bad case is very similar to the proof of Lemma 9, with only one case in that proof that is no longer true. This kind of structural requirement for a bad case gives some hope that a workaround can be found.

6. Extended Halosets: A Potential Framework for 2-to-3 Augmentation

In trying to extend the ideas of [Nutov, 2009] to the case of general requirements for 2 to 3 augmentation in VC-SNDP, it is no longer as easy to define a family $\mathcal{F}$ of tight sets such that, as was done in that paper:

(1) $\mathcal{F}$ can be decomposed into a constant number of biuncrossable families.
(2) Every terminal t for which there exists a tight set containing t is contained in some tight set $X \in \mathcal{F}$.

Condition 1 is necessary for the use of a 2-approximation algorithm for covering biuncrossable families that is the basis of the algorithm presented in [Nutov, 2009]. Condition 2 is satisfied in the case of rooted requirements in that paper, and is related the the "progression" of the cores at each iteration of the covering.

We recall some definitions from [Nutov, 2009] here: the cores C_i are the minimal tight sets, and the haloset M_i is the maximal tight set containing C_i and no other core. The family $\mathcal{F}$ is defined as $\mathcal{F} = \{X : X \subseteq M_i \text{ for some } M_i, X \text{ is tight}\}$. Using the basic lemmas presented in the paper, we conclude that Condition 2 is satisfied. This allows us to ignore terminals that are not contained in some C_i, as the tight sets containing those terminals will be covered when we cover the set $\mathcal{F}$. This step is important, as it allows us to make important conclusions about the number of cores after covering $\mathcal{F}$.

The situation with general requirements is more complicated. Basic modifications to the definition of M_i for the general case that satisfies Condition 2 leaves many difficulties for Condition 1. Definitions that satisfy Condition 1 more easily do not satisfy Condition 2. In the latter case, covering $\mathcal{F}$ may not make very much progress in a solution at all, since we may end up covering very few sets. Thus we need an approach that gives us a family $\mathcal{F}$ satisfying Condition 2 that we can properly identify the difficulties for Condition 1. We present one possibility here.

Many details are omitted from the following exposition. Many of the proofs are very similar to proofs in [Nutov, 2009] or elsewhere in these documents, and by the exploratory nature of this discussion, most of the ideas are not fully fleshed out. The author extends his apologies to the reader.

Definition 6. *For each terminal $t_i \in T$, let C_i be the inclusion-minimal tight set containing t_i. The set T_i is the set of terminals for which C_i is tight. The halo-family of C_i is defined as $H(C_i) = \{X : C_i \subseteq X, X \text{ is tight }, X \cap T = C_i \cap T_i\}$. The haloset of C_i, M_i, is the inclusion-maximal member of $H(C_i)$. Then $\mathcal{F} = \bigcup_i H(C_i)$.*

It is clear by this definition that Condition 2 is satisfied, as every terminal is in a core. We refer to this formulation of cores as "extended halosets", as it is an attempt to cover sets beyond the halosets of the minimal tight sets.

Unfortunately, we have lost the true minimality property of the cores with this definition. The first issue that this causes is that basic lemmas about the structure of the cores no longer hold. In particular, Lemma 10 from Section 4 no longer holds, even in the case where X is itself a core. See a counterexample in Figure 4. Core C_1 is minimal for t_1, t_3 and core C_2 is minimal for t_2, t_4 (these pairs are denoted with a blue dashed line). The other dashed red lines are the other requirements. The gray and black lines are edges; the gray is used only for visibility purposes.

Note that in this counterexample, we required that $t_1 \in \Gamma(C_2)$ and $t_2 \in \Gamma(C_1)$. When considering the proof of Lemma 10, we can see that this must be the case for a counterexample. Since $|\Gamma(C_1)| = |\Gamma(C_2)| = 2$, we conjecture that we can partition the cores into families such that Lemma 10 holds within a family, by using a colouring argument on an auxiliary graph as in [Nutov, 2009].

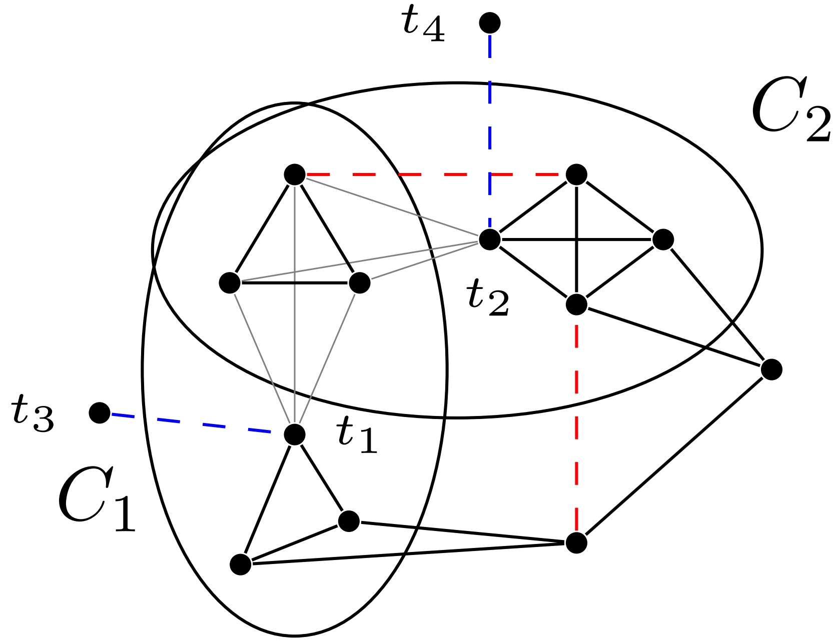

FIGURE 4

We now move on to identifying the reasons why $\mathcal{F}$ may not be biuncrossable, and suggest how to partition $\mathcal{F}$ to avoid these.

Let $X, Y \in \mathcal{F}$. Suppose $X, Y \in H(C_i)$ for some C_i. Then $X \cap Y$ and $X \cup Y$ are tight.

If $X \in H(C_i)$ and $Y \in H(C_j)$ for some C_i and C_j, then we consider several different cases.

Firstly, if $C_i \cap C_j \cap (T_i \cup T_j) = \emptyset$, then this case is very similar to what was shown in [Nutov, 2009]; either $X \cap Y^*$ and $X^* \cap Y$ are tight, or we have $C_i \cap T_i \subseteq \Gamma(M_j)$ or $C_j \cap T_j \subseteq \Gamma(M_i)$. Then the cores satisfying one of the two latter conditions are partitioned into different families using a colouring argument. Given the results of Lemma 10, we can do the same, and partition into a constant number of families such that, within a family, neither $C_i \cap T_i \subseteq \Gamma(M_j)$ nor $C_j \cap T_j \subseteq \Gamma(M_i)$ is satisfied between cores.

If $C_i \cap C_j \cap (T_i \cup T_j) \neq \emptyset$, then using Lemma 10 we can show that one core must be contained in the other, say $C_j \subseteq C_i$.

If C_i is tight for some terminal in C_j, then X and Y are tight for some terminal in common, so $X \cap Y$ and $X \cup Y$ are both tight. Otherwise, we assume that C_i is not tight for any terminal in C_j.

Note that if $M_j \cap C_i$ and $M_j \cup C_i$ are tight, then so are $X \cap Y$ and $X \cup Y$, so we will identify the cases when this is not true. Since $M_j \cap C_i$ separates all the terminal pairs for which C_j is tight, then $M_j \cup C_i$ cannot also do so. The only way for this to be the case is if $C_i^* \cap T_i \subseteq \Gamma(M_j)$, i.e. the other ends of the terminal pairs for C_i are neighbours of M_j. This condition is similar to the previous two identified, and so we hope to make a similar colouring argument here so that we can avoid this case.

Unfortunately, the in-degree colouring argument made in [Nutov, 2009] does not apply to the auxiliary graph made by adding arcs from C_j to C_i when $C_j \subseteq C_i$ and $C_i^* \cap T_i \subseteq \Gamma(M_j)$, as both the in-degree and out-degree of some node can be unbounded. Illustrations of these cases are in Figures 5 and 6. The illustrations do not include all vertices, but rather just terminals and important vertices for illustrating the neighbourhoods of sets.

However, it does appear that some kind of colouring argument may be possible. The cases that occur in the two figures cannot happen simultaneously, as the layout of the terminal pairs must be different. That

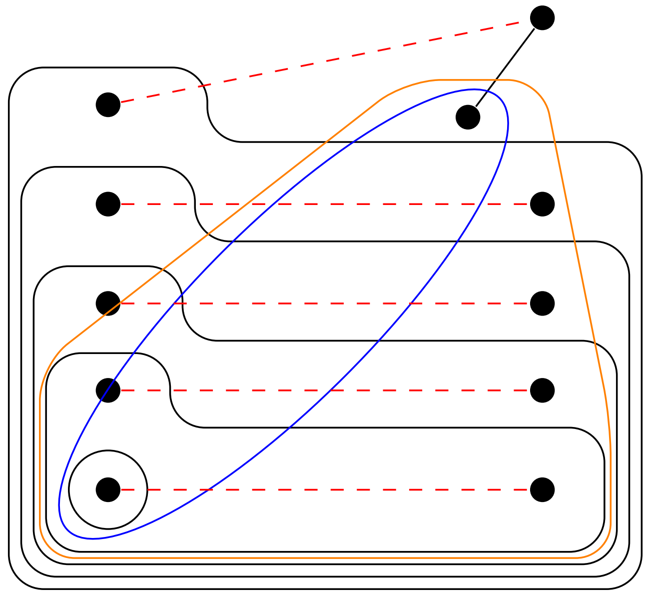

FIGURE 5. The black outlines are different cores, the red dashed lines denote requirement pairs, and the blue and orange outlines denote the halosets of the smallest and second smallest cores respectively. Using an arbitrarily large construction of this type, we see the in-degree in the auxiliary graph of the largest core pictured could be unbounded.

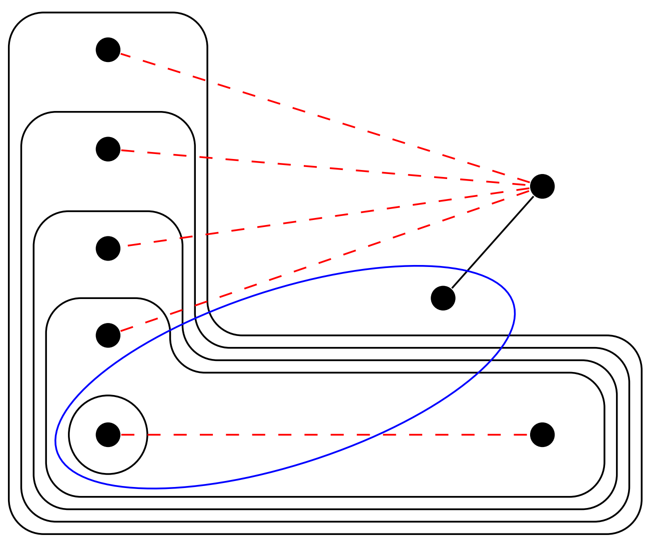

FIGURE 6. The black outlines are different cores, the red dashed lines denote requirement pairs, and the blue outline denotes the haloset of the smallest core. Using an arbitrarily large construction of this type, we see the out-degree in the auxiliary graph of the smallest core pictured could be unbounded.

is to say, while one node of the auxiliary graph can have both unbounded in- and out-degree, not all nodes can. It is possible that we could find a bound on the number of arcs in a subgraph of the auxiliary graph.

A final complication arises from the fact that a core can be contained in several other cores. However, the number of cores that intersect is bounded by a constant, as shown in Section 7, so this could potentially also be avoided.

Thus we conclude with the following conjecture.

Conjecture 11. *The family $\mathcal{F}$ can be partitioned into subfamilies such that, within a subfamily, if $C_j \subseteq C_i$, then $C_i^* \cap T_i \not\subseteq \Gamma(M_j)$.*

Together with the previously mentioned partitions, we then get:

Corollary 12. *The family $\mathcal{F}$ can be partitioned into a constant number of biuncrossable families.*

We can then cover these families using the 2-approximation algorithm.

Unfortunately, this would still not give us a constant approximation factor for the 2-to-3 augmentation. In the rooted case, as shown in [Nutov, 2012a], after covering $\mathcal{F}$, we can eventually guarantee that each core is tight for a certain number of terminals. In our case, we do not have this guarantee, so the best we can achieve is a logarithmic factor in the number of terminals $|T|$. Nonetheless, depending on what constant may come out of the required partitioning arguments, this method may still be a small improvement on the otherwise best-known algorithm found in [Chuzhoy and Khanna, 2009].

7. An Upper Bound on the Maximum Number of Intersecting Cores

In this section, we continue to work in the framework of the 2-to-3 augmentation, using the definitions of Section 4, and prove a lemma that may be useful in future work.

One property that holds in the case of rooted connectivity is that minimal tight sets (cores) do not intersect on terminals, i.e. $C_i \cap C_j \cap T = \emptyset$. Lemma 10 essentially says something similar for the 2-to-3 augmentation; we can conclude that cores do not intersect on terminals for which one of the cores is tight. One further result we get using this lemma is a bound on the number of cores that contain any given vertex. While we have not explicitly used this result anywhere, there are many places where it appeared it could be necessary in the future, so we record it here.

Proposition 13. *Given some $v \in V$, there are at most 4 cores that contain v.*

Before we give the proof, we will note that main fact that we use to prove the proposition. By Lemma 10, every core must have some vertex not contained in any other core, namely, some terminal for which it is tight. Also, the subgraph induced by the vertices of a core C_i is connected; otherwise, the connected component of C_i containing some terminal in T_i must be tight, contradicting minimality. Thus there must be a path P_i from v to $C_i \setminus \bigcup_{j \neq i} C_j$ that is contained in C_i for all cores C_i containing v.

Together with a restriction on the neighbourhood size of the cores, this fact bounds the number of cores that can contain a single vertex. Figure 7 gives an illustration that the above condition can be satisfied with 4 cores in the 2-to-3 augmentation.

The proof of Proposition 14 essentially demonstrates that this construction of the paths is the best possible. The main argument is in the proof of the following lemma.

Lemma 14. *Suppose there are 4 sets $X_1, X_2, X_3, X_4 \subseteq V$ such that $v \in X_1 \cap X_2 \cap X_3 \cap X_4$ for some $v \in V$, and $|\Gamma(X_i)| = 2$ for each i. Suppose there are 4 paths P_1, P_2, P_3, and P_4 from v to $\overline{X}_1, \overline{X}_2, \overline{X}_3$, and $\overline{X}_4$ respectively, where $\overline{X}_i = X_i \setminus \bigcup_{j \neq i} X_j$, such that $P_i \subseteq X_i$. Then $|\Gamma(X_i) \cap \bigcup_i P_i| \geq 2$ for $i \in \{1, 2, 3, 4\}$.*

Proof. The proof is illustrated in Figure 8. We proceed by contradiction and assume that $|\Gamma(X_1) \cap \bigcup_{i \in \{2,3,4\}} P_i| = 1$. Note that this quantity cannot be 0, as there must be some path leaving X_1 (say, P_2). P_1 then contains at least one neighbour of each of X_2, X_3 and X_4. Also, the neighbour of X_1 in the union of the paths must be in $X_2 \cap X_3 \cap X_4 \setminus X_1$, as otherwise there would be no path P_i contained in X_i for some other X_i. Lastly, the remainder of the paths P_1, P_2 and P_3 must have no vertices in X_1, as otherwise $|\Gamma(X_1)| \geq 2$.

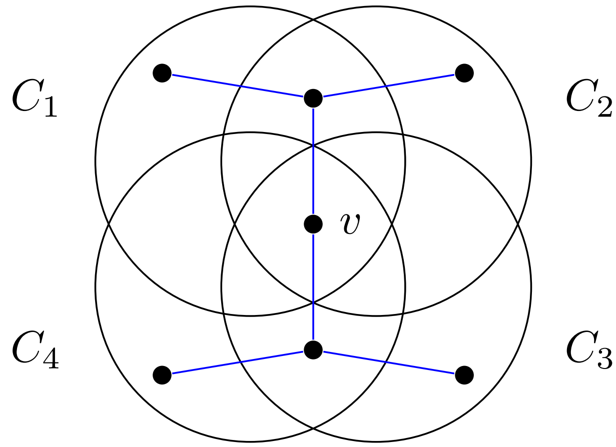

FIGURE 7. The indicated edges and vertices are those that belong to one of the paths P_i

The remainder of the proof is essentially a repetition of this argument with the remaining three sets, starting from the single vertex in $\Gamma(X_1)$. Since X_2 has a neighbour in P_1, it can only have one more neighbour in the remaining paths. Thus this neighbour must be in $X_3 \cap X_4 \setminus (X_1 \cup X_2)$, for the same reason as above. The remainder of the path P_2 then contains one more neighbour for each of X_3 and X_4.

From this point, the paths P_3 and P_4 require at least one more vertex each to be completed. However, these extra vertices would add more vertices to the neighbourhoods of X_3 and X_4, each of which are already size 2. Thus the paths cannot exist, and the lemma follows.

□

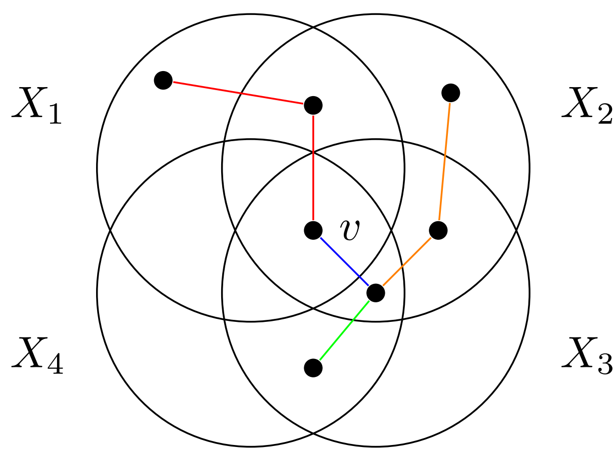

FIGURE 8. Proof of Lemma 3. The red path is an example of a path P_1. The remainder of the paths must then use the blue edge. The orange path is an example of the remainder of P_2. The remaining paths must then use the green edge. From this point, we cannot complete the construction of the paths without adding more than two neighbours to some set.

The proof of Proposition 14 then follows by noting that the cores satisfy the conditions for Lemma 15, and any core C_5 that would also contain v must have its own path P_5 as well. But then P_5 would add more neighbours to at least one of the existing 4 cores, contradicting the fact that they are tight sets.

8. Generalizing the Steiner Forest Algorithm

This section is concerned with a different approach to VC-SNDP which will require more time to explore fully. We consider a potential alternative to finding a constant factor approximation algorithm for the edge-connectivity version of SNDP, EC-SNDP, with the hope that some ideas may transfer to VC-SNDP.

For reference, the EC-SNDP problem is as follows: given a graph $G = (V, E)$ with edge costs c_e for $e \in E$, and requirements r_{st} for each pair of vertices $s, t \in V$, we desire a set of edges $F \subseteq E$ of minimum cost such that each pair of vertices s, t is connected by r_{st} edge-disjoint paths.

When $r_{st} \in \{0, 1\}$, this is known as the Steiner Tree, or Generalized Steiner Forest, Problem. This problem has a quite simple primal-dual 2-approximation algorithm (see [Williamson and Shmoys, 2011], page 170 for the algorithm and its analysis). Here, we explore a direct generalization of this primal-dual 2-approximation algorithm in hopes that it will lead to an alternate 2-approximation algorithm for EC-SNDP (an existing 2-approximation algorithm can be found in [Jain, 2001]).

The following LP models the problem described above, where we assume that requirements are either 0 or k and $\mathcal{S}$ denotes the family of subsets of V satisfying $|\delta(S)| < k$ for all $S \in \mathcal{S}$:

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in E} c_e x_e \\ \text{subject to} \quad & \sum_{e \in \delta(S)} x_e \geq k, \quad \forall S : s \in \mathcal{S} \\ & x_e \geq 0, \quad \forall e \in E \\ & -x_e \geq -1, \quad \forall e \in E \end{aligned}$$

The following is the dual:

$$\begin{aligned} \text{maximize} \quad & \sum_{S \in \mathcal{S}} y_S - \sum_{e \in E} z_e \\ \text{subject to} \quad & \sum_{S : e \in \delta(S)} y_S - z_e \leq c_e, \quad \forall e \in E \\ & y_S \geq 0, \quad \forall S \in \mathcal{S} \\ & z_e \geq 0, \quad \forall e \in E \end{aligned}$$

Let $\mathcal{C}$ denote the family of minimal members of $\mathcal{S}$. The algorithm proceeds as follows:

(1) Initialize all dual variables to 0, and out solution F to $\emptyset$.
(2) Uniformly increase the dual variables y until the the dual constraint for some new edge e is tight. Denote the amount of this increase by ε.
(3) Increase z_e by $\varepsilon \cdot |\{C \in \mathcal{C} : e \in \delta(C)\}|$ for every edge $e \in F$.
(4) Add e to F.
(5) Repeat until the set $\mathcal{C}$ is empty. Perform a "reverse-delete" step to discard unnecessary edges.
(6) Return F.

Throughout the algorithm, we have maintained a feasible dual solution through increases in the z variables.

Before we proceed with the analysis, we note one property of the edges for which we are increasing the variables z in step 3 by a nonzero amount. Denote the set of these edges $|E_z|$.

Lemma 15. *At any iteration of the algorithm, we have that*

$$\sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}| \leq (k-1)|\mathcal{C}|$$

Proof. Noting that for each set $C \in \mathcal{C}$, $|\delta(C)| \leq k-1$, we see that each C can be counted at most $k-1$ times in the L.H.S. Thus

$$\begin{aligned}\sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}| &\leq (k-1)|\{C \in \mathcal{C} : e \in \delta(C) \text{ for some } e \in E_z\}| \\ &\leq (k-1)|\mathcal{C}|\end{aligned}$$

□

We now continue with the primal-dual analysis.

At the end of the algorithm, we have

$$\sum_{e \in F} c_e = \sum_{e \in F} \left(\sum_{S : e \in \delta(S)} y_S - z_e \right) = \sum_S |\delta(S) \cap F| y_S - \sum_{e \in F} z_e.$$

We wish to show that this quantity is less than

$$f(k) \cdot \left(\sum_S k y_S - \sum_{e \in E} z_e \right),$$

since we will then have an $f(k)$-approximation.

At any iteration of the algorithm, the L.H.S of the inequality is increased by

$$\varepsilon \cdot \left(\sum_C |\delta(C) \cap F| - \sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}| \right)$$

and the R.H.S. is increased by

$$\varepsilon \cdot f(k) \cdot \left(k \cdot |\mathcal{C}| - \sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}| \right).$$

Thus to prove the desired inequality we simply need to show that

$$\sum_C |\delta(C) \cap F| - \sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}| \leq k \cdot f(k) \cdot |\mathcal{C}| - f(k) \cdot \sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}|.$$

Equivalently, we can show

$$\sum_C |\delta(C) \cap F| \leq k \cdot f(k) \cdot |\mathcal{C}| - (f(k) - 1) \cdot \sum_{e \in E_z} |\{C \in \mathcal{C} : e \in \delta(C)\}|.$$

Note that from Lemma 1, we have that the R.H.S. of this inequality is greater than or equal to

$$\begin{aligned}k \cdot f(k) \cdot |\mathcal{C}| - (f(k) - 1) \cdot (k-1) \cdot |\mathcal{C}| &= f(k) \cdot |\mathcal{C}| + (k-1) \cdot |\mathcal{C}| \\ &= (f(k) + k - 1) \cdot |\mathcal{C}|.\end{aligned}$$

Therefore it suffices to show the following:

Conjecture 16. *At any iteration of the algorithm, we have*

$$\sum_C |\delta(C) \cap F| \leq (f(k) + k - 1) \cdot |\mathcal{C}|$$

For the case of $k = 1$, that is, the Steiner forest problem, this is proven with $f(k) = 2$ in order to get the known 2-approximation. It is not clear whether the inequality holds in this more general case, what $f(k)$ is needed, or whether the same techniques from the proof of the $k = 1$ case can be generalized. We leave further work on this approach to the reader.

9. Conclusion

In this project, we explored a few different approaches focused answering the open problem of whether VC-SNDP with requirements $r_{st} \in \{0, 1, 2, 3\}$ admits an approximation algorithm with constant approximation factor. While no conclusive results were found, we identified a few areas of interest and proved several basic lemmas that may be helpful in future work.

The open questions in this area remain open, and this document records one first attempt towards a solution. Hopefully this record can prove useful to future researchers interested in answering these questions.

References

[Chuzhoy and Khanna, 2009] Chuzhoy, J. and Khanna, S. (2009). An O($k^3 logn$)-approximation algorithm for vertex-connectivity survivable network design. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 437–441.

[Fleischer et al., 2006] Fleischer, L., Jain, K., and Williamson, D. P. (2006). Iterative rounding 2-approximation algorithms for minimum-cost vertex connectivity problems. *Journal of Computer and System Sciences*, 72(5):838 – 867. Special Issue on FOCS 2001.

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

[Karp, 1972] Karp, R. M. (1972). *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA.

[Nutov, 2009] Nutov, Z. (2009). A note on rooted survivable networks. *Information Processing Letters*, 109(19):1114 – 1119.

[Nutov, 2012a] Nutov, Z. (2012a). Approximating minimum-cost connectivity problems via uncrossable bifamilies. *ACM Trans. Algorithms*, 9(1):1:1–1:16.

[Nutov, 2012b] Nutov, Z. (2012b). Approximating subset k-connectivity problems. *Journal of Discrete Algorithms*, 17:51 – 59. Selected papers from the 2011 meeting of LSD/LAW.

[Williamson and Shmoys, 2011] Williamson, D. P. and Shmoys, D. B. (2011). *The Design of Approximation Algorithms*. Cambridge University Press.