A Note on Iterated Rounding for the Survivable Network Design Problem[*]

Chandra Chekuri[1] and Thapanapong Rukkanchanunt[†][2]

1 Dept. of Computer Science, University of Illinois, Urbana, 61801, USA
chekuri@illinois.edu

2 Chiang Mai University, 239 Huay Kaew Road, Chiang Mai 50200, Thailand
thapanapong.r@cmu.ac.th

Abstract

In this note we consider the survivable network design problem (SNDP) in undirected graphs. We make two contributions. The first is a new counting argument in the iterated rounding based 2-approximation for edge-connectivity SNDP (EC-SNDP) originally due to Jain [10]. The second contribution is to make some connections between hypergraphic version of SNDP (Hypergraph-SNDP) introduced in [17] and edge and node-weighted versions of EC-SNDP and element-connectivity SNDP (Elem-SNDP). One useful consequence is a 2-approximation for Elem-SNDP that avoids the use of set-pair based relaxation and analysis.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems: Computations on Discrete Structures

Keywords and phrases survivable network design, iterated rounding, approximation, element connectivity

Digital Object Identifier 10.4230/OASIcs.SOSA.2018.2

1 Introduction

The *survivable network design problem* (SNDP) is a fundamental problem in network design and has been instrumental in the development of several algorithmic techniques. The input to SNDP is a graph $G = (V, E)$ and an integer requirement $r(uv)$ between each unordered pair of nodes uv. The goal is to find a minimum-cost subgraph H of G such that for each pair uv, the connectivity in H between u and v is at least $r(uv)$. We use $r_{\max}$ to denote $\max_{uv} r(uv)$, the maximum requirement. We restrict attention to undirected graphs in this paper. There are several variants depending on whether the costs are on edges or on nodes, and whether the connectivity requirement is edge, element or node connectivity. Unless otherwise specified we will assume that G has edge-weights $c : E \to \mathbb{R}_+$. We refer to the three variants of interest based on edge, element and vertex connectivity as EC-SNDP, Elem-SNDP and VC-SNDP. All of them are NP-Hard and APX-hard to approximate even in very special cases.

The seminal work of Jain [10] obtained a 2-approximation for EC-SNDP via the technique of iterated rounding that was introduced in the same paper. A 2-approximation for Elem-SNDP was obtained, also via iterated rounding, in [7, 5]. For VC-SNDP the current best approximation bound is $O(r_{\max}^3 \log |V|)$ [6]; it is also known from hardness results in [2] that the approximation bound for VC-SNDP must depend polynomially on $r_{\max}$ under standard hardness assumptions.

[*] Work on this paper supported in part by NSF grants CCF-1016684, CCF-1319376 and CCF-1526799.

[†] The second author was an undergraduate student in the Dept. of Computer Science at Univ. of Illinois when he worked on results of this paper.

1st Symposium on Simplicity in Algorithms (SOSA 2018).
Editor: Raimund Seidel; Article No. 2; pp. 2:1–2:10
OASICS Open Access Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Our contribution: In this note we revisit the iterated rounding framework that yields a 2-approximation for EC-SNDP and Elem-SNDP. The framework is based on arguing that for a class of covering problems, a basic feasible solution to an LP relaxation for the covering problem has a variable of value at least $\frac{1}{2}$. This variable is then rounded up to 1 and the residual problem is solved inductively. A key fact needed to make this iterative approach work is that the residual problem lies in the same class of covering problems. This is ensured by working with the class of skew-supermodular (also called weakly-supermodular) requirement functions which capture EC-SNDP as a special case. The proof of existence of an edge with large value in a basic feasible solution for this class of requirement functions has two components. The first is to establish that a basic feasible solution is characterized by a laminar family of sets in the case of EC-SNDP (and set pairs in the case of Elem-SNDP). The second is a counting argument that uses this characterization to obtain a contradiction if no variable is at least $\frac{1}{2}$. The counting argument of Jain [10] has been simplified and streamlined in subsequent work via fractional token arguments [1, 13]. These arguments have been applied for several related problems for which iterated rounding has been shown to be a powerful technique; see [12]. The fractional token argument leads to short and slick proofs. At the same time we feel that it is hard to see the intuition behind the argument. Partly motivated by pedagogical reasons, in this note, we provide a different counting argument along with a longer explanation. The goal is to give a more *combinatorial* flavor to the argument. We give this argument in Section 2.

The second part of the note is on Elem-SNDP. A 2-approximation for this problem has been derived by generalizing the iterated rounding framework to a set-pair based relaxation [7, 5]. The set-pair based relaxation and arguments add substantial notation to the proofs although one can see that there are strong similarities to the proofs used in EC-SNDP. The notational overhead limits the ability to teach and understand the proof for Elem-SNDP. Interestingly, in a little noticed paper, Zhao, Nagamochi and Ibaraki [17] defined a generalization of EC-SNDP to hypergraphs which we refer to as Hypergraph-SNDP. They observed that Elem-SNDP can be easily reduced to Hypergraph-SNDP in which the only non-zero weight hyperedges are of size 2 (regular edges in a graph). The advantage of this reduction is that one can derive a 2-approximation for Elem-SNDP by essentially appealing to the same argument as for EC-SNDP with a few minor details. We believe that this is a useful perspective. Second, there is a simple and well-known connection between node-weighted network design in graphs and network design problems on hypergraphs. We explicitly point these connections which allows us to derive some results for Hypergraph-SNDP. Section 3 describes these connections and results.

This note assumes that the reader has some basic familiarity with previous literature on SNDP and iterated rounding.

2 Iterated rounding for EC-SNDP

The 2-approximation for EC-SNDP is based on casting it as a special case of covering a skew-supermodular requirement function by a graph. We set up the background now. Given a finite ground set V an integer valued set function $f : 2^V \to \mathbb{Z}$ is skew-supermodular if for all $A, B \subseteq V$ one of the following holds:

$$\begin{aligned} f(A) + f(B) &\leq f(A \cap B) + f(A \cup B) \\ f(A) + f(B) &\leq f(A - B) + f(B - A) \end{aligned}$$

Given an edge-weighted graph $G = (V, E)$ and a skew-supermodular requirement function $f : 2^V \to \mathbb{Z}$, we can consider the problem of finding the minimum-cost subgraph $H = (V, F)$

of G such that H covers f; that is, for all $S \subseteq V$, $|\delta_F(S)| \geq f(S)$. Here $\delta_F(S)$ is the set of all edges in F with one endpoint in S and the other outside. Given an instance of EC-SNDP with input graph $G = (V, E)$ and edge-connectivity requirements $r(uv)$ for each pair uv, we can model it by setting $f(S) = \max_{u \in S, v \notin S} r(uv)$. It can be verified that f is skew-supermodular. The important aspect of skew-supermodular functions that make them well-suited for the iterated rounding approach is the following.

▶ **Lemma 1** ([10]). *Let $G = (V, E)$ be a graph and $f : 2^V \to \mathbb{Z}$ be a skew-supermodular requirement function, and $F \subseteq E$ be a subset of edges. The residual requirement function $g : 2^V \to \mathbb{Z}$ defined by $g(S) = f(S) - |\delta_F(S)|$ for each $S \subseteq V$ is also skew-supermodular.*

Although the proof is standard by now we will state it in a more general way.

▶ **Lemma 2.** *Let $f : 2^V \to \mathbb{Z}$ be a skew-supermodular requirement function and let $h : 2^V \to \mathbb{Z}_+$ be a symmetric submodular function. Then $g = f - h$ is a skew-supermodular function.*

Proof. Since h is submodular we have that for all $A, B \subseteq V$,

$$h(A) + h(B) \geq h(A \cup B) + h(A \cap B).$$

Since h is also symmetric it is posi-modular which means that for all $A, B \subseteq V$,

$$h(A) + h(B) \geq h(A - B) + h(B - A).$$

Note that h satisfies both properties for each A, B. It is now easy to check that $f - h$ is skew-supermodular. ◀

Lemma 1 follows from Lemma 2 by noting that the cut-capacity function $|\delta_F| : 2^V \to \mathbb{Z}_+$ is submodular and symmetric in undirected graphs. We also note that the same property holds for the more general setting when G is a hypergraph.

The standard LP relaxation for covering a function by a graph is described below where there is variable $x_e \in [0, 1]$ for each edge $e \in E$.

$$\begin{array}{llll} \min \sum_{e \in E} c_e x_e & & & \\ \sum_{e \in \delta(S)} x_e & \geq & f(S) & S \subset V \\ x_e & \in & [0,1] & e \in E \end{array}$$

The technical theorem that underlies the 2-approximation for EC-SNDP is the following.

▶ **Theorem 3** ([10]). *Let f be a non-trivial[1] skew-supermodular function. In any basic feasible solution $\overline{x}$ to the LP relaxation of covering f by a graph G there is an edge e such that $\overline{x}_e \geq \frac{1}{2}$.*

To prove the preceding theorem it suffices to focus on basic feasible solutions $\overline{x}$ that are fully fractional; that is, $\overline{x}_e \in (0, 1)$ for all e. For a set of edges $F \subseteq E$ let $\chi(F) \in \{0, 1\}^{|E|}$ denote the characteristic vector of F; that is, a $|E|$-dimensional vector that has a 1 in each position corresponding to an edge $e \in F$ and a 0 in all other positions. Theorem 3 is built upon the following characterization of basic feasible solutions and is shown via uncrossing arguments.

[1] We use the term non-trivial to indicate that there is at least one set $S \subset V$ such that $f(S) > 0$.

▶ **Lemma 4** ([10])**.** *Let $\overline{x}$ be a fully-fractional basic feasible solution to the the LP relaxation. Then there is a laminar family of vertex subsets $\mathcal{L}$ such that $\overline{x}$ is the unique solution to the system of equalities*

$$x(\delta(S)) = f(S) \qquad S \in \mathcal{L}.$$

In particular this also implies that $|\mathcal{L}| = |E|$ and that the vectors $\chi(\delta(S))$, $S \in \mathcal{L}$ are linearly independent.

The second part of the proof of Theorem 3 is a counting argument that relies on the characterization in Lemma 4. The rest of this section describes a counting argument which we believe is slightly different from the previous ones in terms of the main invariant. The goal is to derive it organically from simpler cases.

With every laminar family we can associate a rooted forest. We use terminology for rooted forests such as leaves and roots as well as set terminology. We refer to a set $C \in \mathcal{L}$ as a child of a set S if $C \subset S$ and there is no $S' \in \mathcal{L}$ such that $C \subset S' \subset S$; If C is the child of S then S is the parent of C. Maximal sets of $\mathcal{L}$ correspond to the roots of the forest associated with $\mathcal{L}$.

2.1 Counting Argument

The proof is via contradiction where we assume that $0 < \overline{x}_e < \frac{1}{2}$ for each $e \in E$. We call the two nodes incident to an edge as the endpoints of the edges. We say that an endpoint u *belongs* to a set $S \in \mathcal{L}$ if u is the minimal set from $\mathcal{L}$ that contains u.

We consider the simplest setting where $\mathcal{L}$ is a collection of disjoint sets, in other words, all sets are maximal. In this case the counting argument is easy. Let $m = |E| = |\mathcal{L}|$. For each $S \in \mathcal{L}$, $f(S) \geq 1$ and $\overline{x}(\delta(S)) = f(S)$. If we assume that $\overline{x}_e < \frac{1}{2}$ for each e, we have $|\delta(S)| \geq 3$ which implies that each S contains at least 3 distinct endpoints. Thus, the m disjoint sets require a total of $3m$ endpoints. However the total number of endpoints is at most $2m$ since there are m edges, leading to a contradiction.

Now we consider a second setting where the forest associated with $\mathcal{L}$ has k leaves and h internal nodes but each internal node has at least two children. In this case, following Jain, we can easily prove a weaker statement that $\overline{x}_e \geq 1/3$ for some edge e. If not, then each leaf set S must have four edges leaving it and hence the total number of endpoints must be at least $4k$. However, if each internal node has at least two children, we have $h < k$ and since $h + k = m$ we have $k > m/2$. This implies that there must be at least $4k > 2m$ endpoints since the leaf sets are disjoint. But m edges can have at most $2m$ endpoints. Our assumption on each internal node having at least two children is obviously a restriction. So far we have not used the fact that the vectors $\chi(\delta(S)), S \in \mathcal{L}$ are linearly independent. We can handle the general case to prove $\overline{x}_e \geq 1/3$ by using the following lemma.

▶ **Lemma 5** ([10])**.** *Suppose C is a unique child of S. Then there must be at least two endpoints in S that belong to S.*

Proof. If there is no endpoint that belongs to S then $\delta(S) = \delta(C)$ but then $\chi(\delta(S))$ and $\chi(\delta(C))$ are linearly dependent. Suppose there is exactly one endpoint that belongs to S and let it be the endpoint of edge e. But then $\overline{x}(\delta(S)) = \overline{x}(\delta(C)) + \overline{x}_e$ or $\overline{x}(\delta(S)) = \overline{x}(\delta(C)) - \overline{x}_e$. Both cases are not possible because $\overline{x}(\delta(S)) = f(S)$ and $\overline{x}(\delta(C)) = f(C)$ where $f(S)$ and $f(C)$ are positive integers while $\overline{x}_e \in (0, 1)$. Thus there are at least two end points that belong to S. ◀

Using the preceding lemma we prove that $\overline{x}_e \geq 1/3$ for some edge e. Let k be the number of leaves in $\mathcal{L}$ and h be the number of internal nodes with at least two children and let ℓ be the number of internal nodes with exactly one child. We again have $h < k$ and we also have $k + h + \ell = m$. Each leaf has at least four endpoints. Each internal node with exactly one child has at least two end points which means the total number of endpoints is at least $4k + 2\ell$. But $4k + 2\ell = 2k + 2k + 2\ell > 2k + 2h + 2\ell > 2m$ and there are only $2m$ endpoints for m edges. In other words, we can ignore the internal nodes with exactly one child since there are two endpoints in such a node/set and we can effectively charge one edge to such a node.

We now come to the more delicate argument to prove the tight bound that $\overline{x}_e \geq \frac{1}{2}$ for some edge e. Our main contribution is to show an invariant that effectively reduces the argument to the case where we can assume that $\mathcal{L}$ is a collection of leaves. This is encapsulated in the claim below which requires some notation. Let $\alpha(S)$ be the number of sets of $\mathcal{L}$ contained in S including S itself. Let $\beta(S)$ be the number of edges whose *both* endpoints lie inside S. Recall that $f(S)$ is the requirement of S.

▶ **Claim.** *For all $S \in \mathcal{L}$, $f(S) \geq \alpha(S) - \beta(S)$.*

Assuming that the claim is true we can do an easy counting argument. Let $R_1, R_2, \ldots, R_h$ be the maximal sets in $\mathcal{L}$ (the roots of the forest). Note that $\sum_{i=1}^{h} \alpha(R_i) = |\mathcal{L}| = m$. Applying the claim to each R_i and summing up,

$$\sum_{i=1}^{h} f(R_i) \geq \sum_{i=1}^{h} \alpha(R_i) - \sum_{i=1}^{h} \beta(R_i) \geq m - \sum_{i=1}^{h} \beta(R_i).$$

Note that $\sum_{i=1}^{h} f(R_i)$ is the total requirement of the maximal sets. And $m - \sum_{i=1}^{h} \beta(R_i)$ is the total number of edges that cross the sets $R_1, \ldots, R_h$. Let E' be the set of edges crossing these maximal sets. Now we are back to the setting with h disjoint sets and E' edges with $\sum_{i=1}^{h} f(R_i) \geq |E'|$. This easily leads to a contradiction as before if we assume that $\overline{x}_e < \frac{1}{2}$ for all $e \in E'$. Formally, each set R_i requires $> 2f(R_i)$ edges crossing it from E' and therefore R_i contains at least $2f(R_i) + 1$ endpoints of edges from E'. Since $R_1, \ldots, R_h$ are disjoint the total number of endpoints is at least $2\sum_i f(R_i) + h$ which is strictly more than $2|E'|$.

Thus, it remains to prove the claim which we do by inductively starting at the leaves of the forest for $\mathcal{L}$.

Case 1: S is a leaf node. We have $f(S) \geq 1$ while $\alpha(S) = 1$ and $\beta(S) = 0$ which verifies the claim.

Case 2: S is an internal nodes with k children $C_1, C_2, \ldots, C_k$. See Figure 1 for the different types of edges that are relevant. E_{cc} is the set of edges with end points in two different children of S. E_{cp} be the set of edges that cross exactly one child but do not cross S. E_{po} be the set of edges that cross S but do not cross any of the children. E_{co} is the set of edges that cross both a child and S. This notation is borrowed from [15].

Let $\gamma(S)$ be the number of edges whose both endpoints belong to S but not to any child of S. Note that $\gamma(S) = |E_{cc}| + |E_{cp}|$.

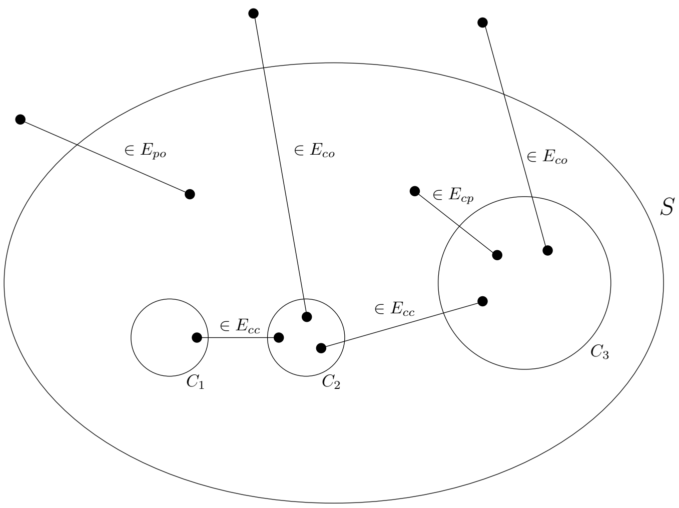

Figure 1 S is an internal node with several children. Different types of edges that play a role. p refers to parent set S, c refer to a child set and o refers to outside.

Then,

$$\begin{aligned}
\beta(S) &= \gamma(S) + \sum_{i=1}^{k} \beta(C_i) \\
&\geq \gamma(S) + \sum_{i=1}^{k} \alpha(C_i) - \sum_{i=1}^{k} f(C_i) \qquad (1) \\
&= \gamma(S) + \alpha(S) - 1 - \sum_{i=1}^{k} f(C_i)
\end{aligned}$$

(1) follows by applying the inductive hypothesis to each child. From the preceding inequality, to prove that $\beta(S) \geq \alpha(S) - f(S)$ (the claim for S), it suffices to show the following inequality.

$$\gamma(S) \geq \sum_{i=1}^{k} f(C_i) - f(S) + 1. \tag{2}$$

The right hand side of the above inequality can be written as:

$$\sum_{i=1}^{k} f(C_i) - f(S) + 1 = \sum_{e \in E_{cc}} 2x_e + \sum_{e \in E_{cp}} x_e - \sum_{e \in E_{po}} x_e + 1. \tag{3}$$

We consider two subcases.

Case 2.1: $\gamma(S) = 0$. This implies that E_{cc} and E_{cp} are empty. Since $\chi(\delta(S))$ is linearly independent from $\chi(\delta(C_1)), \ldots, \chi(\delta(C_k))$, we must have that E_{po} is not empty and hence $\sum_{e \in E_{po}} x_e > 0$. Therefore, in this case,

$$\sum_{i=1}^{k} f(C_i) - f(S) + 1 = \sum_{e \in E_{cc}} 2x_e + \sum_{e \in E_{cp}} x_e - \sum_{e \in E_{po}} x_e + 1 = - \sum_{e \in E_{po}} x_e + 1 < 1.$$

Since the left hand side is an integer, it follows that $\sum_{i=1}^{k} f(C_i) - f(S) + 1 \leq 0 = \gamma(S)$.

Case 2.2: $\gamma(S) \geq 1$. Recall that $\gamma(S) = |E_{cc}| + |E_{cp}|$.

$$\sum_{i=1}^{k} f(C_i) - f(S) + 1 = \sum_{e \in E_{cc}} 2x_e + \sum_{e \in E_{cp}} x_e - \sum_{e \in E_{po}} x_e + 1 \leq \sum_{e \in E_{cc}} 2x_e + \sum_{e \in E_{cp}} x_e + 1$$

By our assumption that $\overline{x}_e < \frac{1}{2}$ for each e, we have $\sum_{e \in E_{cc}} 2x_e < |E_{cc}|$ if $|E_{cc}| > 0$, and similarly $\sum_{e \in E_{cp}} x_e < |E_{cp}|/2$ if $|E_{cp}| > 0$. Since $\gamma(S) = |E_{cc}| + |E_{cp}| \geq 1$ we conclude that

$$\sum_{e \in E_{cc}} 2x_e + \sum_{e \in E_{cp}} x_e < \gamma(S).$$

Putting together we have

$$\sum_{i=1}^{k} f(C_i) - f(S) + 1 \leq \sum_{e \in E_{cc}} 2x_e + \sum_{e \in E_{cp}} x_e + 1 < \gamma(S) + 1 \leq \gamma(S)$$

as desired.

This completes the proof of the claim.

3 Connections between Hypergraph-SNDP, EC-SNDP and Elem-SNDP

Zhao, Nagamochi and Ibaraki [17] considered the extension EC-SNDP to hypergraphs. In a hypergraph $G = (V, \mathcal{E})$ each edge $e \in \mathcal{E}$ is a subset of V. The degree d of a hypergraph is $\max_{e \in \mathcal{E}} |e|$. Graphs are hypergraphs of degree 2. Given a set of hyperedges $F \subseteq \mathcal{E}$ and a vertex subset $S \subset V$, we use $\delta_F(S)$ to denote the set all of all hyperedges in F that have at least one endpoint in S and at least one endpoint in $V \setminus S$. It is well-known that $|\delta_F| : 2^V \to \mathbb{Z}_+$ is a symmetric submodular function.

Hypergraph-SNDP is defined as follows. The input consists of an edge-weighted *hypergraph* $G = (V, \mathcal{E})$ and integer requirements $r(uv)$ for each vertex pair uv. The goal is to find a minimum-cost hypergraph $H = (V, \mathcal{E}')$ with $\mathcal{E}' \subseteq \mathcal{E}$ such that for all uv and all S that separate u, v (that is $|S \cap \{u, v\}| = 1$), we have $|\delta_{\mathcal{E}'}(S)| \geq r(uv)$. Hypergraph-SNDP is a special case of covering a skew-supermodular requirement function by a hypergraph. It is clear that Hypergraph-SNDP generalizes EC-SNDP. Interestingly, [17] observed, via a simple reduction, that Hypergraph-SNDP generalizes Elem-SNDP as well. We now describe Elem-SNDP formally and briefly sketch the reduction from [17], and subsequently describe some implications of this connection.

In Elem-SNDP the input consists of an undirected edge-weighted graph $G = (V, E)$ with V partitioned into terminals T and non-terminals N. The "elements" are the edges and non-terimals, $N \cup E$. For each pair uv of terminals there is an integer requirement $r(uv)$, and the goal is to find a min-cost subgraph H of G such that for each pair uv of terminals there are $r(uv)$ element-disjoint paths from u to v in H. Note that element-disjoint paths can intersect in terminals. The notion of element-connectivity and Elem-SNDP have been useful in several settings in generalizing edge-connectivity problems while having some features of vertex connectivity. In particular, the current approximation for VC-SNDP relies on Elem-SNDP [6].

The reduction of [17] from Elem-SNDP to Hypergraph-SNDP is quite simple. It basically replaces each non-terminal $u \in N$ by a hyperedge. The reduction is depicted in Figure 2.

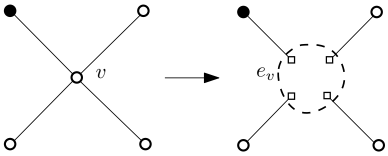

Figure 2 Reducing Elem-SNDP to Hypergraph-SNDP. Each non-terminal v is replaced by a hyperedge e_v by introducing dummy vertices on each edge incident to v. The original edges retain their cost while the new hyperedges are assigned a cost of zero.

The reduction shows that an instance of Elem-SNDP on G can be reduced to an instance of Hypergraph-SNDP on a hypergraph G' where the only hyperedges with non-zero weights in G' are the edges of the graph G. This motivates the definition of $d^+(G)$ which is the maximum degree of a hyperedge in G that has non-zero cost. Thus Elem-SNDP reduces to instances of Hypergraph-SNDP with $d^+ = 2$. In fact we can see that the same reduction proves the following.

▶ **Proposition 6.** *Node-weighted Elem-SNDP in which weights are only on non-terminals can be reduced in an approximation preserving fashion to Hypergraph-SNDP. In this reduction d^+ of the resulting instance of Hypergraph-SNDP is equal to Δ, the maximum degree of a non-terminal with non-zero weight in the instance of node-weighted Elem-SNDP.*

3.1 Reducing Elem-SNDP to problem of covering skew-supermodular functions by graphs

We saw that an instance of Elem-SNDP on a graph H can be reduced to an instance of Hypergraph-SNDP on a graph G where $d^+(G) = 2$. Hypergraph-SNDP on $G = (V, \mathcal{E})$ corresponds to covering a skew-supermodular function $f : 2^V \to \mathbb{Z}$ by G. Let $\mathcal{E} = F \uplus \mathcal{E}'$ where $\mathcal{E}'$ is the set of all hyperedges in G with degree more than 2; thus F is the set of all hyperedges of degree 2 and hence (V, F) is a graph. Since each edge in $\mathcal{E}'$ has zero cost we can include all of them in our solution, and work with the residual requirement function $g = f - |\delta_{\mathcal{E}'}|$. From Lemma 2 and the fact that the cut-capacity function of a hypergraph is also symmetric and submodular, g is a skew-supermodular function. Thus covering f by a min-cost sub-hypergraph of G can be reduced to covering g by a min-cost sub-graph of $G' = (V, F)$. We have already seen a 2-approximation for this in the context of EC-SNDP. The only issue is whether there is an efficient separation oracle for solving the LP for covering g by G'. This is a relatively easy exercise using flow arguments and we omit them. The main point we wish to make is that this reduction avoids working with set-pairs that are typically used for Elem-SNDP. It is quite conceivable that the authors of [17] were aware of this simple connection but it does not seem to have been made explicitly in their paper or in [16].

3.2 Approximating Hypergraph-SNDP

[17] derived a $d^+ H_{r_{\max}}$ approximation for Hypergraph-SNDP where $H_k = 1 + 1/2 + \ldots + 1/k$ is the k'th harmonic number. They obtain this bound via the augmentation framework for network design [9] and a primal-dual algorithm in each stage. In [16] they also observe that Hypergraph-SNDP can be reduced to Elem-SNDP via the following simple reduction. Given a hypergraph $G = (V, \mathcal{E})$ let $H = (V \cup N, E)$ be the standard bipartite graph representation of G where for each hyperedge $e \in \mathcal{E}$ there is a node $z_e \in N$; z_e is connected by edges in H

to each vertex $a \in e$. Let $r(uv)$ be the hyperedge connectivity requirement between a pair of vertices uv in the original instance of Hypergraph-SNDP. In H we label V as terminals and N as non-terminals. For any pair of vertices uv with $u, v \in V$, it is not hard to verify that the element-connectivity betwee u and v in H is the same as the hyperedge connectivity in G. See [16] for details. It remains to model the costs such that an approximation algorithm for element-connectivity in H can be translated into an approximation algorithm for hyperedge connectivity in G. This is straightforward. We simply assign cost to non-terminals in H; that is each node $z_e \in N$ corresponding to a hyperedge $e \in \mathcal{E}$ is assigned a cost equal to c_e. We obtain the following easy corollary.

▶ **Proposition 7.** *Hypergraph-SNDP can be reduced to node-weighted Elem-SNDP in an approximation preservation fashion.*

[16] do not explicitly mention the above but note that one can reduce Hypergraph-SNDP to (edge-weighted) Elem-SNDP as follows. Instead of placing a weight of c_e on the node z_e corresponding to the hyperedge $e \in \mathcal{E}$, they place a weight of $c_e/2$ on each edge incident to z_e. This transformation loses an approximation ratio of $d^+(G)/2$. From this they conclude that a β-approximation for Elem-SNDP implies a $d^+\beta/2$-approximation for Hypergraph-SNDP; via the 2-approximation for Elem-SNDP we obtain a d^+approximation for Hypergraph-SNDP. One can view this as reducing a node-weighted problem to an edge-weighted problem by transferring the cost on the nodes to all the edges incident to the node. Since a non-terminal can only be useful if it has at least two edges incident to it, in this particular case, we can put a weight of half the node on the edges incident to the node. A natural question here is whether one can directly get a d^+ approximation for Hypergraph-SNDP without the reduction to Elem-SNDP. We raise the following technical question.

▶ **Problem 8.** *Suppose f is a non-trivial skew-supermodular function on V and $G = (V, \mathcal{E})$ be a hypergraph. Let $\overline{x}$ be a basic feasible solution to the LP for covering f by G. Is there an hyperedge $e \in \mathcal{E}$ such that $\overline{x}_e \geq \frac{1}{d}$ where d is the degree of G?*

The preceding propositions show that Hypergraph-SNDP is essentially equivalent to node-weighted Elem-SNDP where the node-weights are only put on non-terminals. Node-weighted Steiner tree can be reduced to node-weighted Elem-SNDP and it is known that Set Cover reduces in an approximation preserving fashion to node-weighted Steiner tree [11]. Hence, unless $P = NP$, we do not expect a better than $O(\log n)$-approximation for Hypergraph-SNDP where $n = |V|$ is the number of nodes in the graph. Thus, the approximation ratio for Hypergraph-SNDP cannot be a constant independent of d^+. Node-weighted Elem-SNDP admits an $O(r_{\max} \log |V|)$ approximation; see [14, 3, 4, 8]. For planar graphs, and more generally graphs from a proper minor-closed family, an improved bound of $O(r_{\max})$ is claimed in [3]. The $O(r_{\max} \log |V|)$ bound can be better than the bound of d^+ in some instances. Here we raise a question based on the fact that planar graphs have constant average degree which is used in the analysis for node-weighted network design.

▶ **Problem 9.** *Is there an $O(1)$-approximation for node-weighted EC-SNDP and Elem-SNDP in planar graphs, in particular when $r_{\max}$ is a fixed constant?*

Finally, we hope the counting argument and the connections between Hypergraph-SNDP, EC-SNDP and Elem-SNDP will be useful for related problems including the problems involving degree constraints in network design.

References

1 Nikhil Bansal, Rohit Khandekar, and Viswanath Nagarajan. Additive guarantees for degree-bounded directed network design. *SIAM Journal on Computing*, 39(4):1413–1431, 2009.

2 Tanmoy Chakraborty, Julia Chuzhoy, and Sanjeev Khanna. Network design for vertex connectivity. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 167–176. ACM, 2008.

3 Chandra Chekuri, Alina Ene, and Ali Vakilian. Node-weighted network design in planar and minor-closed families of graphs. In *Automata, Languages, and Programming*, pages 206–217. Springer, 2012.

4 Chandra Chekuri, Alina Ene, and Ali Vakilian. Prize-collecting survivable network design in node-weighted graphs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 98–109. Springer, 2012.

5 J. Cheriyan, S. Vempala, and A. Vetta. Network design via iterative rounding of setpair relaxations. *Combinatorica*, 26(3):255–275, 2006.

6 Julia Chuzhoy and Sanjeev Khanna. An $O(k^3 \log n)$-approximation algorithm for vertex-connectivity survivable network design. *Theory of Computing*, 8:401–413, 2012. Preliminary version in Proc. of IEEE FOCS, 2009.

7 L. Fleischer, K. Jain, and D.P. Williamson. Iterative rounding 2-approximation algorithms for minimum-cost vertex connectivity problems. *Journal of Computer and System Sciences*, 72(5):838–867, 2006.

8 Takuro Fukunaga. Spider covers for prize-collecting network activation problem. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, pages 9–24. SIAM, 2015. URL: http://dl.acm.org/citation.cfm?id=2722129.2722131.

9 M.X. Goemans, A.V. Goldberg, S. Plotkin, D.B. Shmoys, E. Tardos, and D.P. Williamson. Improved approximation algorithms for network design problems. In *Proc. of ACM-SIAM SODA*, pages 223–232, 1994.

10 K. Jain. A factor 2 approximation algorithm for the generalized Steiner network problem. *Combinatorica*, 21(1):39–60, 2001. Preliminary version in FOCS 1998.

11 P. Klein and R. Ravi. A nearly best-possible approximation algorithm for node-weighted Steiner trees. *J. Algorithms*, 19(1):104–115, 1995. Preliminary version in IPCO 1993.

12 Lap Chi Lau, Ramamoorthi Ravi, and Mohit Singh. *Iterative methods in combinatorial optimization*, volume 46. Cambridge University Press, 2011.

13 Viswanath Nagarajan, R Ravi, and Mohit Singh. Simpler analysis of lp extreme points for traveling salesman and survivable network design problems. *Operations Research Letters*, 38(3):156–160, 2010.

14 Z. Nutov. Approximating minimum cost connectivity problems via uncrossable bifamilies and spider-cover decompositions. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 417–426. IEEE, 2009.

15 David P Williamson and David B Shmoys. *The design of approximation algorithms.* Cambridge university press, 2011.

16 Liang Zhao, Hiroshi Nagamochi, and Toshihide Ibaraki. A note on approximating the survivable network design problem in hypergraphs. *IEICE TRANSACTIONS on Information and Systems*, 85(2):322–326, 2002.

17 Liang Zhao, Hiroshi Nagamochi, and Toshihide Ibaraki. A primal–dual approximation algorithm for the survivable network design problem in hypergraphs. *Discrete applied mathematics*, 126(2):275–289, 2003. Preliminary version appeared in *Proc. of STACS*, 2001.